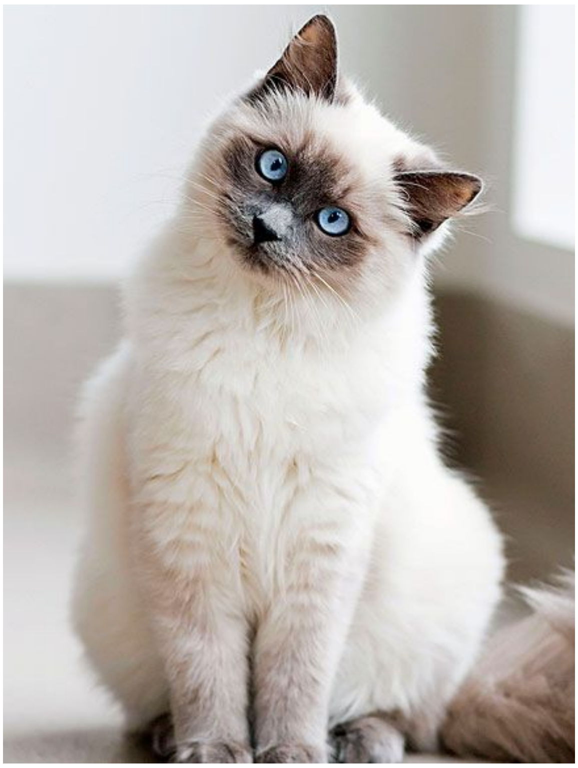


MEET Y1 - Module 2 - Lab 2

Variables and Errors

In this lab, you will learn how to assign values to variables, how to change the values of a variable, and how to recognize errors in code.

Follow these instructions:

0. Open a **Linux** terminal by double clicking on this icon:

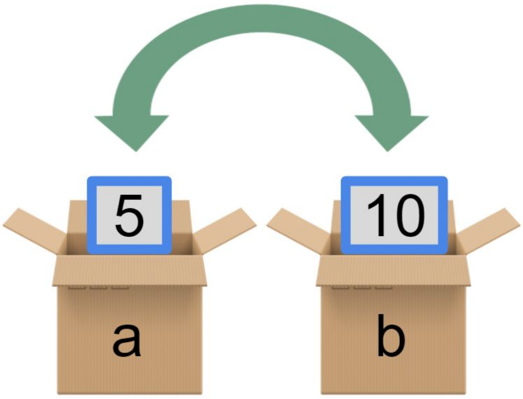
1. Cd into the correct folder and type `startlab`. Enter your username and module ***2***!

2. Type `idle3 &` to open **IDLE3**. A window should pop up like last time.

3. Define two variables, `a = 5` and `b = 10`. We want to switch the values of `a` and `b` so `a = 10` and `b = 5`.

4. Type the following to try to switch a and b:

```
>>> a = b
>>> b = a
```

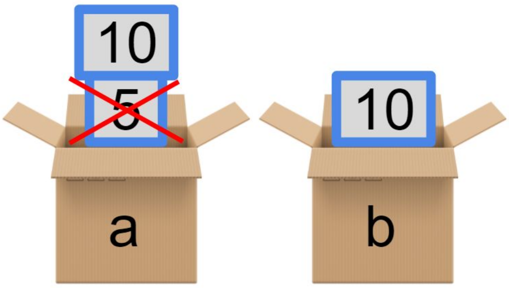


5. What are the new values of a and b? Find out by typing a and pressing the Enter key.

```
>>> a

______
>>> b

______
```

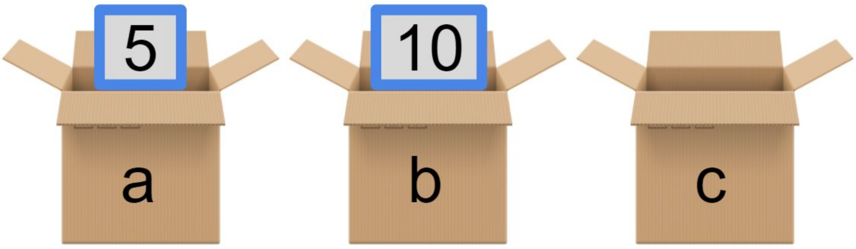
6. What happened? Type the following:

```
>>> a = 5
>>> b = 10
>>> a = b
>>> a

_______
>>> b

_______
```

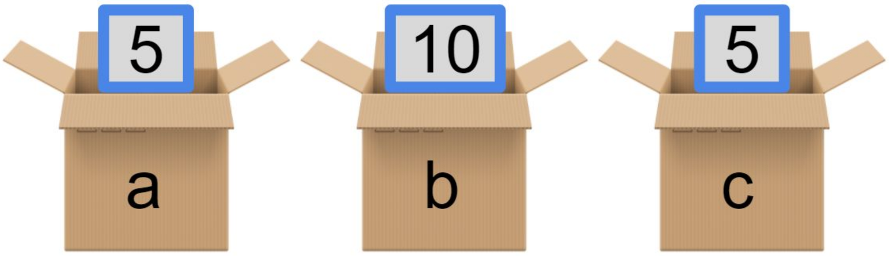


When we set `a = b`, it is like we took the `10` out of the `b` box and put it over the `5`. Then, when we did `b = a`, `a` was already `10`.

7. Now let’s add a third box called `c`.

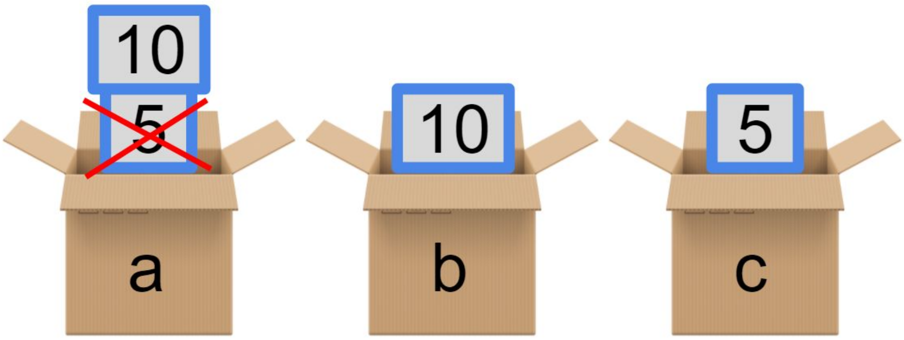


8. Let’s use `c` to hold the value of `a`. Type `c = a`.



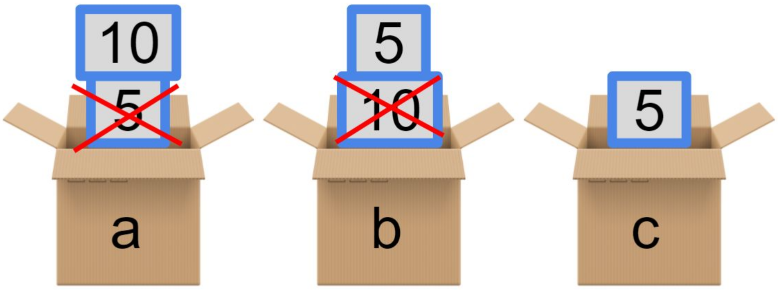
9. Now let `a` be equal to the value of `b`. Type `a = b`.

```
>>> _________
```



10. Finally, let `b` be equal to the value of `c` by typing `b = c`.

```
>>> _________
```



11. You did it! Find the values of `a` and `b`.

```
>>> a

_______
>>> b

_______
```

12. Strings

Now set a variable named four equal to the string 4.

```
>>> four = ‘4’
```

13. What happens when you type `print(four*3)`?

____

Why did this happen?

14. Now set a variable called `five` equal to the integer `4`.

```
>>> five = 4
>>> print(five)
________
>>> print(five*3)
________
```

15. Talk with the person next to you to figure out why `print(five*3)` did not output `15`.

16. Errors

Read the intention of each program. Type the bad code into **IDLE3** to see what the bad code output is. Then write how the correct code in the fourth box. The first table is filled out as an example.

17.

Intention	Print out “My name is student”

Bad Code	`>>> my_name = ‘student’`
	`>>> print(“My name is ’ + ‘my_name”)`

Bad Code Output	`My name is ’ + ‘my_name`
Correct code	`>>> print(“My name is ” + my_name)`

18.

Intention	Print out “Hi, student”
Bad Code	>>> my_name = ‘student’ >>> print(“Hi,” + myName’)
Bad Code Output	
Correct code	

19.

Intention	Print out “I am 15 years old”
Bad Code	>>> my_age = 15 >>> print(‘I am ‘ + my_age + ‘years old’)
Bad Code Output	
Correct code	

20.

Intention	Print out “I am 15 years old”
Bad Code	>>> my_age = 15 >>> print(‘I am ‘ + my_age + ‘years old’)
Bad Code Output	
Correct code	

21.

Intention	Print out the total score

Bad Code	`>>> score = 1`
	`>>> total = score + (count * 2)`
	`>>> print(total)`

Bad Code Output	
Correct code	

22. BONUS

Make a square with Turtle with a variable as a side length.

23. Go to *File* → *New File*
24. Then go to *File* → *Save As…*
25. Name your file `SquareinTurtle.py` .
26. Copy the following into the file and then press F5 to draw a square:

```
import turtle

#Insert variable here

turtle.penup() #Pick up the pen so it doesn’t draw
turtle.goto(0,0) #Move the turtle to the
                       #position, -200, -100, on
                       #the screen
turtle.pendown() #Put the pen down to start
                 #drawing
#Draw the square:
turtle.goto(0,100)
turtle.goto(100,100)
turtle.goto(100,0)
turtle.goto(0,0)

# ...and end it before the next line.
turtle.mainloop()
```

27. Now add a variable named `side_length` and set it equal to 100.
28. Replace the `turtle.goto(0,100)` with `turtle.goto(0,side_length).`
29. Replace the other `turtle.goto()` lines using `side_length`.
30. Press F5 to run your code! The square should appear just as before.

31. Change the value of `side_length` to make the square smaller or larger.

When you are done, don't forget to type `endlab` to finish!