In [ ]:
```
#Functions:
# re-use the same code
# 20 ppl are hvaing bday
# omkar hpy bday
# suresh hapy bday
```

In [1]:
```
num1 = eval(input("enter the number 1:"))
num2 = eval(input("enter the number 2:"))
print(num1+num2)
```

```
enter the number 1:85
enter the number 2:69
154
```

In [ ]:
```
def <function_name>():
    #write your code
```

In [2]:
```
def addition():
    num1 = eval(input("enter the number 1:"))
    num2 = eval(input("enter the number 2:"))
    print(num1+num2)

addition()
```

```
enter the number 1:565
enter the number 2:565
1130
```

- defining function will not give the error
- you will identify the error when you call the function only
- syntax error only will get, when define the function call

In [5]:
```
#wap ask the user enter three numbers
#fnd the average
# implement the function call
def average():

    n1=eval(input('Enter the num1:'))
    n2=eval(input('Enter the num2:'))
    n3=eval(input('Enter the num3:'))
    avg = (n1+n2+n3)/3 #avg = round((n1+n2+n3)/3,2)
    out=round(avg,2)
    print(f"the average of the {n1},{n2} and {n3} is {out}")
average()
```

```
Enter the num1:5
Enter the num2:5
Enter the num3:5
the average of the 5,5 and 5 is 5.0
```

In [7]:
```
#wap ask the user enter bill amount
# ask the user enter tip percentage
#calculate total bill
#implement the function
def totalbill():
    bill = eval(input("Enter the bill:"))
    tip =eval(input("Enter the tip percentage :"))
    tip_percent =(bill*tip)/100
    total_bill=bill+tip_percent
    print(f"The total bill is {total_bill} for the tip percent {tip}")

totalbill()
```

```
Enter the bill:85000
Enter the tip percentage :56
The total bill is 132600.0 for the tip percent 56
```

In [8]:
```
import random
random.randint()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[8], line 2
      1 import random
----> 2 random.randint()

TypeError: Random.randint() missing 2 required positional arguments: 'a' and 'b'
```

- Functions and Methods both are same

In [9]:
```
print("hello")
print(1)
def addition1():
    num1 = eval(input("enter the number 1:"))
    num2 = eval(input("enter the number 2:"))
    print(num1+num2)
print("hello")
addition1()
print("bye")
```

```
hello
1
hello
enter the number 1:45
enter the number 2:63
108
bye
```

In [14]:
```
def summ():
    try:

        num1 = eval(input("enter the number 1:"))
        num2 = eval(input("enter the number 2:"))
        print(num1+num2)
    except Exception as e:
        print(e)

summ()
```

```
enter the number 1:69
enter the number 2:6
75
```

In [ ]:
```
#Basic codes Assignment : 3 days
```

In [ ]:
```
# WAP ask the user enter a number
# find it is an even or odd
# create a function on this
```

In [1]:
```
def user():
    no=eval(input("Enter a number to check if its even or odd"))
    if no%2==0:
        print(f"The number {  no} is even")
    else:
        print(f"The number {  no} is odd")

user()
```

```
Enter a number to check if its even or odd56
The number 56 is even
```

In [2]:
```
def user():
    try:

        no=eval(input("Enter a number to check if its even or odd"))
        if no%2==0:
            print(f"The number {  no} is even")
        else:
            print(f"The number {  no} is odd")
    except  Exception as e:
        print(e)

user()
```

```
Enter a number to check if its even or oddddd
name 'ddd' is not defined
```

In [3]:
```
def user():
    try:

        no=eval(input("Enter a number to check if its even or odd"))
        if no%2==0:
            print(f"The number {  no} is even")
        else:
            print(f"The number {  no} is odd")
    except  Exception as e:
        print(e)

user()
```

```
Enter a number to check if its even or odd54513922122222
The number 54513922122222 is even
```

In [8]:
```
#implement the above code by providing a random value

import random
def user():
    try:

        no=random.randint(1,1000)
        if no%2==0:
            print(f"The number {  no} is even")
        else:
            print(f"The number {  no} is odd")
    except  Exception as e:
        print(e)

user()
```

```
The number 35 is odd
```

In [ ]:
```
#wap ask the user get a random number: num 1
#ask the user enter a number from keyboard:num 2
# if num1 == num2:print("in")
#otherwise print("out")
```

In [13]:
```
import random
def user1():
    try:
        n1=random.randint(1,1000)
        n2=eval(input("Enter the number 2:"))
        if(n1==n2):
            print("in")
        else:
            print("out")
    except Exception as e:
        print(e)

user1()
```

```
Enter the number 2:585236
out
```

In [ ]:
```
def user(): # inside brackets there is nothing hence its called fuction without argument
    no=eval(input("Enter a number to check if its even or odd"))
    if no%2==0:
        print(f"The number {  no} is even")
    else:
        print(f"The number {  no} is odd")

user()
```

- function without arguments
- until now whatever functions we developed those functions were without values
- whatever you provide inside bracket is called as function without aguments or prameters.

In [ ]:
```
def addition():
    num1 = eval(input("enter the number 1:"))
    num2 = eval(input("enter the number 2:"))
    print(num1+num2)

addition()

# Q1] in above function how many variables are there
# num1 num2 add
#Q2] how many input variables are there: 2 num1 num2
#Q3] how many output variables are there: 1 add
```

In [15]:
```
def sum1(num1,num2):
    add=num1+num2
    print(add)


#in above inside the function arguamnets are there and there are two input values
#these are called as functions with arguments


sum1(45,56)
```

```
101
```

In [ ]:
```
def average():

    n1=eval(input('Enter the num1:'))
    n2=eval(input('Enter the num2:'))
    n3=eval(input('Enter the num3:'))
    avg = (n1+n2+n3)/3 #avg = round((n1+n2+n3)/3,2)
    out=round(avg,2)
    print(f"the average of the {n1},{n2} and {n3} is {out}")
average()
```

In [16]:
```
def average(n1,n2,n3):
    avg = (n1+n2+n3)/3 #avg = round((n1+n2+n3)/3,2)
    out=round(avg,2)
    print(f"the average of the {n1},{n2} and {n3} is {out}")
average(78,58,69)
```

```
the average of the 78,58 and 69 is 68.33
```

In [20]:
```
def average(n1,n3):
    print("num1",n1)
    n2=85
    print("num3",n3)
    avg = (n1+n2+n3)/3
    out=round(avg,2)
    print(f"the average of the {n1},{n2} and {n3} is {out}")
average(78,69)
```

```
num1 78
num3 69
the average of the 78,85 and 69 is 77.33
```

In [ ]:
```
num1=500
##################################
def summ(num1,num2):
    num1=2000
    add=num1+num2
    print(add)
##########################################
num1 = 1000
summ(150,50)

# have you initialized any values before funation call
# what are the new values and when are you calling the function
# what are the nw values when the function is executing
#Function will not return any values until and unless you mention  retuen keyword inside the functio
```

In [22]:
```
def totalbill(bill,tip):
    tip_percent =(bill*tip)/100
    total_bill=bill+tip_percent
    print(f"The total bill is {total_bill} for the tip percent {tip}")

totalbill(eval(input("Enter the bill:")),eval(input("Enter the tip percent")))

#Implement the above function with arguments
```

```
Enter the bill:10000
Enter the tip percent10
The total bill is 11000.0 for the tip percent 10
```

In [24]:
```
def user(no1):

    if no1%2==0:
        print(f"The number {no1} is even")
    else:
        print(f"The number {no1} is odd")

user(eval(input("Enter a number to check if its even or odd")))
```

```
Enter a number to check if its even or odd89
The number 89 is odd
```

In [ ]:

In [ ]:

In [ ]: