Learning Python in Jupyter Notebook

01_First Program

In [1]:
```
print ('Hello')
print (2+3)
print ('welldone')
print ('dil dil PaKiStAn')
```

```
Hello
5
welldone
dil dil PaKiStAn
```

02_Operators

In [2]:
```
print (2+3) #Add
print (2 + 3) #No effect because of spacing between operators
print (3-2) #Subtract
print (3**2) # Exponent
print (3*2) #Multiply
print (8/2) #gives float value
print (8//2) #Gives Integer Value
print (9%2) #Gives Remainder

print (3**2/6+3-2)
```

```
5
5
1
9
6
4.0
4
1
2.5
```

Follows PEMDAS (Parantheses,Exponents,Multiply,Divide,Add,Subtract) Left to Right

03_Strings

In [3]:
```
print ('welldone')
print ('dil dil PaKiStAn')

print ('Testing Single Comma')
print ("Testing Double Comma")
print ('''Testing Tripple Comma''')

print ("what's up    ?")
```

```
welldone
dil dil PaKiStAn
Testing Single Comma
Testing Double Comma
Testing Tripple Comma
what's up    ?
```

04_Comments

In [4]:
```
print('Hello Haider')
```

```
Hello Haider
```

- #print ('Testing Single Comma') This is a comment line by using (ctrl + /)*

05_Variables

In [5]:
```
x=2 # x variable is defined and given a value of 2

print (x)

type(x)   # this is used to know the type of variable

print(type(x)) # this gives us type of variable whether it is a integer or string

x='Bananas' # value of x is updated and changed to a string

print(x) # now this function will give new updated value
print(type(x)) # now this function will give new updated type of variable

x= x + ' mangoes' #this concatenate the new value with previous value of x
print (x)

del x #this deletes the variable and it can't be called again
print (x) # now this will give an error as x variable has already been deleted
```

```
2
<class 'int'>
Bananas
<class 'str'>
Bananas mangoes
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_11364/31255471.py in <module>
     16
     17 del x #this deletes the variable and it can't be called again
---> 18 print (x) # now this will give an error as x variable has already been delet
ed

NameError: name 'x' is not defined
```

06_input_variables

In [6]:
```
#input variable simple

name=input("what is your name? ")  #This will ask you your name
print (name) #this will print out the name which has been put into the terminal


# 2nd stage input variable

name=input("what is your name? ")  #This will ask you your name
```

```
greetings= "Hello!"
print (greetings, name)


#3rd stage input variable

name=input("what is your name? ")  #This will ask you your name
age=input("How old are you? ") #This will ask you your age

print (greetings, name,", you are so young")
```

```
what is your name? Haider
Haider
what is your name? Haider
Hello! Haider
what is your name? Haider
How old are you? 28
Hello! Haider , you are so young
```

07_Conditional_Operators

In [7]:

```
print (3<5)
print (3==3)
print (4!=3)
print (4==4)
print (5>=3) # this is true in logical language but is not true in mathematical lang
print (3<=6) # this is true in logical language but is not true in mathematical lang

#application of logical operator

# we want to check if a student is elgible for admission in school or not by checkin

age_at_school=5
student_age=3
print(age_at_school==student_age)

# use combination of input function and logical operators

age_at_school=5
student_age=input("what is the age of the student? ")

print(age_at_school==student_age) #even if we give 5 as input age it will give us fa
print(type(student_age)) # because type of student_age variable is string, therefore
student_age=int(student_age) # here we change the variable value to integer
print(age_at_school==student_age) #now it gives us true
```

```
True
True
True
True
True
True
False
what is the age of the student? 5
False
<class 'str'>
True
```

08_Type_conversion

In [8]:
```
x=5
y=5.6
z="Haider"

print(x, type(x))
print(y, type(y))
print(z, type(z))

# what happens if an integer is added, multiplied or divided by a float, --------- W
r= x*y

print (r, type(r)) # here you get a float, this is called implicit conversion

# how to convert from one type to another type explicitly

print (r, type(int (r))) # now it will give us an integer because we have introduced
                         #of the arithmatic expression
```

```
5 <class 'int'>
5.6 <class 'float'>
Haider <class 'str'>
28.0 <class 'float'>
28.0 <class 'int'>
```

09_if_else_elif

In [12]:
```
student_age=input("what is the age of the student? ")
student_age=int(student_age)

age_at_school=5

if student_age==age_at_school:
    print("Yes, he can join the school in Kinder Garten")
elif student_age<=age_at_school:
    print("He is too young for school")
elif 18>student_age>=age_at_school:
    print("He should join higher class")
else:
    print("Sorry! he is too old for the school")
```

```
what is the age of the student? 5
Yes, he can join the school in Kinder Garten
```

10_Functions

In [13]:
```
print("We are learning python and it is a very interesting language")
print("We are learning python and it is a very interesting language")
print("We are learning python and it is a very interesting language")

# Now we can do the same thing by defining a function.

# 1st method

def print_python_learning():  #def is used to define a new function
    print("We are learning python and it is a very interesting language")
    print("We are learning python and it is a very interesting language")
    print("We are learning python and it is a very interesting language")

print_python_learning()
```

```
# 2 method

def print_python_learning():
    text= "we are defining a new function"
    print(text)
    print(text)
    print(text)

print_python_learning()

# 3 method

def print_python_learning(text):
    # text= "we are defining a new function"
    print(text)
    print(text)
    print(text)

print_python_learning("Hello, this is the 3rd method to define a new function")

#4 Now we will make an age calculator

def age_calculator():

    age=input("what is your current age? ")

    age=int(age)

    age=age+20

    print("After 20 years you will be", age ,"years old")

age_calculator()



def age_calculator(age):

    age=age+20

    print("After 20 years you will be", age ,"years old")

age_calculator(28)

# Now we define a new function to check if the baby is able to join the school or no

def school_join():

    baby_age=input("what is the baby's age? ")
    baby_age=int(baby_age)

    if baby_age==5:
        print("yes, you can join the school")
    elif baby_age<5:
        print("No, he is too young for the school")
    else:
        print("Sorry, we can't help you as it is only a kinder garten school")


school_join()
```

```
We are learning python and it is a very interesting language
We are learning python and it is a very interesting language
```

```
We are learning python and it is a very interesting language
We are learning python and it is a very interesting language
We are learning python and it is a very interesting language
We are learning python and it is a very interesting language
we are defining a new function
we are defining a new function
we are defining a new function
Hello, this is the 3rd method to define a new function
Hello, this is the 3rd method to define a new function
Hello, this is the 3rd method to define a new function
what is your current age? 18
After 20 years you will be 38 years old
After 20 years you will be 48 years old
what is the baby's age? 6
Sorry, we can't help you as it is only a kinder garten school
```

11_loops

In [25]:

```python
#while loop, it keeps on going until a condition stays true

x=671

while (x<675):
    print(x)
    x=x+1  # if we donot put this line, the loop will keep on executing indefinitely

# For loop

for x in range(5,15):
    print (x)


# arrays

months=["Jan", "Feb", "Mar", "Apr", "May", "Jun"]

for i in months:
    if (i=="Apr"):
        break # it will break the loop when April comes in the loop
    print(i)

days=["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"]

for i in days:
    if (i=="Tue"):
        continue #skips "May" and the loop continues
    print(i)
```

```
671
672
673
674
5
6
7
8
9
10
11
12
13
14
```

```
Jan
Feb
Mar
Mon
Wed
Thu
Fri
Sat
Sun
```

12_libraries

In [26]:

```python
#we want to know the value of pi or use any other mathatmatical function, for that w

import math

print("the value of pi is ", math.pi) # this gives us the value of pi

print(math.sin((math.pi/2))) # this gives us the value of sin (pi/2) or 90 degree

# similarly we can import statistics library

import statistics

x=[1,2,3,4,5,6,7,8,9,10]

print(statistics.mean(x)) # it gives us the mean value of array x
```

```
the value of pi is  3.141592653589793
1.0
5.5
```

13_troubleshooting

In [29]:

```python
# syntax error,

# print(hello my name is haider) here it has syntax error and program will give an e

print("hello my name is haider")

# runtime error

# print(25/0) # Runtime error
print (25/5)

#Semantics error, very hard to track and troubleshoot

name="Haider"
print ("hello name") # it contains a semantic error because name is a variable,
                     # if you want to print its value it should be out of commas
print ("hello", name)
```

```
hello my name is haider
5.0
hello name
hello Haider
```

In [ ]:

```python

```