WEEK 2:
VARIABLES

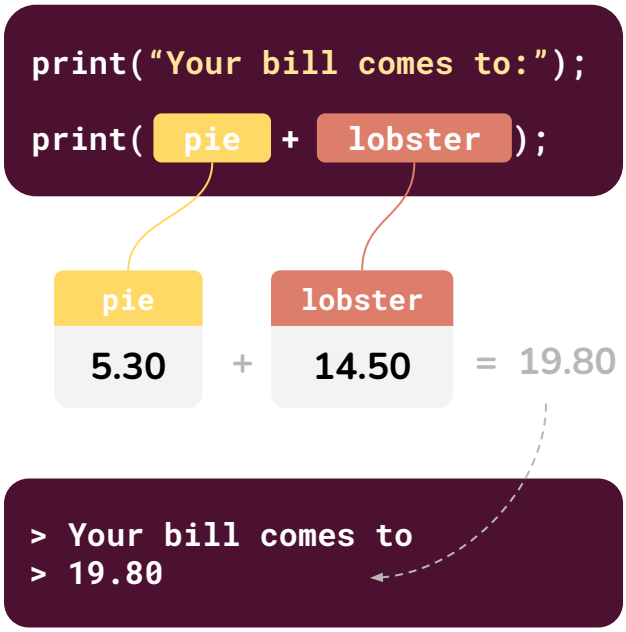
What are variables?

This week we looked at *variables*, a very fundamental part of Python programming. Variables are a way of associating a name with a particular value in the code. For example, we could tell Python to replace the name PI with 3.1415 whenever it sees it.

To assign a variable to a value in Python, we use the equals operator (=) and give a *name* on the left and a *value* on the right.

For example, the two variables seen on the right could be assigned like so:

```
pie = 5.30
lobster = 14.50
```



Experiment with creating variables in Python:

1. Can you create a variable called "PI" with the value 3.14159? Can you use it in a calculation?
2. What happens if you try create a variable with the name "1test"? Why do you think you get an error?
3. Make a variable called "name" with the value of your name. Try print this out this variable, to show your name.
4. Following on from the previous task, add another variable called "second_name" with your surname. Using concatenation (the + operator with strings) can you print out your full name?
5. Recreate the example on the top right of this page -- the restaurant bill example. Can you create say, 5 variables, all with different menu items and prices, and summate them for a total value?

Assigning values to variables

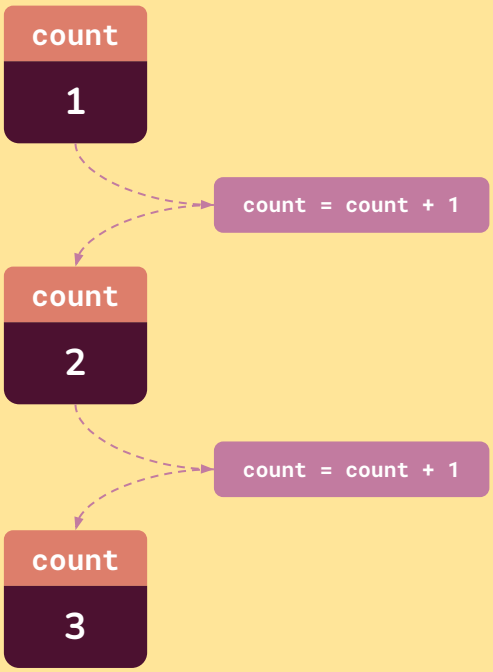
If variables don't change their value ever, we usually call these *constants*. For example, PI is a constant -- it never changes its value. However, variables can have their value changed (hence the name "variables"!). To do this, we can *assign* a variable a new value.

To update (or "assign") the value of a variable, it's pretty easy. We just use the equals (=) operator to assign a new value, just like how we create it!

For example, if we had a variable with the name "total", we could set its value like so:

```
# Total is initially 5:
total = 5

# But now, it's 4:
total = 4
```



Experiment with assigning variables in Python:

1. Create a variable called **total**, and set its value to 5. Print its value to the console to make sure. Now change its value with something like "total = 4". What happens to the output in the console, if you print it out again?
2. Create a variable called **count**, give it the value 1, and print its value to the console. Underneath, use "count = count + 1" to add one to its value. Print out this new value. What happens to **count**? Can you get this to continue all the way up to 10?
3. Alter the code in the previous task to show a countdown from 10 to 1.
4. Create three variables: **lobster**, **pie** and **steak**. Set their values to what you believe each one would cost in a restaurant (in pounds). Create another variable called total. Using three variable assignments, can you get total to equal the sum of lobster, pie and steak? (hint: add to the total variable)
5. Investigate "compound assignment operators" in Python. Create a variable **test** and set its value to 5, then:
    a. Try "test += 5". What do you think += does here? What about -=?
    b. Try "test *= 5". What do you think *= does here? What about /=?

NEXT WEEK:

SELECTION + ITERATION!