

5.1.4.4 - Getting Elements by Position

Let's look at a new analysis question: the events company recently ran an advertising campaign on `web.com`, and they are curious whether it paid off. To do this, they need to determine how many sales were made by people with `web.com` email addresses.

From the class's Task Plan, we are using `string-split-all`. The authors also show us `string-split`, which splits the first input String at the first occurrence of the second input String. Either will work for this case...

```
>>> string-split-all("this-has-hyphens", "-")
[list: "this", "has", "hyphens"]

>>> string-split("bonnie@pyret.org", "@")
[list: "bonnie", "pyret.org"]
```

1. **Pair Program:** Make up a string and experiment with both functions.

```
string-split-all(_____, _____)
#returns:
```

```
string-split(_____, _____)
#returns:
```

We use the function `get` to return the element at the index. This is more like a “method” in object-oriented programming so we use it like this:

```
>>> string-split("bonnie@pyret.org", "@").get(1)
"pyret.org"
```

2. **Ask the Group:** Why did this return a String? (Hint: what kind of data is on the left side of the `“ .get(1)”`)

Our group said:

Do Now!

Why do we use **1** as the input to **get** if we want the second item in the list?

3. **Pair Program:** Experiment with **get** to answer the above question:

_____.get(_____) #returns:	_____.get(_____) #returns:
_____.get(_____) #returns:	_____.get(_____) #returns:

Our group said:

Make a function called **web-com-address** which takes in an email **String** and determines whether it ends in "web.com"

4. **Whiteboard:** Contract

web-com-address :: _____ -> _____

5. **Whiteboard:** Purpose Statement. Restate the contract in your own words

Examples:

Here are just the results for some emails:

"bonnie@pyret.org" is false

"lmcqueen@web.com" is true

6. **Whiteboard then Pair Program:** Rewrite results to show how **string-split-all** or **string-split**, and **.get()** can be used to get the answer:

"bonnie@pyret.org" is _____
"lmcqueen@web.com" is _____

There is an error on this page!

7. **Pair Program:** “Write the function already!” -- That’s you when you are diligent about writing examples...

```
fun web-com-address(____):  
  
end _____
```

8. **Pair Program:** Make sure the code compiles and the examples pass!

- Ok
 Honestly we didn’t do it

9. **Pair Program:** Instead of making another function, use `length()` (which takes in a `List` and returns the number of elements) and `filter` to write the solution to the original problem in a single line of code. The question again:

“How many sales were made by people with the `web.com` address?”
(The answer is 2.)

Our list of emails is defined in the starter file as `emails...`

```
#our single line of code:
```

10. **Reflect:** What have we learned so far about lists and some of the things we can do with them?

11. Extra problem if we have time...

Exercise

What happens if there is a malformed email address string that doesn’t contain the `@` string? What would happen? What could you do about that?