In [1]:
```
import os
os.getcwd()
```

Out[1]:
```
'C:\\Users\\omkar'
```

name error

- variable not defined
- variable is in black color
- string is in red color
- keywords are in green color

In [3]:
```
q4
a=10
print(a)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[3], line 1
----> 1 q4
      2 a=10
      3 print(a)

NameError: name 'q4' is not defined
```

a=10 a

In [ ]:
```
true
True
TRUE
```

input

In [5]:
```
# variable name can be any thing
round(45.789) #


# How do you say 46
```

Out[5]:
```
46
```

In [6]:
```
round(45.56789,1)  # 45.<value>
```

Out[6]:
```
45.6
```

In [ ]:
```
0.5679 ==== > 0.6
```

In [7]:
```
round(45.56789,2)   # 45.<v1><v2>
                    # 45.5 <0.6789>
```

Out[7]:
```
45.57
```

In [8]:
```
round(45.56789,3)
# 45.<5><6><8>
```

Out[8]:
```
45.568
```

- we can take the values directly in python code
    - ex: num=10
- we can take the values from user like from keyboard
- Imagine there is application form we need to fill
- In the application form name: square box will appear
- you need to enter your name
- This is achieved by a keyword **input**

In [10]:
```
input()
```

Out[10]:
```
'naresh it'
```

- when we type input()
- a square box will appear
- you need to observe 3 things
    - square box
    - "*" mark inside square brackets that is before the code cell
    - python kernel is black color
    - mark and black color means kernel is busy
- first fill the box with some value
- then do enter

In [11]:
```
input()
```

Out[11]:
```
'omkar'
```

In [ ]:
```

```

In [14]:
```
input()
```

Out[14]:
```
'100 abc True'
```

Note

- What ever we type in square brackets , the output display in quotes

- the output default type is string type

In [15]:
```
input()
```

Out[15]:
```
'sandeep'
```

In [17]:
```
input()
input()
input()
```

Out[17]:
```
'data science'
```

In [16]:
```
a=10
a
```

Out[16]:
```
10
```

In [18]:
```
input()
```

Out[18]:
```
'10'
```

input()

In [20]:
```
a=input()
b=input()
print(a)
print(b)
print(f'The value of a is: {a} and b is:{b}')
```

```
10
20
The value of a is: 10 and b is:20
```

In [ ]:
```
# Ask the user enter the name
# city and country
# Print the statement
name='naresh it'
city='Hyd'
Country='India'
```

In [23]:
```
name='Amit'
State='MH'
print(f"My name is {name}, Im from {State}")
```

```
My name is Amit, Im from MH
```

In [24]:
```
name=input()  # The value of the input stored in a varabile 'name'
city=input()
country=input()
print(f'My name is {name} and I am from {city} city which is located in {country
```

```
My name is Fredy and I am from nasik city which is located in india
```

In [25]:
```
input()
input("Enter the Name")
input("Enter the name:")
input("Enter the city:")
input("Enter")
```

Out[25]: 'okay'

In [26]:
```
name = input("Enter the name:")
city = input("Enter the City:")
country = input("Enter the Country:")
print(f"My name is {name}, I'm from {city} which is located in {country}")
```

```
My name is sanat, I'm from hyd which is located in india
```

In [27]:
```
# wap ask the user take two numbers
# n1 and n2
# add the two numbers
#n1=input("E")
n1=input("Enter the Number1:") # n1='100'
n2=input("Enter the Number2:") # n2='200'
n1+n2  # '100'+'200'='100200'
```

Out[27]: '100200'

In [28]:
```
input()
```

Out[28]: '100'

In [29]:
```
int('100')
```

Out[29]: 100

In [31]:
```
n1=int('100')
n2=int('200')
n1+n2
```

Out[31]: 300

In [32]:
```
n1=int(input("Enter the Number1:")) # n1=int('100') =100
n2=int(input("Enter the Number2:")) # n2=int('200') =200
n1+n2  # 100+200=300
```

Out[32]: 300

In [34]:
```
n1=int('100')
n2=int('200')
n1+n2
```

Out[34]: 300

In [35]:
```
n1='100'
n2='200'
int(n1)+int(n2)   #  int('100')+int('200')
```

Out[35]: 300

In [36]:
```
# Wrong
n1='100'
n2='200'
int(n1+n2)   #  int('100'+'200')
```

Out[36]: 100200

In [37]:
```
n1=int(input("Enter the number1:"))
n2=int(input("Enter the number2:"))
n1+n2
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[37], line 2
      1 n1=int(input("Enter the number1:"))
----> 2 n2=int(input("Enter the number2:"))
      3 n1+n2

ValueError: invalid literal for int() with base 10: '200.25'
```

In [39]:
```
n1=int(input("Enter the number1:"))
n2=float(input("Enter the number2:"))
n1+n2
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[39], line 1
----> 1 n1=int(input("Enter the number1:"))
      2 n2=float(input("Enter the number2:"))
      3 n1+n2

ValueError: invalid literal for int() with base 10: '200.235'
```

eval

- whenever you listen eval : evaluate
- evaluate is belongs to math family
- It will convert integer value to the integer
- It will convert float value to the float only
- directly apply the eval and it will convert into corresponding data type
- Wheneever you use eval, dont provide strings: It will give the error

In [43]:
```
n1=eval(input("Enter the number1:"))
n2=eval(input("Enter the number2:"))
print(n1+n2)
```

```
58.300000000000004
```

In [44]:
```
# eval ==== two cases at a time
n1=float(input("Enter the number1:"))
n2=float(input("Enter the number2:"))
n1+n2
```

Out[44]:
```
200.5
```

In [ ]:
```
n1=10
n2=20
n3=30
avg=(10+20+30)/3
avg1=round(avg,2)
```

```
print(f"The average of {n1},{n2} and {n3} is {avg1}")
print("The average of {},{} and {} is {}".format(n1,n2,n3,avg1))
```

In [ ]:

```
# Q1) ASk the user take the 3 numbers from the keyboard find the avg
# Q2) Ask the user take the height and bredath find the Right angle traingle are
# Q3) Ask the user ask the enter length and beadth, calculate area of rectangle
# Q4) Ask the user take the radius calculate area of the circle
# Q5) Ask the user how much billl amount
#         How much tip_per you want to pay
#         calculate the total bill
```