Part 5 Loops

Notes, examples, and the assignment at : **https://code.actor**

"For" loop

A For loop repeats a block of code a specific number of times. The current count of the current loop is stored in an "induction variable." You can use the value of the induction variable in logic inside the loop. Each lap around the loop is called an "iteration."

```
3
4  print("\n Looping 5 times. From 0 to 4")
5  for my_variable in range(5):
6      print(my_variable)
7
```

```
Looping 5 times. From 0 to 4
0
1
2
3
4
```

```
2  penup()
3  setpos(-50,20)
4  pendown()
5
6  for x in range(8):
7      forward(50)
8      left(45)
9
10 penup()
11 setpos(50,150)
12
```

"While" loop

A While loop repeats a block of code as long as a specific condition remains true. It's like an "if" statement with a loop. It iterates zero or more times because the "if" could be false already.

```
2
3  favorite_animal = ""
4  favorite_animal = input("What is the best animal? ")
5
6  while (favorite_animal != "dog"):
7      print("Wrong! Try again!")
8      favorite_animal = input("What is the best animal? ")
9
10
11 print("\nyay!\n\n")
```

```
What is the best animal? cat
Wrong! Try again!
What is the best animal? fish
Wrong! Try again!
What is the best animal? mouse
Wrong! Try again!
What is the best animal? dog

yay!
```

```
2  balloon_size = 1
3  balloon_max = 10
4
5  while balloon_size < balloon_max:
6      print(f"The balloon is now size {balloon_size}!")
7      balloon_size += 2
8
9  print("POP! The balloon popped!")
10
```

```
The balloon is now size 1!
The balloon is now size 3!
The balloon is now size 5!
The balloon is now size 7!
The balloon is now size 9!
POP! The balloon popped!
```

The "Do-While" loop always executes at least once and checks for true at the end of the loop. Python doesn't have a Do-While loop.

The "For-Each" loop iterates over a list of items (sometimes called an "array") and executes one loop *iteration* per item.