Introduction To Python Programming

Introduction to Python programming Course Outline

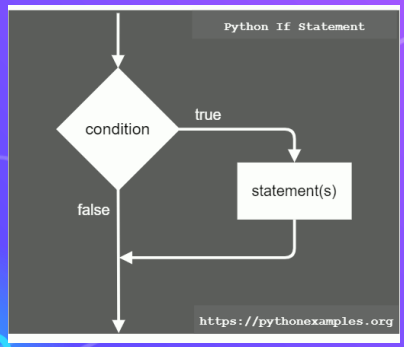
- Intro to Computer Science
- Environment Setup (Anaconda)
- Command Line
- Conda & pip package managers
- Jupyter Notebook
- Input & Output
- Variables
- Data types
- Numbers & Math
- Boolean & Comparison and Logic
- **If Conditions**
- For & While Loops
- Strings
- Lists
- Tuples
- Sets
- Dictionaries
- Advanced If and Loops
- List Comprehensions
- Dictionary Comprehensions
- Exceptions
- File Handling
- Functions
- Built-in functions & Operators (zip, enumerate, range, …)
- Map, Filter, Reduce
- Lambda Expressions

If Conditions

Previously, when we run our code, it would execute all statements in order

It's time to apply flow control

If statements allow us to control the flow of the code based on a certain condition




```
person = 'George'

if person == 'Sammy':
    print('Welcome Sammy!')
elif person =='George':
    print('Welcome George!')
else:
    print("Welcome, what's your name?")

# Welcome George!
```

Quiz Time!

What will be the output of the following if statements:

Q1

```
number1 = 5
number2 = 1
if (number1 + number2) < 3:
  print("Sloths")
else:
  print("Cats")
```

- A. Sloths
- B. Cats
- C. No print

Q2

```
num = 15
if num / 7 == 7:
  print("It divides by 7")
elif num / 3 == 5:
  print("It divides by 3")
else:
  print("Doesn't divide")
```

- A. It divides by 7
- B. It divides by 3
- C. Doesn’t divide

Practice #1

Write a program that asks the user to input a string ‘name’ and a number ‘num’ (either 0 or 1) and return "Hello" + name if num is 1, otherwise return "Bye" + name

Solution Practice #1

Write a program that asks the user to input a string ‘name’ and a number ‘num’ (either 0 or 1) and return "Hello" + name if num is 1, otherwise return "Bye" + name

```
name = input("Please Enter your name")
num = int(input("Please enter 0 or 1 "))

if num == 1:
    print ("Hello "+name)
else:
    print("Bye "+ name)
```

Practice #2

Write a program that asks the user to input a number ‘num’ and print “even number” if the number is even and print “odd number” if the number is odd

Solution Practice #2

Write a program that asks the user to input a number ‘num’ and print “even number” if the number is even and print “odd number” if the number is odd

```python
x = int(input('enter ur number: '))

if x % 2 == 0:
    print('even')
elif x % 2 != 0:
    print('odd')
```

Introduction to Python programming Course Outline

- Intro to Computer Science
- Environment Setup (Anaconda)
- Command Line
- Conda & pip package managers
- Jupyter Notebook
- Input & Output
- Variables
- Data types
- Numbers & Math
- Boolean & Comparison and Logic
- If Conditions
- **For & While Loops**
- Strings
- Lists
- Tuples
- Sets
- Dictionaries
- Advanced If and Loops
- List Comprehensions
- Dictionary Comprehensions
- Exceptions
- File Handling
- Functions
- Built-in functions & Operators (zip, enumerate, range, …)
- Map, Filter, Reduce
- Lambda Expressions

For Loops

Loops are used to repeat a certain block of code

For loops can repeat the code for a known number of times

They should be used when we know how many times we need the code to repeat

```python
for i in range(10):
  print("i =", i)
  # i = 0
  # i = 1
  # i = 2
  # i = 3
  # i = 4
  # i = 5
  # i = 6
  # i = 7
  # i = 8
  # i = 9
```

```python
# range(start, stop, step)
for i in range(2, 10, 2):
  print("i =", i)
  # i = 2
  # i = 4
  # i = 6
  # i = 8
```

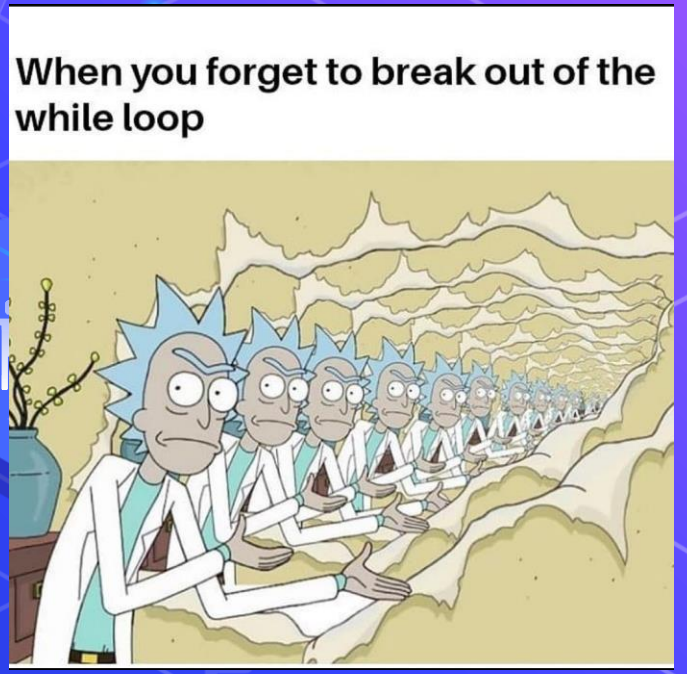
While Loops

While loops keep repeating the code while a given condition is True

It will break out of the loop once the condition turns to False

```
x = 20
while x > 0:
  print("x =", x)
  x -= 5 # Update

  # x = 20
  # x = 15
  # x = 10
  # x = 5
```



Make sure your condition will turn False after a while, or you're getting stuck with an infinite loop!

Quiz Time!

What will be the output of the following statements:

Q1

```
for num in range(2,-5,-1):
    print(num)
```

- A. 2, 1, 0
- B. 2, 1, 0, -1, -2, -3, -4, -5
- C. 2, 1, 0, -1, -2, -3, -4

Q2

```
counter = 1
sum = 0
while counter <= 6:
    sum = sum + counter
    counter = counter + 2
print(sum)
```

- A. 12
- B. 9
- C. 7

Practice #4

- Write a program that asks the user to enter his first and second name in one string then print each letter of the name in separate line using a loop.

Introduction to Python programming Course Outline

- Intro to Computer Science
- Environment Setup (Anaconda)
- Command Line
- Conda & pip package managers
- Jupyter Notebook
- Input & Output
- Variables
- Data types
- Numbers & Math
- Boolean & Comparison and Logic
- If Conditions
- For & While Loops
- Strings
- **Lists**
- Tuples
- Sets
- Dictionaries
- Advanced If and Loops
- List Comprehensions
- Dictionary Comprehensions
- Exceptions
- File Handling
- Functions
- Built-in functions & Operators (zip, enumerate, range, …)
- Map, Filter, Reduce
- Lambda Expressions

Lists

Lists are the most common data structure in Python

You can store multiple values (elements) inside a single variable

Unlike other programming languages, Python lists can have elements of different types

```
my_list = ['A string', 23, 100.232 , 'p', True]

print(my_list[0])   # 'A string'
print(my_list[1])   # 23
print(my_list[2])   # 100.232
print(my_list[3])   # 'p'
print(my_list[4])   # True
```

Lists

List elements can be lists too!

```
my_list = [[1,2,3], [4,5,6], [7,[8,9]]]

print(my_list[0])        # [1,2,3]
print(my_list[0][1])     # 2
print(my_list[2][1][1]) # 9
```

Lists

Some important built-in functions:

- ***.append() and .extend()*** to append values and join list respectively.
- ***.insert() to insert a value in a specific index.***
- ***.remove() and .pop() to remove element from a list by value or index respectively.***
- ***.index() to find the first position of first matching item to the passed value.***
- ***.sort() to sort the items in the list in ascending or descending order.***

Practice #7

- Ask the user to enter five numbers then print the max and min number of those numbers

Solution Practice #7

- Ask the user to enter five numbers then print the max and min number of those numbers

```
numbers = list()
for i in range(5):
    numbers.append(float(input("Enter your employee name: ")))
print("The max number is" , max(numbers))
print("The min number is" , min(numbers))
```