PYTHON MY NOTES

URL used to get logo in ascii format in this course- <https://patorjk.com/software/taag/#p=display&f=Graffiti&t=Type%20Something%20>

Important useful packages

Pypi.org

1. To print the output in the console use the function print(). The function print should be in lower case,

print (“Hello World!”) //Hello World!

Note: We should not leave any space at the start of the line in python and semi colon is not required at the end.

1. To print the output with quoted text in it use the function print() as

print('e.g. print("Hello " + "world")')

output -> e.g. print("Hello " + "world")

1. To concatenate two string use + symbol between 2 strings

print("Hello " + "world") //Hello world

1. How to take input from user during run time in python?

input(“What is your name?”)

upon running the above code, it will print the statement in the console and wait for the user to enter some text. After user enters a text and press enter, the entered text will replace the entire statement. For eg., if you want to say Hello Ragavendran V, then

print(“Hello” + input(“What is your name?”)+”!”)

1. How to print the length of the string?

print(len(“Raga”)) //4

print(len(input(“What is your name?”))) //takes user input and print length

1. How to comment a line in python?

Use # //# print(“Raga”) 🡪 this line will not get executed.

Short cut key ctrl+/

1. How to create a variable and assign value to it.

Name = “Ragavendran V”

Print(name) // Ragavendran V

Name1=input(“what is your name?”)

Print (name1) // user entered text will be printed.

Length=len(“Raga”)

Print(length) //4

1. How to insert a new line in the string?

\n -> this will introduce a new line character in the code.

1. How to declare a string in python?

Name=”Ragavendran V”

1. How to declare integer?

A=123 // no need to give any variable name

B= 123\_456\_789 // it will be treated as 123,456,789

1. How to declare a float?

C=123.456

1. How to find the data type of the given variable using type()?

A=100

Print(Type(a)) // <class ‘int’>

b=”100”

Print(Type(b)) // <class ‘str>

c=100.20

Print(Type(c)) // <class ‘float>

1. How to convert a data type from one type to another?

A=100 // it is an integer

New\_a=str(A) // it is a string now

New\_b=float(a) // it is a float now

C=”100” // it is a string

New\_c=int(c) // it is an integer

1. How will you get an input from end user and show the result by summing up all the digits entered by user. For e.g., if user enters 12 then the result should be 3 (since 1+2)

# 🚨 Don't change the code below 👇

two\_digit\_number = input("Type a two digit number: ")

# 🚨 Don't change the code above 👆

print(type(two\_digit\_number)) //check the user entered type.

#Write your code below this line 👇

first\_number=int(two\_digit\_number[0])

second\_number=int(two\_digit\_number[1])

print(first\_number+second\_number)

1. Math operators and order of priority

3+3 //it is used to add and the result is 6

5-1 //it is used to sub and the result is 4

5\*6 //it is used to mul and the result is 30

6/6 //it is used to div and the result is 1

6\*\*2 //it is used to calculate power of value (6\*6) = 36

Order of Priority PEMDAS and it happens from left to right.

Parentheses

Exponential

Multiplication & Division

Addition and Subtraction

1. BMI calculator program

Formula to calculate = weight (kg)/height\*height (mtr)

# 🚨 Don't change the code below 👇

height = input("enter your height in m: ")

weight = input("enter your weight in kg: ")

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

new\_weight=int(weight)

new\_height=float(height)

bmi=new\_weight/(new\_height\*\*2) // squaring height

new\_bmi=int(bmi)

print(new\_bmi)

1. How to round the value of a floating number?

Use the function round.

Print (round(2.6666)) // output is 3

Print(round(2.6666,2)) // output is 2.67 since we asked to round of the 2 places of decimals.

Note: When round function is used, it will either round up (if decimal place is >0.5) and down (if decimal place is <0.5). When you always want to round up irrespective of the decimal value, use math.ceil() (we need to import math). If you enter math.ceil(1.1) then the output is 2.

1. When user divides 2 numbers, the result will always be a floating number. To round it off, we need to use either the round() or divide using 2 forward slashed instead of 1. i.e., 5//2. By this we can get the output as a whole number instead of integer.
2. To manipulate the previous value by adding a new value to it or subtracting it, use

Result = 4/2

Result /= 2 // here we are saying use the previous value of result variable and divide it by 2 again. Similarly

Result += 2 or Result -= 2

1. How to print different data type in one print statement?

In general, when user tries to print string and integer together in print statement using variable names, then we will get type error while running it. To overcome this, we can use f-string i.e., we can use letter “f” before double quotes inside print.

Name=”Raga”

Age=38

Height = 1.8

isLikeable = True

**print(f”Your name is {name}, age is {age}, height is {height} and you are likeable is {islikeable}”)**

-or- assign it to a variable and then print it.

**Message**= **f”Your name is {name}, age is {age}, height is {height} and you are likeable is {islikeable}”**

**Print(Message)**

1. Write a program to find the amount payable for each person including the tip.

print ("Welcome to the tip calculator.")

bill\_amount=float(input("What was the total bill? $"))

tip\_percent=int(input("What percentage tip would you like to give? 10, 12 or 15? "))

persons=int(input("How many people to split the bill? "))

tip\_amount=float(bill\_amount\*(tip\_percent/100))

total\_Payable = tip\_amount+bill\_amount

payable\_forOnePerson = round(total\_Payable/persons,2)

print(f"Each person should pay: ${payable\_forOnePerson}")

1. How to print the 2 decimal places when the 2nd decimal is zero. For e.g., 12.30 or 13.40

Final\_amount=12.30

Print(round(final\_amount,2)) // will print just 12.3 and not 12.30

To overcome this issue

Final\_amout=”{:.2f}”.format(final\_amount)

Print(final\_amount) // will print 12.30 as expected.

1. How to use if else in python?

The indent is required after if and else part also the semi-colon.

Weight = 82

If weight>=80:

Print(“Overweight”)

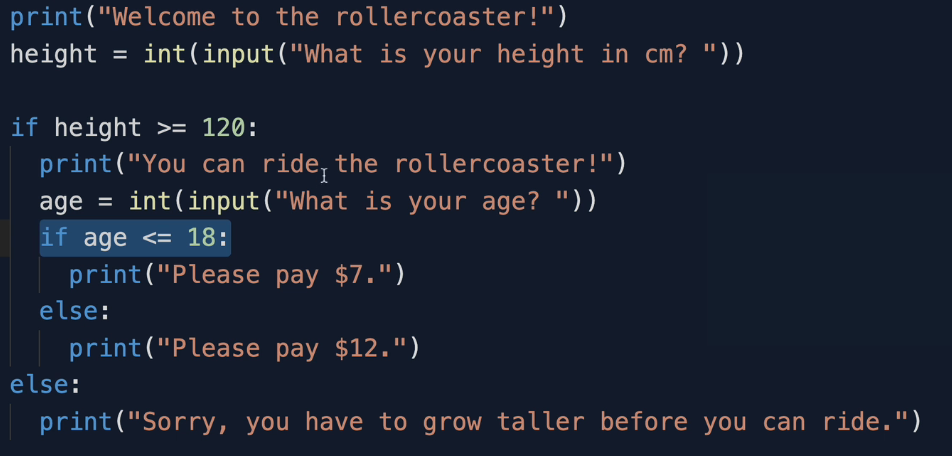
Elif Weight>65: // note this is elif and not elseif

Print(“Normal weight”)

Else:

Print(“normal weight”)

1. Nested if/else example



1. How to find the given year is a leap year or not?

A screenshot of a computer program

Description automatically generated

1. Island Treasure

print('''

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

| | | |

\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.=""\_;=.\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_

| | ,-"\_,="" `"=.| |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_"=.\_o`"-.\_ `"=.\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| `"=.\_o`"=.\_ \_`"=.\_ |

\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:=.\_o "=.\_."\_.-="'"=.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_

| | \_\_.--" , ; `"=.\_o." ,-"""-.\_ ". |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_.\_" ,. .` ` `` , `"-.\_"-.\_ ". '\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| |o`"=.\_` , "` `; .". , "-.\_"-.\_; ; |

\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_| ;`-.o`"=.\_; ." ` '`."\` . "-.\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_

| | |o; `"-.o`"=.\_`` '` " ,\_\_.--o; |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_| ; (#) `-.o `"=.`\_.--"\_o.-; ;\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_|o;.\_ " `".o|o\_.--" ;o;\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_

/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_"=.\_o--.\_ ; | ; ; ;/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_

\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_"=.\_o--.\_ ;o|o; \_.\_;o;\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_

/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_"=.\_o.\_; | ;\_.--"o.--"\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_

\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_"=.o|o\_.--""\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_

/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_ /

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

''')

print("Welcome to Treasure Island.")

print("Your mission is to find the treasure.")

#https://www.draw.io/?lightbox=1&highlight=0000ff&edit=\_blank&layers=1&nav=1&title=Treasure%20Island%20Conditional.drawio#Uhttps%3A%2F%2Fdrive.google.com%2Fuc%3Fid%3D1oDe4ehjWZipYRsVfeAx2HyB7LCQ8\_Fvi%26export%3Ddownload

#Write your code below this line 👇

direction=""

commute=""

door\_color=""

direction=input("You are at Jungle junction. Which direction do you want to take? Right or Left : \n")

if direction == "Left":

commute=input('You are in front of a lake. Enter "Wait" to cross the lake by boat or "Swim" to swim. : \n')

if commute == "Wait":

door\_color = input("You successfully crossed the dangerous lake and reached a Island. Which door do you want to open? Red, Blue or Yellow. \n")

if door\_color == "Yellow":

print("You rock 🤘 and won the game by finding the treasure 🏆!")

elif door\_color == "Blue":

print("You entered a room of 🐍. See you in hell. Game Over 😈 ")

else:

print("Opened the hell of 🔥 . Game Over 😈 ")

else:

print("You jumped into a lake full of 🐊. Nice food for 🐊 🤤. Game Over 😈")

else:

print("You take a Wrong Turn! Game Over 😈")

1. How to count the occurrence of a particular letter in a string?

Use the method count().

Name = “Raga”

Print(Name.count(“a”)) //2

1. **Random function**

It is used to create random integer or random float. Remember we need to import the random module in script at the first line by saying import random.

**To create random integer between 1 to 100 including them**

Int\_Number = random.randint(1,100)

**To create random float between 0 to 1 excluding 1(it is the default range)**

Float\_num = random.random() // max num can be 0.999999999

Note: no need to mention the range since it is defaulted between 0 to 1

**How to get number greater than 1 in float random**

Generate the number and then multiply by corresponding number. Let us say the user wants to create number >9.XXX

Then Float\_num = random.random()

Float\_num\*10`

1. **List:**
   1. How to create a list

Names = [“Ragav”, “Sharanya”, “Niharika”] or Names = [] (for empty list)

* 1. How to get the element at a particular index

Print(Names[1]) // Sharanya. Print the element at the 1st index.

Print(Names[-1]) // Niharika. Print the element at the last index.

Names[-1] = “Nikki” // it replaces the content at the given index

* 1. How to convert a string into a list using split

Name = “My name is Ragav”

List\_name = Name.split(“ “) // similar to java and o/p is [“My”, “Name”]

* 1. How to pick a random name from the list using random function.

To achieve this first find the length of the list using len(). Then use random method(0,size of list-1) to pick any number between first and last index including both indexes.

Pick\_name=list\_name.random(0,len(List\_name)-1)

Or

Random.choice(list\_name)

* 1. How to convert a string into a character list.

**“abcd”🡪 ‘a’, ‘b’, ‘c’, ‘d’**

String1 = “abcd”

Print(list(string1)) **🡪 ‘a’, ‘b’, ‘c’, ‘d’**

* 1. Nested List

Fruits=[“apple”, “Orange”, “Grape”]

Vege =[“brinjal”,”carrot”,”beans”]

Nested\_list = [Fruits, vege] //o/p is [[“apple”, “Orange”, “Grape”],[ [“brinjal”,”carrot”,”beans”]]

Print(Nested\_list[0][0]) = apple //0th row and 0th column in a matrix

* 1. How to get the number of occurrences of an element in a list

fruits = ['orange', 'apple', 'pear', 'banana', 'kiwi', 'apple', 'banana']

fruits.count('apple') // 2 bcoz apple occurs 2 times

* 1. How to reverse the order of the list

fruits.reverse()

fruits

['banana', 'apple', 'kiwi', 'banana', 'pear', 'apple', 'orange']

* 1. How to sort the list

fruits.sort()

fruits

['apple', 'apple', 'banana', 'banana', 'grape', 'kiwi', 'orange', 'pear']

* 1. How to remove the element at a particular index

fruits

['apple', 'apple', 'banana', 'banana', 'grape', 'kiwi', 'orange', 'pear']

Fruits.pop(2)

fruits

['apple', 'apple', 'banana', 'grape', 'kiwi', 'orange', 'pear']

Note: If index was not mentioned, it will remove the last item

* 1. Sum(list\_name) 🡪 gives the sum of all the integer elements inside the list.
  2. Max(list\_name) 🡪 gives the max value from the given elements inside the list.
  3. Min(list\_name) 🡪 gives the min value from the given elements inside the list.
  4. How to iterate through a list:

For I in range(len(ListName)):

Print (listname[i])

**Loop:**

List can be iterated using for loop. To implement for loop, use the syntax as below.

Fruits[“Apple”, “Orange”, “Grape”]

**For fruit in fruits:**

**Print(fruit)**

1. How to use range() in “for loop”?

Range function is used to type the number between the given range. For e.g.,

Range(1,10) // o/p is from 1 to 9 excluding the second range

Range(1,10,3) // o/p is 1,4,7,10 i.e., it increments by 3 based on the range

Fruits=["Apple","Lemon","Orange"]

for fruit in range(0,3): for fruit in range(len(Fruits)):

print(Fruits[fruit]) print(Fruits[fruit])

Note: when range has 2 number, then it will print one number before the second range where as when range has only one number, then it will print till that number.

**ii.** How to iterate through the letters in a string?

Name=”Ragavendran”

For I in Name:

Print(I) // it will print each letter one after the other.

**Functions:**

Function is a block of code used to perform a specific task and which can be called anywhere in the code.

Built-in functions in python - Reference 🡪 https://docs.python.org/3/library/functions.html

**How to define a function:**

def my\_function:

print(“hello”)

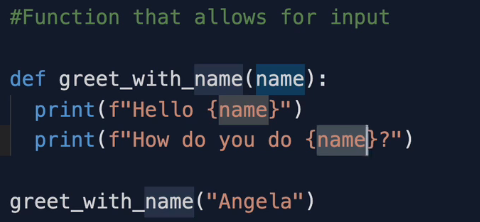
print(“world”)

**How to call the function:**

Print(“Hello”)

My\_function()

**Functions with input:**



**Syntax**

def greet\_with\_name(**username**): // where **username** is a parameter.

print(f”hello {**username**}”)

Calling the function

greet\_with\_name(“**Ragav**”) // hello Ragav.

**Note:** here “**Ragav**” is argument.

**Function with multiple input:**

def greet\_name\_loc(name, location):

print(f"hello {name}")

print(f"What is it like in {location}?")

**greet\_name\_loc(“Ragav”,”Chennai”)**

Note: here the order of parameter always equals to the order of arguments. This is called positional argument.

**Greet\_name\_loc(name=”Raga”,location=”Chennai”)**

In the above example we are mapping the parameter and arguments.ro

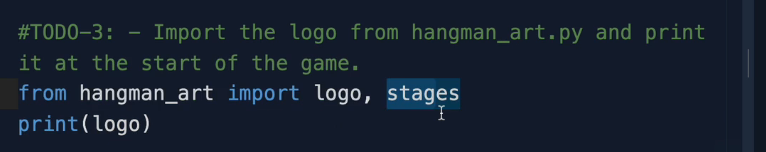
**Import:** When a variable is put between the brackets then it is called parameter and when we pass value to that parameter while calling it in the code, then it is called arguments.

**How to import from other file with .py extension into main.py**

Let us say you have a file name main.py where you want to perform addition of 2 numbers and the logic to perform addition is on another file say add.py (logic is c=a+b).

In main.py simply say

**“from add import c”** as a single line or “**import add”** in one lineand then in next line say **add.c**



Where “**hangman\_art**” is a python file in the package and “**logo and stages**” are variable from hangman\_art

A screen shot of a computer

Description automatically generated

Where hangman\_words is .py file and hangman\_words.word\_list is a variable fromhangman\_words file.

**Indentation:**

By default, there will be 4 spaces while python indents a code of line, and we use spaces over tab when it comes to priority. Otherwise, both can be used.

**Dictionary:**

Dictionary are used to store data in terms of key and value pair. To create a dictionary

**Syntax**: Dictionary\_name= {key:value} key can be a string or integer

**Name**: Emp\_info={“Name”:”Ragav”, “Designation”:”TechLead”}

**How to get a value from a dictionary**: Emp\_info(“Name”) // Ragav

**How to add new value:** Emp\_info[“Phone”] = “9962385854”. To check whether it works or not simply “print(Emp\_info)”

**Note:** When key is not found in the dictionary while adding, it will create one else it will update the value for the existing key.

**How to create empty dictionary:** empty\_dictionary = {}

**How to wipe the data out from dictionary:** Emp\_info = {} // this will set the dictionary to blank

**How to iterate through a dictionary:**

for variable\_name in Emp\_info:

print (variable\_name) //this will print all the keys

**How to iterate through a dictionary and print its value:**  
 for variable\_name in Emp\_info:

print (Emp\_info[variable\_name])

**Nesting in Dictionary:**

Nesting is a concept of having the collections inside the dictionary like list inside dictionary or both list and dictionary inside dictionary etc.

**Syntax for list inside a dictionary**

Dictionary = {

“India” : [“TamilNadu”, “Kerala”, “AP”]}

**Syntax for dictionary inside a dictionary**

Travel\_log={

“France” : {“cities\_visited” : [”Paris”, ”Lille”, ”Dijon”], “Total\_Visits” : 12}

}

France is key of travel\_log dictionary, cities visited is again a key for France, values are paris, lill,

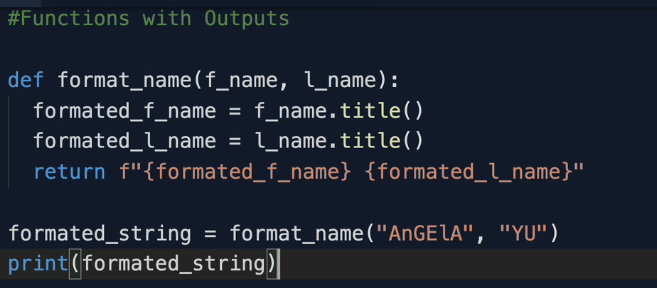


Dictionary in a list in more readable form:

A screenshot of a computer program

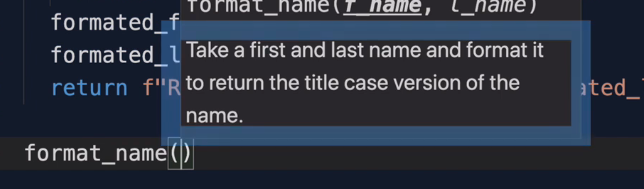
Description automatically generated

**Functions with output or with return value:**

How to convert a string into a Title case (first case of each word is capital) using title() and function with output (function returns some value when it is called)?

**Note**: Anything that is written after the return statement inside the def block, will not be executed.

**DocString:**

Doc String are used to give a short description about the use of the function whether it is in-built or user defined i.e., when user hovers the mouse over the function name, it will display a short note about the use of the function.

**How to define doc string for user defined function:**

It is very simple just write your definition in the very next line as soon as you def a function with in “”” “”” (triple quotes).

def title\_case\_converter(f\_name,l\_name):

“””This function is used to convert the given argument into a title case.”””

See the above example in blue colour. It appears as soon as user enters the function name and hover over the brackets.

Note: The text must be the immediate line in triple quotes.

**Name Scope:**

Anything (function or a variable) defined by a user inside a .py file with different scope is called namescope. It can be of “Local” and “Global”.

**Global Scope:**

When a user defined a variable or a function outside the function i.e., the variable which are not intended is called global and it can be accessible anywhere in the .py file.

**Local Scope:**

When a user defined a variable or a function inside the function then it called local scope and it is accessible only by that function when you call it and couldn’t be used in print statement i.e., unintended. If user likes to print it, then it must be intended and it has to be inside the function when you still want to use it as a variable name.

**Block Scope:**

This is not applicable in python. When you create a variable inside the loops like for, if and while, it is still considered as global and not local as long as that loop is not defined inside the function.

A computer screen shot of a code

Description automatically generated

Here enemies is defined at the global level and that is why it can be accessed even inside the function increase\_enemies (also we need to understand that the variable inside fn is new and not the same as global one though they both named as same 😊. Look at the below screen shot to understand in a better way). Again, when you run this program, you will get 2 different values for enemies and not 2 for both.

A screenshot of a computer program

Description automatically generated

Increase\_enemies() // will print 2 because we have increased inside the fn and that line is intended. Hence it will not affect the value at the global level. If user want to change it then the line has to be at the same intend level where the global level is declared.

**Can’t we change the value of global variable inside a function?**

As mentioned above, though we use the same variable name inside the function as global, the python will internally consider that as a new variable. So, what if user wanted to change it inside the function. Let understand with an example.

Score = 1

Def increase():

Score += 1 // this line will give compile error.

**Solution**:

You can call the global variable inside the function as shown below which is not recommended.

Score = 1

Def increase():

Global score

Score += 1 // now this will work.

**Best way to do this:**

Score = 1

Def increase():

Return score+1 // now this will work.

Score = increase() // this will set the global variable (score) to 2 using function.

Note: The best place to use global variables are when user uses constant value like PI, URL, USERNAME etc., as a good practice when something is going to be constant, then it is in caps so that user aware that we should not alter it.

***OOPS in Python:***

Attributes (same as variables but difference is, it is associated to a particular object) and Methods (same as function but difference is, it is associated to a particular object).

**Attributes are what the object have and methods are what the object does.**

**How to create an object in Python:**

Assume we have a class named Tutle insude turtle module, then to creatre an object we need to import the turtle module. Class should always be written in pascal’s caqse (first letter of each word in caps)

**from turtle import Turtle // where turtle is module and Turtle is class.**

**timmy = Turtle() // timmy is the object and Turtle is class**

**How to access and method from Turtle class using object Timmy**

timmy.shape(“turtle”)

**How to install an external package:**

Pycharm 🡪 file 🡪 settings 🡪 Project:[projectname] 🡪 Python Interpreter 🡪

A screenshot of a computer

Description automatically generated

Click + symbol and search for the external package something like prettytable 🡪 select the package and click install packages.

**How to create a class:**

Syntax 🡪 class ClassName:

Anything you write inside the class should be indented by 4 spaces similar to functions, if and while loops.

**Use of keyword pass:**

Let us you want to write piece of code without declaring anything inside the class, then the Ide will throw an error that the newly written code should be indented though it is not related to class. To overcome this, user has to use the keyword pass as below.

A screen shot of a computer

Description automatically generated

**Constructor:**

It is a special method which will get called once we create an object for a class. It is generally used to initialize the value of the attributes of an object.

**Syntax:**

Def \_\_init\_\_(self):

**Importance of constructor:**

We can initialize the values of the attributes without constructor too but the problem is we might do some typos. For e.g.,

Class User:

Pass

User1 = User()

User1.id = “100”

User1.name= “Ragav”

**Same line of code will repeat for n number of users.**

**The same can be simplified by using constructor.**

Class User:

Def\_\_init\_\_(Self,id,name):

Self.id = id

Self.name = name

User1 = User(“100”, “Ragav”)

User2 = User()(“200”,”Ragul”) //if user fails to pass arguments, then this line will throw error

Note: in below, followers are something which can be used to set the initial value.

A screenshot of a computer program

Description automatically generated

**Adding methods to class:**

To add a method to class

Syntax

def method\_name\_in\_lowercase(self):

Importance of self:

Self will be a default and first parameter in both **constructor** and **methods** (**both** are attached to an object) when we declare.

How to call a method:

Class User:

Def \_\_init\_\_(Seld, id, name):

Self.id = id

Self.name = name

def method1(self):

print(“from method:)

user1 = User(“1”. “Ragav”)

user1.method1: