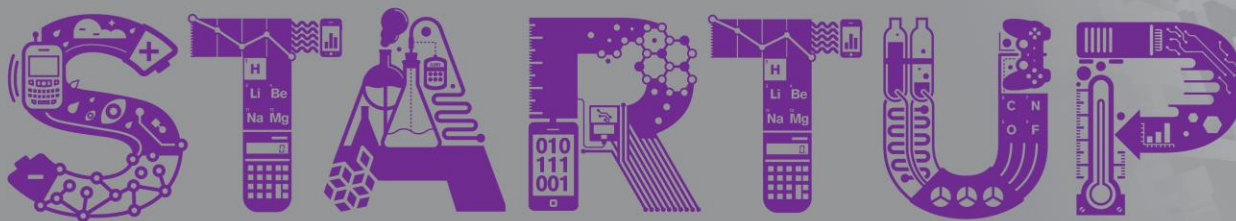


# Bloomberg



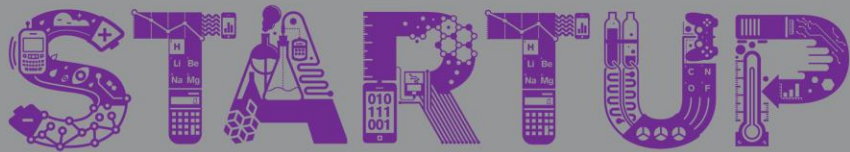
Copyright 2017 Bloomberg L.P.

Licensed under Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0)

Non-commercial use only. Modifications must not be distributed.

Please see <http://creativecommons.org/licenses/by-nc-nd/4.0/>

Bloomberg



# Python Lesson 8

---

## For Loops



# Review Lists

What does this code do?

```
colors = ["red", "blue", "orange", "purple"]  
print(colors[1])
```



# Review Lists

What does this code do?

```
colors = ["red", "blue", "orange", "purple"]  
print(colors[1])
```

blue



# Review Lists

What does this code do?

```
numbers = [3, -5, -7, 1]
sum = 0;
for number in numbers:
    sum = sum + number
print(sum)
```



# Review Lists

What does this code do?

```
numbers = [3, -5, -7, 1]
sum = 0;
for number in numbers:
    sum = sum + number
print(sum)
```

-8

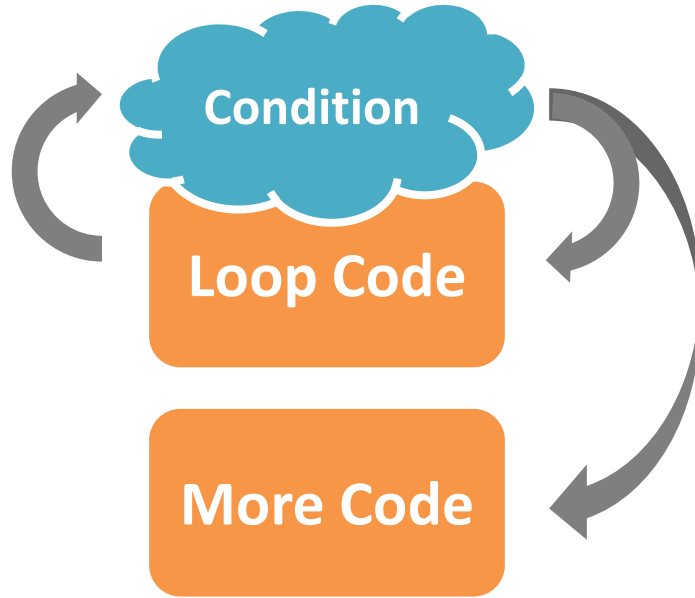


# For Loops: Introduction

For loops allow us to repeat a block of code a certain number of times.

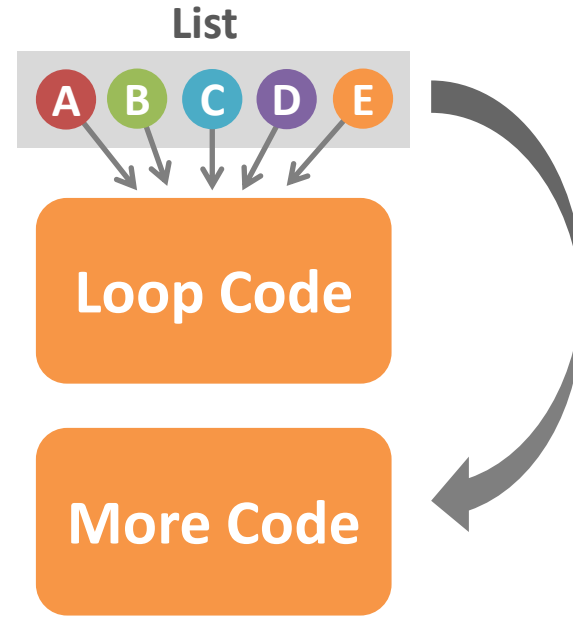


## While Loops



VS

## For Loops







# Anatomy of a For loop

**for** indicates that the block of code will repeat in a loop

The variable that we will be using to access each item

The list that we are going through

```
numbers = 5, -5, -7, 11
sum = 0
for number in numbers:
    sum = sum + number
print(sum)
```

The loop code goes here. It must all be indented the same amount.

The : tells the program you are starting the block of code for the for loop



## range(start,end)

- The function range allows us to generate a list of numbers.
- **start**: the number that you want to start at (inclusive)
- **end**: the number that you want to end at (exclusive)

**Ex:** range(1,5) gives you 1, 2, 3, 4



# Iterate through a range of numbers

We can use **for loops** to execute the code block for each number in the range of numbers.

```
for num in range(1,11):  
    print(num)
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10



# Iterate through a range of numbers

We get the same output using either **for loops** or **while loops**!

```
for num in range(1,11):  
    print(num)
```

```
counter = 1  
while counter <= 10:  
    print(counter)  
    counter = counter + 1
```



# Iterate through a string

We can also use **for loops** to execute the code block for each character inside of a string.

```
normal = "apple"  
reverse = ""  
for char in normal:  
    reverse = char + reverse  
print(reverse)
```

elppa



# Group Practice



# Recap

- **For loops** let you execute a block of code once for every
  - **Item** in a **list**
  - **Number** in a **range** of numbers
  - **Character** in a **string**



# Python Explorer

How can we use **for loops** in Python Explorer?