Aaron Eberly

CNIT 15501

Review Boolean Operators!

The three Boolean operations are **and**, **or**, and **not**.

This small chart sums up the **and** operator:

|  |  |  |
| --- | --- | --- |
| **Ice Cream** | **Chocolate Chips** | **Chocolate Chip Ice Cream** |
| I don’t have ice cream | I don’t have chocolate chips | I can’t make chocolate chip ice cream |
| I don’t have ice cream | I have chocolate chips | I can’t make chocolate chip ice cream |
| I have ice cream | I don’t have chocolate chips | I can’t make chocolate chip ice cream |
| I have ice cream | I have chocolate chips | I can mix the ice cream ***and***chocolate chips....   to make chocolate chip ice cream |

X and y: True if x and y are true.

X or y: True if x or y are true.

Not x: True if x is false.

Order of operations for Boolean operators: Not, And, Or

Boolean order of operations example:

(True and False) or (not(True) and False) or True or not(False)

First evaluate the not expressions: (True and False) or (False and False) or True or True

Then evaluate the and statements: False or False or True or True

Then, evaluate the or statements from left to right: False or True or True

True or True

Answer: True

F-string

print(f“*Code you wish to print*”)

When you do an f string instead of a normal print line, you may put {} in the string. You can calculate different expressions inside of these {}, and it will be converted to a string.

Example:

x = True

print (f“x is {x}”)

Output: x is True

Notice how even though x isn’t a string, because I used an f-string, and put it in {}, it still printed correctly.

f-strings make it simpler to include variables in print lines.

Random Numbers

First, you must import random

random.random() gives you a number between 0 and 1. (Technically, between 0 and 0.999999…)

Using this logic, the following line will print a random number from 1 to 10.

print(1 + random.random() \* 2)

Remember, you can also use random.randint(1, 11) to print a number between 1 and 10.

While Loops

A while loop designates a segment of code that will continually repeat until the condition is false.

while(*condition*):

Example:

x = 0

while (x < 5):

print(“\*”)

x += 1

The above code would print the following:

\*

\*

\*

\*

\*

Notice how x is increased in the while loop. If we did not increase x, then the loop would continue on forever, since x would always be lower than 5.

After the code runs, the value of x will be 5.

Turtle Module

import turtle

This allows you to draw pictures in python! Remember to import turtle!

Now, you need to create a cursor with the following line of code:

yertle = turtle.Turtle

I used the same Yertle, but you can choose different names for your cursor.

You can use the following lines to draw with the turtle module:

yertle.pendown() This makes your cursor ready to draw

yertle.penup() This allows you to move the turtle without drawing. Remember to use pendown if you want to draw again.

yertle.forward(x) This moves your cursor forwards. If your pen is down, it will draw a line. For x, enter the number of pixels that you want to move the cursor.

yertle.right(X) This will turn the cursor to the right. X is the degree of the turn. For example, Yertle.right(90) will turn in a 90 degree angle.

This is only a sample of what the turtle module can do. A quick google search will reveal code that allows you to change colors, fill in objects, draw stars, and more!