Computing Needs All Voices!

The pioneers pictured below are featured in our Computing Needs All Voices lesson. To learn more about them and their contributions, visit https://bit.ly/bootstrap-pioneers.

We are in the process of expanding our collection of pioneers. If there's someone else whose work inspires you, please let us know at https://bit.ly/pioneer-suggestion.
Notice and Wonder

Write down what you Notice and Wonder from the [What Most Schools Don't Teach](#) video. "Notices" should be statements, not questions. What stood out to you? What do you remember? "Wonders" are questions.

<table>
<thead>
<tr>
<th>What do you Notice?</th>
<th>What do you Wonder?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>


Windows and Mirrors

Think about the images and stories you've just encountered. Identify something(s) that served as a mirror for you, connecting you with your own identity and experience of the world. Write about who or what you connected with and why.

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

Identify something(s) from the film or the posters that served as a window for you, giving you insight into other people's experiences or expanding your thinking in some way.

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
Reflection: Problem Solving Advantages of Diverse Teams

This reflection is designed to follow reading LA Times Perspective: A solution to tech’s lingering diversity problem? Try thinking about ketchup.

1) The author argues that tech companies with diverse teams have an advantage. Why?

____________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________________
Many important questions ("What's the best restaurant in town?", "Is this law good for citizens?", etc.) are answered with data. Data Scientists try and answer these questions by writing programs that ask questions about data.

Data of all types can be organized into Tables.

- Every Table has a header row and some number of data rows.

- Quantitative data is numeric and measures an amount, such as a person's height, a score on a test, distance, etc. A list of quantitative data can be ordered from smallest to largest.

- Categorical data is data that specifies qualities, such as sex, eye color, country of origin, etc. Categorical data is not subject to the laws of arithmetic — for example, we cannot take the “average” of a list of colors.
<table>
<thead>
<tr>
<th>name</th>
<th>species</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>legs</th>
<th>pounds</th>
<th>weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sasha</td>
<td>cat</td>
<td>female</td>
<td>1</td>
<td>false</td>
<td>4</td>
<td>6.5</td>
<td>3</td>
</tr>
<tr>
<td>Snuffles</td>
<td>rabbit</td>
<td>female</td>
<td>3</td>
<td>true</td>
<td>4</td>
<td>3.5</td>
<td>8</td>
</tr>
<tr>
<td>Mittens</td>
<td>cat</td>
<td>female</td>
<td>2</td>
<td>true</td>
<td>4</td>
<td>7.4</td>
<td>1</td>
</tr>
<tr>
<td>Sunflower</td>
<td>cat</td>
<td>female</td>
<td>5</td>
<td>true</td>
<td>4</td>
<td>8.1</td>
<td>6</td>
</tr>
<tr>
<td>Felix</td>
<td>cat</td>
<td>male</td>
<td>16</td>
<td>true</td>
<td>4</td>
<td>9.2</td>
<td>5</td>
</tr>
<tr>
<td>Sheba</td>
<td>cat</td>
<td>female</td>
<td>7</td>
<td>true</td>
<td>4</td>
<td>8.4</td>
<td>6</td>
</tr>
<tr>
<td>Billie</td>
<td>snail</td>
<td>hermaphrodite</td>
<td>0.5</td>
<td>false</td>
<td>0</td>
<td>0.1</td>
<td>3</td>
</tr>
<tr>
<td>Snowcone</td>
<td>cat</td>
<td>female</td>
<td>2</td>
<td>true</td>
<td>4</td>
<td>6.5</td>
<td>5</td>
</tr>
<tr>
<td>Wade</td>
<td>cat</td>
<td>male</td>
<td>1</td>
<td>false</td>
<td>4</td>
<td>3.2</td>
<td>1</td>
</tr>
<tr>
<td>Hercules</td>
<td>cat</td>
<td>male</td>
<td>3</td>
<td>false</td>
<td>4</td>
<td>13.4</td>
<td>2</td>
</tr>
<tr>
<td>Toggle</td>
<td>dog</td>
<td>female</td>
<td>3</td>
<td>true</td>
<td>4</td>
<td>48</td>
<td>1</td>
</tr>
<tr>
<td>Boo-boo</td>
<td>dog</td>
<td>male</td>
<td>11</td>
<td>true</td>
<td>4</td>
<td>123</td>
<td>24</td>
</tr>
<tr>
<td>Fritz</td>
<td>dog</td>
<td>male</td>
<td>4</td>
<td>true</td>
<td>4</td>
<td>92</td>
<td>3</td>
</tr>
<tr>
<td>Midnight</td>
<td>dog</td>
<td>female</td>
<td>5</td>
<td>false</td>
<td>4</td>
<td>112</td>
<td>4</td>
</tr>
<tr>
<td>Rex</td>
<td>dog</td>
<td>male</td>
<td>1</td>
<td>false</td>
<td>4</td>
<td>28.9</td>
<td>9</td>
</tr>
<tr>
<td>Gir</td>
<td>dog</td>
<td>male</td>
<td>8</td>
<td>false</td>
<td>4</td>
<td>88</td>
<td>5</td>
</tr>
<tr>
<td>Max</td>
<td>dog</td>
<td>male</td>
<td>3</td>
<td>false</td>
<td>4</td>
<td>52.8</td>
<td>8</td>
</tr>
<tr>
<td>Nori</td>
<td>dog</td>
<td>female</td>
<td>3</td>
<td>true</td>
<td>4</td>
<td>35.3</td>
<td>1</td>
</tr>
<tr>
<td>Mr. Peanutbutter</td>
<td>dog</td>
<td>male</td>
<td>10</td>
<td>false</td>
<td>4</td>
<td>161</td>
<td>6</td>
</tr>
<tr>
<td>Lucky</td>
<td>dog</td>
<td>male</td>
<td>3</td>
<td>true</td>
<td>3</td>
<td>45.4</td>
<td>9</td>
</tr>
<tr>
<td>Kujo</td>
<td>dog</td>
<td>male</td>
<td>8</td>
<td>false</td>
<td>4</td>
<td>172</td>
<td>30</td>
</tr>
<tr>
<td>Buddy</td>
<td>lizard</td>
<td>male</td>
<td>2</td>
<td>false</td>
<td>4</td>
<td>0.3</td>
<td>3</td>
</tr>
<tr>
<td>Gila</td>
<td>lizard</td>
<td>female</td>
<td>3</td>
<td>true</td>
<td>4</td>
<td>1.2</td>
<td>4</td>
</tr>
<tr>
<td>Bo</td>
<td>dog</td>
<td>male</td>
<td>8</td>
<td>true</td>
<td>4</td>
<td>76.1</td>
<td>10</td>
</tr>
<tr>
<td>Niblet</td>
<td>rabbit</td>
<td>male</td>
<td>6</td>
<td>false</td>
<td>4</td>
<td>4.3</td>
<td>2</td>
</tr>
<tr>
<td>Snuggles</td>
<td>tarantula</td>
<td>female</td>
<td>2</td>
<td>false</td>
<td>8</td>
<td>0.1</td>
<td>1</td>
</tr>
<tr>
<td>Daisy</td>
<td>dog</td>
<td>female</td>
<td>5</td>
<td>true</td>
<td>4</td>
<td>68</td>
<td>8</td>
</tr>
<tr>
<td>Ada</td>
<td>dog</td>
<td>female</td>
<td>2</td>
<td>true</td>
<td>4</td>
<td>32</td>
<td>3</td>
</tr>
<tr>
<td>Miaulis</td>
<td>cat</td>
<td>male</td>
<td>7</td>
<td>false</td>
<td>4</td>
<td>8.8</td>
<td>4</td>
</tr>
<tr>
<td>Heathcliff</td>
<td>cat</td>
<td>male</td>
<td>1</td>
<td>true</td>
<td>4</td>
<td>2.1</td>
<td>2</td>
</tr>
<tr>
<td>Tinkles</td>
<td>cat</td>
<td>female</td>
<td>1</td>
<td>true</td>
<td>4</td>
<td>1.7</td>
<td>3</td>
</tr>
<tr>
<td>Tinkles</td>
<td>cat</td>
<td>female</td>
<td>1</td>
<td>true</td>
<td>4</td>
<td>1.7</td>
<td>3</td>
</tr>
<tr>
<td>Maple</td>
<td>dog</td>
<td>female</td>
<td>3</td>
<td>true</td>
<td>4</td>
<td>51.6</td>
<td>4</td>
</tr>
</tbody>
</table>
Quantitative data measures an amount and can be ordered from smallest to largest.

Categorical data specifies qualities and is not subject to the laws of arithmetic — for example, we cannot take the "average" of a list of colors.

Note: Numbers can be sometimes be categorical rather than quantitative!

For each piece of data below, circle whether it is Categorical or Quantitative data.

1. Hair color: Categorical
2. Age: Categorical
3. ZIP Code: Categorical
4. Date: Categorical
5. Height: Categorical
6. Sex: Categorical
7. Street Name: Categorical

For each question, circle whether it will be answered by Categorical or Quantitative data.

8. We'd like to find out the average price of cars in a lot: Quantitative
9. We'd like to find out the most popular color for cars: Categorical
10. We'd like to find out which puppy is the youngest: Categorical
11. We'd like to find out which cats have been fixed: Categorical
12. We want to know which people have a ZIP code of 02907: Categorical
13. We'd like to sort a list of phone numbers by area code: Categorical
Questions and Column Descriptions

What questions can you ask about the animals dataset? For each question, can it be answered by this dataset? Make sure you have at least two questions that can be answered, and at least one that cannot.

<table>
<thead>
<tr>
<th>Notice</th>
<th>Wonder</th>
<th>Answered by this dataset?</th>
</tr>
</thead>
<tbody>
<tr>
<td>I notice that</td>
<td>...so I wonder</td>
<td>Yes No</td>
</tr>
<tr>
<td>I notice that</td>
<td>...so I wonder</td>
<td>Yes No</td>
</tr>
<tr>
<td>I notice that</td>
<td>...so I wonder</td>
<td>Yes No</td>
</tr>
<tr>
<td>I notice that</td>
<td>...so I wonder</td>
<td>Yes No</td>
</tr>
<tr>
<td>I notice that</td>
<td>...so I wonder</td>
<td>Yes No</td>
</tr>
<tr>
<td>I notice that</td>
<td>...so I wonder</td>
<td>Yes No</td>
</tr>
</tbody>
</table>

Describe the table, and two of the columns, by filling in the blanks below.

1. This dataset is ________________________, which contains ________ data rows.
2. Some of the columns are:
   a. ________________________, which contains ______________________ data. Some example values are:  
      ________________________.
   b. ________________________, which contains ______________________ data. Some example values are:  
      ________________________.
The following is a dataset of a bicycle rider's training rides.

<table>
<thead>
<tr>
<th>date</th>
<th>miles</th>
<th>time (w/stops)</th>
<th>weather</th>
<th>average speed</th>
<th>max speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>04/10/2018</td>
<td>10</td>
<td>44</td>
<td>&quot;cloudy&quot;</td>
<td>13</td>
<td>30</td>
</tr>
<tr>
<td>05/30/2018</td>
<td>15</td>
<td>66</td>
<td>&quot;sunny&quot;</td>
<td>13.5</td>
<td>22</td>
</tr>
<tr>
<td>06/12/2018</td>
<td>12</td>
<td>61</td>
<td>&quot;rainy&quot;</td>
<td>11.2</td>
<td>25</td>
</tr>
<tr>
<td>07/04/2018</td>
<td>24</td>
<td>103</td>
<td>&quot;sunny&quot;</td>
<td>14</td>
<td>26</td>
</tr>
<tr>
<td>07/12/2018</td>
<td>24</td>
<td>120</td>
<td>&quot;windy&quot;</td>
<td>12.5</td>
<td>26</td>
</tr>
</tbody>
</table>

1) Decide whether each question below can or cannot be answered with the given data and circle your selection.

2) In the space provided below, explain how you could answer the question using the data or why you cannot answer the question.

<table>
<thead>
<tr>
<th>Question</th>
<th>Answered by this dataset?</th>
</tr>
</thead>
<tbody>
<tr>
<td>How many miles did the cyclist ride June 12th?</td>
<td>Yes No</td>
</tr>
<tr>
<td>What tire pressure produces the highest average speed?</td>
<td>Yes No</td>
</tr>
<tr>
<td>What is the average time it takes this cyclist to ride 1 mi?</td>
<td>Yes No</td>
</tr>
<tr>
<td>Does this cyclist ride slower when it is rainy?</td>
<td>Yes No</td>
</tr>
<tr>
<td>Does this cyclist ride faster when they are late to an appointment?</td>
<td>Yes No</td>
</tr>
<tr>
<td>How many miles has the cyclist ridden in total as part of their training?</td>
<td>Yes No</td>
</tr>
</tbody>
</table>

Are there any questions that you could find the answers to more than one way?
The Editor is a software program we use to write Code. Our Editor allows us to experiment with Code on the right-hand side, in the **Interactions Area**. For Code that we want to keep, we can put it on the left-hand side in the **Definitions Area**. Clicking the "Run" button causes the computer to re-read everything in the Definitions Area and erase anything that was typed into the Interactions Area.

### Data Types

Programming languages involve different *data types*, such as Numbers, Strings, Booleans, and even Images.

- **Numbers** are values like 1, 0.4, 1/3, and -8261.003.
  - Numbers are *usually* used for quantitative data and other values are *usually* used as categorical data.
  - In Pyret, any decimal *must* start with a 0. For example, 0.22 is valid, but .22 is not.
- **Strings** are values like "Emma", "Rosanna", "Jen and Ed", or even "08/28/1980".
  - All strings *must* be surrounded in quotation marks.
- **Booleans** are either true or false.

All values evaluate to themselves. The program 42 will evaluate to 42, the String "Hello" will evaluate to "Hello", and the Boolean false will evaluate to false.

### Operators

Operators (like +, -, *, <, etc.) work the same way in Pyret that they do in math.

- Operators are written between values, for example: 4 + 2.
- In Pyret, operators must always have a space around them. 4 + 2 is valid, but 4+2 is not.
- If an expression has different operators, parentheses must be used to show order of operations. 4 + 2 + 6 and 4 + (2 * 6) are valid, but 4 + 2 * 6 is not.

### Applying Functions

Applying functions works much the way it does in math. Every function has a name, takes some inputs, and produces some output. The function name is written first, followed by a list of arguments in parentheses.

- In math this could look like f(5) or g(10, 4).
- In Pyret, these examples would be written as f(5) and g(10, 4).
- Applying a function to make images would look like star(50, "solid", "red").
- There are many other functions, for example num-sqr, num-sqrt, triangle, square, string-repeat, etc.

Functions have *contracts*, which help explain how a function should be used. Every Contract has three parts:

- The **Name** of the function - literally, what it’s called.
- The **Domain** of the function - what types of values the function consumes, and in what order.
- The **Range** of the function - what type of value the function produces.
Strings and Numbers

Make sure you’ve loaded the code.pyret.org (CPO), clicked “Run”, and are working in the Interactions Area.

Strings

String values are always in quotes.
- Try typing your name (in quotes).
- Try typing a sentence like "I’m excited to learn to code!" (in quotes).
- Try typing your name with the opening quote, but without the closing quote. Read the error message!
- Now try typing your name without any quotes. Read the error message!

1) Explain what you understand about how strings work in this programming language. ____________________________

Numbers

2) Try typing 42 into the Interactions Area and hitting “Enter”.

3) Is 42 the same as "42"? Why or why not? Write your answer below:
   ____________________________________________________________

4) What is the largest number the editor can handle?
   ____________________________________________________________

5) Try typing 0.5. Then try typing .5. Then try clicking on the answer. Experiment with other decimals. Explain what you understand about how decimals work in this programming language.
   ____________________________________________________________

   ____________________________________________________________

6) What happens if you try a fraction like 1/3?
   ____________________________________________________________

7) Try writing negative integers, fractions and decimals. What do you learn?
   ____________________________________________________________

   ____________________________________________________________

Operators

8) Just like math, Pyret has operators like +, -, * and /. Try typing in 4 + 2, and then 4+2 (without the spaces). What can you conclude from this?
   ____________________________________________________________

9) Type in the following expressions, one at a time: 4 + 2 * 6, (4 + 2) * 6, 4 + (2 * 6). What do you notice?
   ____________________________________________________________

10) Try typing in 4 + "cat", and then "dog" + "cat". What can you conclude from this?
   ____________________________________________________________
### Booleans

Boolean-producing expressions are yes-or-no questions and will always evaluate to either **true** ("yes") or **false** ("no"). What will each of the expressions below evaluate to? Write down your prediction in the blanks provided and then type the code into the Interactions Area to see what it returns.

<table>
<thead>
<tr>
<th>Prediction</th>
<th>Result</th>
<th>Prediction</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) $3 \leq 4$</td>
<td>$2$</td>
<td>2) &quot;a&quot; &gt; &quot;b&quot;</td>
<td></td>
</tr>
<tr>
<td>3) $3 \equiv 2$</td>
<td>$3$</td>
<td>4) &quot;a&quot; &lt; &quot;b&quot;</td>
<td></td>
</tr>
<tr>
<td>5) $2 &lt; 4$</td>
<td>$6$</td>
<td>6) &quot;a&quot; == &quot;b&quot;</td>
<td></td>
</tr>
<tr>
<td>7) $5 \geq 5$</td>
<td>$5$</td>
<td>8) &quot;a&quot; &lt;&gt; &quot;a&quot;</td>
<td></td>
</tr>
<tr>
<td>9) $4 \geq 6$</td>
<td>$4$</td>
<td>10) &quot;a&quot; &gt;= &quot;a&quot;</td>
<td></td>
</tr>
<tr>
<td>11) $3 &lt;&gt; 3$</td>
<td>$3$</td>
<td>12) &quot;a&quot; &lt;&gt; &quot;b&quot;</td>
<td></td>
</tr>
<tr>
<td>13) $4 &lt;&lt;= 3$</td>
<td>$4$</td>
<td>14) &quot;a&quot; &gt;= &quot;b&quot;</td>
<td></td>
</tr>
</tbody>
</table>

15) In your own words, describe what `<` does.

16) In your own words, describe what `>=` does.

17) In your own words, describe what `<>` does.

<table>
<thead>
<tr>
<th>Prediction:</th>
<th>Result:</th>
</tr>
</thead>
<tbody>
<tr>
<td>18) string-contains(&quot;catnap&quot;, &quot;cat&quot;)</td>
<td></td>
</tr>
<tr>
<td>19) string-contains(&quot;cat&quot;, &quot;catnap&quot;)</td>
<td></td>
</tr>
</tbody>
</table>

20) In your own words, describe what `string-contains` does. Can you generate another expression using `string-contains` that returns true?

21) There are infinite numbers values out there (…-2,-1,0,1,2…) and infinite string values ("a", "aa", "aaa"…) But how many different Boolean values are there?
Open the **Animals Starter File** and click “Run”.
In the Interactions Window on the right, type `animals-table` and hit “Enter” to see the default view of the table.

### sort

Suppose we wanted to see the names of the animals in alphabetical order...

The **sort** function takes in three pieces of information:

1. A table
2. A column we want to sort the table by (declared using a String)
3. The order in which we want the column sorted (declared using a Boolean)

Test out these two expressions in the Interactions Area and record what you learn about ordering below:

- `sort(animals-table, "species", true)`
- `sort(animals-table, "species", false)`

1) `true` sorts the table... __________________________________________________________________________
2) `false` sorts the table... __________________________________________________________________________

Suppose we wanted to sort the **animals-table** by the **weeks** column to determine which animals were adopted quickest...

3) Would you use `true` or `false`? Explain. __________________________________________________________________________
4) Test it out, and write your thinking about quantitative columns at the end of your explanations of `true` and `false` above.

5) Which animal(s) were adopted the quickest? __________________________________________________________________________
6) Some functions produce Numbers, some produce Strings, some produce Booleans. What did the **sort** function produce? ____________

There are many other functions available to us in Pyret. We can describe them using contracts. The Contract for **sort** is:

```pyret
# sort :: Table, String, Boolean -> Table
```

- Each Contract begins with the function name: **in this case sort**
- Lists the data types required to satisfy its Domain: **in this case Table, String, Boolean**
- And then declares the data type of the Range it will return. **in this case Table**

Contracts can also be written with more detail, by adding variable names in the Domain:

```pyret
# sort :: (Table, String, Boolean) -> Table
```

**table-name** :: Table **column-name** :: String **order** :: Boolean

Suppose we wanted to sort the **animals-table** by the **legs** column to determine which animals had the most legs...

7) Fill in the blanks below with the code you’d use (We’ve put pieces of the Contract below each line to help you!):

```
function-name( table-name::Table, column-name::String, order::Boolean)
```

8) Which animal(s) had the most legs? __________________________________________________________________________
9) Think of another question you might answer quickly by sorting the table.

________________________________________________________________________________________

10) What code would you write to answer your question?

```
function-name( table-name::Table, column-name::String, order::Boolean)
```
Functions for Tables (continued)

**count**

# count :: Table, String \(\rightarrow\) Table

1) What is the Domain of count? ____________________________

2) What is the Range of count? ____________________________

3) What do you suspect the String in the Domain will describe? ____________________________

**Suppose we wanted to know how many animals had 4 legs...**

Type `count(animals-table, "legs")` into the Interactions Area and click "Enter"

4) What did the expression produce? ____________________________

5) How many animals had 4 legs? ____________________________

6) Think of another question you might be able to answer with the `count` function.

7) Fill in the blanks with the code you’d write.

`function-name(__________, table-name :: Table, ____________, column-name :: String)`

8) Tables that summarize data with a count are commonly used in the real world. Give two examples of where you’ve seen them before:

   • Example 1: ____________________________

   • Example 2: ____________________________

9) Newscasters and journalists often incorporate data into their reporting. How else might they display this information, besides using a table?

**first-n-rows**

10) Type `first-n-rows(animals-table, 5)`. What happens? ____________________________

11) If we wanted a table of the first 3 rows of the `animals-table`, what code would you write? ____________________________

12) What is the Contract for `first-n-rows`? ____________________________

★ What happens when you type `first-n-rows(sort(animals-table, "pounds", true), 5)`?

**Note:** The Domain for `first-n-rows` is Table, Number. In this case, the output of `sort(animals-table, "pounds", true)` is the Table!

★ See if you can figure out how to compose the code that would generate a table of the 10 oldest animals!

`function-name(__________, table-name :: Table, ____________, number :: Number)`. 

14
### Circles of Evaluation: Count, Sort, First-n-rows

For each scenario below, draw the Circle of Evaluation and then use it to write the code. When you're done, test your code out in the *Animals Starter File* and make sure it does what you'd expect it to.

```haskell
# count :: Table, String -> Table
# first-n-rows :: Table, Number -> Table
# sort :: Table, String, Boolean -> Table
```

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Circle of Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) We want to see the 10 animals who were adopted the quickest.</td>
<td>code:</td>
</tr>
<tr>
<td>2) We want to see the heaviest animal</td>
<td>code:</td>
</tr>
<tr>
<td>3) We want to take the first 8 animals from the table and put them in alphabetical order (by name).</td>
<td>code:</td>
</tr>
<tr>
<td>4) You notice that the lightest 16 animals weigh under 10 pounds and you want to know the count by species of those animals.</td>
<td>code:</td>
</tr>
</tbody>
</table>
# Exploring Displays

In the [Animals Starter File](#), use the contracts below to make each type of display. Then answer the questions below.

<table>
<thead>
<tr>
<th>Displays</th>
<th>Function Name</th>
<th>Table Arguments</th>
<th>Column Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bar Charts</strong></td>
<td><code>bar-chart :: Table, String -&gt; Image</code></td>
<td></td>
<td></td>
<td>Sketch a bar chart below. Bar charts summarize 1 column of <strong>categorical/quantitative</strong> data. This kind of display tells us...</td>
</tr>
<tr>
<td><strong>Pie Charts</strong></td>
<td><code>pie-chart :: Table, String -&gt; Image</code></td>
<td></td>
<td></td>
<td>Sketch a pie chart below. Pie charts summarize 1 column of <strong>categorical/quantitative</strong> data. This kind of display tells us...</td>
</tr>
<tr>
<td><strong>Box Plots</strong></td>
<td><code>box-plot :: Table, String -&gt; Image</code></td>
<td></td>
<td></td>
<td>Sketch a box plot below. Box plots summarize 1 column of <strong>categorical/quantitative</strong> data. This kind of display tells us...</td>
</tr>
<tr>
<td><strong>Histograms</strong></td>
<td><code>histogram :: Table, String, String, Number -&gt; Image</code></td>
<td></td>
<td></td>
<td>Sketch a histogram below. Histograms summarize 1 column of <strong>categorical/quantitative</strong> data. This kind of display tells us...</td>
</tr>
</tbody>
</table>
Using the Contracts below as a reference, draw the Circle of Evaluation for each prompt.

# pie-chart :: Table, String -> Image
# bar-chart :: Table, String -> Image
# histogram :: Table, String, String, Number -> Image
# box-plot :: Table, String -> Image
# first-n-rows :: Table, Number -> Table
# sort :: Table, String, Boolean -> Table

1) Make a bar-chart of the lightest 16 animals by sex.

★) What other bar chart might you want to compare this to?

2) Take the heaviest 20 animals and make a histogram of weeks to adoption (use "species" for your labels).

★) What other histogram might you want to compare this to?

3) Make a box-plot of age for the 11 animals who spent the most weeks in the shelter.

★) What other box plot might you want to compare this to?

4) Make a pie-chart of species for the 18 animals who spent the fewest weeks in the shelter.

★) What other pie chart might you want to compare this to?
In the [Animals Starter File](#), use the contracts below to make each type of display. Then answer the questions below.

### Line Graphs

```haskell
# line-graph :: Table, String, String -> Image
```

Sketch a line graph below.

Line Graphs summarize 2 columns of data.
This kind of display tells us...

### Scatter Plots

```haskell
# scatter-plot :: Table, String, String, String -> Image
```

Sketch a scatter plot below.

Scatter Plots summarize 2 columns of data.
This kind of display tells us...

### LR Plots

```haskell
# lr-plot :: Table, String, String, String -> Image
```

Sketch an Linear Regression Plot below.

LR Plots summarize 2 columns of data.
This kind of display tells us...

---

---
Match Display Descriptions to Circles of Evaluation

Match each prompt on the left to the Circle of Evaluation used to answer it.

1. Make a pie-chart, showing the species of the 4 oldest animals.

2. Take the 4 heaviest animals and make a box plot of their weight.

3. Make a table showing the count of the species in this dataset, sorted from most to least.

4. Make a table showing the count of the 4 species with the most animals.
Using the Contracts below as a reference, draw the Circle of Evaluation for each prompt.

- \texttt{pie-chart :: Table, String \rightarrow Image}
- \texttt{bar-chart :: Table, String \rightarrow Image}
- \texttt{histogram :: Table, String, String, Number \rightarrow Image}
- \texttt{box-plot :: Table, String \rightarrow Image}
- \texttt{first-n-rows :: Table, Number \rightarrow Table}
- \texttt{sort :: Table, String, Boolean \rightarrow Table}

1) Take the youngest 12 animals and make a box-plot of pounds.

What other box plot might you want to compare this to? ________________________________

2) Make a pie-chart of legs for the 10 oldest animals.

What other pie chart might you want to compare this to? ________________________________

★) Take the 20 lightest animals, then take the 10 youngest of those animals and make a bar-chart of species

What other pie chart might you want to compare this to? ________________________________
Data Scientists use **displays** to visualize data. You've probably seen some of these charts, graphs and plots yourselves! When it comes to displaying **Categorical Data**, there are two displays that are especially useful.

1. **Bar charts** show the *count or percentage* of rows in each category.
   - Bar charts provide a visual representation of the frequency of values in a categorical column.
   - Bar charts have a bar for every category in a column.
   - The more rows in a category, the taller the bar.
   - Bars in a bar chart can be shown in *any order*, without changing the meaning of the chart. However, bars are usually shown in some sensible order (bars for the number of orders for different t-shirt sizes might be presented in order of smallest to largest shirt).

2. **Pie charts** show the *percentage* of rows in each category.
   - Pie charts provide a visual representation of the relative frequency of values in a categorical column.
   - Pie charts have a slice for every category in a column.
   - The more rows in a category, the larger the slice.
   - Slices in a pie chart can be shown in *any order*, without changing the meaning of the chart. However, slices are usually shown in some sensible order (e.g. slices might be shown in alphabetical order or from the smallest to largest slice).
Bar & Pie Chart - Notice and Wonder

What do you Notice and Wonder about the displays below?

San Francisco Unified School District, CA

Albuquerque Public Schools, NM

Hawaii DOE, HI

Hartford School District, CT

What do you Notice?

What do you Wonder?
Match each bar chart below to the pie chart that displays the racial demographic data from the same school district.

Cleveland Municipal School District

San Diego City Unified School District

Houston Independent School District

New York City Dept of Education
Stacked & Multi Bar Charts - Notice and Wonder

What do you Notice and Wonder about the displays below?

### Refugees in the Americas

![Bar Chart](image)

What do you Notice?

What do you Wonder?

### Distribution of sexes across Cats, Dogs and Rabbits

![Bar Chart](image)

What do you Notice?

What do you Wonder?
Matching Stacked and Multi Bar Charts

Match each stacked bar chart below to the multi bar chart that displays the same information.
Open the Animals Starter File and click "Run".

Part A:
Test the following expressions in the Interactions Area:
- `count(animals-table, "species")`
- `bar-chart(animals-table, "species")`

1) How do they compare?

2) Which do you like better: the bar chart or the table? Why?

Now test out the expression `pie-chart(animals-table, "species")`

3) How does this pie chart compare to the bar chart you just made?

When you first build a bar chart or pie chart in pyret, they are interactive displays. That means that you can mouse over them for more information. Hit the up arrow in the interactions area to reload your last expression and test it out!

Part B:

4) Make a `pie-chart` and `bar-chart` for the pounds column. Why isn't grouping the pounds column very useful?

5) Look at the list of columns in the Definitions Area. For which columns do you expect pie charts to be most useful?

Part C: (Best completed after Bar & Pie Chart - Notice and Wonder and Matching Bar and Pie Charts)

6) How is `pie-chart` similar to `bar-chart`?

7) How are `pie-chart` and `bar-chart` different?

8) When would you want to use one chart instead of another?

9) What information is provided in bar charts that is hidden in pie charts?

10) Why might this sometimes be problematic?
Introducing Displays for Subgroups

This page is designed to be used with the Animals Starter File.

Part A

1) How many cats are male? ________________________

2) How many cats are female? ________________________

3) Would you imagine that there are about as many male dogs as female dogs? Why or why not? ________________________

4) Would you imagine that the distribution of male and female animals will be similar for every species at the shelter? Why or why not?

Part B

Comparing groups is great, but sometimes we want to compare sub-groups across groups. In this example, we want to compare the distribution of sexes across each species.

Fortunately, Pyret has two functions that let us specify both a group and a subgroup:

```
# stacked-bar-chart :: ( Table, String, String ) -> Image

# multi-bar-chart :: ( Table, String, String ) -> Image
```

5) Make a stacked-bar-chart showing the distribution of sexes across species in our shelter.

6) Make a multi-bar-chart showing the distribution of sexes across species in our shelter.

7) What do you notice? ________________________

8) What do you wonder? ________________________

9) Do you learn anything surprising from these displays? ________________________

10) Which display do you think is easier to read? Why? ________________________
Bar Chart - Notice and Wonder

<table>
<thead>
<tr>
<th>Washington, D.C. Public Schools</th>
<th>West Warwick Schools, RI</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Bar Chart" /></td>
<td><img src="image2.png" alt="Bar Chart" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Providence Schools, RI</th>
<th>Oxnard Union HS District, CA</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image3.png" alt="Bar Chart" /></td>
<td><img src="image4.png" alt="Bar Chart" /></td>
</tr>
</tbody>
</table>

**What do you Notice?**

**What do you Wonder?**
What do you Notice and Wonder about the pie charts below?

<table>
<thead>
<tr>
<th>Boston School District, MA</th>
<th>Santa Fe Public Schools, NM</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.9%</td>
<td>44.6%</td>
</tr>
<tr>
<td>19.8%</td>
<td>51.5%</td>
</tr>
<tr>
<td>22.8%</td>
<td>44.4%</td>
</tr>
</tbody>
</table>

- "White"
- "Black"
- "Hispanic or Latinx"
- "Asian"
- "Some other race alone"
- "Two or more races"

<table>
<thead>
<tr>
<th>Washington, D.C. Public Schools</th>
<th>Hawaii DOE, HI</th>
</tr>
</thead>
<tbody>
<tr>
<td>4%</td>
<td>11%</td>
</tr>
<tr>
<td>11.1%</td>
<td>22%</td>
</tr>
<tr>
<td>37.4%</td>
<td>37%</td>
</tr>
</tbody>
</table>

- "White"
- "Black"
- "Hispanic or Latinx"
- "Asian"
- "Two or more races"

<table>
<thead>
<tr>
<th>What do you Notice?</th>
<th>What do you Wonder?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Data Cycle

Data Science is all about asking questions of data. Sometimes the answer is easy to compute. Sometimes the answer to a question is already in the dataset - no computation needed. And sometimes the answer just sparks more questions! Data Scientists ask a lot of questions, and each one adds a chapter to the story of their research. Even if a question is a "dead-end", it’s valuable to share what the question was and what work you did to answer it!

We start by Asking Questions - questions that can be answered with data. It’s useful to think of all questions as falling into four categories:

- **Lookup questions** can be answered simply by looking up a single value in the table. Examples of lookup questions might be "How many legs does Felix have?" or "What species is Sheba?"

- **Arithmetic questions** can be answered by computing an answer within a single column. Examples of arithmetic questions might be "How much does the heaviest animal weigh?" or "What is the average age of animals from the shelter?"

- **Statistical questions** are where things get interesting! The answers are never black-and-white. If we asked, "How old are animals at the shelter?" we could report back the average age, the age that shows up most frequently or the range of the ages. There are also some statistical questions that deal with relationships between two columns: "Do cats tend to be adopted faster than dogs?" or "Are older animals heavier than young ones?"

- **Questions we can’t answer** are pretty common, too! We might wonder where the animal shelter is located, or what time of year the data was gathered! But the data in the table won’t help us answer that question, so as Data Scientists we might need to do some research beyond the data. And if nothing turns up, we simply recognize that there are limits to what we can analyze.

Next, we Consider Data. This could be done by conducting a survey, observing and recording data, or finding a dataset that meets our needs. Since our data is contained in a table, it’s useful to start by asking two questions:

- What rows do we care about? - Is it all the animals? Just the lizards?

- What columns do we need? - Are we examining the ages of the animals? Their weights?

Then, we Analyze the Data, by producing data displays and new tables of filtered or transformed data and using them to identify patterns and relationships.

- Are we making a pie chart? A bar chart? Something else?

Finally, we Interpret the Data, by answering our questions and summarizing the results. As we’ve already seen from the Animals Dataset, these interpretations often lead to new questions… and the cycle begins again.
Take a moment to look at the table of Pokemon data below:

<table>
<thead>
<tr>
<th>name</th>
<th>type1</th>
<th>hitpoint</th>
<th>attack</th>
<th>defense</th>
<th>speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulbasaur</td>
<td>Grass</td>
<td>45</td>
<td>49</td>
<td>49</td>
<td>45</td>
</tr>
<tr>
<td>Ivysaur</td>
<td>Grass</td>
<td>60</td>
<td>62</td>
<td>63</td>
<td>60</td>
</tr>
<tr>
<td>Venusaur</td>
<td>Grass</td>
<td>80</td>
<td>82</td>
<td>83</td>
<td>80</td>
</tr>
<tr>
<td>Mega Venusaur</td>
<td>Grass</td>
<td>80</td>
<td>100</td>
<td>123</td>
<td>80</td>
</tr>
<tr>
<td>Charmander</td>
<td>Fire</td>
<td>39</td>
<td>52</td>
<td>43</td>
<td>65</td>
</tr>
<tr>
<td>Charmeleon</td>
<td>Fire</td>
<td>58</td>
<td>64</td>
<td>58</td>
<td>80</td>
</tr>
<tr>
<td>Charizard</td>
<td>Fire</td>
<td>78</td>
<td>84</td>
<td>78</td>
<td>100</td>
</tr>
<tr>
<td>Mega Charizard X</td>
<td>Fire</td>
<td>78</td>
<td>130</td>
<td>111</td>
<td>100</td>
</tr>
<tr>
<td>Mega Charizard Y</td>
<td>Fire</td>
<td>78</td>
<td>104</td>
<td>78</td>
<td>100</td>
</tr>
<tr>
<td>Squirtle</td>
<td>Water</td>
<td>44</td>
<td>48</td>
<td>65</td>
<td>43</td>
</tr>
<tr>
<td>Wartortle</td>
<td>Water</td>
<td>59</td>
<td>63</td>
<td>80</td>
<td>58</td>
</tr>
</tbody>
</table>

For each question about this data below, write down whether it’s a Lookup, Arithmetic, Statistical Question, or one that cannot be answered with the table.

<table>
<thead>
<tr>
<th>Question</th>
<th>Question Type</th>
<th>Which Rows?</th>
<th>Column(s)?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>What type is Charizard?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Which Pokemon is the fastest?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>What is Wartortle’s attack score?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>What is the average defense score?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>What is a typical defense score?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Is Ivysaur faster than Venusaur?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Is speed related to attack score?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>What is the most common type?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Does one type tend to be faster than others?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Are hit-points (hp) similar for all Pokemon?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Which Pokemon has a speed of 43?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Data Cycle: Consider Data

For each question about the animals dataset below, identify the type of question and fill in the Rows and Columns needed to answer it. Then, for the final two cycles, think of your own questions and fill in any additional information needed.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question</th>
<th>Question Type</th>
<th>Question Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>How old is Boo-boo?</td>
<td>What question do you have?</td>
<td>(circle one):</td>
<td>Lookup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Arithmetic</td>
<td>Statistical</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Are there more cats than dogs in the shelter?</td>
<td>What question do you have?</td>
<td>(circle one):</td>
<td>Lookup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Arithmetic</td>
<td>Statistical</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>What question do you have?</td>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
<td>(circle one):</td>
<td>Lookup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Arithmetic</td>
<td>Statistical</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Rows**: All the rows, just the cats, fixed dogs, etc.

**Columns**: age, weight-in-kilograms, weeks, etc.
Data Cycle: Analyzing with Displays

For each question below, complete the first three steps of the Data Cycle. For the final Data Cycle, develop your own question and complete the remaining steps. Once you know what code to write, type it into Pyret and try it out!

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>How many of each species are at the shelter?</strong></td>
<td>What question do you have?</td>
</tr>
<tr>
<td><strong>How many of each sex are at the shelter?</strong></td>
<td>What question do you have?</td>
</tr>
<tr>
<td><strong>What question do you have?</strong></td>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
<tr>
<td><strong>Which Rows should we investigate?</strong> (All the rows, just the cats, fixed dogs, etc.)</td>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
<tr>
<td><strong>Which Rows should we investigate?</strong> (All the rows, just the cats, fixed dogs, etc.)</td>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
<tr>
<td><strong>Ask Questions</strong></td>
<td><strong>Consider Data</strong></td>
</tr>
<tr>
<td><strong>Analyze Data</strong></td>
<td><strong>What code will make the table or display you want?</strong></td>
</tr>
<tr>
<td><strong>What code will make the table or display you want?</strong></td>
<td><strong>What code will make the table or display you want?</strong></td>
</tr>
</tbody>
</table>
A subset of the whole Animals Dataset is shown in the table below.

<table>
<thead>
<tr>
<th>name</th>
<th>species</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>legs</th>
<th>pounds</th>
<th>weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sasha</td>
<td>cat</td>
<td>female</td>
<td>1</td>
<td>false</td>
<td>4</td>
<td>6.5</td>
<td>3</td>
</tr>
<tr>
<td>Sunflower</td>
<td>cat</td>
<td>female</td>
<td>5</td>
<td>true</td>
<td>4</td>
<td>8.1</td>
<td>6</td>
</tr>
<tr>
<td>Felix</td>
<td>cat</td>
<td>male</td>
<td>16</td>
<td>true</td>
<td>4</td>
<td>9.2</td>
<td>5</td>
</tr>
<tr>
<td>Sheba</td>
<td>cat</td>
<td>female</td>
<td>7</td>
<td>true</td>
<td>4</td>
<td>8.4</td>
<td>6</td>
</tr>
<tr>
<td>Billie</td>
<td>snail</td>
<td>hermaphrodite</td>
<td>0.5</td>
<td>false</td>
<td>0</td>
<td>0.1</td>
<td>3</td>
</tr>
<tr>
<td>Snowcone</td>
<td>cat</td>
<td>female</td>
<td>2</td>
<td>true</td>
<td>4</td>
<td>6.5</td>
<td>5</td>
</tr>
<tr>
<td>Wade</td>
<td>cat</td>
<td>male</td>
<td>1</td>
<td>false</td>
<td>4</td>
<td>3.2</td>
<td>1</td>
</tr>
<tr>
<td>Hercules</td>
<td>cat</td>
<td>male</td>
<td>3</td>
<td>false</td>
<td>4</td>
<td>13.4</td>
<td>2</td>
</tr>
<tr>
<td>Toggle</td>
<td>dog</td>
<td>female</td>
<td>3</td>
<td>true</td>
<td>4</td>
<td>48</td>
<td>1</td>
</tr>
</tbody>
</table>

Using this table - or the full dataset - write three questions of each type below.

<table>
<thead>
<tr>
<th>Question</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lookup</td>
</tr>
<tr>
<td>2</td>
<td>Lookup</td>
</tr>
<tr>
<td>3</td>
<td>Lookup</td>
</tr>
<tr>
<td>4</td>
<td>Arithmetic</td>
</tr>
<tr>
<td>5</td>
<td>Arithmetic</td>
</tr>
<tr>
<td>6</td>
<td>Arithmetic</td>
</tr>
<tr>
<td>7</td>
<td>Statistical</td>
</tr>
<tr>
<td>8</td>
<td>Statistical</td>
</tr>
<tr>
<td>9</td>
<td>Statistical</td>
</tr>
<tr>
<td>Ask Questions</td>
<td>Question Type (circle one):</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>What question do you have?</td>
<td>Lookup</td>
</tr>
<tr>
<td></td>
<td>Arithmetic</td>
</tr>
<tr>
<td></td>
<td>Statistical</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>What did you find out? What can you infer?</td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
</tr>
</tbody>
</table>

35
Probability, Inference, and Sample Size

How can you tell if a coin is fair, or designed to cheat you? Statisticians know that a fair coin should turn up "heads" about as often as "tails", so they begin with the null hypothesis: they assume the coin is fair, and start flipping it over and over to record the results.

A coin that comes up "heads" three times in a row could still be fair! The odds are 1-in-8, so it’s totally possible that the null hypothesis is still true. But what if it comes up "heads" five times in a row? Ten times?

Eventually, the chances of the coin being fair get smaller and smaller, and a Data Scientist can say “this coin is a cheat! The chances of it being fair are one in a million!”

By sampling the flips of a coin, we can infer whether the coin itself is fair or not. Inference plays a major role in Data Science and Statistics!

- If we survey pet owners about whether they prefer cats or dogs, the null hypothesis is that the odds of someone preferring dogs are about the same as them preferring cats. And if the first three people we ask vote for dogs (a 1-in-8 chance), the null hypothesis could still be true! But after five people? Ten?

- If we’re looking for gender bias in hiring, we might start with the null hypothesis that no such bias exists. If the first three people hired are all men, that doesn’t necessarily mean there’s a bias! But if 30 out of 35 hires are male, this is evidence that undermines the null hypothesis and suggests a real problem.

- If we poll voters for the next election, the null hypothesis is that the odds of voting for one candidate are the same as voting for the other. But if 80 out of 100 people say they’ll vote for the same candidate, we might reject the null hypothesis and infer that the population as a whole is biased towards that candidate!

Sample size matters! The more bias there is, the smaller the sample we need to detect it. Major biases might need only a small sample, but subtle ones might need a huge sample to be found. However, choosing a good sample can be tricky!

Random Samples are a subset of a population in which each member of the subset has an equal chance of being chosen. A random sample is intended to be a representative subset of the population. The larger the random sample, the more closely it will represent the population and the better our inferences about the population will tend to be.

Grouped Samples are a subset of a population in which each member of the subset was chosen for a specific reason. For example, we might want to look at the difference in trends between two groups ("Is the age of a dog a bigger factor in adoption time v. the age of a cat?"). This would require making grouped samples of just the dogs and just the cats.
Finding the Trick Coin

Open the Fair Coins Starter File, which defines coin1, coin2, and coin3. Click "Run". You can flip each coin by evaluating \( \text{flip(coin1)} \) in the Interactions Area (repeat for coins 2 and 3). One of these coins is fair, one will land on "heads" 75% of the time, and one will land on "heads" 90% of the time. Which one is which?

1) In the table below, record five flips of each coin and the number of "heads" you saw. Then convert that number into a percentage. Finally, write a "Y" or "N" depending on whether you think each coin is fair based on your sample.

<table>
<thead>
<tr>
<th>Sample</th>
<th>coin1</th>
<th>coin2</th>
<th>coin3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#heads</td>
<td>/5</td>
<td>/5</td>
<td>/5</td>
</tr>
<tr>
<td>% heads</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>fair?</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2) In the table below, record 15 more flips of each coin and the number of "heads" you saw. Then convert that number into a percentage. Finally, write a "Y" or "N" depending on whether you think each coin is fair based on your sample.

<table>
<thead>
<tr>
<th>Sample</th>
<th>coin1</th>
<th>coin2</th>
<th>coin3</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#heads</td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
</tr>
<tr>
<td>% heads</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>fair?</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3) Which weighted coin did you identify first: fair, 75%, or 90%?

4) Why was that coin the easiest to find?
Sampling and Inference

Open the Expanded Animals Starter File, and save a copy.

1) Evaluate the more-animals table in the Interactions Area. This is the complete population of animals from the shelter!

Here is a true statement about that population: The population is 47.7% fixed and 52.3% unfixed.

Type each of the following lines into the Interactions Area and hit “Enter”.
random-rows(more-animals, 10)
random-rows(more-animals, 40)

2) What do you get? _______________________________________________________

3) What is the Contract for random-rows? ______________________________________

4) What does the random-rows function do? ______________________________________

5) In the Definitions Area, define small-sample and large-sample to be random-rows(more-animals, 10) and random-rows(more-animals, 40), respectively.

6) Make a pie-chart for the animals in each sample, showing percentages of fixed and unfixed.
   - The percentage of fixed animals in the entire population is 47.7%
   - The percentage of fixed animals in small-sample is _____________________________
   - The percentage of fixed animals in large-sample is _____________________________

7) Make a pie-chart for the animals in each sample, showing percentages for each species.
   - The percentage of tarantulas in the entire population is roughly 5%
   - The percentage of tarantulas in small-sample is _____________________________
   - The percentage of tarantulas in large-sample is _____________________________

8) Click “Run” to direct the computer to generate a different set of random samples of these sizes. Make a new pie-chart for each sample, showing percentages for each species.
   - The percentage of tarantulas in the entire population is roughly 5%
   - The percentage of tarantulas in small-sample is _____________________________
   - The percentage of tarantulas in large-sample is _____________________________

9) Which sample size gave us a more accurate inference about the whole population? Why?
   _______________________________________________________
   _______________________________________________________
   _______________________________________________________
   _______________________________________________________
Predictions from Samples

1) In the Definitions Area of the Expanded Animals Starter File, define the following samples:

- tiny-sample = random-rows(more-animals, 10)
- small-sample = random-rows(more-animals, 20)
- medium-sample = random-rows(more-animals, 40)
- large-sample = random-rows(more-animals, 80)

2) Click "Run" and make a pie-chart of the species in the tiny-sample. What animals are in the sample? ________________

- Click "Run" for a new random tiny-sample, and make another pie-chart for species. What animals are there? ________________
- Click "Run" for a new random sample, and make yet another pie-chart for species. Based on these 3 samples, how many species do you think are at the shelter? ________________
- Which is the most common species at the shelter? ________________

3) What did you learn from taking multiple samples that you wouldn't have known if you'd only taken one?

________________________

4) Repeat the steps above, but for small-sample. What animals are in the sample?

________________________

5) Now that you've seen small-sample, how has your sense of the distribution of the species changed?

________________________

6) Now use medium-sample to make a pie-chart of the species. If there are about 400 animals at the shelter, how many of each species would you predict there to be?

________________________

7) Now use large-sample to make a pie-chart of the species. If there's anything you'd like to change about your prediction now that you've seen large-sample, record it here.

________________________

8) Let's see how accurate your prediction is... feel free to click "Run" and build a few more pie charts from your samples if you want to collect more information first! When you're ready, make a pie-chart of more-animals.

- Which predictions were closest? ________________
- Which predictions were off? ________________
- Were there any surprises? ________________

________________________

9) In the real world, we usually don't have access to a whole dataset to check predictions against! How could we test...

- Every giraffe on the planet?
- Everyone who has ever come in contact with a covid-positive person?
- Every person who identifies as queer?

What strategies can we use to make sure that predictions from samples are as close to accurate as possible?
Choosing Your Dataset

When selecting a dataset to explore, *pick something that matters to you!* You’ll be working with this data for a while, so you don’t want to pick something at random just to get it done.

When choosing a dataset, it’s a good idea to consider a few factors:

1. Is it **interesting**? Pick a dataset you’re genuinely interested in, so that you can explore questions that matter to you!

2. Is it **relevant**? Pick a dataset that deals with something personally relevant to you and your community! Does this data impact you in any way? Are there questions you have about the dataset that mean something to you or someone you know?

3. Is it **familiar**? Pick a dataset you know about, so you can use your expertise to deepen your analysis! You wouldn’t be able to make samples of the Animals Dataset properly if you didn’t know that some animals are much bigger or longer-lived than others.
Fill in the tables below by considering the rows and columns you need. Look up the Contract for the display and record the Pyret code you’d need to make it. If time allows, type your code into code.pyret.org (CPO) to see your display! The first column has been filled in for you.

1) A pie-chart showing the species of animals from the shelter.

<table>
<thead>
<tr>
<th>Which Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>All the animals</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

code: 

2) A bar-chart showing the sex of animals from the shelter.

<table>
<thead>
<tr>
<th>Which Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>All the animals</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

code: 

3) A histogram of the number of pounds that animals weigh.

<table>
<thead>
<tr>
<th>Which Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>All the animals</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

code: 

4) A box-plot of the number of pounds that animals weigh.

<table>
<thead>
<tr>
<th>Which Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>All the animals</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

code: 

5) A scatter-plot, using the animals’ species as the labels, age as the x-axis, and pounds as the y-axis.

<table>
<thead>
<tr>
<th>Which Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>All the animals</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

code: 

6) A scatter-plot, using the animals’ name as the labels, pounds as the x-axis, and weeks as the y-axis.

<table>
<thead>
<tr>
<th>Which Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>All the animals</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

code: 


My Dataset

The _____________________ dataset contains _______ data rows.

1) I’m interested in this data because __________________________

2) My friends, family or neighbors would be interested because __________________________

3) Someone else should care about this data because __________________________

4) In the table below, write down what you Notice and Wonder about this dataset.

<table>
<thead>
<tr>
<th>What do you NOTICE?</th>
<th>What do you WONDER?</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lookup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Arithmetic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Statistical</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Can't Answer</td>
</tr>
</tbody>
</table>

5) Look at each Wonder you wrote above. Is it an arithmetic question? A question you can’t answer from the data? Circle the type of question. Choose two columns to describe below.

6) ______________________, which contains __________________ data. Example values from this column include: column name categorical/quantitative

7) ______________________, which contains __________________ data. Example values from this column include: column name categorical/quantitative
Data Cycle: Categorical Data

Use the Data Cycle to explore the distribution of one or more categorical columns using **pie-charts and bar-charts**, and write down your findings.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one):</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Lookup</td>
</tr>
<tr>
<td></td>
<td>Arithmetic</td>
</tr>
<tr>
<td></td>
<td>Statistical</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th>What code will make the table or display you want?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>What did you find out? What can you infer?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>What new question(s) does this raise?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
Histograms

To best understand histograms, it's helpful to contrast them first with bar charts. Bar charts show the number of rows belonging to a given category. The more rows in each category, the taller the bar.

- **Bar charts provide a visual representation of the frequency of values in a categorical column.**

- There's no strict numerical way to order these bars, but **sometimes there's an order** that makes sense. For example, bars for the sales of different t-shirt sizes might be presented in order of smallest to largest shirt.

Histograms show the number of rows that fall within certain intervals, or "bins", on a horizontal axis. The more rows that fall within a particular "bin", the taller the bar.

- **Histograms provide a visual representation of the frequencies (or relative frequencies) of values in a quantitative column.**

- Quantitative data **can always be ordered**, so the bars of a histogram always progress from smallest (on the left) to largest (on the right).

- When dealing with histograms, it's important to select a good **bin size**. If the bins are too small or too large, it is difficult to see the shape of the dataset. Choosing a good bin size can take some trial and error!

The **shape** of a dataset tells us which values are more or less common.

- In a **symmetric** dataset, values are just as likely to occur a certain distance above the mean as below the mean. Each side of a symmetric distribution looks almost like a mirror-image of the other.

- Some extreme values may be far greater or far lower than the other values in a dataset. These extreme values are called **outliers**.

- A dataset that is **skewed left** has a few values that are unusually low. The histogram for a skewed left dataset has a few data points that are stretched out to the left (lower) end of the x-axis.

- A dataset that is **skewed right** has a few values that are unusually high. The histogram for a skewed right dataset has a few data points that are stretched out to the right (higher) end of the x-axis.

- One way to visualize the difference between a histogram of data that is **skewed left** or **skewed right** is to think about the lengths of our toes on our left and right feet. Much like the bar lengths of a histogram that is "skewed left", our left feet have smaller toes on the left and a bigger toe on the right. Our right feet have the big toe on the left and smaller toes on the right, more closely resembling the shape of a histogram of "skewed right" data.
### Summarizing Columns

<table>
<thead>
<tr>
<th>name</th>
<th>species</th>
<th>age</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Sasha&quot;</td>
<td>&quot;cat&quot;</td>
<td>1</td>
<td>6.5</td>
</tr>
<tr>
<td>&quot;Boo-boo&quot;</td>
<td>&quot;dog&quot;</td>
<td>11</td>
<td>12.3</td>
</tr>
<tr>
<td>&quot;Felix&quot;</td>
<td>&quot;cat&quot;</td>
<td>16</td>
<td>9.2</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;dog&quot;</td>
<td>6</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Wade&quot;</td>
<td>&quot;cat&quot;</td>
<td>1</td>
<td>3.2</td>
</tr>
<tr>
<td>&quot;Nibblet&quot;</td>
<td>&quot;rabbit&quot;</td>
<td>6</td>
<td>4.3</td>
</tr>
<tr>
<td>&quot;Maple&quot;</td>
<td>&quot;dog&quot;</td>
<td>3</td>
<td>51.6</td>
</tr>
</tbody>
</table>

The two displays below both summarize this table. The display on the left is a **Bar Chart**, the one on the right is a **Histogram**. What is similar about them? What is different?

<table>
<thead>
<tr>
<th>Similarities</th>
<th>Differences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. How many cats are there in the table above?
2. How many dogs are there?
3. How many animals weigh between 0-20 pounds?
4. How many animals weigh between 20-40 pounds?
5. Are there more animals weighing 40-60 than 60-140 pounds?
Making Histograms

Suppose we have a dataset for a group of 50 adults, showing the number of teeth each person has:

<table>
<thead>
<tr>
<th>Number of teeth</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
</tr>
<tr>
<td>26</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>4</td>
</tr>
<tr>
<td>29</td>
<td>3</td>
</tr>
<tr>
<td>30</td>
<td>5</td>
</tr>
<tr>
<td>31</td>
<td>3</td>
</tr>
<tr>
<td>32</td>
<td>27</td>
</tr>
</tbody>
</table>

Draw a histogram for the table in the space below. For each row, find which interval (or "bin") on the x-axis represents the right number of teeth. Then fill in the box so that the height of the box is equal to the sum of the counts that fit into that interval. One of the intervals has been completed for you.
Students watched 5 videos, and rated them on a scale of 1 to 10. While the average score for every video is the same (5.5), the shapes of the ratings distributions were very different! Match the summary description (left) with the shape of the histogram of student ratings (right). For each histogram, the x-axis is the score, and the y-axis is the number of students who gave it that score. These axes are intentionally unlabeled - focusing on the shape is what matters here!

1. Most of the students were fine with the video, but a couple of them gave it an unusually low rating.

2. Most of the students were okay with the video, but a couple students gave it an unusually high rating.

3. Students tended to give the video an average rating, and they weren't likely to stray far from the average.

4. Students either really liked or really disliked the video.

5. Reactions to the video were all over the place: high ratings and low ratings and inbetween ratings were all equally likely.
Choosing the Right Bin Size

Open your saved Animals Starter File, or make a new copy, and click "Run".

```python
# histogram :: ( Table, String, String, Number ) -> Image
```

Make a histogram for the "weeks" column in the animals-table, using a bin size of 10 and the "name" column for your labels.

1) How many animals took between 0 and 10 weeks to be adopted? _____________________________________________________________________________

2) How many animals took between 10 and 20 weeks to be adopted? _____________________________________________________________________________

Try some other bin sizes (be sure to experiment with bigger and smaller bins!)

3) What shape emerges? _______________________________________________________________________________________

4) What bin size gives you the best picture of the distribution? (Note: ideally your histogram should have between 5 and 10 bars) __________

5) Are there any outliers? If so, are they high or low? _______________________________________________________________________________________

6) How many animals took between 0 and 5 weeks to be adopted? _____________________________________________________________________________

7) How many animals took between 5 and 10 weeks to be adopted? _____________________________________________________________________________

8) What else do you Notice? What do you Wonder?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
Identifying Shape - Histograms

Describe the shape of the histograms on the left. Do your best to incorporate the vocabulary you've been introduced to.
## Data Cycle: Shape of the Animals Dataset

Describe two histograms made from columns of the animals dataset. The first question is given - you'll need to come up with the second question on your own!

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>What is the distribution of weight among all animals at the shelter? What question do you have?</th>
<th>Question Type (circle one):</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lookup, Arithmetic, Statistical</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th>What code will make the table or display you want?</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th>What did you find out? What can you infer?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>What new question(s) does this raise?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>What question do you have?</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th>What code will make the table or display you want?</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th>What did you find out? What can you infer?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>What new question(s) does this raise?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question Type (circle one):</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lookup, Arithmetic, Statistical</td>
</tr>
</tbody>
</table>
Outliers: Should they Stay or Should they Go?

Tahli and Fernando are looking at a scatter plot showing the relationship between poverty and test scores at schools in Michigan. They find a trend, with low-poverty schools generally having higher test scores than high-poverty schools. However, one school is an extreme outlier: the highest poverty school in the state also has higher test scores than most of the other schools!

Tahli thinks the outlier should be removed before they start analyzing, and Fernando thinks it should stay. Here are their reasons:

<table>
<thead>
<tr>
<th>Tahli’s Reasons:</th>
<th>Fernando’s Reasons:</th>
</tr>
</thead>
<tbody>
<tr>
<td>This outlier is so far from every other school - it has to be a mistake.</td>
<td>Even if it’s a magnet, those are still Michigan students and their data should not be erased. Maybe it’s not a mistake or a special school! Maybe the school has an amazing new strategy that’s different from other schools! Instead of removing an inconvenient data point from the analysis, we should be focusing our analysis on what is happening there.</td>
</tr>
<tr>
<td>Maybe someone entered the poverty level or the test scores incorrectly! We</td>
<td></td>
</tr>
<tr>
<td>don’t want those errors to influence our analysis. Or maybe it’s a magnet,</td>
<td></td>
</tr>
<tr>
<td>exam or private school that gets all the top-performing students. It’s not</td>
<td></td>
</tr>
<tr>
<td>right to compare that to non-magnet schools.</td>
<td></td>
</tr>
</tbody>
</table>

1) Do you think this outlier should stay or go? Why?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

2) What information would help you make your decision?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
Data Cycle: Shape of My Dataset

Use the Data Cycle to explore the distribution of one or more quantitative columns using histograms, and write down your findings.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one):</th>
</tr>
</thead>
<tbody>
<tr>
<td>What question do you have?</td>
<td>Lookup, Arithmetic, Statistical</td>
</tr>
</tbody>
</table>

| Consider Data                                      |                             |
| Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.) |                             |
| What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.) |                             |

| Analyze Data                                       |                             |
| What code will make the table or display you want? |                             |

| Interpret Data                                      |                             |
| What did you find out? What can you infer?         |                             |
| What new question(s) does this raise?              |                             |
Measures of Center and Spread

There are three ways to measure the center of a dataset, to summarize a whole column of quantitative data using just one number:

- The **mean** of a dataset is the average of all the numbers.

- The **median** of a dataset is a value that is smaller than half the dataset, and larger than the other half. In an ordered list the median will either be the middle number or the average of the two middle numbers.

- The **mode(s)** of a dataset is the value (or values) occurring most often. When all of the values occur equally often, a dataset has no mode.

In a symmetric dataset, values are just as likely to occur a certain distance above the mean as below the mean, and the median and mean are usually close together.

When a dataset is asymmetric, the median is a more descriptive measure of center than the median.

- A dataset with **left skew** has a few values that are unusually low, which pull the mean **below** the median.

- A dataset with **right skew** has a few values that are unusually high, which pull the mean **above** the median.

When a dataset contains a small number of values, the mode may be the most descriptive measure of center.

Data Scientists can also measure the **spread** of a dataset using a **five-number summary**:

- The **minimum** – the lowest value in the dataset

- The **first, or “lower” quartile (Q1)** – the middle of the lower half of values, which separates the lowest quarter from the next smallest quarter

- The **second quartile (Q2)** – the middle value, which separates the entire dataset into “top” and “bottom” halves

- The **third, or “upper” quartile (Q3)** – the middle of the higher half of values which separates the second highest quarter from the highest quarter

- The **maximum** – the largest value in the dataset
The five-number summary can be used to draw a box plot.

- Each of the four sections of the box plot contains 25% of the data. If the values are distributed evenly across the range, the four sections of the box plot will be equal in width. Uneven distributions will show up as differently-sized sections of a box plot.

- The left whisker extends from the minimum to Q1.

- The box, or interquartile range, extends from Q1 to Q3. It is divided into 2 parts by the median. Each of those parts contains 25% of the data, so the whole box contains the central 50% of the data.

- The right whisker extends from Q3 to the maximum.

The box plot above, for example, tells us that:

- The minimum weight is about 165 pounds. The median weight is about 220 pounds. The maximum weight is about 310 pounds.
  - 1/4 of the players weigh roughly between 165 and 195 pounds
  - 1/4 of the players weigh roughly between 195 and 220 pounds
  - 1/4 of the players weigh roughly between 220 and 235 pounds
  - 1/4 of the players weigh roughly between 235 and 310 pounds
  - 50% of the players weigh roughly between 165 and 220 pounds
  - 50% of the players weigh roughly between 195 and 235 pounds
  - 50% of the players weigh roughly between 220 and 310 pounds
- The densest concentration of players’ weights is between 220 and 235 pounds.
- Because the widest section of the box plot is between 235 and 310 pounds, we understand that the weights of the heaviest 25% fall across a wider span than the others. 310 may be an outlier, the weights of the players weighing between 235 pounds and 310 pound could be evenly distributed across the range, or all of the players weighing over 235 pounds may weigh around 310 pounds.
Summarizing Columns in the Animals Dataset

Find the measures of center and spread to summarize the ___ pounds ___ column of the Animals Starter File.

If you're using a printed workbook, be sure to add examples to your Contracts pages in the back as you work.

Measures of Center

The three measures of center for this column are:

<table>
<thead>
<tr>
<th>Mean (Average)</th>
<th>Median</th>
<th>Mode(s)</th>
</tr>
</thead>
</table>

Since the mean is __________ compared to the median, this suggests the shape is ____________

- higher/lower/about equal
- skewed right (or high outliers) / skewed left (or low outliers) / symmetric

Measures of Spread

My five-number summary is:

<table>
<thead>
<tr>
<th>Minimum</th>
<th>Q1</th>
<th>Median</th>
<th>Q3</th>
<th>Maximum</th>
</tr>
</thead>
</table>

Displaying Center and Spread with a Box Plot

Draw a box plot from this summary on the number line below. Be sure to label the number line with consistent intervals.

From this summary and box plot, I conclude:

_____________________________________________________________________________________________________________________________________________
Complete both of the Data Cycles shown here, which have questions defined to get you started.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the mean age for cats at the shelter?</td>
<td></td>
</tr>
<tr>
<td>What question do you have?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
<td></td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>What did you find out? What can you infer?</td>
<td></td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the median time it takes for an animal to be adopted?</td>
<td></td>
</tr>
<tr>
<td>What question do you have?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
<td></td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>What did you find out? What can you infer?</td>
<td></td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
<td></td>
</tr>
</tbody>
</table>

56
Consider the following dataset, representing the heaviest bench press (in lbs) for ten powerlifters:
135, 95, 230, 135, 203, 55, 1075, 135, 110, 185

1) In the space below, rewrite this dataset in sorted order.

2) In the table below, compute the measures of center for this dataset.

<table>
<thead>
<tr>
<th>Mean (Average)</th>
<th>Median</th>
<th>Mode(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3) The following statements are correct ... but misleading. Write down the reason why.

<table>
<thead>
<tr>
<th>Statement</th>
<th>Why it's misleading</th>
</tr>
</thead>
<tbody>
<tr>
<td>“More personal records are set at 135 lbs than any other weight!”</td>
<td></td>
</tr>
<tr>
<td>“The average powerlifter can bench press 235 lbs.”</td>
<td></td>
</tr>
<tr>
<td>“With a median of 135, that means that half the people in this group can’t even lift 135 lbs.”</td>
<td></td>
</tr>
</tbody>
</table>
Describe the shape of the box plots on the left. Do your best to incorporate the vocabulary you’ve been introduced to.
Matching Box Plots to Histograms

Students watched 5 videos, and rated them on a scale of 1 to 10. For each video, their ratings were used to generate box plots and histograms. Match the box plot to the histogram that displays the same data.
Directions: Connect each item on this page to at least one other item by drawing an arrow and writing an explanation of how they are connected along the arrow. (Arrows may curve.)

<table>
<thead>
<tr>
<th>Minimum</th>
<th>25%</th>
<th>50%</th>
<th>Median</th>
<th>Upper Quartile</th>
<th>Interquartile Range</th>
<th>Lower Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>25%</td>
<td>50%</td>
<td>50%</td>
<td>Median</td>
<td>Upper Quartile</td>
<td>Interquartile Range</td>
<td>Lower Quartile</td>
<td>Maximum</td>
</tr>
</tbody>
</table>
Data Cycle: Shape of My Dataset

Use the Data Cycle to explore the distribution of one or more quantitative columns using histograms and box plots, and write down your findings.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Consider Data</th>
<th>Analyze Data</th>
<th>Interpret Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>What question do you have?</td>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
<td>What code will make the table or display you want?</td>
<td>What did you find out? What can you infer?</td>
</tr>
<tr>
<td>Question Type (circle one):</td>
<td></td>
<td></td>
<td>What new question(s) does this raise?</td>
</tr>
<tr>
<td>Lookup</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arithmetic</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Statistical</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Match each box-plot to the histogram that displays the same data.

1

2

3

4

5
Computing Standard Deviation

The histogram below shows the ages of ten cats at the shelter:

1) Describe the shape of this histogram.

2) How many cats are 1 year old? 2 years old? Fill in the table below by reading the histogram. The first column has been done for you.

<table>
<thead>
<tr>
<th>age</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3) What is the mean age of the cats in this histogram?

4) Draw a star to locate the mean on the x-axis of the histogram above.

5) For each cat in the histogram above, draw a horizontal arrow under the axis from your star to the cat's interval, and label the arrow with its distance from the mean. (For example, if the mean is 3 and a cat is in the 1yr interval, your arrow would stretch from 1 to 3, and be labeled with the distance “2”)

6) We want to summarize all distances (N=10) from the mean into a single number. What number do you think will best summarize those distances?
   To compute the standard deviation we square each distance and take the average, then take the square root of the average.

7) We've recorded the ages (N=10) shown in the histogram above in the table below and listed the distance-from-mean for the four 1-year-old cats for you. As you can see, 1 year-olds are 2 years away from the mean, so their squared distance is 4. Complete the table.

<table>
<thead>
<tr>
<th>age of cat</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>3</th>
<th>5</th>
<th>7</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>distance from mean</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>squared distance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8) Add all the squared distances. What is their sum?

9) There are N=10 distances. What is N-1? Divide the sum by N-1. What do you get?

10) Take the square root to find the **standard deviation**!
The Effect of an Outlier

The histogram below shows the ages of eleven cats at the shelter:

1) Describe the shape of this histogram. 

2) How many cats are 1 year old? 2 years old? Fill in the table below by reading the histogram. The first column has been done for you.

<table>
<thead>
<tr>
<th>age</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3) What is the mean age of the cats in this histogram? 

4) Draw a star to identify the mean on the histogram above.

5) For each cat in the histogram above, draw a horizontal arrow from the mean to the cat’s interval, and label the arrow with its distance from the mean. (For example, if the mean is 2 and a cat is 5 years old, your arrow would stretch from 2 to 5, and be labeled with the distance “3”)

To compute the standard deviation we square each distance and take the average, then take the square root of the average.

6) Recorded the 11 ages shown in the histogram in the first row of the table below. For each age, compute the distance from the mean and the squared distance.

<table>
<thead>
<tr>
<th>age of cat</th>
<th>distance from mean</th>
<th>squared distance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7) Add all the squared distances. What is their sum? ____

8) Divide the sum by N-1. What do you get? 

9) Take the square root to find the standard deviation! 

10) How did the outlier impact the standard deviation?
The mean time-to-adoption is 5.75 weeks. Does that mean most animals generally get adopted in 4-6 weeks? Use the Data Cycle to find out. Write your findings on the lines below, in response to the question.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>What question do you have?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do the animals all get adopted in around the same length of time?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>What did you find out? What can you infer?</td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
</tr>
</tbody>
</table>

Turn the Data Cycle above into a Data Story, which answers the question “If the average adoption time is 5.75 weeks, do all the animals get adopted in roughly 4-6 weeks?”
**Data Cycle: Standard Deviation in My Dataset**

What questions do you have about the spread of your dataset? Use the Data Cycle to find the standard deviation in two distributions, and write down your thinking and findings.

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>What question do you have?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
<td></td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>What did you find out? What can you infer?</td>
<td></td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ask Questions</th>
<th>Question Type (circle one): Lookup Arithmetic Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>What question do you have?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consider Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
<td></td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Analyze Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
<td></td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interpret Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>What did you find out? What can you infer?</td>
<td></td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
<td></td>
</tr>
</tbody>
</table>
Computing Standard Deviation (2)

Shown below are ten different family incomes:
$43k, $62k, $39k, $141k, $58k, $82k, $41k, $73k, $68, $73k

1) Draw a **number line** representing the distribution of this data.

2) What is the mean income of the families in this dataset?

3) **Draw a star to locate the mean on the number line above.**

4) For each family on the number line you drew, **draw a horizontal arrow** under the axis from your star to that family's income, and **label the arrow with its distance from the mean**. (For example, if the mean is $50k and a family's income is $82k, your arrow would stretch from $50k to $82k, and be labeled with the distance "$32k")

To compute the standard deviation we square each distance and take the average, then take the square root of the average.

5) For each of the 10 incomes in the table below, list the distance-from-mean for each income, using the mean you computed above. Then fill in the squared distance in the next row.

<table>
<thead>
<tr>
<th>income (in 10s of thousand)</th>
<th>39</th>
<th>41</th>
<th>43</th>
<th>58</th>
<th>62</th>
<th>68</th>
<th>73</th>
<th>73</th>
<th>82</th>
<th>141</th>
</tr>
</thead>
<tbody>
<tr>
<td>distance from mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>squared distance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6) Add all the squared distances. What is their sum?

7) There are N=10 distances. What is N-1? Divide the sum by N-1. What do you get?

8) Take the square root to find the **standard deviation**!
## Matching Mean & Standard Deviation to Data

In the table below, match the mean and standard deviation to the list of data it describes.

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>StDev</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>0</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-1, -2, -3, -4, -5, -6, -7</td>
</tr>
<tr>
<td>2</td>
<td>-5</td>
<td>-5.66</td>
<td>B</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1, 2, 3, 4, 5, 6, 7</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>-2.16</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-1, -9</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>-2.65</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0, 2, 3, 4, 5, 6, 8</td>
</tr>
<tr>
<td>5</td>
<td>-4</td>
<td>-2.16</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4, 4, 4, 4, 4</td>
</tr>
</tbody>
</table>
Correlations in Scatter Plots

Scatter Plots can be used to show a relationship between two quantitative columns. Each row in the dataset is represented by a point, with one column providing the x-value and the other providing the y-value. The resulting "point cloud" makes it possible to look for a relationship between those two columns.

- If the points in a scatter plot appear to follow a straight line, it suggests that a linear relationship exists between those two columns. A number called a correlation can be used to summarize this relationship.

- \( r \) is the name of the correlation statistic. The \( r \)-value will always fall between −1 and +1. The sign tells us whether the correlation is positive or negative. Distance from 0 tells us the strength of the correlation.
  - −1 is the strongest possible negative correlation.
  - +1 is the strongest possible positive correlation.
  - 0 means no correlation.
  - ±0.65 or ±0.70 or more is typically considered a "strong correlation".
  - ±0.35 and ±0.65 is typically considered "moderately correlated".
  - Anything less than about ±0.25 or ±0.35 may be considered weak.
  - However, these cutoffs are not an exact science! In some contexts an \( r \)-value of ±0.50 might be considered impressively strong!

- The correlation is positive if the point cloud slopes up as it goes farther to the right. This means larger y-values tend to go with larger x-values. The correlation is negative if the point cloud slopes down as it goes farther to the right.

- It is a strong correlation if the points are tightly clustered around a line. In this case, knowing the x-value gives us a pretty good idea of the y-value. It is a weak correlation if the points are loosely scattered and the y-value doesn't depend much on the x-value.

- Points that do not fit the trend line in a scatter plot are called unusual observations.

- We graphically summarize this relationship by drawing a straight line through the data cloud, so that the vertical distance between the line and all the points taken together is as small as possible. This line is called the line of best fit and allows us to predict y-values based on x-values.

- Correlation is not causation! Correlation only suggests that two column variables are related, but does not tell us if one causes the other. For example, hot days are correlated with people running their air conditioners, but air conditioners do not cause hot days!
Creating a Scatter Plot

1) The table below has some new animals! Choose one and plot their age/weeks values by adding a dot to the scatter plot on the right (be sure to check your axes!). Then write their name next to your dot.

<table>
<thead>
<tr>
<th>name</th>
<th>species</th>
<th>age</th>
<th>weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Alice&quot;</td>
<td>&quot;cat&quot;</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>&quot;Bob&quot;</td>
<td>&quot;dog&quot;</td>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>&quot;Callie&quot;</td>
<td>&quot;cat&quot;</td>
<td>16</td>
<td>4</td>
</tr>
<tr>
<td>&quot;Diver&quot;</td>
<td>&quot;lizard&quot;</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>&quot;Eddie&quot;</td>
<td>&quot;dog&quot;</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>&quot;Fuzzy&quot;</td>
<td>&quot;cat&quot;</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>&quot;Gary&quot;</td>
<td>&quot;rabbit&quot;</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>&quot;Hazel&quot;</td>
<td>&quot;dog&quot;</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

2) Plot the rest of the animals - one at a time - labeling each point as you go. After each animal, ask yourself whether or not you see a pattern in the data.

3) After how many animals did you begin to see a pattern?

4) Use a straight edge to draw a line on the graph that best represents the pattern you see, then circle the cloud of points around that line.

5) Are the points tightly clustered around the line or loosely scattered?

6) Does this display support the claim that younger animals get adopted faster? Why or why not?

7) Place points on the graph to create a scatter plot with NO relationship.
## Data Cycle: Relationships in the Animals Dataset

Use the Data Cycle to search for relationships! The first cycle has a question to get you started. What question will you ask for the second?

| Ask Questions |  | Question Type (circle one): |
|---------------|  | ____________________________ |
| Is there a relationship between weight and adoption time? |  | Lookup Arithmetic Statistical |
| What question do you have? |  | ____________________________ |

| Consider Data |  |  |
|---------------|  | ____________________________ |
| Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.) |  | ____________________________ |
| What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.) |  | ____________________________ |

| Analyze Data |  |  |
|--------------|  | ____________________________ |
| What code will make the table or display you want? |  | ____________________________ |

| Interpret Data |  |  |
|----------------|  | ____________________________ |
| What did you find out? What can you infer? |  | ____________________________ |
| What new question(s) does this raise? |  | ____________________________ |

| Ask Questions |  | Question Type (circle one): |
|---------------|  | ____________________________ |
| What question do you have? |  | Lookup Arithmetic Statistical |

| Consider Data |  |  |
|---------------|  | ____________________________ |
| Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.) |  | ____________________________ |
| What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.) |  | ____________________________ |

| Analyze Data |  |  |
|--------------|  | ____________________________ |
| What code will make the table or display you want? |  | ____________________________ |

| Interpret Data |  |  |
|----------------|  | ____________________________ |
| What did you find out? What can you infer? |  | ____________________________ |
| What new question(s) does this raise? |  | ____________________________ |
Data Cycle: Relationships in Your Dataset

Use the Data Cycle to search for relationships between columns in your dataset!

**Ask Questions**
- What question do you have?
- Question Type (circle one):
  - Lookup
  - Arithmetic
  - Statistical

**Consider Data**
- Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)
- What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)

**Analyze Data**
- What code will make the table or display you want?

**Interpret Data**
- What did you find out? What can you infer?
- What new question(s) does this raise?
Identifying Form, Direction and Strength

What do your eyes tell you about the Form, Direction, & Strength of these displays?

Note: If the form is nonlinear, we shouldn't report direction - a curve may rise and then fall.

A
Form: Linear
Direction: Positive
Strength: Strong

B
Form: Linear
Direction: Positive
Strength: Strong

C
Form: Linear
Direction: Positive
Strength: Strong

D
Form: Linear
Direction: Positive
Strength: Strong

E
Form: Linear
Direction: Positive
Strength: Strong

F
Form: Linear
Direction: Positive
Strength: Strong
1) What has to be true about the shape of a relationship in order to start talking about a correlation?

________________________________________________________________________________________

2) What is the difference between a weak relationship and a negative relationship?

________________________________________________________________________________________

3) What is the difference between a strong relationship and a positive relationship?

________________________________________________________________________________________

4) If we find a strong relationship in a sample from a larger population, will that relationship always hold for the whole population? Why or why not?

________________________________________________________________________________________

________________________________________________________________________________________

5) If two correlations are both positive, the stronger one is more positive (steeper slope) than the other

________________________________________________________________________________________

________________________________________________________________________________________

6) A news report claims that after surveying 10 million people, a positive correlation was found between how much chocolate a person eats and how happy they are. Does this mean eating chocolate almost certainly makes you happier? Why or why not?

________________________________________________________________________________________

________________________________________________________________________________________
Identifying Form and $r$-Values

What do your eyes tell you about the Form and Direction of the data? If the form is linear, approximate the $r$-value.

Reminder:

- $-1$ is the strongest possible negative correlation, and $+1$ is the strongest possible positive correlation
- 0 means no correlation
- ±0.65 or ±0.70 or more is typically considered a "strong correlation"
- ±0.35 and ±0.65 is typically considered "moderately correlated"
- Anything less than about ±0.25 or ±0.35 may be considered weak

Form: $r$ close to:

A

B

C

D

E

F
Correlation Does Not Imply Causation!

Here are some possible correlations and the nonsense headlines a confused journalist might report as a result. In reality, the correlations have absolutely no causal relationship; they come about because both of them are related to another variable that's lurking in the background.

Can you think of another variable for each situation that might be the actual cause of the correlation and explain why the headlines the paper ran based on the correlations are nonsense?

1) **Correlation:** For a certain psychology test, the amount of time a student studied was negatively correlated with their score!
   **Headline:** "Students who study less do better!"

2) **Correlation:** Weekly data gathered in a city throughout the year showed a positive correlation between ice cream consumption and drowning deaths.
   **Headline:** "Drowning Deaths are Good for Ice Cream Sales!"

3) **Correlation:** A negative correlation was found between how much time students talked on the phone and how much they weighed.
   **Headline:** "Students Lose Weight by Talking on the Phone!"
Correlations in the Animals Dataset

1) In the Interactions Area, create a scatter plot for the Animals Starter File, using "pounds" as the xs and "weeks" as the ys.
   - Form: Does the point cloud appear linear or nonlinear?  
   - Direction: If it's linear, does it appear to go up or down as you move from left to right?  
   - Strength: Is the point cloud tightly packed, or loosely dispersed?  
   - Would you predict that the r-value is positive or negative?  
   - Will it be closer to zero, closer to ±1, or in between?  
   - What r-value does Pyret compute when you type r-value(animals-table, "pounds", "weeks")?  
   - Does this match your predictions?  

2) In the Interactions Area, create a scatter plot for the Animals Dataset, using "age" as the xs and "weeks" as the ys.
   - Form: Does the point cloud appear linear or nonlinear?  
   - Direction: If it's linear, does it appear to go up or down as you move from left to right?  
   - Strength: Is the point cloud tightly packed, or loosely dispersed?  
   - Would you predict that the r-value is positive or negative?  
   - Will it be closer to zero, closer to ±1, or in between?  
   - What r-value does Pyret compute?  
   - Does this match your prediction?  

3) Is this correlation stronger or weaker than the correlation for "pounds"?  

4) What does that mean?  


Correlations in My Dataset

1) There may be a correlation between ___________________________ and ___________________________.
I think it is a ___________________________ correlation, because ___________________________.
It might be stronger if I looked at ___________________________.

2) There may be a correlation between ___________________________ and ___________________________.
I think it is a ___________________________ correlation, because ___________________________.
It might be stronger if I looked at ___________________________.

3) There may be a correlation between ___________________________ and ___________________________.
I think it is a ___________________________ correlation, because ___________________________.
It might be stronger if I looked at ___________________________.

4) There may be a correlation between ___________________________ and ___________________________.
I think it is a ___________________________ correlation, because ___________________________.
It might be stronger if I looked at ___________________________.

78
Identifying Form, Direction and Strength (Matching)

Match the description (left) with the scatter plot (right).

Note: The computer won’t tell us if the relationship we see in a scatter plot is linear, so we must train our eyes to decide this ourselves. For linear relationships, we should train our eyes to assess their direction and get a feel for their strength, rather than relying completely on what numbers the computer reports.

The relationship appears to be linear, negative, and of moderate strength. 1 A

This relationship is nonlinear. 2 B

The x and y variables in this dataset do not appear to be related. 3 C

The relationship appears to be linear, positive, and strong. 4 D

The relationship appears to be linear, negative, and strong. 5 E
Linear Regression

- We compute **linear relationships to predict the future**! Well...sort of. Given a dataset, like ages of animals v. how long before they're adopted, we try to compute the relationship between **age** and **weeks** so that we can **predict** how long a new animal might stay, based on their age.

- When we compute linear relationships, we're talking about **straight-line patterns** that appear on a scatter plot.

- A scatter plot has an **x**- and **y**-axis, which get special names when looking for relationships. The **y**-axis is called the **response variable**, and the **x**-axis is called the **explanatory variable**. In our example, we are trying to figure out how much of the **weeks** variable is **explained by** the **age** variable.

- **Linear Regression** is a way of computing the **line of best fit**, which tries to draw a line as close as possible to all the points. (Want details? It minimizes the **sum of the squares** of the vertical distances from the points to the line. There's a reason we use computers to do this!)

- **Slope** is how much we predict the **response variable** will increase or decrease for each unit that the **explanatory variable** increases. In our example, a slope of 0.5 would mean "we predict that each additional year of age means an extra half-week in the shelter". (What would a slope of 3 mean?)

- **Sample size matters**! The number of data values is also relevant. We'd be more convinced of a positive relationship in general between cat age and time to adoption if a correlation of +0.57 were based on 50 cats instead of 5.
Introduction to Linear Regression

**How much can one point move the line of best fit?**

Open the [Interactive Regression Line (Geogebra)](https://www.geogebra.org). Move the blue point “P”, and see what effect it has on the red line.

1) Move P so that it is **centered amongst** the other points. Now move it all the way to top and bottom of the screen.

2) Move P so that it is **far to the left or right** of the other points. Now move it all the way to top and bottom of the screen. How - if at all - does the x-position of P impact on the line of best fit?

3) Could the regression line ever be above or below **all** the points **including the blue one you're dragging**? Why or why not?

4) Would it be possible to have a line with more points on one side than the other? Why or why not?

5) What is the highest \( r \)-value you can get? Where did you place P? (______, ______)

6) What function describes the regression line with this value of P? \( y = \ldots \) x + ______

7) What is the lowest \( r \)-value you can get? Where did you place P? (______, ______)

8) What function describes the regression line with this value of P? \( y = \ldots \) x + ______

**Predictions from Scatter Plots**

9) Draw the line of best fit for age-v-weeks (on the left). Is this a strong correlation that will allow us to make a good prediction of an animal’s adoption time just by knowing how old it is?

10) Draw the line of best fit for pounds-v-weeks (on the right). Is this a strong correlation that will allow us to make a good prediction of an animal’s adoption time just by knowing how heavy it is?

11) Do either or both of the relationships appear to be linear?
Drawing Predictors

Remember what we learned about r-values...

<table>
<thead>
<tr>
<th>$r$</th>
<th>perfect negative correlation</th>
<th>moderate negative association</th>
<th>no correlation</th>
<th>moderate positive association</th>
<th>perfect positive correlation</th>
</tr>
</thead>
</table>

For each of the scatter plots below, draw a predictor line that seems like the best fit. Describe the correlation in terms of Direction and whether you think it is generally stronger or weaker, then estimate the $r$-value as being close to -1, -0.5, 0, +0.5, or +1.

### A
- **Direction:** Positive
- **Strength:** Stronger
- **r:**

### B
- **Direction:** Positive
- **Strength:** Stronger
- **r:**

### C
- **Direction:** Positive
- **Strength:** Stronger
- **r:**

### D
- **Direction:** Positive
- **Strength:** Stronger
- **r:**
Exploring `lr-plot`

**Part 1:** You should already have plotted `lr-plot(animals-table, "name", "age", "weeks")` in the Animals Starter File.

1) What is the predictor function? \(y = \ldots\) 

2) What is the slope? 

3) What is the y-intercept? 

4) How long would our line of best fit predict it would take for a 5 year-old animal to be adopted? 

5) What if they were a newborn, or just 0 years old? 

6) Does it make sense to find the adoption time for a newborn using this predictor function? Why or why not?

**Part 2:** Make another `lr-plot`, but this time use the animals' weight as our explanatory variable instead of their age.

7) How long would our line of best fit predict it would take for an animal weighing 21 pounds to be adopted? 

8) What if they weighed 0.1 pounds? 

**Part 3:** `lr-plot` with filtered tables

9) Make another `lr-plot`, comparing the `age` v. `weeks` columns for only the cats using the following code:

```r
fun is-cat(r): r["species"] == "cat" end
lr-plot(filter(animals-table, is-cat), "name", "age", "weeks")
```

10) What is the predictor function? \(y = \ldots\)

11) What is the slope? 

12) What is the y-intercept? 

13) How does this line of best fit for cats compare to the line of best fit for all animals? 

14) How long would our line of best fit predict it would take for a 5 year-old cat to be adopted? 

★ *Make another `lr-plot`, comparing the `age` v. `weeks` columns for only the dogs.*
1) About how many inches are kids in this dataset expected to grow per year? 

2) At that rate, if a child were 45” tall at age eight, how tall would you expect them to be at age twelve? 

3) At that rate, if a ten-year-old were 55” tall, how tall would you expect them to have been at age 9? 

4) Using the equation, how tall would you expect a seven-year-old child to be? 

5) How many of the seven-year-olds in this sample are actually that height? 

6) Using the equation, determine the expected height of someone who is...

<table>
<thead>
<tr>
<th></th>
<th>7.5 years old</th>
<th>13 years old</th>
<th>6 years old</th>
<th>newborn</th>
<th>90 years old</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7) For which ages is this predictor function likely to be the most accurate? Why? 

8) For which ages is this predictor function likely to be the least accurate? Why?
### Interpreting Regression Lines & r-Values

Each description on the left is written about the linear regression findings on the right. Fill in the blanks using the information in the line of best fit and the r-value.

<table>
<thead>
<tr>
<th></th>
<th>Interpreting Regression Lines &amp; r-Values</th>
</tr>
</thead>
</table>
| 1 | For every additional Marvel Universe movie released each year, the average person is predicted to consume __________ pounds of sugar! This correlation is __________.  
   | sugar(m) = -3.19m + 12  
   | r = -0.05  
| 2 | Shoe size and height are __________ correlated. If person A is one size bigger than person B, we predict that they will be roughly __________ inches taller than person B as well.  
   | height(s) = 1.65s + 52  
   | r = 0.89  
| 3 | There is __________ relationship found between the number of Uber drivers in a city and the number of babies born each year.  
   | babies(u) = 0.012u + 7.8  
   | r = 0.01  
| 4 | The correlation between weeks-of-school-missed and SAT score is __________ and __________. For every week a student misses, we predict a __________ point __________ in their SAT score.  
   | score(w) = -15.3w + 1150  
   | r = -0.65  
| 5 | There is a __________ correlation between the number of streaming video services someone has, and how much they weigh. For each service, we expect them to be roughly __________ pounds heavier.  
   | weight(n) = 1.6n + 160  
   | r = 0.12 |
# Data Cycle: Regression Analysis (Example)

## Ask Questions

**How big of a factor is age in determining adoption time?**

- What question do you have?

## Consider Data

- **all animals at the shelter**
  - Which rows should we investigate? [All the rows, just the cats, fixed dogs, etc.]
  - **name, age, and weeks**
    - What columns do we need? (age, weight-in-kilograms, weeks, etc.)

## Analyze Data

- If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)  
- If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)

\[
lr-plot (\text{animals-table}, \ "name", \ "age", \ "weeks")
\]

- What code will make the table or display you want?

## Interpret Data

- The slope is positive (0.789) and \( R = 0.448 \), so there is a moderate, positive relationship

- What did you find out? What can you infer?

- Is weight a bigger factor?

- What new question(s) does this raise?

---

This was a statistical question.

---

**Write your Data Story below:**

I performed a linear regression on a sample of [dataset or subset] and found a [moderate / strong / weak] correlation between [x-axis, units] and [y-axis, units]. I would predict that a 1 [x-axis units] increase in [x-axis] is associated with a [slope, y-units] increase / decrease in time to adoption [y-axis].
Describing Relationships

A small sample of people were surveyed about their coffee drinking and sleeping habits. Does drinking coffee impact one’s amount of sleep? NOTE: this data is made up for instructional purposes!

<table>
<thead>
<tr>
<th>Daily Cups of Coffee</th>
<th>Sleep (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>400</td>
</tr>
<tr>
<td>0</td>
<td>480</td>
</tr>
<tr>
<td>8</td>
<td>310</td>
</tr>
<tr>
<td>1</td>
<td>300</td>
</tr>
<tr>
<td>1</td>
<td>390</td>
</tr>
<tr>
<td>2</td>
<td>360</td>
</tr>
<tr>
<td>1</td>
<td>410</td>
</tr>
<tr>
<td>0</td>
<td>500</td>
</tr>
<tr>
<td>2</td>
<td>390</td>
</tr>
<tr>
<td>1</td>
<td>480</td>
</tr>
<tr>
<td>3</td>
<td>360</td>
</tr>
<tr>
<td>4</td>
<td>430</td>
</tr>
<tr>
<td>0</td>
<td>450</td>
</tr>
<tr>
<td>5</td>
<td>240</td>
</tr>
<tr>
<td>1</td>
<td>420</td>
</tr>
<tr>
<td>2</td>
<td>380</td>
</tr>
<tr>
<td>1</td>
<td>480</td>
</tr>
</tbody>
</table>

1) Describe the relationship between coffee intake and minutes of sleep shown in the data above.

2) Why is the y-axis of the display above misleading?
Ask a question about your data to tell your Data Story.

**Ask Questions**

What question do you have?

**Consider Data**

Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)

What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)

**Analyze Data**

If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)

If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)

What code will make the table or display you want?

**Interpret Data**

What did you find out? What can you infer?

What new question(s) does this raise?

Write your Data Story below:

I performed a linear regression on a sample of ___________________________ and found ______ correlation between ______ and ______. I would predict that a 1 ______ of ______ is associated with a ______ increase / decrease in ______.
Open the [Age vs. Height Starter File](#) and click 'Run' to interact with data from another sample of students.

1) Take a look at the code in the Definitions Area. What do you notice? What do you wonder?

2) Build `image-scatter-plot(h-table, "age", "height", dot)`. Try to visualize the line of best fit for just the blue dots. Then try to visualize the line of best fit for just the red stars. How do you think they would compare? Which line do you think would be steeper?

3) Make three linear regression plots comparing `age` and `height`, and record the results for each in the table below:
   - The whole population: `lr-plot(h-table, "gender-id", "age", "height")`
   - Females only: `lr-plot(filter(h-table, is-f), "gender-id", "age", "height")`
   - Males only: `lr-plot(filter(h-table, is-m), "gender-id", "age", "height")`

<table>
<thead>
<tr>
<th>Sample</th>
<th>rate of change</th>
<th>y-intercept</th>
<th>R value</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Females</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Males</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4) What makes it difficult to compare these plots visually?

Rebuild `lr-plot(filter(h-table, is-f), "gender-id", "age", "height")`, adjust the window of the interactive plot using the numbers in the table below, and click Redraw.

<table>
<thead>
<tr>
<th>x-min:</th>
<th>x-max:</th>
<th>y-min:</th>
<th>y-max:</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.5</td>
<td>12.5</td>
<td>45</td>
<td>70</td>
</tr>
</tbody>
</table>

Then, do the same for `lr-plot(filter(h-table, is-m), "gender-id", "age", "height")`.

5) How do the plots compare now that their windows match?

6) What happens if you compare the students' height in inches to their height in centimeters by plotting `lr-plot(h-table, "gender-id", "height-cm", "height")`?
Are people more satisfied with a purchase on which they spent more money? A fictional group of people were surveyed about their most recent purchase, rating their satisfaction from 1 (very unsatisfied) to 5 (extremely satisfied).

<table>
<thead>
<tr>
<th>Dollars</th>
<th>Satisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>15.5</td>
<td>4</td>
</tr>
<tr>
<td>280</td>
<td>5</td>
</tr>
<tr>
<td>0.99</td>
<td>1</td>
</tr>
<tr>
<td>2.3</td>
<td>3</td>
</tr>
<tr>
<td>39</td>
<td>3</td>
</tr>
<tr>
<td>82</td>
<td>4</td>
</tr>
<tr>
<td>215</td>
<td>4</td>
</tr>
<tr>
<td>700</td>
<td>4</td>
</tr>
<tr>
<td>25</td>
<td>3</td>
</tr>
<tr>
<td>79</td>
<td>4</td>
</tr>
<tr>
<td>99.99</td>
<td>5</td>
</tr>
<tr>
<td>30</td>
<td>1</td>
</tr>
<tr>
<td>75</td>
<td>5</td>
</tr>
<tr>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>320</td>
<td>5</td>
</tr>
<tr>
<td>260</td>
<td>5</td>
</tr>
<tr>
<td>150</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>1</td>
</tr>
<tr>
<td>45</td>
<td>2</td>
</tr>
<tr>
<td>65</td>
<td>2</td>
</tr>
</tbody>
</table>

Describe the relationship between dollars spent and satisfaction shown in the data above.
Ask a question about your data to tell your Data Story.

**Ask Questions**

What question do you have?

---

**Question Type**

(circle one):

- Lookup
- Arithmetic
- Statistical

---

**Consider Data**

Which rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)

---

What columns do we need? (age, weight-in-kilograms, weeks, etc.)

---

**Analyze Data**

If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)

---

If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)

---

What code will make the table or display you want?

---

**Interpret Data**

What did you find out? What can you infer?

---

What new question(s) does this raise?

---

**Write your Data Story below:**

I performed a linear regression on a sample of ___ and found ___ correlation between ___ and ___.

A weak/strong/moderate (R=…), positive/negative

___ is associated with a ___ increase in ___.

___ is associated with a ___ decrease in ___.

---
Case Study: Ethics, Privacy, and Bias

These questions are designed to accompany one of the case studies provided in the Ethics, Privacy, and Bias lesson.

My Case Study is ____________________________________________________________

1) Read the case study you were assigned, and write your summary here.

__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________

2) Is this a good thing or a bad thing? Why?

__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________

3) What are the arguments on each side?  
Data Science used for this purpose is good because...

__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________

Data Science used for this purpose is bad because...

__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
Analyzing Survey Results When Data is Dirty

1) Paolo made a pie-chart of the dessert column and was suprised to discover that Fruit was the most popular dessert among 8th graders! Make the pie-chart. Why is this display misleading? How is the data “dirty”?

2) What ideas do you have for how the survey designer could have made sure that the data in the dessert column would have been cleaner?

3) Shani made a bar-chart of the gender-id column. In her analysis she stated that the most common gender identity among eighth graders in her class is male. Make the pie-chart. Do you agree? Why or Why Not?

4) Make a chart showing the ages of the 8th graders surveyed. What "dirty" data problems do you spot and how are they misleading?

5) What ideas do you have for how the survey designer could have made sure that the data in the age column would have been cleaner?
Dirty Data!

A bunch of new animals are coming to the shelter, and that means more data! Open the New Animals Dataset and take a careful look.

<table>
<thead>
<tr>
<th>What do you Notice?</th>
<th>What do you Wonder?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

There are many different ways that data can be dirty!

1. **Missing Data** - A column containing some cells with data, but some cells left blank.
2. **Inconsistent Types** - A column with inconsistent data types. For example, a `years` column where almost every cell is a Number, but one cell contains the string "5 years old".
3. **Inconsistent Units** - A column with consistent data types, but inconsistent units. For example, a `weight` column where some entries are in pounds but others are in kilograms.
4. **Inconsistent Naming** - Inconsistent spelling and capitalization for entries lead to them being counted as different. For example, a `species` column where some entries are "cat" and others are "Cat" will not give us a full picture of the cats.

1) Which animals' row(s) have **missing data**? __________________________

2) Which column(s) have **inconsistent types**? __________________________

3) Which column(s) have **inconsistent units**? __________________________

4) Which column(s) have **inconsistent naming**? __________________________

5) If we want to analyze this data, what should we do with the rows for Tanner, Toni, and Lizzy? ____________________________________________

6) If we want to analyze this data, what should we do with the rows for Chanel and Bibbles? ____________________________________________

7) If we want to analyze this data, what should we do with the rows for Porche and Boss? ____________________________________________

8) If we want to analyze this data, what should we do with the row for Niko? ____________________________________________

9) If we want to analyze this data, what should we do with rows for Mona, Rover, Susie Q, and Happy? ____________________________________________

10) Sometimes data cleaning is straightforward. Sometimes the problem is evident but the solution is less certain. For which questions were you certain of your data cleaning suggestion? For which were you less certain? Why?

__________________________

94
The **Height vs. Wingspan Survey (make a copy)** has lots of problems, which can lead to many kinds of dirty data: Missing Data, Inconsistent Types, Inconsistent Units and Inconsistent Language! Try filling it out with bad data. Record the problems and make some recommendations for how to improve the survey!

**NOTE:** make sure your teachers gives you a link to their *own copy* of the survey. You should be able to the access the data after you submit the form!

<table>
<thead>
<tr>
<th>Q</th>
<th>What examples of bad data were you able to submit?</th>
<th>How could the survey be improved to avoid bad data?</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Looking up Rows and Columns

We can define names for values in Pyret, the same way we do in math:

```pyret
ame = "Flannery"
age = 16
logo = star(50, "solid", "red")
```

When looking up a data Row from a Table, programmers use the `row-n` function. This function takes a Table and a Number as its inputs. The numbers tell the computer which Row we want from the Table. Note: Rows are numbered starting at zero!

For example:

```pyret
sasha = row-n(animals-table, 0)  # define Sasha to be the first row
mittens = row-n(animals-table, 2)  # define Mittens to be the third row
```

When we define these rows, it's more useful to name them based on their properties, rather than their identifiers:

```pyret
cat-row = row-n(animals-table, 0)  # Sasha is a cat
dog-row = row-n(animals-table, 10)  # Toggle is a dog
```

When looking up a column from a Row, programmers use square brackets and the name of the column they want.

For example:

```pyret
# these two lines do the same thing! We can use the defined name to simplify our code
row-n(animals-table, 0)["age"]  # look up Sasha's age (in row 0)
cat-row["species"]  # look up Sasha's age (using the defined name)
dog-row["age"]  # look up Toggle's age (using the defined name)
```
# Lookup Questions

The table below represents four pets at an animal shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>3</td>
<td>48</td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>92</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Maple&quot;</td>
<td>&quot;female&quot;</td>
<td>3</td>
<td>51.6</td>
</tr>
</tbody>
</table>

1) Match each Lookup Question (left) to the code that will give the answer (right).

<table>
<thead>
<tr>
<th>Question</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>“How much does Maple weigh?”</td>
<td>A row-n(pets-table, 3)</td>
</tr>
<tr>
<td>“Which is the last row in the table?”</td>
<td>B row-n(pets-table, 2)</td>
</tr>
<tr>
<td>“What is Fritz’s sex?”</td>
<td>C row-n(pets-table, 1)</td>
</tr>
<tr>
<td>“What’s the third animal’s name?”</td>
<td>D row-n(pets-table, 3)</td>
</tr>
<tr>
<td>“How much does Nori weigh?”</td>
<td>E row-n(pets-table, 3)</td>
</tr>
<tr>
<td>“How old is Maple?”</td>
<td>F row-n(pets-table, 0)</td>
</tr>
<tr>
<td>“What is Toggle’s sex?”</td>
<td>G row-n(pets-table, 2)</td>
</tr>
<tr>
<td>“What is the first row in the table?”</td>
<td>H row-n(pets-table, 0)</td>
</tr>
</tbody>
</table>

2) Write the Pyret code that will produce each value on the right.

<table>
<thead>
<tr>
<th>Value</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Maple&quot;</td>
<td>row-n(pets-table, 3)[&quot;name&quot;]</td>
</tr>
<tr>
<td>&quot;male&quot;</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td></td>
</tr>
</tbody>
</table>
More Practice with Lookups

Consider the table below, and the four value definitions that follow:

<table>
<thead>
<tr>
<th>name</th>
<th>corners</th>
<th>is-round</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;triangle&quot;</td>
<td>3</td>
<td>false</td>
</tr>
<tr>
<td>&quot;square&quot;</td>
<td>4</td>
<td>false</td>
</tr>
<tr>
<td>&quot;rectangle&quot;</td>
<td>4</td>
<td>false</td>
</tr>
<tr>
<td>&quot;circle&quot;</td>
<td>0</td>
<td>true</td>
</tr>
</tbody>
</table>

shapeA = row-n(shapes-table, 0)
shapeB = row-n(shapes-table, 1)
shapeC = row-n(shapes-table, 2)
shapeD = row-n(shapes-table, 3)

1) *Match* each Pyret expression (left) to the description of what it evaluates to (right).

   a. shapeD 1 A Evaluates to 4
   b. shapeA 2 B Evaluates to the last row in the table
   c. shapeB["corners"] 3 C Evaluates to "square"
   d. shapeC["is-round"] 4 D Evaluates to true
   e. shapeB["name"] 5 E Evaluates to false
   f. shapeA["corners"] 6 F Evaluates to 3
   g. shapeD["name"] == "circle" 7 G Evaluates to the first row in the table

2) Fill in the blanks (left) with the Pyret lookup code that will produce the value (right).

   a. "rectangle"
   b. "square"
   c. 4
   d. 0
   e. true
Defining Rows

We've already given you two row definitions: cat-row and dog-row:

```
cat-row = row-n(animals-table, 0)  # Sasha is a cat
dog-row = row-n(animals-table, 10) # Toggle is a dog
```

Remember: rows start at index zero! Use this to answer the questions below. (HINT: turn to The Animals Dataset and number the data rows first, then answer the questions below.)

1) The index of a row containing a lizard is

2) The index of a row containing a rabbit is

3) The index of a row containing a fixed animal is

4) The index of a row containing a male animal

5) The index of a row containing a female animal is

6) The index of a row containing a hermaphroditic animal is

7) The index of a row containing an unfixed animal is

8) The index of a row containing a young animal (<2 years) is

9) The index of a row containing an old animal (>10 years) is

10) What code would you write to define lizard-row?

11) What code would you write to define rabbit-row?

12) What code would you write to define fixed-row?

13) What code would you write to define male-row?

14) What code would you write to define female-row?

15) What code would you write to define hermaphrodite-row?

16) What code would you write to define young-row?

17) What code would you write to define old-row?

Add this code to your Animals Starter File! You'll want these rows for later!
The Great gt domain debate!

Kermit: The domain of \( \text{gt} \) is \( \text{Number, String, String} \).

Oscar: The domain of \( \text{gt} \) is \( \text{Number} \).

Ernie: I'm not sure who's right!

In order to make a triangle, we need a size, a color and a fill style...
but all we had to tell our actor was \( \text{gt}(20) \)...and they returned \( \text{triangle}(20, "solid", "green") \).

Please help us!

1) What is the correct domain for \( \text{gt} \)?

________________________________________________________________________

2) What could you tell Ernie to help him understand how you know?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
Let's Define Some New Functions!

1) Let's define a function `rs` to generate solid red squares of whatever size we give them!
If I say `rs(5)`, what would our actor need to say?

Let's write a few more examples:

<table>
<thead>
<tr>
<th><code>rs(____)→</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>rs(____)→</code></td>
</tr>
<tr>
<td><code>rs(____)→</code></td>
</tr>
</tbody>
</table>

What changes in these examples? Name your variable(s):

Let's define our function using the variable:

`fun rs(____):_____________ end`

2) Let's define a function `bigc` to generate big solid circles of size 100 in whatever color we give them!
If I say `bigc("orange")`, what would our actor need to say?

Let's write a few more examples:

<table>
<thead>
<tr>
<th><code>bigc(____)→</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bigc(____)→</code></td>
</tr>
<tr>
<td><code>bigc(____)→</code></td>
</tr>
</tbody>
</table>

What changes in these examples? Name your variable(s):

Let's define our function using the variable:

`fun bigc(____):_____________ end`

3) Let's define a function `ps` to build a pink star of size 50, with the input determining whether it's solid or outline!
If I say `ps("outline")`, what would our actor need to say?

Write examples for all other possible inputs:

<table>
<thead>
<tr>
<th><code>ps(____)→</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ps(____)→</code></td>
</tr>
</tbody>
</table>

What changes in these examples? Name your variable(s):

Let's define our function using the variable:

`fun ps(____):_____________ end`

4) Add these new function definitions to your `gt Starter File` and test them out!
Let’s Define Some More New Functions!

1) Let’s define a function *sun* to write SUNSHINE in whatever color and size we give it!
If I say `sun(5, "blue")`, what would our actor need to say?

Let’s write a few more examples:

- `sun(___ , ___________ ) → _____________________________`
- `sun(___ , ___________ ) → _____________________________`
- `sun(___ , ___________ ) → _____________________________`

What changes in these examples? Name your variable(s): ________________________________
Let’s define our function using the variable(s):

```plaintext
fun sun(_____________, _______________):
    _________________________________ end
```

2) Let’s define a function *me* to generate your name in whatever size and color we give it!
If I say `me(18, "gold")`, what would our actor need to say?

Let’s write a few more examples:

- `me(___ , ___________ ) → _____________________________`
- `me(___ , ___________ ) → _____________________________`
- `me(___ , ___________ ) → _____________________________`

What changes in these examples? Name your variable(s): ________________________________
Let’s define our function using the variable(s):

```plaintext
fun me(_____________, _______________):
    _________________________________ end
```

3) Let’s define a function *gr* to build a solid, green rectangle of whatever height and width we give it!
If I say `gr(10, 80)`, what would our actor need to say?

Let’s write a few more examples:

- `gr(___ , ___ ) → rectangle(___ , ___ , "solid", "green")`
- `gr(___ , ___ ) → rectangle(___ , ___ , "solid", "green")`
- `gr(___ , ___ ) → rectangle(___ , ___ , "solid", "green")`

What changes in these examples? Name your variable(s): ________________________________
Let’s define our function using the variable(s):

```plaintext
fun gr(_______________, _______________):
    _________________________________ end
```

4) Add these new function definitions to your *gt Starter File* and test them out!
Describe and Define Your Own Functions!

1) Let's define a function ________________ to generate...

__________________________________________________________

If I say ________________, what would our actor need to say? ____________________________________________________________

Let's write a few more examples:

____ (____________) → ______ (__________________________)
____ (____________) → ______ (__________________________)
____ (____________) → ______ (__________________________)

What changes in these examples? Name your variable(s): ______________________________________________________________

Let's define our function using the variable.

```plaintext
fun ______ (__________) : __________ (__________________________) end
```

2) Let's define a function ________________ to generate...

__________________________________________________________

If I say ________________, what would our actor need to say? __________________________________________________________

Let's write a few more examples:

____ (____________) → ______ (__________________________)
____ (____________) → ______ (__________________________)
____ (____________) → ______ (__________________________)

What changes in these examples? Name your variable(s): ______________________________________________________________

Let's define our function using the variable.

```plaintext
fun ______ (__________) : __________ (__________________________) end
```

3) Let's define a function ________________ to generate...

__________________________________________________________

If I say ________________, what would our actor need to say? __________________________________________________________

Let's write a few more examples:

____ (____________) → ______ (__________________________)
____ (____________) → ______ (__________________________)
____ (____________) → ______ (__________________________)

What changes in these examples? Name your variable(s): ______________________________________________________________

Let's define our function using the variable.

```plaintext
fun ______ (__________) : __________ (__________________________) end
```

4) Add your new function definitions to your `gt Starter File` and test them out!
Defining Functions

Functions can be viewed in *multiple representations*. You already know one of them: *Contracts*, which specify the Name, Domain, and Range of a function. Contracts are a way of thinking of functions as a *mapping* between one set of data and another. For example, a mapping from Numbers to Strings:

```
# f :: Number -> String
```

Another way to view functions is with *Examples*. Examples are essentially input-output tables, showing what the function would do for a specific input:

In our programming language, we focus on the last two columns and write them as code:

```
examples:
  f(1) is 1 + 2
  f(2) is 2 + 2
  f(3) is 3 + 2
  f(4) is 4 + 2
end
```

Finally, we write a formal function definition ourselves. The pattern in the Examples becomes *abstract* (or "general"), replacing the inputs with *variables*. In the example below, the same definition is written in both math and code:

```
f(x) = x + 2
fun f(x): x + 2 end
```

Look for connections between these three representations!

- The function name is always the same, whether looking at the Contract, Examples, or Definition.
- The number of inputs in the Examples is always the same as the number of types in the Domain, which is always the same as the number of variables in the Definition.
- The "what the function does" pattern in the Examples is almost the same in the Definition, but with specific inputs replaced by variables.
# Matching Examples and Contracts

Match each set of examples (left) with the Contract that best describes it (right).

<table>
<thead>
<tr>
<th>Examples</th>
<th>Contract</th>
</tr>
</thead>
<tbody>
<tr>
<td>examples:</td>
<td>1</td>
</tr>
<tr>
<td>f(5) is 5 / 2</td>
<td>A # f :: Number -&gt; Number</td>
</tr>
<tr>
<td>f(9) is 9 / 2</td>
<td></td>
</tr>
<tr>
<td>f(24) is 24 / 2</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>examples:</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>f(1) is rectangle(1, 1, &quot;outline&quot;, &quot;red&quot;)</td>
<td>B # f :: String -&gt; Image</td>
</tr>
<tr>
<td>f(6) is rectangle(6, 6, &quot;outline&quot;, &quot;red&quot;)</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>examples:</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>f(&quot;pink&quot;, 5) is star(5, &quot;solid&quot;, &quot;pink&quot;)</td>
<td>C # f :: Number -&gt; Image</td>
</tr>
<tr>
<td>f(&quot;blue&quot;, 8) is star(8, &quot;solid&quot;, &quot;blue&quot;)</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>examples:</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>f(&quot;Hi!&quot;) is text(&quot;Hi!&quot;, 50, &quot;red&quot;)</td>
<td>D # f :: Number, String -&gt; Image</td>
</tr>
<tr>
<td>f(&quot;Ciao!&quot;) is text(&quot;Ciao!&quot;, 50, &quot;red&quot;)</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>examples:</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>f(5, &quot;outline&quot;) is star(5, &quot;outline&quot;, &quot;yellow&quot;)</td>
<td>E # f :: String, Number -&gt; Image</td>
</tr>
<tr>
<td>f(5, &quot;solid&quot;) is star(5, &quot;solid&quot;, &quot;yellow&quot;)</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>
**Matching Examples and Function Definitions**

Find the variables in `gt` and label them with the word "size".

**Examples:**

```
gt(20) is triangle(20, "solid", "green")
gt(50) is triangle(50, "solid", "green")
end
```

```
fun gt(size): triangle(size, "solid", "green") end
```

Highlight and label the variables in the example lists below. Then, using `gt` as a model, match the examples to their corresponding function definitions.

<table>
<thead>
<tr>
<th>Examples</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>A</td>
</tr>
</tbody>
</table>
| `f("solid")` is circle(8, "solid", "red")
`f("outline")` is circle(8, "outline", "red") | `fun f(s): star(s, "outline", "red") end` |
| **2**    | B          |
| `f(2) is 2 + 2`
`f(4) is 4 + 4`
`f(5) is 5 + 5` | `fun f(num): num + num end` |
| **3**    | C          |
| `f("red")` is circle(7, "solid", "red")
`f("teal")` is circle(7, "solid", "teal") | `fun f(c): star(9, "solid", c) end` |
| **4**    | D          |
| `f("red")` is star(9, "solid", "red")
`f("grey")` is star(9, "solid", "grey")
`f("pink")` is star(9, "solid", "pink") | `fun f(s): circle(8, s, "red") end` |
| **5**    | E          |
| `f(3) is star(3, "outline", "red")`
`f(8) is star(8, "outline", "red")` | `fun f(c): circle(7, "solid", c) end` |
Creating Contracts From Examples

Write the contracts used to create each of the following collections of examples. The first one has been done for you.

1) # big-triangle :: Number, String -> Image

```
examples:
    big-triangle(100, "red") is triangle(100, "solid", "red")
    big-triangle(200, "orange") is triangle(200, "solid", "orange")
end
```

2) __________

```
examples:
    purple-square(15) is rectangle(15, 15, "outline", "purple")
    purple-square(6) is rectangle(6, 6, "outline", "purple")
end
```

3) __________

```
examples:
    banner("Game Today!") is text("Game Today!", 50, "red")
    banner("Go Team!") is text("Go Team!", 50, "red")
    banner("Exit") is text("Exit", 50, "red")
end
```

4) __________

```
examples:
    twinkle("outline", "red") is star(5, "outline", "red")
    twinkle("solid", "pink") is star(5, "solid", "pink")
    twinkle("outline", "grey") is star(5, "outline", "grey")
end
```

5) __________

```
examples:
    half(5) is 5 / 2
    half(8) is 8 / 2
    half(900) is 900 / 2
end
```
gt

Directions: Define a function called gt, which makes solid green triangles of whatever size we want.

Every contract has three parts...

<table>
<thead>
<tr>
<th>#</th>
<th>gt:</th>
<th>Number</th>
<th>-&gt;</th>
<th>Image</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Domain</td>
<td></td>
<td>Range</td>
</tr>
</tbody>
</table>

Write some examples, then circle and label what changes...

examples:

<table>
<thead>
<tr>
<th>#</th>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>gt(10)</td>
<td></td>
<td>triangle(10, &quot;solid&quot;, &quot;green&quot;)</td>
</tr>
<tr>
<td>2</td>
<td>gt(20)</td>
<td></td>
<td>triangle(20, &quot;solid&quot;, &quot;green&quot;)</td>
</tr>
</tbody>
</table>

Write the definition, giving variable names to all your input values...

fun gt( size ): 

triangle(size, "solid", "green") what the function does with those variable(s)

end

bc

Directions: Define a function called bc, which makes solid blue circles of whatever radius we want.

Every contract has three parts...

<table>
<thead>
<tr>
<th>#</th>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Domain</td>
<td></td>
</tr>
</tbody>
</table>

Write some examples, then circle and label what changes...

examples:

<table>
<thead>
<tr>
<th>#</th>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Write the definition, giving variable names to all your input values...

fun | variable(s): 

what the function does with those variable(s)

end
sticker

Directions: Define a function called `sticker`, which consumes a color and draws a 50px solid star of the given color.

Every contract has three parts...

```
# function name :: Domain -> Range

Write some examples, then circle and label what changes...

examples:

    function name (input(s)) is what the function produces

end

Write the definition, giving variable names to all your input values...

fun

    function name (variable(s)):

    what the function does with those variable(s)

end

```

gold-star

Directions: Define a function called `gold-star`, which takes in a radius and draws a solid gold star of that given size.

Every contract has three parts...

```
# function name :: Domain -> Range

Write some examples, then circle and label what changes...

examples:

    function name (input(s)) is what the function produces

end

Write the definition, giving variable names to all your input values...

fun

    function name (variable(s)):

    what the function does with those variable(s)

end

```
name-color

Directions: Define a function called `name-color`, which makes an image of your name at size 50 in whatever color is given.

Every contract has three parts...

```
# function name :: Domain -> Range
```

Write some examples, then circle and label what changes...

examples:

```
function name (input(s)) is what the function produces

function name (input(s)) is what the function produces
```

directions:

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```

name-size

Directions: Define a function called `name-size`, which makes an image of your name in your favorite color (be sure to specify your name and favorite color!) in whatever size is given.

Every contract has three parts...

```
# function name :: Domain -> Range
```

Write some examples, then circle and label what changes...

examples:

```
function name (input(s)) is what the function produces

function name (input(s)) is what the function produces
```

directions:

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```

110
Do the Examples Have the Same Contracts?

For each pair of Examples below, decide whether the two examples have the same Contract. If they do, fill in the Contract in the space provided. If not, write a few words explaining how you know their contracts aren't the same.

1) examples:
   mystery(30) is 30 * 50
   mystery(10) is text("Welcome!", 10, "darkgreen")
   end

2) examples:
   mystery(30, 40) is 40 - (2 * 30)
   mystery(10, 15) is 15 - (2 * 10)
   end

3) examples:
   mystery("New York") is text("New York", 20, "red")
   mystery(20) is text("New York", 20, "red")
   end

4) examples:
   mystery("green", 32) is circle(32, "outline", "green")
   mystery(18, "green") is circle(18, "outline", "green")
   end

5) examples:
   mystery(6, 9, 10) is 6 / (9 + 10)
   mystery(3, 7) is 3 / (7 + 10)
   end

6) examples:
   mystery("red", "blue") is text("blue", 25, "red")
   mystery("purple", "Go Team!") is text("Go Team!", 25, "purple")
   end
Do the Examples Have the Same Contracts? (2)

For each pair of Examples below, decide whether the two examples have the same Contract. If they do, fill in the Contract in the space provided. If not, write a few words explaining how you know their contracts aren't the same.

1)

```
examples:
  mystery(triangle(70, "solid", "green")) is triangle(140, "solid", "green")
  mystery(circle(100, "solid", "blue")) is circle(200, "solid", "blue")
end
```

2)

```
examples:
  mystery("red") is triangle(140, "solid", "red")
  mystery("blue", "circle") is circle(140, "solid", "blue")
end
```

3)

```
examples:
  mystery("+", 4, 5) is 4 + 5
  mystery("sqrt", 25) is num-sqrt(25)
end
```

4)

```
examples:
  mystery("circle", 4) is num-pi * num-sqr(4)
  mystery("square", 5) is num-sqr(5)
end
```

5)

```
examples:
  mystery("dog") is 3
  mystery("cat") is "kitten"
end
```

6)

```
examples:
  mystery("dog") is 3
  mystery("kitten") is 6
end
```
## Matching Contracts and Examples

Match each Example on the left with its Contract on the right. NOTE: Multiple examples may match to the same Contract!

<table>
<thead>
<tr>
<th>Contract</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>examples: match(circle(10, &quot;solid&quot;, &quot;green&quot;)) is rotate(37, circle(10, &quot;solid&quot;, &quot;green&quot;)) end</td>
<td>1</td>
</tr>
<tr>
<td>examples: match(triangle(20, &quot;solid&quot;, &quot;blue&quot;), 3) is scale(3, triangle(20, &quot;solid&quot;, &quot;blue&quot;)) end</td>
<td>2</td>
</tr>
<tr>
<td>examples: match(circle(20, &quot;outline&quot;, &quot;gold&quot;)) is rotate(37, circle(20, &quot;outline&quot;, &quot;gold&quot;)) end</td>
<td>3</td>
</tr>
<tr>
<td>examples: match(30, &quot;red&quot;) is 30 + string-length(&quot;red&quot;) end</td>
<td>4</td>
</tr>
<tr>
<td>examples: match(circle(10, &quot;solid&quot;, &quot;orange&quot;), 22) is scale(22, circle(10, &quot;solid&quot;, &quot;orange&quot;)) end</td>
<td>5</td>
</tr>
<tr>
<td>examples: match(10, &quot;blue&quot;) is 10 + string-length(&quot;blue&quot;) end</td>
<td>6</td>
</tr>
<tr>
<td>examples: match(5, star(20, &quot;solid&quot;, &quot;red&quot;)) is rotate(90 - 5, star(20, &quot;solid&quot;, &quot;red&quot;)) end</td>
<td>7</td>
</tr>
<tr>
<td>examples: match(num-abs(-4), &quot;45&quot;) is 4 end</td>
<td>8</td>
</tr>
</tbody>
</table>

**A** # match :: Number, Image -> Image

**B** # match :: Image, Number -> Image

**C** # match :: Image -> Image

**D** # match :: Number, String -> Number
Matching Contracts and Examples (2)

Match each Example on the left with its Contract on the right. NOTE: Multiple examples may match to the same Contract!

<table>
<thead>
<tr>
<th>Contract</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>examples:</td>
<td>match(1.5) is &quot;greater than 1&quot;</td>
</tr>
<tr>
<td>end</td>
<td>1</td>
</tr>
<tr>
<td>examples:</td>
<td>match(24) is star(24 * 2, &quot;outline&quot;, &quot;purple&quot;)</td>
</tr>
<tr>
<td>end</td>
<td>2</td>
</tr>
<tr>
<td>examples:</td>
<td>match(string-length(&quot;tabletop&quot;)) is &quot;8&quot;</td>
</tr>
<tr>
<td>end</td>
<td>3</td>
</tr>
<tr>
<td>examples:</td>
<td>match(circle(10, &quot;solid&quot;, &quot;silver&quot;), 16) is 16 * image-height(circle(10, &quot;solid&quot;, &quot;silver&quot;))</td>
</tr>
<tr>
<td>end</td>
<td>4</td>
</tr>
<tr>
<td>examples:</td>
<td>match(&quot;triangle&quot;, &quot;blue&quot;) is triangle(40, &quot;outline&quot;, &quot;blue&quot;)</td>
</tr>
<tr>
<td>end</td>
<td>5</td>
</tr>
<tr>
<td>examples:</td>
<td>match(30) is star(30 * 2, &quot;outline&quot;, &quot;purple&quot;)</td>
</tr>
<tr>
<td>end</td>
<td>6</td>
</tr>
<tr>
<td>examples:</td>
<td>match(string-length(&quot;coffee&quot;), string-length(&quot;tea&quot;)) is 6 + 3</td>
</tr>
<tr>
<td>end</td>
<td>7</td>
</tr>
<tr>
<td>examples:</td>
<td>match(30) is star(30 * 2, &quot;outline&quot;, &quot;purple&quot;)</td>
</tr>
<tr>
<td>end</td>
<td>8</td>
</tr>
</tbody>
</table>
Solving Word Problems

Being able to see functions as Contracts, Examples or Definitions is like having three powerful tools. These representations can be used together to solve word problems!

1) When reading a word problem, the first step is to figure out the **Contract** for the function you want to build. Remember, a Contract must include the Name, Domain and Range for the function!

2) Then we write a **Purpose Statement**, which is a short note that tells us what the function *should do*. Professional programmers work hard to write good purpose statements, so that other people can understand the code they wrote! Programmers work on teams; the programs they write must outlast the moment that they are written.

3) Next, we write at least two **Examples**. These are lines of code that show what the function should do for a **specific** input. Once we see examples of at least two inputs, we can *find a pattern* and see which parts are changing and which parts aren’t.

4) To finish the Examples, we circle the parts that are changing, and label them with a short **variable name** that explains what they do.

5) Finally, we **define the function** itself! This is pretty easy after you have some examples to work from: we copy everything that didn’t change, and replace the changeable stuff with the variable name!
Match each word problem below to its corresponding purpose statement.

1. Annie got a new dog, Xavier, that eats about 5 times as much as her little dog, Rex, who is 10 years old. She hasn't gotten used to buying enough dog food yet. Write a function that generates an estimate for how many pounds of food Xavier will eat, given the amount of food that Rex usually consumes in the same amount of time.

   D. Consume the pounds of food Rex eats and divide by 5.

2. Adrienne's raccoon, Rex, eats 5 more pounds of food each week than her pet squirrel, Lili, who is 7 years older. Write a function to determine how much Lili eats in a week, given how much Rex eats.

   B. Consume the pounds of food Rex eats and subtract 5.

3. Alejandro's rabbit, Rex, poops about 3/5 of what it eats. His rabbit Hutch is 10 cubic feet. Write a function to figure out how much rabbit poop Alejandro's rabbit, Rex, poops about 3/5 of what it eats. His rabbit Hutch is about 1/2 of a week each week that Rex eats.

   C. Consume the pounds of food Rex eats and multiply by 5.

4. Max's turtle, Rex, eats 5 pounds less per week than his turtle, Harry, who is 2 inches taller. Write a function to calculate how much food Harry eats.

   A. Consume the pounds of food Rex eats and divide by 5.
Writing Examples from Purpose Statements

We’ve provided contracts and purpose statements to describe two different functions. Write examples for each of those functions.

Contract and Purpose Statement

Every contract has three parts...

# triple:: Number -> Number

function name          Domain       Range

# Consumes a Number and triples it.

Examples

Write some examples, then circle and label what changes...

examples:

function name   (input(s)) is what the function produces

end

triple

<table>
<thead>
<tr>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
</table>

Contract and Purpose Statement

Every contract has three parts...

# upside-down:: Image -> Image

function name          Domain       Range

# Consumes an image, and turns it upside down by rotating it 180 degrees.

Examples

Write some examples, then circle and label what changes...

examples:

function name   (input(s)) is what the function produces

end

upside-down

<table>
<thead>
<tr>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
</table>
Fixing Purpose Statements

Beneath each of the word problems below is a purpose statement (generated by ChatGPT!) that is either missing information or includes unnecessary information. Write an improved version of each purpose statement beneath the original, then explain what was wrong with the ChatGPT-generated Purpose Statement.

1) **Word Problem:** The New York City ferry costs $2.75 per ride. The Earth School requires two chaperones for any field trip. Write a function `fare` that takes in the number of students in the class and returns the total fare for the students and chaperones.

   **ChatGPT’s Purpose Statement:** Take in the number of students and add 2.

   **Improved Purpose Statement:**

   Problem with ChatGPT’s Purpose Statement:

2) **Word Problem:** It is tradition for the Green Machines to go to Humpy Dumpty’s for ice cream with their families after their soccer games. Write a function `cones` to take in the number of kids and calculate the total bill for the team, assuming that each kid brings two family members and cones cost $1.25.

   **ChatGPT’s Purpose Statement:** Take in the number of kids on the team and multiply it by 1.25.

   **Improved Purpose Statement:**

   Problem with ChatGPT’s Purpose Statement:

3) **Word Problem:** The cost of renting an ebike is $3 plus an additional $0.12 per minute. Write a function `ebike` that will calculate the cost of a ride, given the number of minutes ridden.

   **ChatGPT’s Purpose Statement:** Take in the number of minutes and multiply it by 3.12.

   **Improved Purpose Statement:**

   Problem with ChatGPT’s Purpose Statement:

4) **Word Problem:** Suleika is a skilled house painter at only age 21. She has painted hundreds of rooms and can paint about 175 square feet an hour. Write a function `paint` that takes in the number of square feet of the job and calculates how many hours it will take her.

   **ChatGPT’s Purpose Statement:** Take in the number of square feet of walls in a house and divide them by 175 then add 21 years.

   **Improved Purpose Statement:**

   Problem with ChatGPT’s Purpose Statement:
Word Problem: rocket-height

Directions: A rocket blasts off, and is now traveling at a constant velocity of 7 meters per second. Use the Design Recipe to write a function rocket-height, which takes in a number of seconds and calculates the height.

### Contract and Purpose Statement
Every contract has three parts...

<table>
<thead>
<tr>
<th>function name</th>
<th>::</th>
<th>Domain</th>
<th>-&gt;</th>
<th>Range</th>
</tr>
</thead>
</table>

# what does the function do?

### Examples
Write some examples, then circle and label what changes...

**examples:**

```plaintext
function name (input(s)) is what the function produces
```

**end**

### Definition
Write the definition, giving variable names to all your input values...

```plaintext
fun function name (variable(s)):
```

```plaintext
what the function does with those variable(s)
```

**end**


We've provided contracts and purpose statements to describe two different functions. Write examples for each of those functions.

**Contract and Purpose Statement**

Every contract has three parts...

<table>
<thead>
<tr>
<th>function name</th>
<th>Image</th>
<th>Domain</th>
<th>Range</th>
</tr>
</thead>
</table>

# half-image:

Consumes an image, and produces that image scaled to half its size.

**Examples**

Write some examples, then circle and label what changes...

examples:

<table>
<thead>
<tr>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
</table>

end

---

**Contract and Purpose Statement**

Every contract has three parts...

<table>
<thead>
<tr>
<th>function name</th>
<th>Number, Number</th>
<th>Domain</th>
<th>Range</th>
</tr>
</thead>
</table>

# product-squared:

Consumes two numbers and squares their product

**Examples**

Write some examples, then circle and label what changes...

examples:

<table>
<thead>
<tr>
<th>function name</th>
<th>input(s)</th>
<th>what the function produces</th>
</tr>
</thead>
</table>

end
Rocket Height Challenges

1) Can you make the rocket fly faster?

2) Can you make the rocket fly slower?

3) Can you make the rocket sink down instead of fly up?

4) Can you make the rocket accelerate over time, so that it moves faster the longer it flies?

5) Can you make the rocket blast off and then land again?

6) Can you make the rocket blast off, reach a maximum height of exactly 1000 meters, and then land?

7) Can you make the rocket blast off, reach a maximum height of exactly 1000 meters, and then land after exactly 100 seconds?

8) Can you make the rocket fly to the edge of the universe?
Directions: Use the Design Recipe to write a function `split-tab` that takes in a cost and the number of people sharing the bill and splits the cost equally.

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name :: Domain -> Range
#
# what does the function do?
```

**Examples**

Write some examples, then circle and label what changes...

```
examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)

end
```

---

Directions: Use the Design Recipe to write a function `tip-calculator` that takes in the cost of a meal and returns the 15% tip for that meal.

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name :: Domain -> Range
#
# what does the function do?
```

**Examples**

Write some examples, then circle and label what changes...

```
examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)

end
```
The Design Recipe (Direct Variation)

Directions: Use the Design Recipe to write a function wage, that takes in a number of hours worked and returns the amount a worker will get paid if their rate is $10.25/hr.

Contract and Purpose Statement
Every contract has three parts...

```
# function name :: Domain -> Range
#

what does the function do?

Examples
Write some examples, then circle and label what changes...
examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition
Write the definition, giving variable names to all your input values...

fun

function name (variable(s)):

what the function does with those variable(s)

end

Directions: On average, people burn about 11 calories/minute riding a bike. Use the Design Recipe to write a function calories-burned that takes in the number of minutes you bike and returns the number of calories burned.

Contract and Purpose Statement
Every contract has three parts...

```
# function name :: Number -> Number
#

what does the function do?

Examples
Write some examples, then circle and label what changes...
examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition
Write the definition, giving variable names to all your input values...

fun

function name (variable(s)):

what the function does with those variable(s)

end
The Design Recipe (Slope/Intercept)

Directions: For his birthday, James’ family decided to open a savings account for him. He started with $50 and committed to adding $10 a week from his afterschool job teaching basketball to kindergartners. Use the Design Recipe to write a function `savings` that takes in the number of weeks since his birthday and calculates how much money he has saved.

Contract and Purpose Statement
Every contract has three parts...

```
# function name :: Domain -> Range
#
```

Examples
Write some examples, then circle and label what changes...

```
examples:
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
end
```

Definition
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

function name (variable(s)) what the function does with those variable(s)
end
```

Directions: Use the Design Recipe to write a function `moving` that takes in the days and number of miles driven and returns the cost of renting a truck. The truck is $45 per day and each driven mile is 15¢.

Contract and Purpose Statement
Every contract has three parts...

```
# function name :: Domain -> Range
#
```

Examples
Write some examples, then circle and label what changes...

```
examples:
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
end
```

Definition
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

function name (variable(s)) what the function does with those variable(s)
end
```
The Design Recipe (Negative Slope/Intercept)

Directions: An Olympic pool holds 660,000 gallons of water. A fire hose can spray about 250 gallons per minute. Use the Design Recipe to write a function pool that takes in the number of minutes that have passed and calculates how much water is still needed to fill it.

Contract and Purpose Statement
Every contract has three parts...

# function name :: Domain -> Range
#
# what does the function do?

Examples
Write some examples, then circle and label what changes...
examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition
Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end

Directions: The community arts fund awards a $1500 grant each month to support a new mural. They started with $50000 in their account. Use the Design Recipe to write a function funds-available that takes in the number of months and calculates how much money they have left.

Contract and Purpose Statement
Every contract has three parts...

# function name :: Domain -> Range
#
# what does the function do?

Examples
Write some examples, then circle and label what changes...
examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition
Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end
The Design Recipe (Geometry - Rectangles)

Directions: Use the Design Recipe to write a function `lawn-area` that takes in the length and width of a rectangular lawn and returns its area.

**Contract and Purpose Statement**
Every contract has three parts...

```
# function name :: Domain -> Range
#
```

**Examples**
Write some examples, then circle and label what changes...

```
examples:
```

```
function name (input(s)) is what the function produces
```

```
function name (input(s)) is what the function produces
```

end

**Definition**
Write the definition, giving variable names to all your input values...

```
fun (variable(s)):
```

```
function name (variable(s)):
```

```
what the function does with those variable(s)
```

end

Directions: Use the Design Recipe to write a function `rect-perimeter` that takes in the length and width of a rectangle and returns the perimeter of that rectangle.

**Contract and Purpose Statement**
Every contract has three parts...

```
# function name :: Domain -> Range
#
```

**Examples**
Write some examples, then circle and label what changes...

```
examples:
```

```
function name (input(s)) is what the function produces
```

```
function name (input(s)) is what the function produces
```

end

**Definition**
Write the definition, giving variable names to all your input values...

```
fun (variable(s)):
```

```
function name (variable(s)):
```

```
what the function does with those variable(s)
```

end
The Design Recipe (Geometry - Rectangular Prisms)

**Directions:** Use the Design Recipe to write a function `rect-prism-vol` that takes in the length, width, and height of a rectangular prism and returns the Volume of a rectangular prism.

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name ::: Domain -> Range
#
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```

---

**Directions:** Use the Design Recipe to write a function `rect-prism-sa` that takes in the width, length and height of a rectangular prism and calculates its surface area (the sum of the areas of each of its six faces)

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name ::: Domain -> Range
#
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```
The Design Recipe (Geometry - Circles)

Directions: Use the Design Recipe to write a function circle-area-dec that takes in a radius and uses the decimal approximation of pi (3.14) to return the area of the circle.

Contract and Purpose Statement

Every contract has three parts...

# function name ::= Domain -> Range
#
# what does the function do?

Examples

Write some examples, then circle and label what changes...

eamples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

Definition

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end

Directions: Use the Design Recipe to write a function circumference that takes in a radius and uses the decimal approximation of pi (3.14) to return the circumference of the circle.

Contract and Purpose Statement

Every contract has three parts...

# function name ::= Domain -> Range
#
# what does the function do?

Examples

Write some examples, then circle and label what changes...

eamples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

Definition

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end
The Design Recipe (Geometry - Cylinders)

Directions: Use the Design Recipe to write a function `circle-area` that takes in a radius and uses the fraction approximation of π (\(\frac{22}{7}\)) to return the area of the circle.

**Contract and Purpose Statement**
Every contract has three parts...

```
# function name :: Domain -> Range
#
# what does the function do?
```

**Examples**
Write some examples, then circle and label what changes...

```
examples:

  function name (input(s)) is what the function produces
  function name (input(s)) is what the function produces

end
```

**Definition**
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):
  function name variable(s)
  what the function does with those variable(s)
end
```

Directions: Use the Design Recipe to write a function `cylinder` that takes in a cylinder's radius and height and calculates its volume, making use of the function `circle-area`.

**Contract and Purpose Statement**
Every contract has three parts...

```
# function name :: Domain -> Range
#
# what does the function do?
```

**Examples**
Write some examples, then circle and label what changes...

```
examples:

  function name (input(s)) is what the function produces
  function name (input(s)) is what the function produces

end
```

**Definition**
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):
  function name variable(s)
  what the function does with those variable(s)
end
```
The Design Recipe (Breaking Even)

Directions: The Swamp in the City Festival is ordering t-shirts. The production cost is $75 to set up the silk screen and $9 per shirt. Use the Design Recipe to write a function min-shirt-price that takes in the number of shirts to be ordered, \( n \), and returns the minimum amount the festival should charge for the shirts in order to break even. (Assume that they will sell all of the shirts.)

Contract and Purpose Statement

Every contract has three parts...

# function name :: Domain -> Range
#
#
# what does the function do?

Examples

Write some examples, then circle and label what changes...

eamples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces
end

Definition

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)
end
The Design Recipe (Marquee & Cubing)

Directions: Use the Design Recipe to write a function `marquee` that takes in a message and returns that message in large gold letters.

**Contract and Purpose Statement**

Every contract has three parts...

<table>
<thead>
<tr>
<th>function name</th>
<th>Domain</th>
<th>Range</th>
</tr>
</thead>
</table>

- what does the function do?

**Examples**

Write some examples, then circle and label what changes...

**examples:**

- function name (input(s)) is what the function produces
- function name (input(s)) is what the function produces

**Definition**

Write the definition, giving variable names to all your input values...

**fun** function name (variable(s)):

- what the function does with those variable(s)

Directions: Use the Design Recipe to write a function `num-cube` that takes in a number and returns the cube of that number.

**Contract and Purpose Statement**

Every contract has three parts...

<table>
<thead>
<tr>
<th>function name</th>
<th>Domain</th>
<th>Range</th>
</tr>
</thead>
</table>

- what does the function do?

**Examples**

Write some examples, then circle and label what changes...

**examples:**

- function name (input(s)) is what the function produces
- function name (input(s)) is what the function produces

**Definition**

Write the definition, giving variable names to all your input values...

**fun** function name (variable(s)):

- what the function does with those variable(s)
Defining Table Functions

The steps of the Design Recipe don't change just because we're working with Rows, but we can make some adjustments when using Row-consuming functions to filter tables and build columns! Let's try a concrete example:

Write a function `is-lizard`, which tells us whether an animal is a lizard.

Contract and Purpose

- We still want to pick good names. Are we writing a function to check if an animal is a lizard? Call it `is-lizard`!
- The Domain is a lot easier — it's always a Row!
- The Range is easier, too. If we're writing a function to filter a Table, we know the Range has to be a Boolean. (What would it be if we were building a column of Numbers? Images? Strings?)

Examples

The goal of the Examples step is to find the pattern that represents what the function does. When working with Rows, sometimes we have to start by just focusing on what the answer should be.

Suppose `animalC` is a lizard, and `animalD` is a cat. We can imagine the answers for an `is-lizard` to be...

```
examples:
  is-lizard(animalC) is true
  is-lizard(animalD) is false
end
```

But how do we know these are true and false? Well, we KNOW `animalC` is a lizard, and we KNOW `animalD` is a cat. So let's replace those answers with the Boolean expressions that compare their species:

```
examples:
  is-lizard(animalC) is "lizard" == "lizard"  # will produce true
  is-lizard(animalD) is "cat" == "lizard"     # will produce false
end
```

But what work gives us "lizard" and "cat"? Well, we can look in the species column!

```
examples:
  is-lizard(animalC) is animalC["species"] == "lizard" # will produce true
  is-lizard(animalD) is animalD["species"] == "lizard" # will produce false
end
```

Sometimes we can go straight to this final form, doing the whole thing in one step. But it's nice to know we can break it down into pieces if we have to.

Once we see the pattern, we can circle and label what changes. In this case, only the Row representing the animal changes! We might use `r` as label, to represent the Row.

Definition

The final step in the Design Recipe is to take the pattern from our examples and generalize it to work with any input. It's no different when working with Rows.

Once again, our previous step is a huge help. We can simply copy everything that stays the same, and replace the part that changes with the label we used:

```
fun is-lizard(r): r["species"] == "lizard" end
```
Open the Row Functions Starter File on your computer, save a copy, and Click "Run"!

1) Find the definition for cat-row. What do you get back when you evaluate cat-row in the Interactions Area? 

2) How many weeks did it take for cat-row to be adopted? How many for young-row? 

3) Scroll down in the starter file until you find the Contract, Purpose, Examples and Definition for weeks-dot. Discuss with your partner:
   - What is the Domain of this function? The Range? 
   - The Purpose Statement is a way of describing the function in detail. Does the Purpose Statement make it clear what this function should do, when given a Row? 
   - How many examples do you see defined for this function? 

4) Look at the first two examples. How do they satisfy the Contract and Purpose Statement?

These examples show us exactly what should be produced for cat-row and young-row - the two Rows representing "Sasha" and "Wade", based on their weeks to adoption. But these examples only tell us part of the story! Where does the computer get the number of weeks from?

5) Now look at the last two examples. How do they satisfy (connect to) to the first two?

6) Now look at the definition. How is this connected to our examples?

7) We've learned that representations of functions have to match. 

Look at the Examples carefully - there's a mistake, where the Examples and Definition don't match the Contract and Purpose. 

What's the mistake?

When you're done, fix the mistake in the code!
Design Recipe Telephone Set 2: is-old

Directions: Define a function called is-old, which consumes a Row of the animals table and computes whether it is more than 10 years old. HINT: use predefined rows like old-row to make your examples easier!

Contract and Purpose Statement

Every contract has three parts...

# is-old:: Row -> Boolean

# what does the function do?

Examples

Write some examples, then circle and label what changes...

examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

Definition

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end
Design Recipe Telephone Set 2: kilos

Directions: Define a function called kilos, which consumes a Row of the animals table and computes the number of kilograms the animal weighs (1 kg = 2.2lbs). HINT: use predefined rows like cat-row to make your examples easier!

Contract and Purpose Statement
Every contract has three parts...

```
# kilos:: Row -> Number

# what does the function do?

Examples
Write some examples, then circle and label what changes...
examples:
```

```
function name (input(s)) is what the function produces
```

end

Definition
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):
```

```
what the function does with those variable(s)
```

end
Design Recipe Telephone Set 2: is-cat

Directions: Define a function called is-cat, which consumes a Row of the animals table and computes whether the animal is a cat. HINT: use predefined rows like cat-row to make your examples easier!

**Contract and Purpose Statement**

Every contract has three parts...

```
# is-cat:: Row -> Boolean

# what does the function do?
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

function name (variable(s)) what the function does with those variable(s)

end
```
Design Recipe

Directions:

Contract and Purpose Statement
Every contract has three parts...

```
# function name :: Domain --> Range
#
```

Examples
Write some examples, then circle and label what changes...

elements:

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
```

definition
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```

Directions:

Contract and Purpose Statement
Every contract has three parts...

```
# function name :: Domain --> Range
#
```

Examples
Write some examples, then circle and label what changes...

elements:

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
```

definition
Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```

137
1) On the three lines below, write the code to lookup the value of the age column from dog-row, old-row, and young-row.

2) On the three lines below, write the code that uses the circle function to draw a solid, blue circle whose radius is the age of the animal. You can find the Contract for circle on the Contracts Page. If you're working with a printed workbook, the contracts pages are included in the back.

3) When you are confident with the code you have written, check with your partner or another student. Do you have the same code? Why or why not?

Instead of writing this out over and over for each animal, let's define a function to do it for us!

### Defining the Function

**Directions:** Define a function called age-dot, which takes in a row from the Animals Table and draws a solid, blue circle whose radius is the age of the animal. **HINT:** Use of the rows from above in your examples!

**Every contract has three parts...**

<table>
<thead>
<tr>
<th>function name</th>
<th>Domain</th>
<th>Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>age-dot</td>
<td>Row</td>
<td>-&gt; Image</td>
</tr>
</tbody>
</table>

**Write some examples, then circle and label what changes...**

examples:

```plaintext
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
```

**end**

**Write the definition, giving variable names to all your input values...**

```plaintext
fun age-dot(input(s)):
  function name variable(s)
  what the function does with those variable(s)

end
```
To help you with this page, we've re-printed the Contract for the `text` function, and an example of how to use it. (Don't forget, you can always refer to the [Contracts Pages](#). If you're working with a printed workbook, they are included in the back.)

```plaintext
# text :: (String, Number, String) -> Image

text("hello", 24, "green")
```

1) On the three lines below, write the code to lookup the value of the `species` column from `dog-row`, `old-row`, and `young-row`.

```plaintext
(species `dog-row`) (species `old-row`) (species `young-row`)
```

2) On the three lines below, write the code that uses the `text` function to show the species of those same three rows in red, 15px letters.

```plaintext
of (text 24 "red"
```

3) When you are confident with the code you have written, check with your partner or another student. Do you have the same code? Why or why not?

*Instead of writing this out over and over for each animal, let's define a function to do it for us!*

### Defining the Function

**Directions:** Define a function called `species-tag`, which takes in a row from the Animals Table and draws its name in red, 15px letters.  
*HINT: Use of the rows from above in your examples!*

<table>
<thead>
<tr>
<th>Row</th>
<th>-&gt; Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>function name</td>
<td>Domain</td>
</tr>
<tr>
<td># species-tag:</td>
<td>Range</td>
</tr>
</tbody>
</table>

**Every contract has three parts...**

- **Write some examples, then circle and label what changes...**

  **examples:**

  ```plaintext
  function name (input(s)) is what the function produces
  ```

- **Write the definition, giving variable names to all your input values...**

  **fun** `species-tag`:

  ```plaintext
  function name variable(s):
  ```

  ```plaintext
  what the function does with those variable(s)
  ```

end

end
Exploring Image Scatter Plots

Look at the code in the Custom Scatter Plot Starter File.

1) Compare the definitions of age-dot and species-tag to what you wrote. Are they the same? If not, what is different?

Answer the following questions about the last line of code in the file, which is commented out.

2) What is the name of the function being used here? How many things are in its Domain?

3) What is the 1st argument? What is its data type?

4) What is the 2nd argument? What is its data type?

5) What is the 3rd argument? What is its data type?

6) What is the 4th argument?

7) What is the data type for the fourth thing in the Domain? If you're not sure, write down your thinking. What can you rule out? What do you think it might be?

8) Uncomment the last line at the bottom of the file, and click "Run". What does image-scatter-plot do with its 4th argument?

9) Try changing your age-dot function to use different colors, or even different shapes! Can you make the size of the shape be one half the age of the animal?

10) On a new line in the Definitions Area, try making an image-scatter-plot using the species-tag function. Click run, and describe what you see.

Understanding Custom Displays

11) Look at the image scatter plot with differently sized dots. Can you draw any conclusions about animals that are both young and lightweight?

12) Looking at the scatter plot, the director of the shelter says: "Animals that are older and that weigh greater than 50 pounds generally take at least 5 weeks to be adopted." Do you agree with this statement? Explain.

13) Now, look at your image scatter plot with species nametags in red. What does this chart reveal that we couldn't see on the original (pounds v. weeks) scatter plot?
Exploring Conditional / Piecewise Functions

Here's an example of a piecewise function with 3 "pieces" (or "conditions"):

```python
# species-dot :: (Row) -> Image
fun species-dot(r):
  if (r["species"] == "dog"):       square(5, "solid", "black")
  else if (r["species"] == "cat"): square(5, "solid", "orange")
  else if (r["species"] == "lizard"): square(5, "solid", "green")
end
end
```

<table>
<thead>
<tr>
<th>What do you Notice about this code?</th>
<th>What do you Wonder?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1) What will this function produce for a dog? ________________________________

2) What will this function produce for a cat? ________________________________

Open the Piecewise Displays Starter File, and click "Run".

3) Compare the regular scatter plot with the image scatter plot. What can you see now that you couldn't see before?
_________________________________________________________________________
_________________________________________________________________________

4) Compare the regular histogram with the image histogram. What can you see now that you couldn't see before?
_________________________________________________________________________
_________________________________________________________________________

5) What do you think will happen if we run the function on a species that it has no condition for? ________________________________

6) On line 45, add a comment (#) to "turn off" the condition for snails. Click Run and test your prediction. In your own words, describe how piecewise / conditional functions work.
_________________________________________________________________________
_________________________________________________________________________

★ Optional: Make a new function (don't delete species-dot!), which uses piecewise functions to draw something different! For example, have it draw different shapes depending on whether an animal is younger than 3 years old or not.
Word Problem: fixed-dot

Directions: We want to generate a custom dot for our image-scatter-plot, to distinguish between animals that are fixed and unfixed. Write a function called fixed-dot, which takes in a Row from the animals table and generates a 5-pixel solid green dot if the animal is fixed and a 5-pixel black dot if the animal is not fixed.

Every contract has three parts...

# function name :: Domain -> Range

Write some examples, then circle and label what changes...

examples:

function name (input(s)) is what the function produces

[example 1]

function name (input(s)) is what the function produces

end

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

[example 2]

what the function does with those variable(s)

end
Open the Animal Images Starter File and look at the code.

<table>
<thead>
<tr>
<th>What do you Notice?</th>
<th>What do you Wonder?</th>
</tr>
</thead>
</table>

1) How is this program similar to the last one we looked at?

2) How is the code different from the other image-scatter-plot programs we've seen?

3) Click "Run". How does using clip art help us to better understand the data?

4) What risks might there be to using clip art for data that describes humans?

5) We have seen image-scatter-plot used to make a lot of different custom scatter plots. How could you use image-scatter-plot in your own analysis?
Open the Dots for Value Ranges Starter File and look at the code.

<table>
<thead>
<tr>
<th>What do you Notice?</th>
<th>What do you Wonder?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1) How is this program similar to the last one we looked at?

________________________________________________________________________________

2) How is the code for `weight-dot` different from other functions we’ve seen?

________________________________________________________________________________

3) Can you explain how this code works?

________________________________________________________________________________

________________________________________________________________________________

________________________________________________________________________________
Defining Row Functions & Using Table Functions

Pyret has special functions that we can use to manipulate Tables.

- In this course, the functions we’ll be using are
  - `sort` - consumes the name of a column and a Boolean value to determine if that table should be sorted by that column in ascending order
  - `row-n` - consumes an index (starting with zero!) and produces a row from a table
  - `filter` - consumes a Boolean-producing function, and produces a table containing only rows for which the function returns true
  - `build-column` - consumes the name of a new column, and a function that produces the values in that column for each Row

**Order Matters!** Table operations must be applied in the appropriate order. For example, trying to order a table by a column that hasn’t been built yet will result in an error.
Open the Table Functions Starter File on your computer, save a copy, and click "Run".

1) What is the name of the Table defined on line 5? ________________________________

2) How many columns does it have? _____________________________________________

3) What is the name of the Row defined on line 17? ________________________________

4) Is red-circle a Number, String, Image, Boolean, Table, or Row? __________________

5) Type red-circle into the Interactions Area. What do you get? ______________________

6) In the space provided on lines 19 and 20, add new definitions for two more Rows from this table.

7) A Contract for a function is written on line 22. What is its name? __________________

8) What is its Domain? __________________________________________________________

9) What is its Range? ___________________________________________________________

10) What other functions are defined in the starter file?

______________________________________________________________________________

11) Use the function is-red, passing in a Row. For example, type is-red(blue-triangle). What do you get? ______________________

12) What does the is-red function do? ____________________________________________

For the remaining functions, read the code and try to guess what it does before testing it out by passing in a Row.

13) What does is-polygon do? ____________________________________________________

14) What does lookup-name do? _________________________________________________

15) What does is-triangle do? ___________________________________________________

16) Define two new functions: is-green and is-blue.

★ There is a hidden function called draw-shape. What is its Domain and Range? What does it do?

______________________________________________________________________________

______________________________________________________________________________

★ Is there another way to define is-polygon, so that it doesn't use the "polygon" column at all?

______________________________________________________________________________
Exploring Table Functions

This page refers to the Table Functions Starter File.

Filtering Rows

1) What does \texttt{filter(shapes-table, is-red)} evaluate to? Describe the value you get back below.

2) What does \texttt{filter(shapes-table, is-polygon)} evaluate to? Describe the value you get back below.

3) Write the code to generate a table of only triangles.

4) In the Definitions Area, define \texttt{triangles} by writing \texttt{triangles = filter(shapes-table, is-triangle)}. Click "Run" and evaluate \texttt{triangles} in the Interactions Area.

5) Define \texttt{reds} to be a table of only red shapes.

6) What do the contracts for \texttt{is-red, is-polygon, and is-triangle} have in common?

7) Find the Contract for \texttt{filter} on the Contracts Page. If you’re working with a printed workbook, the contracts pages are included in the back. Explain how \texttt{filter} uses the two inputs specified in the Domain.

8) What happens if you evaluate \texttt{filter(shapes-table, lookup-name)}? Why?

Building Columns

9) What does \texttt{build-column(shapes-table, "red", is-red)} evaluate to?

10) What does \texttt{build-column(shapes-table, "img", draw-shape)} evaluate to?

11) Find the Contract for \texttt{build-column} on the Contracts Page. If you’re working with a printed workbook, the contracts pages are included in the back. Explain how \texttt{build-column} uses the three inputs specified in the Domain.

Challenge!

★ In the Definitions Area, define a table of your own using \texttt{filter} or \texttt{build-column}. Add a comment to describe what’s in it!
You have the following functions defined below:

```plaintext
fun lookup-fixed(animal): animal["fixed"]
fun is-dog(animal): animal["species"] == "dog"
fun is-old(animal): animal["age"] > 10
fun label(animal): text(animal["name"], 20, "red")
```

The table below represents four animals from the shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>species</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>12</td>
<td>true</td>
<td>&quot;dog&quot;</td>
<td>48</td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>false</td>
<td>&quot;dog&quot;</td>
<td>92</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>true</td>
<td>&quot;dog&quot;</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Sunflower&quot;</td>
<td>&quot;female&quot;</td>
<td>2</td>
<td>false</td>
<td>&quot;cat&quot;</td>
<td>51.6</td>
</tr>
</tbody>
</table>

Below is a list of expressions, each using one table function. Match each expression to the description of the table it will produce.

1. sort(t, "age", true)  
   - A  Produces a table with Toggle, Fritz, and Nori - but not Sunflower.

2. sort(t, "pounds", false)  
   - B  Produces a table of all four animals, sorted youngest-to-oldest

3. build-column(t, "sticker", label)  
   - C  Produces a table, with only Toggle.

4. filter(t, is-old)  
   - D  Produces an identical table with an extra column called "dog", whose values are true, true, true, false

5. filter(t, lookup-fixed)  
   - E  Produces a table containing only Nori and Toggle.

6. filter(t, is-dog)  
   - F  Produces a table with all four animals, sorted from heaviest to lightest.

7. build-column(t, "dog", is-dog)  
   - G  Won't run: will produce an error. (Why?)

8. filter(t, label)  
   - H  Produces an identical table with an extra column called "sticker", whose values are images
The Design Recipe: is-cat / is-male

Directions: Define a function called is-cat, which consumes a Row of the animals table and computes whether the animal is a cat. HINT: use predefined rows like cat-row to make your examples easier!

**Contract and Purpose Statement**

Every contract has three parts...

```plaintext
#  is-cat:: Row -> Boolean

# Consumes an animal, and checks whether the species == "cat"
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
is-cat( cat-row ) is cat-row["species"] == "cat"
```

**Definition**

Write the definition, giving variable names to all your input values...

```plaintext
fun is-cat( variable(s) ):

what the function does with those variable(s)
```

---

Directions: Define a function called is-male, which consumes a Row of the animals table and returns true if the animal is male. HINT: use predefined rows like male-row to make your examples easier!

**Contract and Purpose Statement**

Every contract has three parts...

```plaintext
#  is-male:: Row -> Boolean

# Consumes an animal, and checks whether the species == "male"
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
is-male( male-row ) is male-row["species"] == "male"
```

**Definition**

Write the definition, giving variable names to all your input values...

```plaintext
fun is-male( variable(s) ):

what the function does with those variable(s)
```
The Design Recipe: is-young / is-fixed

Directions: Define a function called is-young, which consumes a Row of the animals table and computes whether it is less than 5 years old. HINT: use predefined rows like young-row to make your examples easier!

Contract and Purpose Statement

Every contract has three parts...

```
# function name :: Function Signature -> Range
#
```

Examples

Write some examples, then circle and label what changes...

```
Examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

definition
```

Definition

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)): what the function does with those variable(s)
definition
```

Directions: Define a function called is-fixed, which returns true if an animal is fixed

Contract and Purpose Statement

Every contract has three parts...

```
# function name :: Function Signature -> Range
#
```

Examples

Write some examples, then circle and label what changes...

```
Examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

definition
```

Definition

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)): what the function does with those variable(s)
definition
```
The Design Recipe: is-dog / is-female

Directions: Define a function called is-dog, which consumes a Row of the animals table and computes whether the animal is a dog. HINT: use predefined rows like dog-row to make your examples easier!

Contract and Purpose Statement

Every contract has three parts...

# is-dog:: Row -> Boolean

# Consumes an animal, and checks whether the species == "dog"

Examples

Write some examples, then circle and label what changes...

eamples:

is-dog( dog-row ) is dog-row["species"] == "dog"

Definition

Write the definition, giving variable names to all your input values...

fun : ( )


Directions: Define a function called is-female, which consumes a Row of the animals table and returns true if the animal is female. HINT: use predefined rows like female-row to make your examples easier!

Contract and Purpose Statement

Every contract has three parts...

# is-female:: Row -> Boolean

# Consumes an animal, and checks whether the species == "female"

Examples

Write some examples, then circle and label what changes...

eamples:

is-female( female-row ) is female-row["species"] == "female"

Definition

Write the definition, giving variable names to all your input values...

fun : ( )


The Design Recipe: is-old / name-has-s

Directions: Define a function called `is-old`, which consumes a Row of the animals table and computes whether it is more than 12 years old. HINT: use predefined rows like `old-row` to make your examples easier!

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name :: Domain -> Range
# what does the function do?
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
```

---

Directions: Define a function called `name-has-s`, which returns true if an animal’s name contains the letter "s"

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name :: Domain -> Range
# what does the function do?
```

**Examples**

Write some examples, then circle and label what changes...

**examples:**

```
name-has-s(cat-row) is
function name input(s) what the function produces
name-has-s(dog-row) is
function name input(s) what the function produces
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun name-has-s(r variable(s)):

what the function does with those variable(s)
```

end
Composing Table Operations

The table $t$ below represents four animals from the shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>3</td>
<td>true</td>
<td>48</td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>true</td>
<td>92</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>true</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Sasha&quot;</td>
<td>&quot;female&quot;</td>
<td>1</td>
<td>false</td>
<td>6.5</td>
</tr>
</tbody>
</table>

You have the following functions defined below (read them carefully!):

```
fun is-fixed(r): r["fixed"]
fun is-young(r): r["age"] < 4
fun nametag(r): text(r["name"], 20, "red")
```

For each table on the left, match it to the Circle of Evaluation that will produce it.

A table containing only Toggle and Sasha 1

A

- sort
- t "age" true

Produces a table of only young, fixed animals 2

B

- filter
- t is-fixed

Produces a table, sorted youngest-to-oldest 3

C

- build-column
- t "sticker" nametag

Produces a table with an extra column, named "sticker" 4

D

- filter
- t is-young

Produces a table containing Toggle and Sasha, in that order 5

E

- filter
- t is-fixed
- t is-young

- filter

Produces a table containing Toggle, Fritz, and Nori 6

F

- sort
- t is-young
- "pounds" false

Won't run: will produce an error (why?) 7

G

- sort
- build-column "age" true
- t "label" nametag

Produces a table with an extra "label" column, sorted youngest-to-oldest 8

H

- sort
- t "cuteness" false
The table below represents four animals from the shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>species</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>12</td>
<td>true</td>
<td>&quot;dog&quot;</td>
<td>48</td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>false</td>
<td>&quot;dog&quot;</td>
<td>92</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>true</td>
<td>&quot;dog&quot;</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Sunflower&quot;</td>
<td>&quot;female&quot;</td>
<td>2</td>
<td>false</td>
<td>&quot;cat&quot;</td>
<td>51.6</td>
</tr>
</tbody>
</table>

For each of the Circles of Evaluation below, convert them into Pyret code. What do you think the resulting table will be? The first one has been done for you.

<table>
<thead>
<tr>
<th>Circle of Evaluation</th>
<th>Pyret code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>sort(filter(t, is-dog), &quot;name&quot;, false)</code></td>
</tr>
<tr>
<td>2</td>
<td><code>sort(t, &quot;name&quot;, false)</code></td>
</tr>
<tr>
<td>3</td>
<td><code>build-column(filter(t, is-cat), &quot;large&quot;, is-big)</code></td>
</tr>
<tr>
<td>4</td>
<td><code>filter(sort(t, &quot;pounds&quot;, true), is-dog)</code></td>
</tr>
<tr>
<td>5</td>
<td><code>filter(build-column(t, is-cat), &quot;large&quot;, is-big)</code></td>
</tr>
</tbody>
</table>

Define the functions specified below by filling in the blanks. The first one has been done for you.

6. A function `is-cat`, which returns true if the animal is a cat.
   ```pyret```
   ```
   fun is-cat(r): r["species"] == "cat" end
   ```

7. A function `is-dog`, which returns true if the animal is a dog.
   ```pyret```
   ```
   fun is-dog(r): ____________________________ end
   ```

8. A function `is-big`, which returns true if an animal weighs more than 50 pounds.
   ```pyret```
   ```
   fun is-big(r): ____________________________ end
   ```
Planning Table Operations

The table $t$ below represents four animals from the shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>3</td>
<td>true</td>
<td>48</td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>true</td>
<td>92</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>true</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Sasha&quot;</td>
<td>&quot;female&quot;</td>
<td>1</td>
<td>false</td>
<td>6.5</td>
</tr>
</tbody>
</table>

You have the following functions defined below (read them carefully!):

```latex
fun is-female(r): r["sex"] == "female" end
fun is-young(r): r["age"] < 4 end
fun is-fixed(r): r["fixed"] end
```

For each prompt on the left, draw the Circle of Evaluation that will produce the desired table or display.

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Circle of Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Produce a Table containing all young, fixed animals</td>
<td>![Circle of Evaluation](filter is-young is-fixed t)</td>
</tr>
<tr>
<td>Produce a Table showing all fixed female animals, sorted by age</td>
<td></td>
</tr>
<tr>
<td>Produce a box-plot for all fixed female animals, showing the distribution of age</td>
<td><img src="filter" alt="Circle of Evaluation" /></td>
</tr>
<tr>
<td>Produce a pie-chart for all young, fixed animals, showing the distribution of sex</td>
<td><img src="filter" alt="Circle of Evaluation" /></td>
</tr>
</tbody>
</table>
Composing Table Operations: Order Matters!

The table $t$ below represents four animals from the shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>pounds</th>
<th>kilos</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>3</td>
<td>true</td>
<td>48</td>
<td></td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>true</td>
<td>92</td>
<td></td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>true</td>
<td>35.3</td>
<td></td>
</tr>
<tr>
<td>&quot;Sasha&quot;</td>
<td>&quot;female&quot;</td>
<td>1</td>
<td>false</td>
<td>6.5</td>
<td></td>
</tr>
</tbody>
</table>

You have the following functions defined below (read them carefully!):

```
fun is-female(r): r["sex"] == "female" end
fun kilograms(r): r["pounds"] / 2.2 end
fun is-heavy(r): r["kilos"] > 25 end
```

**TIP:** To help you complete this page, fill in the "kilos" values in the table above by dividing each animal's `pounds` column by 2.2.

For each table on the left, **match** it to the Circle of Evaluation that will produce it.

1. **A**
   - Produces a table containing Toggle, Nori and Sasha, with an extra column showing their weight in kilograms

2. **B**
   - Produces a table containing Sasha, Nori and Toggle (in that order)

3. **C**
   - Produces a table with an extra column for kilos, but only animals with more than 25 kilos

4. **D**
   - Won't run: will produce an error (if so, why?)

5. **E**
   - Produces a table containing Sasha, Toggle, and Nori (in that order)

6. **F**
   - Produces an empty table - with no animals at all!
The table $t$ below represents four animals from the shelter:

<table>
<thead>
<tr>
<th>name</th>
<th>sex</th>
<th>age</th>
<th>fixed</th>
<th>species</th>
<th>pounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Toggle&quot;</td>
<td>&quot;female&quot;</td>
<td>12</td>
<td>true</td>
<td>&quot;dog&quot;</td>
<td>48</td>
</tr>
<tr>
<td>&quot;Fritz&quot;</td>
<td>&quot;male&quot;</td>
<td>4</td>
<td>false</td>
<td>&quot;dog&quot;</td>
<td>92</td>
</tr>
<tr>
<td>&quot;Nori&quot;</td>
<td>&quot;female&quot;</td>
<td>6</td>
<td>true</td>
<td>&quot;dog&quot;</td>
<td>35.3</td>
</tr>
<tr>
<td>&quot;Sunflower&quot;</td>
<td>&quot;female&quot;</td>
<td>2</td>
<td>false</td>
<td>&quot;cat&quot;</td>
<td>51.6</td>
</tr>
</tbody>
</table>

For each of the Circles of Evaluation below, convert them into Pyret code. What do you think the resulting table will be? The first one has been done for you.

<table>
<thead>
<tr>
<th>Circle of Evaluation</th>
<th>Pyret code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>sort(build-column($t$, &quot;kg&quot;, kilograms), &quot;name&quot;, true)</td>
</tr>
<tr>
<td>2</td>
<td>sort(filter($t$, &quot;name&quot;, true), is-fixed)</td>
</tr>
<tr>
<td>3</td>
<td>build-column(filter($t$, &quot;kg&quot;, kilograms), &quot;name&quot;, true)</td>
</tr>
<tr>
<td>4</td>
<td>filter(sort($t$, &quot;name&quot;, true), is-female)</td>
</tr>
<tr>
<td>5</td>
<td>filter(filter($t$, is-female, is-fixed))</td>
</tr>
</tbody>
</table>

Define the functions specified below by filling in the blanks.

6. A function `is-fixed`, which returns true if the animals is fixed.
   
   ```pyret
   fun is-fixed($r$): ___________________________ end
   ```

7. A function `kilograms`, which returns the weight of the animal in kilograms (1kg is about 2.2lbs).
   
   ```pyret
   fun kilograms($r$): ___________________________ end
   ```

8. A function `is-female`, which returns true if an animals weighs more than 50 pounds.
   
   ```pyret
   fun is-female($r$): ___________________________ end
   ```
Use function composition to define the grouped samples below. Assume that the following helper functions are defined exactly the way they are in the `grouped_samples_starter_file.txt`.

<table>
<thead>
<tr>
<th>Subset</th>
<th>Code to define the subset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kittens</td>
<td><code>kittens = filter(filter(animals-table, is-cat), is-young)</code></td>
</tr>
<tr>
<td>Puppies</td>
<td></td>
</tr>
<tr>
<td>Fixed Cats</td>
<td></td>
</tr>
<tr>
<td>Fixed Kittens</td>
<td></td>
</tr>
<tr>
<td>Old Female Cats</td>
<td></td>
</tr>
<tr>
<td>Old Fixed Female Cats</td>
<td></td>
</tr>
<tr>
<td>Fixed Female Dogs</td>
<td></td>
</tr>
<tr>
<td>Old Fixed Female Cats</td>
<td></td>
</tr>
</tbody>
</table>
Displaying Data

Fill in the tables below, then use Pyret to make the following displays. Record the code you used. The first table has been filled in for you.

1) A **bar-chart** showing how many puppies are fixed or not.

<table>
<thead>
<tr>
<th>What Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
<tbody>
<tr>
<td>puppies</td>
<td>fixed</td>
<td>bar-chart</td>
</tr>
</tbody>
</table>

**code:** `bar-chart(filter(filter(animals-table, is-dog), is-young), "fixed")`

2) A **pie-chart** showing how many heavy dogs are fixed or not.

<table>
<thead>
<tr>
<th>What Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
</table>

**code:**

3) A **histogram** of the number of **weeks** it takes for a random sample of animals to be adopted.

<table>
<thead>
<tr>
<th>What Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
</table>

**code:**

4) A **box-plot** of the number of **pounds** that kittens weigh.

<table>
<thead>
<tr>
<th>What Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
</table>

**code:**

5) A **scatter-plot** of a random sample using **species** as the labels, **age** as the x-axis, and **weeks** as the y-axis.

<table>
<thead>
<tr>
<th>What Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
</table>

**code:**

6) Describe your own **grouped sample** here, and fill in the table below.

<table>
<thead>
<tr>
<th>What Rows?</th>
<th>Which Column(s)?</th>
<th>What will you Create?</th>
</tr>
</thead>
</table>

**code:**
<table>
<thead>
<tr>
<th><strong>Ask Questions</strong></th>
<th><strong>Consider Data</strong></th>
<th><strong>Analyze Data</strong></th>
<th><strong>Interpret Data</strong></th>
<th><strong>Question Type</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>How many of each species are fixed at the shelter?</strong>&lt;br&gt;What question do you have?</td>
<td><strong>Which Rows should we investigate?</strong> [All the rows, just the cats, fixed dogs, etc.]&lt;br&gt;What Column(s) do we need? [age, weight-in-kilograms, weeks, etc.]</td>
<td><strong>If you only need some rows, define your filter function here [Need help? Use the Design Recipe!]</strong>&lt;br&gt;<strong>If you need to make a new column, define your builder function here [Need help? Use the Design Recipe!]</strong>&lt;br&gt;What code will make the table or display you want?</td>
<td><strong>What did you find out? What can you infer?</strong>&lt;br&gt;<strong>What new question(s) does this raise?</strong></td>
<td><strong>(circle one):</strong>&lt;br&gt;Lookup&lt;br&gt;Arithmetic&lt;br&gt;Statistical</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Ask Questions</strong></th>
<th><strong>Consider Data</strong></th>
<th><strong>Analyze Data</strong></th>
<th><strong>Interpret Data</strong></th>
<th><strong>Question Type</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Are there more female cats than male cats at the shelter?</strong>&lt;br&gt;What question do you have?</td>
<td><strong>Which Rows should we investigate?</strong> [All the rows, just the cats, fixed dogs, etc.]&lt;br&gt;What Column(s) do we need? [age, weight-in-kilograms, weeks, etc.]</td>
<td><strong>If you only need some rows, define your filter function here [Need help? Use the Design Recipe!]</strong>&lt;br&gt;<strong>If you need to make a new column, define your builder function here [Need help? Use the Design Recipe!]</strong>&lt;br&gt;What code will make the table or display you want?</td>
<td><strong>What did you find out? What can you infer?</strong>&lt;br&gt;<strong>What new question(s) does this raise?</strong></td>
<td><strong>(circle one):</strong>&lt;br&gt;Lookup&lt;br&gt;Arithmetic&lt;br&gt;Statistical</td>
</tr>
</tbody>
</table>
Think back to when we defined grouped samples from the Animals Table, like "puppies", "old cats", etc. What grouped samples would be useful for your dataset? List a few of these in the first column. Then, for each one, what function will identify if a row \( r \) is in the subset?

<table>
<thead>
<tr>
<th>Grouped Sample</th>
<th>A function that returns true if a row ( r ) is in the subset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fun __________________________(r):</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>fun __________________________(r):</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>fun __________________________(r):</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>fun __________________________(r):</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>fun __________________________(r):</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
</tbody>
</table>
Write helper functions for your dataset, which you can use to define grouped samples. Since all helper functions will consume Rows, their Domains have already been filled in for you.

Directions: Define a function called __________________________, which consumes a Row of the __________________________
table and __________________________.

Contract and Purpose Statement

Every contract has three parts...

# function name :: (r :: Row) -> Boolean

#

Examples

Write some examples, then circle and label what changes...

elements:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition

Write the definition, giving variable names to all your input values...

fun __________________________(variable(s)):

______________________________ what the function does with those variable(s)

end
“Trust, but verify ...”

This page requires that you also open the Trust but Verify Starter File.

A Data Scientist gives you access to the following function:

```
# fixed-cats :: Table -> Table
# consumes a table of animals, and produces a table containing only cats that have been fixed, sorted from youngest-to-oldest
```

You can use the function, but you can’t see the code for it! How do you know if you can trust their code?

- You could make a verification subset that contains one of every species, and make sure that the function filters out everything but cats.
- You could make sure this subset has multiple cats not already ordered youngest-to-oldest, and make sure the function puts them in the right order.

1) What other qualities would this subset need to have?

2) Create your verification subset! In the space below, list the name of each animal in your subset.

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
</table>
A "helpful" Data Scientist gives you access to the following function:

```plaintext
# old-dogs-nametags :: Table -> Table
# consumes a table of animals, and produces a table containing only dogs 5 years or older, with an extra column showing their name in red
```

You can use the function, but you can't see the code for it! How do you know if you can trust their code?

1) What qualities would a verification subset need to have?

2) Create your verification subset! In the space below, list the name of each animal in your subset.

| Name |
We listened to more than 3 hours of US Congress testimony on facial recognition so you didn’t have to go through it

*Long story short: Models are ineffective, racist, dumb...*

**Katyanna Quach**  Wed 22 May 2019 // 23:50 UTC

**ANALYSIS**  Al experts, lawyers, and law enforcement urged US Congress to regulate the use of facial recognition technology during a hearing held by the House Committee on Oversight and Reform on Wednesday, May 22, 2019.

The technical issues and social impacts of using AI software to analyse images or videos are well known. There have been repeated reports of how inaccuracies lead to people being misidentified in research and in real life. San Francisco just passed an ordinance banning the local government using facial recognition technology.

In some cases, like the experiment conducted by the American Civil Liberties Union's (ACLU), a nonprofit based in New York, that showed Amazon Rekognition incorrectly matched members of the US Congress to criminal mugshots, the effects have been minimal. It's simply absurd for elected politicians to be wanted criminals. But what happens when the technology is turned on civilians who have less power?

At a hearing of the House Committee on Oversight and Reform on Wednesday, Joy Buolamwini, founder of Algorithmic Justice League, an activist collective focused on highlighting the shortcomings of facial recognition, found that commercial computer models struggled most when it came to recognizing women with darker skin. IBM’s system was incorrect for 34.7 per cent of the time when it came to identifying black women, she said...

The problem boiled down to biased training datasets, Buolamwini told the House committee. AI systems perform worse on data that they haven’t seen before. So, if most datasets mainly represent white men then it’s not surprising that they find it difficult when faced with an image of women of colour.

When it comes to databases of mugshots, however, the reverse is true. Black people are overrepresented in mugshot databases, explained Clare Garvie, Senior Associate at Georgetown University Law Center's Center on Privacy & Technology. If law enforcement are using these flawed models to target the group of people that it struggles to identify most then it will undoubtedly lead to police stopping and searching the wrong people. “It’s a violation of the first and fourth amendment,” Garvie said during the hearing.

**Law enforcement and lack of transparency**

Cedric Alexander, the former president of the National Organization of Black Law Enforcement Executives who was also a witness at the hearing, estimated that at least a quarter of law enforcement agencies across the US use facial recognition to some degree.

Police from Washington County and Orlando are an example of some bureaus that are using Rekognition. Michael Punke, Amazon’s VP of Global Public Policy, said at the time it has “not received a single report of misuse by law enforcement.” It’s difficult to verify that claim, however, considering that the police haven’t been transparent about how it’s used.

It’s all done in secrecy, according to testimony. Elijah Cummings, the chair of the Oversight Committee, said that 18 states had shared data like passport photos or driver licenses with the FBI without explicit consent. When the witnesses were pressed with questions on what kind of information law agencies share with one another, nobody knew.

Neema Guliani, senior legislative counsel for the ACLU, took a tough stance and called for a moratorium on the technology. She urged the committee to “take steps to halt the use of face recognition for law enforcement and immigration enforcement purposes until Congress passes a law dictating what, if any, uses are permissible and ensures that individuals’ rights can be protected.” Unregulated use of the technology could also potentially lead to an “Orwellian surveillance state,” where citizens are constantly tracked Guliani said.

In the opening statement, Cummings said there are about 50 million surveillance cameras in the US, and that half of all American adults are probably part of facial recognition databases and they don’t even know it.

Andrew Ferguson, professor of law at the University of the District of Columbia, agreed that the Congress needed to act now to prohibit facial recognition until Congress establishes clear rules. “Unregulated facial recognition should not be allowed to continue unregulated. It is too chilling, too powerful. The fourth amendment won’t save us. The Supreme Court is trying to make amendments but it’s not fast enough. Only legislation can react in real time to real time threats,” he warned.

Alexander was more cautious about a blanket ban on the technology, however. He believed that there were still ways that law enforcement could positively use facial recognition. “There is a place for the technology, but the police need to be trained properly. They can’t just be passed the technology by software companies.” Effective policing is about building relationships in the local community, and it can’t afford the effects of misidentifying people. How can we utilise the technology, whilst developing some standards?, he asked.

**Benchmark tests simply aren’t good enough**

The National Institute of Standards and Technology (NIST), a laboratory part of the US Department of Commerce, is currently conducting official benchmark tests for commercial facial recognition systems. But they need to be better, Buolamwini said. She brought up the issue of what she called “pale male datasets”. “The gold standard benchmark dataset is biased and can lead to a false understanding of progress,” she said.

Even if there was a facial recognition system with near-perfect accuracy in the testing phase, it doesn’t solve the problem that most data used by law enforcement is often grainy and low resolution. A recent report by Georgetown University found that in some cases police were even trying to match people by composite artist sketches.

“Faces maybe the final frontier of privacy,” Buolamwini said.

The hearing took place at the same time as Amazon shareholders tried to stop Rekognition being sold to law enforcement. The proposal was defeated, but the vote tallies were not immediately disclosed. © *The Register.*
1) Describe three concerns experts and activists have raised about Artificial Intelligence.

2) What are some solutions that would address these concerns?

3) How would you test whether or not a facial recognition system was equally accurate for everyone?
Threats to Validity can undermine a conclusion, even if the analysis was done correctly.

Some examples of threats are:

- **Selection bias** - identifying the favorite food of the rabbits won't tell us anything reliable about what all the animals eat.

- **Study bias** - If someone is supposed to assess how much cat food is eaten each day on average, but they only measure how much cat food is put in the bowls (instead of how much is actually consumed), they'll end up with an over-estimate.

- **Poor choice of summary** - Suppose a different shelter that had 10 animals recorded adoption times (in weeks) as 1, 1, 1, 7, 7, 8, 8, 9, 9, 10. Using the mode (1) to report what’s typical would make it seem like the animals were adopted much quicker than they really were, since 7 out of 10 animals took at least 7 weeks to be adopted.

- **Confounding variables** - Some shelter workers might prefer cats, and steer people towards cats as a result. This would make it appear that “cats are more popular with people”, when the real variable dominating the sample is what workers at the shelter prefer.
Some volunteers from the animal shelter surveyed a group of pet owners at a local dog park. They found that almost all of the owners were there with their dogs. From this survey, they concluded that dogs are the most popular pet in the state.

What are some possible threats to the validity of this conclusion?

The animal shelter noticed a large increase in pet adoptions between Christmas and Valentine’s Day. They conclude that at the current rate, there will be a huge demand for pets this spring.

What are some possible threats to the validity of this conclusion?
The animal shelter wanted to find out what kind of food to buy for their animals. They took a random sample of two animals and the food they eat, and they found that spider and rabbit food was by far the most popular cuisine!

Explain why sampling just two animals can result in unreliable conclusions about what kind of food is needed.

A volunteer opens the shelter in the morning and walks all the dogs. At mid-day, another volunteer feeds all the dogs and walks them again. In the evening, a third volunteer walks the dogs a final time and closes the shelter. The volunteers report that the dogs are much friendlier and more active at mid-day, so the shelter staff assume the second volunteer must be better with animals than the others.

What are some possible threats to the validity of this conclusion?
## Fake News

There are six separate, unrelated claims below, and ALL OF THEM ARE WRONG! Your job is to figure out why by looking at the data.

<table>
<thead>
<tr>
<th>Data</th>
<th>Claim</th>
<th>What's Wrong</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The average player on a basketball team is 6'1&quot;.</td>
<td>&quot;Most of the players are taller than 6'&quot;.</td>
</tr>
<tr>
<td>2</td>
<td>Linear regression found a positive correlation (r=0.42) between people's height and salary.</td>
<td>&quot;Taller people are more qualified for their jobs.&quot;</td>
</tr>
<tr>
<td>3</td>
<td><img src="image" alt="Graph" /></td>
<td>&quot;According to the predictor function indicated here, the value on the x-axis will predict the value on the y-axis 63.6% of the time.&quot;</td>
</tr>
<tr>
<td>4</td>
<td><img src="image" alt="Bar Chart" /></td>
<td>&quot;According to this bar chart, Felix makes up a little more than 15% of the total ages of all the animals in the dataset.&quot;</td>
</tr>
<tr>
<td>5</td>
<td><img src="image" alt="Histogram" /></td>
<td>&quot;According to this histogram, most animals weigh between 40 and 60 pounds.&quot;</td>
</tr>
<tr>
<td>6</td>
<td>Linear regression found a negative correlation (r= -0.91) between the number of hairs on a person's head and their likelihood of owning a wig.</td>
<td>&quot;Owning wigs causes people to go bald.&quot;</td>
</tr>
</tbody>
</table>
1) Using real data and displays from your dataset, come up with a misleading claim.

2) Trade papers with someone and figure out why their claims are wrong!

<table>
<thead>
<tr>
<th>Data</th>
<th>Claim</th>
<th>Why it's wrong</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Selection Bias or Biased Study?

The school newspaper ran an article stating that chicken was more popular than pork in the East Village.

**Kendell thinks the study was biased.**

*Would you rather eat pork or delicious crispy fried chicken?* That’s such a leading question! It encouraged people to pick chicken. I bet the results would have been different if they had asked about crispy bacon!

**Carson thinks the study suffered from selection bias.**

One of the survey sites was outside of a mosque?! Muslims don’t even eat pork!

Who’s right? How do you know?
Data scientists want to know if listening to music or podcasts reduces symptoms of stress in individuals. They conducted a study of 1,000 people who were brought into a laboratory office for testing. While wearing a heart-rate monitor, participants were asked to listen to either music or a podcast of their choosing while completing a series of complicated puzzles. The data scientists discovered that on average, participants who listened to music had a 5% lower heart rate while completing the tasks than those who listened to podcasts.

Before publishing their findings, the data scientists have asked you to review their claim. In the space below, indicate possible threats to validity faced by this study.
<table>
<thead>
<tr>
<th>Data Cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ask Questions</strong></td>
</tr>
<tr>
<td>What question do you have?</td>
</tr>
<tr>
<td><strong>Consider Data</strong></td>
</tr>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
<tr>
<td><strong>Analyze Data</strong></td>
</tr>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
</tr>
<tr>
<td><strong>Interpret Data</strong></td>
</tr>
<tr>
<td>What did you find out? What can you infer?</td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
</tr>
<tr>
<td>Ask Questions</td>
</tr>
<tr>
<td>---------------</td>
</tr>
<tr>
<td>What question do you have?</td>
</tr>
<tr>
<td>Consider Data</td>
</tr>
<tr>
<td>Which Rows should we investigate? (All the rows, just the cats, fixed dogs, etc.)</td>
</tr>
<tr>
<td>What Column(s) do we need? (age, weight-in-kilograms, weeks, etc.)</td>
</tr>
<tr>
<td>Analyze Data</td>
</tr>
<tr>
<td>If you only need some rows, define your filter function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>If you need to make a new column, define your builder function here (Need help? Use the Design Recipe!)</td>
</tr>
<tr>
<td>What code will make the table or display you want?</td>
</tr>
<tr>
<td>Interpret Data</td>
</tr>
<tr>
<td>What did you find out? What can you infer?</td>
</tr>
<tr>
<td>What new question(s) does this raise?</td>
</tr>
</tbody>
</table>
Design Recipe

Directions:

Contract and Purpose Statement

Every contract has three parts...

# function name :: Domain -> Range
#
# what does the function do?

Examples

Write some examples, then circle and label what changes...

examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end

Directions:

Contract and Purpose Statement

Every contract has three parts...

# function name :: Domain -> Range
#
# what does the function do?

Examples

Write some examples, then circle and label what changes...

examples:

function name (input(s)) is what the function produces

function name (input(s)) is what the function produces

end

Definition

Write the definition, giving variable names to all your input values...

fun function name (variable(s)):

what the function does with those variable(s)

end
Directions:

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name :: Domain -> Range
#
```

**Examples**

Write some examples, then circle and label what changes...

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
end
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
end
```

Directions:

**Contract and Purpose Statement**

Every contract has three parts...

```
# function name :: Domain -> Range
#
```

**Examples**

Write some examples, then circle and label what changes...

```
function name (input(s)) is what the function produces
function name (input(s)) is what the function produces
end
```

**Definition**

Write the definition, giving variable names to all your input values...

```
fun function name (variable(s)):

what the function does with those variable(s)
end
```
Contracts for Data Science

Contracts tell us how to use a function, by telling us three important things:

1. The Name
2. The Domain of the function - what kinds of inputs do we need to give the function, and how many?
3. The Range of the function - what kind of output will the function give us back?

For example: The contract \( \text{triangle} :: (\text{Number}, \text{String}, \text{String}) \rightarrow \text{Image} \) tells us that the name of the function is \( \text{triangle} \), it needs three inputs (a Number and two Strings), and it produces an Image.

With these three pieces of information, we know that typing \( \text{triangle}(20, \text{"solid"}, \text{"green"}) \) will evaluate to an Image.

<table>
<thead>
<tr>
<th>Name</th>
<th>Domain</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td># bar-chart</td>
<td>(table-name, column)</td>
<td>Image</td>
</tr>
<tr>
<td>bar-chart</td>
<td>(animals-table, &quot;species&quot;)</td>
<td></td>
</tr>
<tr>
<td># bar-chart-summarized</td>
<td>(table-name, labels, values)</td>
<td>Image</td>
</tr>
<tr>
<td>bar-chart-summarized</td>
<td>(count(animals-table, &quot;species&quot;), &quot;value&quot;, &quot;count&quot;)</td>
<td></td>
</tr>
<tr>
<td># beside</td>
<td>(left, right)</td>
<td>Image</td>
</tr>
<tr>
<td>beside</td>
<td>(circle(10, &quot;solid&quot;, &quot;black&quot;), square(50, &quot;solid&quot;, &quot;red&quot;))</td>
<td></td>
</tr>
<tr>
<td># box-plot</td>
<td>(table-name, column)</td>
<td>Image</td>
</tr>
<tr>
<td>box-plot</td>
<td>(animals-table, &quot;weeks&quot;)</td>
<td></td>
</tr>
<tr>
<td># build-column</td>
<td>(table-name, column, builder-function)</td>
<td>Table</td>
</tr>
<tr>
<td>build-column</td>
<td>(animals-table, &quot;kilos&quot;, kilograms)</td>
<td></td>
</tr>
<tr>
<td># circle</td>
<td>(radius, fill-style, color)</td>
<td>Image</td>
</tr>
<tr>
<td>circle</td>
<td>(50, &quot;solid&quot;, &quot;purple&quot;)</td>
<td></td>
</tr>
<tr>
<td># count</td>
<td>(table-name, column)</td>
<td>Number</td>
</tr>
<tr>
<td>count</td>
<td>(animals-table, &quot;species&quot;)</td>
<td></td>
</tr>
<tr>
<td># filter</td>
<td>(table-name, tester-function)</td>
<td>Table</td>
</tr>
<tr>
<td>filter</td>
<td>(animals-table, is-dog)</td>
<td></td>
</tr>
<tr>
<td># first-n-rows</td>
<td>(table-name, num-rows)</td>
<td>Table</td>
</tr>
<tr>
<td>first-n-rows</td>
<td>(animals-table, 15)</td>
<td></td>
</tr>
<tr>
<td># histogram</td>
<td>(table-name, labels, values, bin-size)</td>
<td>Image</td>
</tr>
<tr>
<td>histogram</td>
<td>(animals-table, &quot;species&quot;, &quot;weeks&quot;, 2)</td>
<td></td>
</tr>
<tr>
<td># image-bar-chart</td>
<td>(table-name, values, draw-function)</td>
<td>Image</td>
</tr>
<tr>
<td>image-bar-chart</td>
<td>(animals-table, &quot;name&quot;, &quot;pounds&quot;, f)</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Signature</td>
<td>Returns</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td><code>image-histogram</code></td>
<td><code>table-name, String, String, (Row -&gt; Image)</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>image-pie-chart</code></td>
<td><code>table-name, String, (Row -&gt; Image)</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>image-scatter-plot</code></td>
<td><code>table-name, String, String, String, (Row -&gt; Image)</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>line-graph</code></td>
<td><code>table-name, String, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>lr-plot</code></td>
<td><code>table-name, String, String, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>mean</code></td>
<td><code>table-name, Number</code></td>
<td><code>Number</code></td>
</tr>
<tr>
<td><code>median</code></td>
<td><code>table-name, String</code></td>
<td><code>Number</code></td>
</tr>
<tr>
<td><code>modes</code></td>
<td><code>table-name, String</code></td>
<td><code>List</code></td>
</tr>
<tr>
<td><code>modified-box-plot</code></td>
<td><code>table-name, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>multi-bar-chart</code></td>
<td><code>table-name, String, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>pie-chart</code></td>
<td><code>table-name, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>pie-chart-summarized</code></td>
<td><code>table-name, String, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td><code>r-value</code></td>
<td><code>table-name, String, String</code></td>
<td><code>Number</code></td>
</tr>
<tr>
<td><code>random-rows</code></td>
<td><code>table-name, Number</code></td>
<td><code>Table</code></td>
</tr>
<tr>
<td><code>rectangle</code></td>
<td><code>Number, Number, String, String</code></td>
<td><code>Image</code></td>
</tr>
<tr>
<td>Name</td>
<td>Domain</td>
<td>Range</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td># regular-polygon</td>
<td>:: (Number, Number, String, String) -&gt; Image</td>
<td>vertices, size, fill-style, color</td>
</tr>
<tr>
<td>regular-polygon(25, 5, &quot;solid&quot;, &quot;purple&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># right-triangle</td>
<td>:: (Number, Number, String, String) -&gt; Image</td>
<td>leg1, leg2, fill-style, color</td>
</tr>
<tr>
<td>right-triangle(50, 60, &quot;outline&quot;, &quot;blue&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># row-n</td>
<td>:: (Table, Number) -&gt; Row</td>
<td>table-name, index</td>
</tr>
<tr>
<td>row-n(animals-table, 2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># scatter-plot</td>
<td>:: (Table, String, String, String) -&gt; Image</td>
<td>table-name, labels, xs, ys</td>
</tr>
<tr>
<td>scatter-plot(animals-table, &quot;name&quot;, &quot;pounds&quot;,&quot;weeks&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># sort</td>
<td>:: (Table, String, Boolean) -&gt; Table</td>
<td>table-name, column, ascending</td>
</tr>
<tr>
<td>sort(animals-table, &quot;species&quot;, true)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># square</td>
<td>:: (Number, String, String) -&gt; Image</td>
<td>size, fill-style, color</td>
</tr>
<tr>
<td>square(50, &quot;solid&quot;, &quot;red&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># stacked-bar-chart</td>
<td>:: (Table, String, String) -&gt; Image</td>
<td>table-name, group, subgroup</td>
</tr>
<tr>
<td>stacked-bar-chart(animals-table, &quot;species&quot;, &quot;sex&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># star</td>
<td>:: (Number, String, String) -&gt; Image</td>
<td>radius, fill-style, color</td>
</tr>
<tr>
<td>star(50, &quot;solid&quot;, &quot;red&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># stdev</td>
<td>:: (Table, String) -&gt; Number</td>
<td>table-name, column</td>
</tr>
<tr>
<td>stdev(animals-table, &quot;pounds&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># string-contains</td>
<td>:: (String, String) -&gt; Boolean</td>
<td>haystack, needle</td>
</tr>
<tr>
<td>string-contains(&quot;hotdog&quot;, &quot;dog&quot;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td># text</td>
<td>:: (String, Number, String) -&gt; Image</td>
<td>message, size, color</td>
</tr>
<tr>
<td>text(&quot;Zari&quot;, 85, &quot;orange&quot;)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
These materials were developed partly through support of the National Science Foundation (awards 1042210, 1535276, 1648684, and 1738598) and are licensed under a Creative Commons 4.0 Unported License. Based on a work at www.BootstrapWorld.org. Permissions beyond the scope of this license may be available by contacting contact@BootstrapWorld.org.