- Basic python
  - data types
  - type casting
  - print statments
  - eval concept
  - input from key board
- try except
- if elif else
- Functions
  - with out arguments
  - with arguments
  - default arguments
  - return
  - local vs global

In [ ]:
```
- For

- while
```

In [ ]:
```
loop means circle

if you want to iterate same task many number of times , then we go for loop

if we want to use same piece of code , functions
```

In [ ]:
```
def <function_name>():
    # st-1
    # st-2

whenever you have : ===== indentation will be there

if  <condition>:
```

In [ ]:
```
# if i want to print first 10 numbers
```

In [ ]:
```
- intialisation (start point)

- condition     (stop point)

- increment/decrement
```

In [ ]:
```
# if i want to print first 10 numbers

# start point : 1
# stop point: 10
# 1 2 3 4 increment
```

Note

in python the index always start with zero

In [1]:
```
for i in range(10):
    print(i)

#start=0  (python index start with zero)
#  i in range(10) ====== start/stop./incre
# nothing mentioned means zero is the starting point
```

```
0
1
2
3
4
5
6
7
8
9
```

$$pattern - 1$$

range(stop)

- start=0 python index start with zero
- end= stop-1
- increment

In [2]:
```
for i in range(20):
    print(i)
# start=0   end= stop-1= 20-1=19   increment
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
```

In [5]:
```
print(0,end=' ')
print(1,end=' ')
print(2,end=' ')
print(3,end=' ')

# combine the print statements

#print(i,end=' ')
```

```
0 1 2 3
```

In [6]:
```
for i in range(20):
    print(i,end=' ')
```

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
```

$pattern - 2$

range(start,stop)

- start=start
- end=stop-1
- incremental

ex: range(10,20)

- start=10
- stop= 20-1=19
- incremental

In [7]:
```
for i in range(10,20):
    print(i,end=' ')
```

```
10 11 12 13 14 15 16 17 18 19
```

$pattern - 3$

range(start,stop,step)

- start= start
- stop value depends on sign of step value
- step sign is + : incremental
    - end= stop-1
- step sign is - : decremental
    - end= stop+1

In [ ]:
```
for i in range(5,20,3)

# start=5
# stop  value decide on step sign
# step = 3 ===== +ve
# end= stop-1= 20-1=19

for i in range(5,20,-3)
# step=-3 ======== -ve
# start=5
# end=stop+1=20+1=21
```

In [9]:
```
#for i in range(5,20,3)

# =============== Observation -1 : fix the values=========
# start= 5
# step sign =+ve
# end= stop-1= 20-1=19

#==================== Obervation-2: answer is possibile==========


#========== Observation-3 ===========
#5  8  11 14  17

for i in range(5,20,3):
    print(i,end=' ')
```

```
5 8 11 14 17
```

In [10]:
```
for i in range(-5,-20,-3):
    print(i,end=' ')

# obs-1:
# start= -5
# step sign = -ve  ====== > decrement
# end= stop+1= -20+1=-19

# obs-2 : possibile
# -5  -8  -11   -14    -17
```

```
-5 -8 -11 -14 -17
```

In [11]:
```
for i in range(5,20,-3):
    print(i,end=' ')

# obs-1
# start=5
# sign= -
# end= stop+1 = 20+1 =21
# 5           21

# obs-2 : not possibile
```

In [ ]:
```
for i in range(5,20,3):
    print(i,end=' ')

for i in range(-5,-20,-3):
    print(i,end=' ')

for i in range(5,20,-3):
    print(i,end=' ')
```

In [ ]:
```
range(10,30,3) ========= possibile
range(10,30,-3) =======  NP ======= > No answer
range(10,-30,3) =======  NP======= > No answer
range(10,-30,-3) ======  Possib
range(-10,-30,-3) ====   P
range(-10,-30,3)  =====  NP ======= > No error/ No answer
range(-10,30,-3)  =====  NP ======= > No answer
range(-10,30,3)   =====  p
```

In [13]:
```
# start=10, dire=post, stop =30-1=29, step=3
for i in range(10,30,-3):
    print(i,end=' ')
```

In [14]:
```
for i in range(10,-30,3):
    print(i,end=' ')
```

In [16]:
```
for i in range(40,20,-3):
    print(i,end=" ")

#Not possible because reverse direction
```

```
40 37 34 31 28 25 22
```

In [ ]:
```
# WAP ask the user print first 10 square numbers
# 1 to 10
# 1  4   9  16  25  36  49  64  81  100
```

In [17]:
```
for i in range(1,11):
    print("The square of {} is {}".format(i,i*i))
```

```
The square of 1 is 1
The square of 2 is 4
The square of 3 is 9
The square of 4 is 16
The square of 5 is 25
The square of 6 is 36
The square of 7 is 49
The square of 8 is 64
The square of 9 is 81
The square of 10 is 100
```

In [18]:
```
# WAP to print 7th table 7x1=7  tp 7x10=70
for i in range(1,11):
    print("7 * {} = {}".format(i,7*i))
```

```
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
7 * 10 = 70
```

In [19]:
```
for i in range(1,11):
    print(i*7,end=" ")
```

```
7 14 21 28 35 42 49 56 63 70
```

In [22]:
```
for i in range(1,10):
    print('The 7x{} is:{}'.format(i,i*7))
```

```
The 7x1 is:7
The 7x2 is:14
The 7x3 is:21
The 7x4 is:28
The 7x5 is:35
The 7x6 is:42
The 7x7 is:49
The 7x8 is:56
The 7x9 is:63
```

In [23]:
```
for i in range(1,11):
    print("7 * ",i,"=",i*7)
```

```
7 *  1 = 7
7 *  2 = 14
7 *  3 = 21
7 *  4 = 28
7 *  5 = 35
7 *  6 = 42
7 *  7 = 49
7 *  8 = 56
7 *  9 = 63
7 *  10 = 70
```

In [ ]:
```
# WAP ask the user take random number between 1 to 100
# and print sauqre of random numbr
# and this operation to 5 times
```

In [29]:
```
import random
num=random.randint(1,100)
sq_num=num*num
sq_num
```

Out[29]:
```
961
```

In [30]:
```
for i in range(5):

    print(i)
```

```
0
1
2
3
4
```

In [ ]:
```
import random
num=random.randint(1,100)
for i in range(num,num+5):
    print(i*i,end=' ')
```

In [32]:
```
#import random
#num=random.randint(1,100)
for num in range(5):
    #
    print("Square of {} is={}".format(num,num*num))
```

```
Square of 0 is=0
Square of 1 is=1
Square of 2 is=4
Square of 3 is=9
Square of 4 is=16
```

In [35]:
```
for i in range(5):
    print('hello good')
```

```
hello good
hello good
hello good
hello good
hello good
```

In [38]:
```
import random
for i in range(5):
    num=random.randint(1,100)
    print("The sqaure of {} is {}".format(num,num*num))


import random
num=random.randint(1,100)
sq_num=num*num
sq_num
```

```
The sqaure of 85 is 7225
The sqaure of 98 is 9604
The sqaure of 30 is 900
The sqaure of 30 is 900
The sqaure of 35 is 1225
```

In [ ]:
```
# For loop is just a jaint wheel
#
```

In [1]:
```
print('hello')
print('hello')
print('hello')
print('hello')
print('hello')

# I want to run any code multiple times ====> loop
```

```
hello
hello
hello
hello
hello
```

In [3]:
```
# my loop should run 5 times
for i in range(5):
    print('hello')
```

```
hello
hello
hello
hello
hello
```

In [5]:
```
# wap ask the user to get 5 random numbers ==== square
import random
num=random.randint(1,100)
num=random.randint(1,100)
num=random.randint(1,100)
num=random.randint(1,100)
num=random.randint(1,100)
print(num)
```

```
60
```

Normal approach

In [6]:
```
for i in range(5):
    num=random.randint(1,100)
    print("the square of {} is {}".format(num,num*num))
```

```
the square of 78 is 6084
the square of 27 is 729
the square of 93 is 8649
the square of 97 is 9409
the square of 99 is 9801
```

In [ ]:
```
# WAP print the number is even or odd number
# the numbers ranges from 10 to 40

# do for loop ranges between 10,40
# then perform if else concept
```

In [10]:
```
for i in range(10,41):
    if i%2==0:
        print('{} an even number'.format(i))
    else:
        print('{} an odd number'.format(i))
```

```
10 an even number
11 an odd number
12 an even number
13 an odd number
14 an even number
15 an odd number
16 an even number
17 an odd number
18 an even number
19 an odd number
20 an even number
21 an odd number
22 an even number
23 an odd number
24 an even number
25 an odd number
26 an even number
27 an odd number
28 an even number
29 an odd number
30 an even number
31 an odd number
32 an even number
33 an odd number
34 an even number
35 an odd number
36 an even number
37 an odd number
38 an even number
39 an odd number
40 an even number
```

With out argument

In [13]:
```
# when you copied makes sure indeation should maintain
# with out argument
def even_odd():
    for i in range(25,35,3):
        if i%2==0:
            print('{} an even number'.format(i))
        else:
            print('{} an odd number'.format(i))

even_odd()

# Instead of changing numbers inside the code
# we can provide those as arguments
```

```
20 an even number
21 an odd number
22 an even number
23 an odd number
24 an even number
25 an odd number
26 an even number
27 an odd number
28 an even number
29 an odd number
```

In [ ]:
```

```

With arguments

In [14]:
```
# With arguments
def even_odd(start,stop,step):
    for i in range(start,stop,step):
        if i%2==0:
            print('{} an even number'.format(i))
        else:
            print('{} an odd number'.format(i))

even_odd(10,15)
```

```
10 an even number
11 an odd number
12 an even number
13 an odd number
14 an even number
```

Global variable

In [15]:
```
# Global variable
# I want to take start and stop from keyboard
start=eval(input("enter start value:")) # global variable
stop=eval(input("enter stop value")) # global variale

def even_odd():  # parameters are not required
    for i in range(start,stop):
        if i%2==0:
            print('{} an even number'.format(i))
        else:
            print('{} an odd number'.format(i))

even_odd()
```

```
enter stop value:20
enter start value10
10 an even number
11 an odd number
12 an even number
13 an odd number
14 an even number
15 an odd number
16 an even number
17 an odd number
18 an even number
19 an odd number
```

Local variable

In [16]:
```
# Local variable
def oddeven():
    a=eval(input("enter start"))  # Local variable
    b=eval(input("enter stop"))   # local variable
    for i in range (a,b):
        if i%2==0:
            print("{} is even".format(i))
        else:
            print("{} is odd".format(i))
oddeven()
```

```
enter start10
enter stop15
10 is even
11 is odd
12 is even
13 is odd
14 is even
```

In [ ]:
```
start=eval(input("enter the value"))
stop=eval(input("enter the value"))
def fun():
    for i in range(start,stop):
        if i%2==0:
            print("{} is an even number".format(i))
        else:
            print("{} is odd number".format(i))

fun()
```

In [18]:
```
stop =eval(input("enter stop value:"))
start =eval(input("enter start value:"))

def evenodd():
    for i in range(start,stop):
        if i%2 == 0:
            print("Given number {} is even number". format(i))

        else:
            print("Given number {} is odd number". format(i))

evenodd()
```

```
enter stop value:15
enter start value:10
Given number 10 is even number
Given number 11 is odd number
Given number 12 is even number
Given number 13 is odd number
Given number 14 is even number
```

Function in function

In [ ]:
```
# Global variable
# I want to take start and stop from keyboard

    start=eval(input("enter start value:"))
    stop=eval(input("enter stop value"))


def even_odd():  # parameters are not required
    for i in range(start,stop):
        if i%2==0:
            print('{} an even number'.format(i))
        else:
            print('{} an odd number'.format(i))

even_odd()
```

In [22]:
```
def get_values():
    start=eval(input("enter start value:"))
    stop=eval(input("enter stop value"))
    return(start,stop)

def even_odd():
    start,stop=get_values()
    for i in range(start,stop):
        if i%2==0:
            print('{} an even number'.format(i))
        else:
            print('{} an odd number'.format(i))

even_odd()

# step-1: it will call get_values()
# step-2: start= enter the start values
# step-3: stop value
# for loop
```

```
enter start value:20
enter stop value30
20 an even number
21 an odd number
22 an even number
23 an odd number
24 an even number
25 an odd number
26 an even number
27 an odd number
28 an even number
29 an odd number
```

In [21]:
```
def get_values():
    start=eval(input("enter start value:"))
    stop=eval(input("enter stop value"))
    return(start,stop)

get_values()
```

```
enter start value:20
enter stop value30
```

Out[21]:
```
(20, 30)
```

In [25]:
```
start=eval(input("enter start value:"))
stop=eval(input("enter stop value"))
```

```
enter start value:10
enter stop value20
```

In [ ]:
```
range((20,30))
```

In [24]:
```
def values():
    num1=20
    num2=30
    return(num1,num2)

num1,num2=values()
num1,num2
```

Out[24]:
```
(20, 30)
```

In [ ]:
```
# It will not come one day
# You need to avoid the error
# you need understand the concept
# logics will come

# It is okay if you are not able to create function in function
# tryto understand the concept
```

In [ ]:
```

```

In [26]:
```
print('hai')
print('hello')
for i in range(3):
    print('good')
    print('morning')

print('bye')

# hai
# hello
# good
# morning
# good
# morning
# good
# morning
# bye
```

```
hai
hello
good
morning
good
morning
good
morning
bye
```

In [29]:
```
# WAP ask the user get a 10 random numbers between 1 to 100
# and print it is an even or odd
# first write normal code
# keep that code in a loop
# that loop should run 10 times

n=random.randint(1,100)
if n%2==0:
    print("{} is even".format(n))
else:
    print("{} is odd".format(n))
```

```
42 is even
```

In [30]:
```
import random
for n in range(10):
    n=random.randint(1,100)
    if n%2==0:
        print("{} is even".format(n))
    else:
        print("{} is odd".format(n))
```

```
45 is odd
48 is even
42 is even
18 is even
1 is odd
43 is odd
49 is odd
29 is odd
85 is odd
71 is odd
```

In [ ]:
```
# Game: Guessing
# step-1: get a random number between 1 to 20  : num1
# step-2: ask the user enter a number  : num2
# step-3: if the num1==num2 : print( you won)
# step-4: otherwise you lost
# I will give only 3 chances
# # step-2: ask the user enter a number  : num2
#                             check the condition num1==num2
```

In [ ]:
```
# num1= random.randint(1,20)
# for i in range(3):
#    num2=eval()
#    if
```

In [31]:
```
import random
n=random.randint(1,20)
for i in range(3):
    u=eval(input("Enetr u Value:"))
    if n==u:
        print("You own")
    else:
        print("You Lost")
```

```
Enetr u Value:4
You Lost
Enetr u Value:6
You Lost
Enetr u Value:7
You Lost
```

In [32]:
```
import random
for i in range(3):
    num1 = random.randint(1,20)
    num2 = eval(input("enter number: "))
    if num1 == num2:
        print("you won")
    else:
        print("you lost")
```

```
enter number: 2
you lost
enter number: 4
you lost
enter number: 7
you lost
```

In [ ]:
```
```

In [ ]:
```
import random

for i in range(3):
    num1=random.randint(1,20)
    num2=eval()
    if num1==num2:
         print("{} is equal to {} and YOU WON".format(num1,num2))
    else:
        print("{} is not equal to {} and YOU LOST".format(num1,num2))
```

In [40]:
```
import random
num1=random.randint(1,20)
print(num1)
for i in range(3):
    num2=eval(input("enter the number: "))
    if num1==num2:
        print("You won")
        break
    else:
        print('you lost')
        # chances are left
```

```
7
enter the number: 8
you lost
enter the number: 9
you lost
enter the number: 6
you lost
```

In [ ]:
```
#how to write code if first time we are wrong it must return you got n-1 ch
```

In [41]:
```
import random
num1=random.randint(1,20)
for i in range(3):
    num2=eval(input())
    if num2==num1:
        print("You won")
        break
    else:
        print("You lost")
        print("you have {} chances left".format(i))
print("num1 is:",num1)
```

```
4
You lost
you have 0 chances left
5
You lost
you have 1 chances left
7
You lost
you have 2 chances left
num1 is: 14
```

In [45]:
```
import random
chances=eval(input("enter the number of chances:"))
for i in range(chances):
    num1=random.randint(1,15)
    num2=eval(input("Enter the Numnber"))
    if num1==num2:
        print("{} is equal to {} and YOU WON".format(num1,num2))
        break
    else:
        print("{} is not equal to {} and YOU LOST".format(num1,num2))
        print(" {} is left ".format(chances-1-i))  # 3-1-0 =2
```

```
enter the number of chances:4
Enter the Numnber9
13 is not equal to 9 and YOU LOST
 3 is left
Enter the Numnber9
7 is not equal to 9 and YOU LOST
 2 is left
Enter the Numnber9
14 is not equal to 9 and YOU LOST
 1 is left
Enter the Numnber9
15 is not equal to 9 and YOU LOST
 0 is left
```

In [ ]:
```
import random
try:
    num1=random.randint(1,20)

    for i in range(3):
        total_chances=3
        num2=eval(input("Enter a number : "))
        if num1==num2:
            print("You won")
            break
        else:
            print("You lost")
            print("{} attempt is left".format(total_chances-1-i))

except Exception as e:
    print(e)
```

In [ ]:
```
import random
n=eval(input("how many Chances you will give"))
num1 = random.randint(1,20)
for i in range(3):
    num2 = eval(input("enter number: "))
    if num1 == num2:
        print("you won")
        break

    else:
        print("you lost")
        print("{} chances left".format(n-1-i))
```

In [47]:
```
import random
chances=eval(input("enter the number of chances:"))
num1=random.randint(1,15)
for i in range(chances):
    num2=eval(input("Enter the Numnber"))
    if num1==num2:
        print("{} is equal to {} and YOU WON".format(num1,num2))
        break
    else:
        print("{} is not equal to {} and YOU LOST".format(num1,num2))
        print(" {} is left ".format(chances-1-i))

print("your all chances are over,better luck next time!") # problem
```

```
enter the number of chances:3
Enter the Numnber2
12 is not equal to 2 and YOU LOST
 2 is left
Enter the Numnber12
12 is equal to 12 and YOU WON
your all chances are over,better luck next time!
```

In [48]:
```
import random
chances=eval(input("how many Chances you will give"))
num1 = random.randint(1,20)
print(num1)
for i in range(chances):
    num2 = eval(input("enter number: "))
    if num1 == num2:
        print("you won")
        break

    else:
        print("you lost")
        print("{} chances left".format(chances-1-i))
print("All your chances are over!, Better Luck Next Time")
```

```
how many Chances you will give3
2
enter number: 2
you won
All your chances are over!, Better Luck Next Time
```

In [ ]:

```
# ============= Case-1: Guess program==================================
import random
n=random.randint(1,20)
for i in range(3):
    u=eval(input("Enetr u Value:"))
    if n==u:
        print("You own")
    else:
        print("You Lost")
#============= Case-2: No of chances printing=========================
#============= case-3: take the chances from user=====================
import random
chances=eval(input("enter the number of chances:"))
num1=random.randint(1,15)
for i in range(chances):
    num2=eval(input("Enter the Numnber"))
    if num1==num2:
        print("{} is equal to {} and YOU WON".format(num1,num2))
        break
    else:
        print("{} is not equal to {} and YOU LOST".format(num1,num2))
        print(" {} is left ".format(chances-1-i))  # 3-1-0 =2
#============== Case-4: when chances are over better luck next time ========
import random
n1 = random.randint(1,20)
ch = 3
print(n1)
x=0
for i in range(ch):
    n2 = eval(input("Enter a number b/w 1-20:"))
    if (n1 == n2):
        print ("You Won")
        x=1
        break
    else:
        print("You Lost")
        print("{} chances left".format((ch-1)-i))
if x==0 :
    print("All chance are over better luck next time")


# m-2:
import random
try:
    num1=random.randint(1,20)

    for chances in range(3):
        total_chances=3
        num2=eval(input("Enter a number : "))
        if num1==num2:
            chances=total_chances-1
            print("You won")
            break
        else:
            print("You lost")
            print("{} attempt is left".format(total_chances-1-chances))

        if chances==total_chances-1:
             print("You have lost, Better luck next time ")
```

```
except Exception as e:
    print(e)
```

In [49]:
```
import random
chances=eval(input('enter the number of chances:'))
for i in range(chances):
    n1=random.randint(1,20)
    print(n1)
    n2=eval(input('entner a number:'))
    if n1==n2:
        print('you won')
        break
    else:
        print('you lost')
        print('you have {} chances'.format(chances-1-i))
        print('your all chances re over better luck next time')
```

```
enter the number of chances:4
13
entner a number:12
you lost
you have 3 chances
your all chances re over better luck next time
13
entner a number:13
you won
```

In [50]:
```
import random
n1 = random.randint(1,20)
ch = 3
print(n1)
x=0
for i in range(ch):
    n2 = eval(input("Enter a number b/w 1-20:"))
    if (n1 == n2):
        print ("You Won")
        x=1
        break
    else:
        print("You Lost")
        print("{} chances left".format((ch-1)-i))
if x==0 :
    print("All chance are over better luck next time")
```

```
1
Enter a number b/w 1-20:1
You Won
```

In [51]:
```
import random
try:
    num1=random.randint(1,20)

    for chances in range(3):
        total_chances=3
        num2=eval(input("Enter a number : "))
        if num1==num2:
            chances=total_chances-1
            print("You won")
            break
        else:
            print("You lost")
            print("{} attempt is left".format(total_chances-1-chances))

        if chances==total_chances-1:
             print("You have lost, Better luck next time ")



except Exception as e:
    print(e)
```

```
Enter a number : 4
You lost
2 attempt is left
Enter a number : 6
You lost
1 attempt is left
Enter a number : 8
You lost
0 attempt is left
You have lost, Better luck next time
```

In [ ]:
```
# Exercise: For loop implementaion ===== functions
# Exercise: Guess a game code with 4 use cases

# Practice today session
```

In [ ]:
```
# Find the sum of first 10 natural numbers
# natural: countable


```

In [ ]:
```
i=1
i=2     +  1 = 3
i=3     +  3=6
i=4     +  6=10
i=5
i=6
i=7
i=8
i=9
i=10
```

In [7]:
```
sum1=0
for i in range(1,11):    # i=2
    sum1=sum1+i          # sum1=1+2=3

print("The sum of first 10 natural numbers are:",sum1) # out of the loop

# if you keep different variable
# step-1:    sum1=0   i=1    new sum1= 0+1=1
# step-2:    sum1=1   i=2        sum1=1+2=3
# step-3:    sum1=3   i=3        sum1=3+3=6
# step-4:    sum1=6   i=4        sum1=6+4=10    ====>   sum1= (sum1)+i
```

```
The sum of first 10 natural numbers are: 55
```

In [ ]:
```
# WAP ask the user to find a number is an even number and odd number
# between 1 to 100
# this time count the number of even numbers and odd numbers
```

In [ ]:
```
1   2    3
1+2=3
3+3=6   your mind is storing and updating the values
we will use only one variable to store an update the value
```

In [12]:
```
count_e=0
count_o=0
for i in range(1,101):
    if i%2 ==0:
        count_e = count_e+1     # Instead of printing even ==== we are just
    else:
        count_o=count_o+1       # instead of printing odd ==== we are count
print(" the total number of even numbers are", count_e)
print(" The total number of odd numbers are ",count_o)
```

```
 the total number of even numbers are 50
 The total number of odd numbers are  50
```

In [ ]:
```
you are seeing only 3 even numbers
you display  i counted 4 even numbers
```

In [ ]:
```
sir we dont have to take a variable count for counting
count=1
0
0+1=1
1+1=2
2+1=3
3+1=4
4+1=5
5+1=6

(vary)+1

sai=sai+1
```

In [ ]:
```
# wap ask get a 20 random number between 1 to 100  #  20 times
# count the number of values which are greater than 50 and less than 90
# which are between 50 and 90


# you need to get the count out of 20 number how many numbers are betwenn 5
```

In [19]:
```
import random
num=random.randint(1,100) # random is outside
count=0
for num in range(20):
    if 50<num and num<90:
         count=count+1

print("the total number between 50 to 90: ",count)
```

```
the total number between 50 to 90:  0
```

In [22]:
```
import random

numGt50 = 0
numLt50 = 0
i = 1
for i in range (1,20):  #19
    num = random.randint(1, 100)

    if(num >50 and num < 90):
        numGt50 = numGt50 +1
    elif(num < 50):
        numLt50 = numLt50 +1
    else:
        print("invalid number")   # 4 are more than 90

print("Numbers between 50 and 90:", numGt50)
print("Numbers less than 50 :", numLt50)
```

```
invalid number
invalid number
invalid number
invalid number
Numbers between 50 and 90: 10
Numbers less than 50 : 5
```

In [ ]:
```
# first remove try-except
# you are return the values under exception

# what is the use exception:  to capture the error

# in try block if you get the error====== exception
# in try block there is no error ====== > try block will execute

# return(c,d) under exception

# remove try-exception run the code
# keep exact code inside try block
```

$in - operator$

In [23]:
```
'b' in 'banana'
```

Out[23]: True

In [25]:
```
name='python'

'p' in name   # T
'y' in name   # T
't' in name   # T
'o' in name   # T
'n' in name   # T
'P' in name   # F
```

Out[25]: False

In [26]:
```
'py' in name
```

Out[26]: True

In [27]:
```
'pt' in name
```

Out[27]: False

In [28]:
```
'on' in name  # name='python'
```

Out[28]: True

In [ ]:
```
name='python'

'p' in name   # T
'y' in name   # T
't' in name   # T
'o' in name   # T
'n' in name   # T

# generlised expression
# what is vary and what is common

# i in name
```

In [29]:
```
name='python'

for i in name:
    print(i)
```

```
p
y
t
h
o
n
```

range ======= numbers

in ========== strings

eval ======== numbers

input ======= strings

In [ ]:
```
# wap ask the user enter a your name : omkar
# count the number of letters :  using for loop and in operator
# if the count>5 : print(you have lengthy name)
# else:print(you have small name)

# in operator
# loop
# counter
# conditional
```

In [35]:
```
name='python'

count=0
for i in name:
    count=count+1
    print(count,i)

print("total count:",count)

if count>=5:print("it is a lengthy name") # side by side also
else:print("it is a small name")
```

```
1 p
2 y
3 t
4 h
5 o
6 n
total count: 6
it is a lengthy name
```

In [36]:
```
name = input("enter your name")
NumOfA = 0

for i in name:
    if 'a' == i: NumOfA = NumOfA +1
print("number of a", NumOfA)
```

```
enter your namepythona
number of a 1
```

In [37]:
```
name_1=input("Enter your name:")
for i in name_1:
    if i=='a':
        count=0
        count=count+1
print("a letter in your name:",count)
```

```
Enter your name:aaaaa
a letter in your name: 1
```

In [ ]:
```
# wap ask the user enter your name : python
# iterate your name with loop
# check the condition if i=='a': count it
# print how many letter 'a' are there in your name

# Mobin
# DB
# Saurabh
# Anil
# Swetha
# Kishore
# Azeez
# Saud
# Vikas
# Harikrishna
# Sibaprasad
# Tarun
# Jyotsna
# santhosh
# Karim
# srinivas
```

In [ ]:
```
I will upload solution
check the solution ask me

sibaprasd in the group
group pls help sibaprasad
Thank you!
```

In [ ]:
```

```

In [ ]:
```
- Range

- in concept
```

In [ ]:
```
'a'  # ascii

A=65
a=97
```

ord

In [2]:
```
ord('A')
```

Out[2]: 65

In [3]:
```
'Apple'>'apple'

# 'A': 65
# 'a': 97
# 65>97
```

Out[3]: False

In [6]:
```
'Apple'>'Bat'

# 'A'=65    'B'=66
#  65>66  False
```

Out[6]:
```
False
```

In [5]:
```
ord('A'),ord('B')
```

Out[5]:
```
(65, 66)
```

In [9]:
```
'Apple'>'APPLE'

# 'A'   'A'  leave it
#  'p'   'P'
# 112>80
```

Out[9]:
```
True
```

In [8]:
```
ord('p'),ord('P')
```

Out[8]:
```
(112, 80)
```

In [13]:
```
'Sufiyn' >'SUfiyn'

# which word is greater
# A SCII: American starndard code information interchanfe
# A: 65     a=97
# B:66
# C=67
```

Out[13]:
```
True
```

In [11]:
```
ord('S'),ord('s')
```

Out[11]:
```
(83, 115)
```

chr

In [15]:
```
chr(65),ord('A')
```

Out[15]:
```
('A', 65)
```

In [ ]:
```
# Case-1: use for loop 20 to 200
# ASCII : where it will start I dont knwo
#         where it will end
# chr(1)
# chr(2)
```

In [18]:
```
print(chr(65))
print(chr(66))
```

```
A
B
```

In [ ]:
```
# output
# asccii starting
#  1-----?
#  31----?
#  65----->A
#  66---->B
#  ending
```

In [19]:
```
for i in range(1,100):
    print(i)
```

In [24]:
```
for i in range(33,127):
    print(i,'------>',chr(i))    # step-1:  1,chr(1)  2,chr(2)
```

33 ------> !
34 ------> "
35 ------> #
36 ------> $
37 ------> %
38 ------> &
39 ------> '
40 ------> (
41 ------>)
42 ------> *
43 ------> +
44 ------> ,
45 ------> -
46 ------> .
47 ------> /
48 ------> 0
49 ------> 1
50 ------> 2
51 ------> 3
52 ------> 4
53 ------> 5
54 ------> 6
55 ------> 7
56 ------> 8
57 ------> 9
58 ------> :
59 ------> ;
60 ------> <
61 ------> =
62 ------> >
63 ------> ?
64 ------> @
65 ------> A
66 ------> B
67 ------> C
68 ------> D
69 ------> E
70 ------> F
71 ------> G
72 ------> H
73 ------> I
74 ------> J
75 ------> K
76 ------> L
77 ------> M
78 ------> N
79 ------> O
80 ------> P
81 ------> Q
82 ------> R
83 ------> S
84 ------> T
85 ------> U
86 ------> V
87 ------> W
88 ------> X
89 ------> Y
90 ------> Z
91 ------> [
92 ------> \
93 ------>]

```
94 ------> ^
95 ------> _
96 ------> `
97 ------> a
98 ------> b
99 ------> c
100 ------> d
101 ------> e
102 ------> f
103 ------> g
104 ------> h
105 ------> i
106 ------> j
107 ------> k
108 ------> l
109 ------> m
110 ------> n
111 ------> o
112 ------> p
113 ------> q
114 ------> r
115 ------> s
116 ------> t
117 ------> u
118 ------> v
119 ------> w
120 ------> x
121 ------> y
122 ------> z
123 ------> {
124 ------> |
125 ------> }
126 ------> ~
```

In [ ]:
```
# Do the same reverse task for capital letters A to Z
# in operator
```

In [25]:
```
for i in range(A,Z):
    print(i,ord(i))
```

```
---------------------------------------------------------------------------
-
NameError                                 Traceback (most recent call las
t)
Cell In[25], line 1
----> 1 for i in range(A,Z):
      2     print(i,ord(i))

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

In [28]:
```
for i in 'abcdefghijklmnopqrstuvwxyz':
    print(i,"--->",ord(i))
```

```
a ---> 97
b ---> 98
c ---> 99
d ---> 100
e ---> 101
f ---> 102
g ---> 103
h ---> 104
i ---> 105
j ---> 106
k ---> 107
l ---> 108
m ---> 109
n ---> 110
o ---> 111
p ---> 112
q ---> 113
r ---> 114
s ---> 115
t ---> 116
u ---> 117
v ---> 118
w ---> 119
x ---> 120
y ---> 121
z ---> 122
```

In [29]:
```
for i in 'a','b','C','D':
    print(i,ord(i))
```

```
a 97
b 98
C 67
D 68
```

In [30]:
```
for i in 'ABCDEFGHIJKLMNOPQRSTUVWXYZ':
    print(i,ord(i))

for i in range(65,97):
    print(i,chr(i))
```

```
A 65
B 66
C 67
D 68
E 69
F 70
G 71
H 72
I 73
J 74
K 75
L 76
M 77
N 78
O 79
P 80
Q 81
R 82
S 83
T 84
U 85
V 86
W 87
X 88
Y 89
Z 90
```

Package-Name: string

In [33]:
```
import string
```

In [ ]:
```
# step-1: apply dir
# step-2: understand the method name  (with out __version__)
# step-3: try to use the method name
```

In [ ]:
```
# import <package_name>
# dir(package_name)
# help(<package_name>.<method_name>)
# package_name.method_name
#'ABCDEFGHIJKLMONOPQRSTUVWXYZ' :
# 'abc----z'
#'012---9'
#'!@$#%^&*()'
```

In [35]:
```
print(string.ascii_uppercase)
print(string.ascii_lowercase)
```

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
```

In [36]:
```
for i in string.ascii_uppercase:
    print(i,ord(i))
```

```
A 65
B 66
C 67
D 68
E 69
F 70
G 71
H 72
I 73
J 74
K 75
L 76
M 77
N 78
O 79
P 80
Q 81
R 82
S 83
T 84
U 85
V 86
W 87
X 88
Y 89
Z 90
```

In [38]:
```
string.capwords('a')  # capitalise the word
```

Out[38]:
```
'A'
```

In [39]:
```
import string
for i in string.printable:
    print(i, ord(i), chr(i))
```

```
---------------------------------------------------------------------------
-
TypeError                                 Traceback (most recent call las
t)
Cell In[39], line 3
      1 import string
      2 for i in string.printable:
----> 3     print(i, ord(i), chr(i))

TypeError: 'str' object cannot be interpreted as an integer
```

In [40]:
```
string.printable
```

Out[40]:
```
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ!"#$%&\'()*
+,-./:;<=>?@[\\]^_`{|}~ \t\n\r\x0b\x0c'
```

In [ ]:
```
if condition on each letter type ===== ord
else:                                  chr

idea is good
```