**PYTHON**

Python is a multiparadigm programming/scripting language introduced by Guido van Rossum. Its implementation began in December 1989.

**Why to Study Python?**

Easy to learn

Less lines of codes

Simple to implement

Huge library support

**The Features List**

It provides rich data type and easier to read syntax.

It is a platform independent scripted language.

It allows more run-time flexibility.

Libraries in Python are cross-platform compatible with Linux, Macintosh, and Windows.

Python supports functional and structured programming as well as OOP.

In Python, since there is no compilation step, editing, debugging and testing is fast.

**Why Python as a hybrid language?**

One can design programs with a functional or OOPs approach.

Python is a concept with different implementations – Cpython (C background), PyPy, IRON Python(.NET) and Jython(Java background).

We are learning ‘CPython’.

>>> 4+8  
12  
>>> 8.5+9  
17.5  
>>> 4/6  
0.6666666666666666  
>>> 4\*8  
32  
>>> 4\*\*2  
16  
>>> (4\*\*6)  
4096  
>>> a=9  
>>> a  
9  
>>> print(a)  
9  
>>> b  
Traceback (most recent call last):  
  File "<pyshell#9>", line 1, in <module>  
    b  
NameError: name 'b' is not defined  
>>> a,v=4,'dee'  
>>> s="deepthi my name"  
>>> s='i'  
>>> s  
'i'  
>>> a  
4  
>>> v  
'dee'  
>>> r=7.8  
>>> type(a)  
<class 'int'>  
>>> type(v)  
<class 'str'>  
>>> type(r)  
<class 'float'>

>>> i=1  
>>> i+=1  
>>> i  
2  
>>> i+=9.9  
>>> i  
11.9  
>>> 4%2  
0  
>>> 12345//10  
1234  
>>> a,b,c,d=1,12,2.345,6.7  
>>> a+b-c\*d  
-2.711500000000001  
>>> x=(a+b)-(a\*b)/(c-(d\*\*a))  
>>> x  
15.755453501722158

>>> a=2  
>>> a  
2  
>>> type(a)  
<class 'int'>  
>>> v='4'  
>>> type(v)  
<class 'str'>  
>>> v='2'  
>>> a==v  
False  
>>> 'a'=a  
SyntaxError: cannot assign to literal  
>>> a=='a'  
False  
>>> 2=='2'  
False  
>>> 1a=9  
SyntaxError: invalid syntax  
>>> a1=3  
>>> a1  
3  
>>> \_=0  
>>> \_  
0  
>>> \_deepthi='dee'  
>>> \_deepthi  
'dee'  
>>> a=6  
>>> b=9  
>>> complex(a,b)  
(6+9j)  
>>> a='I like Python'  
>>> b="It's true."  
>>> c=a+b  
>>> c  
"I like PythonIt's true."  
>>> c=a+'. '+b  
>>> c  
"I like Python. It's true."  
>>> "I like Python."+" It's true."  
"I like Python. It's true."  
>>> a\*3  
'I like PythonI like PythonI like Python'  
>>> a=(int("1110",2))  
>>> a  
14  
>>> a=int("1110")  
>>> a  
1110  
>>> a=(int(0b1110))  
>>> a  
14  
>>> a=int("0b1110",2)  
>>> a  
14  
>>> a='a'  
>>> ord(a)  
97  
>>> ord('b')  
98  
>>> ord('A')  
65  
>>> x = b.find("e")  
>>> x  
8  
>>> b.find('I')  
0  
>>> b.find('i')  
-1  
>>> b.find('z')  
-1  
>>> b.find("e", 5, 10) It’s true  
8  
>>> b.find("e", 1, 3)  
-1  
>>> print(b.index("q"))  
Traceback (most recent call last):  
  File "<pyshell#47>", line 1, in <module>  
    print(b.index("q"))  
ValueError: substring not found  
>>> b  
"It's true."  
>>> b.index("I")  
0  
>>> len(b)  
10

>>>  a='I like Python'  
  
SyntaxError: unexpected indent  
>>>  a='I like Python'  
  
SyntaxError: unexpected indent  
>>> a='I like Python'  
>>> b="It's true."  
>>> a[0]  
'I'  
>>> a[7]  
'P'  
>>> a[0:4]  
'I li'  
>>> a[2:20]  
'like Python'  
>>> a[-1]  
'n'  
>>> a[-7]  
' '  
>>> a[-7:-1]  
' Pytho'  
>>> a[-7:0]  
''  
>>> a[-7:-2]  
' Pyth'  
>>> a[-7:]  
' Python'  
>>> a[:4]  
'I li'  
>>> a[1:5:2] (I like Python)  
' i'  
>>> a  
'I like Python'  
>>> a[2:9:2]  
'lk y'  
>>> a[::3]  
'Ii tn'  
>>> print("like">"dislike")  
True  
>>> print("aike">"dislike")  
False  
>>> 2>4  
False  
>>> @>=6  
SyntaxError: invalid syntax  
>>> 2>=4  
False  
>>> 2!=9  
True  
>>> print('e' in b)  
True  
>>> 'z' not in a  
True  
>>> c=1  
>>> not c  
False  
>>> new=a.replace('like','love')  
>>> new  
'I love Python'  
>>> z,y='like','love'  
>>> a.replace(z,y)  
'I love Python'  
>>> print(z.capitalize())  
Like  
>>> z.upper()  
'LIKE'  
>>> old='I like HCL'  
>>> print(old + '\t its true')  
I like HCL     its true  
>>> r1=r'I like HCL \n its true'  
>>> r1  
'I like HCL \\n its true'  
>>> r1='I like HCL \n its true'  
>>> r1  
'I like HCL \n its true'  
>>> r1='I like HCL /n its true'  
>>> r1  
'I like HCL /n its true'  
>>> print("integer1:%2d, float1:%5.4f, oct:%5.4o"%(234, 56.546789,(25)))  
integer1:234, float1:56.5468, oct: 0031  
>>> 'deepthi %s'%z  
'deepthi like'  
>>> print("integer1:%d, float1:%5.2f, oct:%5.4o"%(234, 56.546789,(25)))  
integer1:234, float1:56.55, oct: 0031  
>>> print('I Love  {​​​​​1}​​​​​ "{​​​​​0}​​​​​"'.format('HCL','Training'))  
I Love  Training "HCL"  
>>> print('I Love  {​​​​​0}​​​​​ {​​​​​1}​​​​​'.format('HCL','Training'))  
I Love  HCL Training  
>>> print("I Love  {​​​​​0}​​​​​ '{​​​​​0}​​​​​'".format('HCL','Training'))  
I Love  HCL 'HCL'  
>>> print(z.center(50,'#'))  
#######################like#######################

>>> print('gjfgjgj')  
gjfgjgj  
>>> Print('ghfghfgh')  
Traceback (most recent call last):  
  File "<pyshell#1>", line 1, in <module>  
    Print('ghfghfgh')  
NameError: name 'Print' is not defined  
>>> a,c,=2,8  
>>> c/a  
4.0  
>>> c%a  
0  
>>> 12345//210  
58  
>>> 12345//10  
1234  
>>> a='hi all'  
>>> a[5]  
'l'  
>>> a.index('a')  
3  
>>> a.index('i')  
1  
>>> a.index('l')  
4  
>>> a.rindex('l')  
5  
>>> a[4]  
'l'  
>>> a[-1]  
'l'  
>>> a[-7]  
Traceback (most recent call last):  
  File "<pyshell#15>", line 1, in <module>  
    a[-7]  
IndexError: string index out of range  
>>> a[-6]  
'h'  
>>> a[-1:-6] (hi all)  
''  
>>> lla ih  
SyntaxError: invalid syntax  
>>> a[::-1]  
'lla ih'  
>>> a=input('enter the string value for a')  
enter the string value for adeepthi  
>>> a  
'deepthi'  
>>> type(a)  
<class 'str'>  
>>> b=input('enter the integer value for b')  
enter the integer value for b123  
>>> type(b)  
<class 'str'>  
>>> b=int(input('enter the integer value for b'))  
enter the integer value for b123  
>>> type(b)  
<class 'int'>  
>>> c=float(input())  
2.78  
>>> c  
2.78  
>>> 2+6+7  
15  
>>> c=float(input())  
2+4+6  
Traceback (most recent call last):  
  File "<pyshell#30>", line 1, in <module>  
    c=float(input())  
ValueError: could not convert string to float: '2+4+6'  
>>> c=eval(input())  
2+4+5  
>>> c  
11  
>>> type(c)  
<class 'int'>  
>>> s=input()[0:1]  
hi everyone  
>>> s  
'h'

*#If condition*x=0  
y=-8  
**if** x>y:  
 print(**"x is greater than y"**)  
 print(**'yes'**)  
**elif**(x==y):print(**"x and y are equal"**)  
**else**:print(**"y is greater"**)

#for

#for(i=1;i<=5;i++)

**for** nita **in** range(9,10):#nita<10

print(**'i='**,nita)

s=**'yogita'**

**for** y **in** s:

print(**'letter is: '**,y)

s=**'yogita'**

y=**'yogita'**

**if** y **is not** s:#y==s

print(**'same'**)

print(**'done'**)

#while

i=1

**while** i<=5:

print(i)

j=i+0.1

**while** j<i+1:

print(j)

j+=0.1

i+=1

#even-odd

**for** l **in** range(2,22,2):

print(l)

x=**'Sohail'**

print(**'Hi,'**,x, end=**' '**)

f=133

t=56

print(**'and all'**)

#for

x = **"abcd"**

**for** i **in** range(len(x)): print(i)

#fun

print(**'yes'**)

def deepthi(x,y):#argument

print(x)

print(x-y)

deepthi(y=2,x=21)

deepthi(2,7)

print(**'no'**)

deepthi(2,8)

#fun

print(**'yes'**)

def deepthi(x,y=0):#argument #default value

print(x)

print(x-y)

deepthi()

deepthi(2,7)

print(**'no'**)

deepthi(2,8)

def a1(\*dee):

print(dee)

a1(1.1,2,3,4,5,6,5.7)

def out(a=20):#5

def inside():

print(a)

return inside()

out(5)

#recursive function

#factorial 5=5\*4\*3\*2\*1

def fact(n):#0

if n==0: return 1

return n\*fact(n-1)#5\*4\*3\*2\*1\*1

print(fact(5))

>>> javadevops='deepthi','nita','sagarika','macha','jatin'  
>>> javadeops  
Traceback (most recent call last):  
  File "<pyshell#1>", line 1, in <module>  
    javadeops  
NameError: name 'javadeops' is not defined  
>>> javadevops  
('deepthi', 'nita', 'sagarika', 'macha', 'jatin')  
>>> s='sohail'  
>>> s  
'sohail'  
>>> javadevops=['deepthi','nita','sagarika','macha','jatin']  
>>> type(javadevops  
     )  
<class 'list'>  
>>> a=[1,2.3,4.5,1,'dee',5]  
>>> a  
[1, 2.3, 4.5, 1, 'dee', 5]  
>>> a[0]  
1  
>>> a[7]  
Traceback (most recent call last):  
  File "<pyshell#11>", line 1, in <module>  
    a[7]  
IndexError: list index out of range  
>>> a[5]  
5  
>>> len(a)  
6  
>>> a1=[1,2,3,4,5]  
>>> sum(a1)  
15  
>>> v=[1,2,3,[4,5,6],6,7]  
>>> v[2]  
3  
>>> v[4]  
6  
>>> v[5]  
7  
>>> v[3]  
[4, 5, 6]  
>>> v[3][0]  
4  
>>> v[3][0:1]  
[4]  
>>> x=[]  
>>> x.insert(1)  
Traceback (most recent call last):  
  File "<pyshell#24>", line 1, in <module>  
    x.insert(1)  
TypeError: insert expected 2 arguments, got 1  
>>> x.insert(1,0)  
>>> x  
[0]  
>>> q=[i for i in rang(1,11,2)]  
Traceback (most recent call last):  
  File "<pyshell#27>", line 1, in <module>  
    q=[i for i in rang(1,11,2)]  
NameError: name 'rang' is not defined  
>>> q=[i for i in range(1,11,2)]  
>>> q  
[1, 3, 5, 7, 9]

>>> t=list(range(1,5))  
>>> t  
[1, 2, 3, 4]  
>>> s2='yogita'  
>>> e=list(s2)  
>>> e  
['y', 'o', 'g', 'i', 't', 'a']  
>>> e1=split('')  
Traceback (most recent call last):  
  File "<pyshell#35>", line 1, in <module>  
    e1=split('')  
NameError: name 'split' is not defined  
>>> e1=s2.split('')  
Traceback (most recent call last):  
  File "<pyshell#36>", line 1, in <module>  
    e1=s2.split('')  
ValueError: empty separator  
>>> e1=s2.split(' ')  
>>> e1  
['yogita']

>>> t=list(range(1,5))  
>>> t  
[1, 2, 3, 4]  
>>> s2='yogita'  
>>> e=list(s2)  
>>> e  
['y', 'o', 'g', 'i', 't', 'a']  
>>> e1=split('')  
Traceback (most recent call last):  
  File "<pyshell#35>", line 1, in <module>  
    e1=split('')  
NameError: name 'split' is not defined  
>>> e1=s2.split('')  
Traceback (most recent call last):  
  File "<pyshell#36>", line 1, in <module>  
    e1=s2.split('')  
ValueError: empty separator  
>>> e1=s2.split(' ')  
>>> e1  
['yogita']

Lists are just like the arrays.

Lists need not be homogeneous always which makes it a most powerful tool in python.

A single list may contain Datatypes like Integers, Strings, as well as Objects.

Lists are mutable, and hence, they can be altered even after their creation.

List values are not unique.

List is ordered collection of elements.

Syntax:

L=[‘as’, “Iwas”, 2, 4.5, 2,[4,6], [2,5,6]]

>>> a=()  
>>> a  
()  
>>> type(a)  
<class 'tuple'>  
>>> x=[]  
>>> type(x)  
<class 'list'>  
>>> a=(1,2,3.6,4,5,5,'dee')  
>>> a  
(1, 2, 3.6, 4, 5, 5, 'dee')  
>>> a[0]  
1  
>>> a.append(2)  
Traceback (most recent call last):  
  File "<pyshell#8>", line 1, in <module>  
    a.append(2)  
AttributeError: 'tuple' object has no attribute 'append'  
>>> c=(1,2)  
>>> a.extend(c)  
Traceback (most recent call last):  
  File "<pyshell#10>", line 1, in <module>  
    a.extend(c)  
AttributeError: 'tuple' object has no attribute 'extend'  
>>> max(c)  
2  
>>> sum(c)  
3  
>>> len(a)  
7  
>>> a.index(2)  
1  
>>> a.count(5)  
2  
>>> any(a)  
True  
>>> any(z)  
Traceback (most recent call last):  
  File "<pyshell#17>", line 1, in <module>  
    any(z)  
NameError: name 'z' is not defined  
>>> a[1:4]  
(2, 3.6, 4)  
>>> a[1]=(6)  
Traceback (most recent call last):  
  File "<pyshell#19>", line 1, in <module>  
    a[1]=(6)  
TypeError: 'tuple' object does not support item assignment  
>>> tup=('one',)  
>>> a  
(1, 2, 3.6, 4, 5, 5, 'dee')  
>>> b  
Traceback (most recent call last):  
  File "<pyshell#22>", line 1, in <module>  
    b  
NameError: name 'b' is not defined  
>>> b=(1,2,3,'dee')  
>>> (a,e,c,d)=b  
>>> d  
'dee'  
>>> a  
1  
>>> z=1  
>>> any(z)  
Traceback (most recent call last):  
  File "<pyshell#28>", line 1, in <module>  
    any(z)  
TypeError: 'int' object is not iterable  
>>> any(x)  
False  
>>> any(b)  
True  
>>> (1,a,v,u)=b  
SyntaxError: cannot assign to literal  
>>> (tanmay, nita, neha,ashi)=b  
>>> tanmay  
1  
>>> a,b=(5,6),(2,4)  
>>> if (a>b): print('a is greater')  
else: print('b is greater')

a is greater  
>>>

**TUPLE**

A tuple is a data structure that is an immutable, or unchangeable, ordered sequence of elements.

Elements are not unique in a tuple.

Syntax:tup=(‘one’,’two’,3)

>>> menu={}  
>>> type(menu)  
<class 'dict'>  
>>> menu={1:5,'a':3,1.5:4,'chandu':[1,2,3]}  
>>> menu  
{1: 5, 'a': 3, 1.5: 4, 'chandu': [1, 2, 3]}  
>>> munu1={'idli':10,'Idli':20,'idli':5}  
>>> munu1  
{'idli': 5, 'Idli': 20}  
>>>  menu1={'idli':5,'Idli':5,'idli':5}  
  
SyntaxError: unexpected indent  
>>> menu1={'idli':5,'Idli':5,'idli':5}  
>>> menu1  
{'idli': 5, 'Idli': 5}  
>>> a=[1,2,4]  
>>> a[0]  
1  
>>> menu1[0]  
Traceback (most recent call last):  
  File "<pyshell#11>", line 1, in <module>  
    menu1[0]  
KeyError: 0  
>>> menu1['idli']  
5  
>>> a=9  
>>> a=5  
>>> a  
5  
>>> a=[12,465]  
>>> a=(7,8,9)  
>>> a  
(7, 8, 9)  
>>> menu1['idli']=menu1['some']  
Traceback (most recent call last):  
  File "<pyshell#19>", line 1, in <module>  
    menu1['idli']=menu1['some']  
KeyError: 'some'  
>>> menu1['idli']=10  
>>> menu1['idli']  
10  
>>> menu1  
{'idli': 10, 'Idli': 5}  
>>> menu2=menu1.copy()  
>>> menu2  
{'idli': 10, 'Idli': 5}  
>>> menu2.update({'dosa':40})  
>>> menu2  
{'idli': 10, 'Idli': 5, 'dosa': 40}  
>>> del menu2['idli']  
>>> menu2  
{'Idli': 5, 'dosa': 40}  
>>> menu2.items()  
dict\_items([('Idli', 5), ('dosa', 40)])  
>>> menu2.keys()  
dict\_keys(['Idli', 'dosa'])  
>>> menu2.values()  
dict\_values([5, 40])  
>>>

>>> m={'idli':10,'Idli':20,'Dosa':30,'bonda':40,'Vada':50}  
>>> m.sort()  
Traceback (most recent call last):  
  File "<pyshell#1>", line 1, in <module>  
    m.sort()  
AttributeError: 'dict' object has no attribute 'sort'  
>>> m1=list(m.keys())  
>>> m1.sort()  
>>> m1  
['Dosa', 'Idli', 'Vada', 'bonda', 'idli']  
>>> len(m)  
5  
>>> str(m)  
"{'idli': 10, 'Idli': 20, 'Dosa': 30, 'bonda': 40, 'Vada': 50}"  
>>> for i in m: print(':'.join((i,str(m[i]))))

idli:10  
Idli:20  
Dosa:30  
bonda:40  
Vada:50  
>>> m['idli']  
10  
>>>

>>> menu={'idli':1,'dosa':2,'poori':3}  
>>> menu1=menu.copy()  
>>> del menu1['idli']  
>>> menu1.update({'vada':4})  
>>> menu1  
{'dosa': 2, 'poori': 3, 'vada': 4}  
>>> for i in list(menu.keys()):  
    if i in list(menu1.keys()):  
        print(True)  
    else: print(False)

False  
True  
True  
>>> del menu  
>>> menu  
Traceback (most recent call last):  
  File "<pyshell#11>", line 1, in <module>  
    menu  
NameError: name 'menu' is not defined  
>>> len(menu1)  
3  
>>> menu1.clear()  
>>> menu1  
{}  
>>>

'''def sum(a,b):

return a+b

c=sum(3,4)

print(c)

#Annonimus/ Lambda

sum=lambda a,b:a+b

print(sum(1,2))'''

#filter

num=[1,2,5,7,3,4,8]

even=list(filter(lambda n:n%2==0,num))

print(even)

#Map

sum=list(map(lambda n:n+3,num))

print(sum)

#reduce

from functools import reduce

s=reduce(lambda a,b:a+b,num)

print(s)

def addition(n):

return n + n

numbers = (1, 2, 3, 4)

result = map(addition, numbers)

print(list(result))

numbers = (1, 2, 3, 4)

result = map(lambda x: x + x, numbers)

print(list(result))

numbers1 = [1, 2, 3]

numbers2 = [4, 5, 6]

result = map(lambda x, y: x + y, numbers1, numbers2)

print(list(result))

def hotel(x):*#0*

menu={​​​​​​

0:**'idli'**,

1:**'dosa'**,

2:**'roti'**,

}​​​​​​

*#print(menu[0])*

return menu.get(x,**'not available'**)*#menu[0]*

a=int(input(**'what would you like to have?'**))

b=hotel(a)

print(b)

def menu():

print(**'1:dosa** \n **2:idli'**)

choice=int(input(**'enter your choice'**))

def dosa(): print(**'paper,masala'**)

def idli(): print(**'plain,rava'**)

dict={​​​​​​

1:dosa,

2:idli

}​​​​​​

dict.get(choice)()

menu()

def menu():

print(**'i:dosa** \n **2:idli'**)

choice=int(input(**'enter'**))

if choice==1: print(**'paper,masala'**)

elif choice==2: print(**'Plain,rava'**)

else:exit()*#print('not available')*

menu()

print(**'done'**)

>>> import math  
>>> math.ceil(-33.1)  
-33  
>>> math.ceil(5.7)  
6  
>>> v=5  
>>> v.\_\_ceil\_\_()  
5  
>>> math.floor(7)  
7  
>>> math.floor(7.9)  
7  
>>> round(1.1)  
1  
>>> round(1.8)  
2  
>>> rount(10.6)  
Traceback (most recent call last):  
  File "<pyshell#9>", line 1, in <module>  
    rount(10.6)  
NameError: name 'rount' is not defined  
>>> round(10.6)  
11  
>>> round(10.4)  
10  
>>> round(10.5)  
10  
>>> round(11.5)  
12  
>>> round(2.5)  
2  
>>> math.copysign(3.4,0.0)  
3.4  
>>> math.copysign(3.4,-0.0)  
-3.4  
>>>  
>>> math.factorial(5)  
120  
>>> math.fabs(-8.9)  
8.9  
>>> math.isnan(5)  
False  
>>> math.isnan('anusha')  
Traceback (most recent call last):  
  File "<pyshell#21>", line 1, in <module>  
    math.isnan('anusha')  
TypeError: must be real number, not str  
>>> math.lcm(3,13)  
39  
>>> math.gcd(3,13)  
1  
>>> a=[0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1]  
>>> math.fsum(a)  
0.8  
>>> sum(a)  
0.7999999999999999  
>>>

*#module=collection of functions*

*#package=collection of modules*

import math

a=math.sqrt(8)

print(a)

from math import sqrt

print(sqrt(5))

print(math.ceil(9))

import sys

sys.stdout.write(**'deeps'**)

print(sys.version)

print(sys.argv)

import sys

def print\_to\_stderr(\*a):

# Here a is the array holding the objects

# passed as the arguement of the function

print(\*a, file = sys.stderr)

print\_to\_stderr(**"Hello World"**)

import sys

age = 19

if age < 18:

# exits the program

sys.exit(**"Age less than 18"**)

else:

print(**"Age is not less than 18"**)

print(**'done'**)

print(sys.path)

print(sys.modules)

#os

import os

**'''os.chdir('C:**\\**Users**\\**S KRISHNA**\\**PycharmProjects**\\**mypython project')**

**print(os.getcwd())'''**

os.mkdir(**'dir1'**)

#os.rmdir('dir1')

import random

list1 = [1, 2, 3, 4, 5, 6]

print(random.choice(list1))

string = **"striver"**

print(random.choice(string))

print(random.random())

random.seed(100)

print(**"The mapped random number with 5 is : "**, end=**""**)

print(random.random())

sample\_list = [**'A'**, **'B'**, **'C'**, **'D'**, **'E'**]

random.shuffle(sample\_list)

print(sample\_list)

print(**"The random floating point number between 5 and 10 is : "**, end=**""**)

print(random.uniform(5, 10))

print(random.randint(1,10))

print(random.randrange(12))

print(random.randrange(1,10,2))

from datetime import date

from datetime import datetime

from datetime import time

from datetime import timedelta

import time

import datetime

print(datetime.datetime.today())

print(time.time())

print(time.localtime())

print(date.today())

#print(datetime.now())'''

i=date.today()

print(**"today'd date is:"**, i.day,**'/'**,i.month, i.year,i.weekday())

i=datetime.datetime.now()

print(i.strftime(**"%a, %d, %b, %y"**))

print(i.strftime(**"%A, %d, %B, %Y"**))

print(i.strftime(**'%c'**))

print(i.strftime(**'%x'**))

print(i.strftime(**'%X'**))

i=datetime.datetime.now()

print(**'today:'**,i)

print(timedelta(days=365,hours=8,minutes=45))

print(**'one year from now:'**+str(datetime.datetime.now()+timedelta(days=455,hours=3)))

from salary\_increment import increment,nothing

salary= increment(45000,10)

print(salary)

nothing()

import saldecreament

salary= saldecreament.decrement(45000,10)

print(salary)

import salary\_increment as i

salary= i.increment(45000,10)

print(salary)

import dee.salary\_increment as i

import dee.saldecreament

salary= i.increment(45000,10)

print(salary)

i.nothing()

salary= dee.saldecreament.decrement(45000, 10)

print(salary)

#from dee import salary\_increment as i, saldecreament

import sys

sys.path.append(**'C:/Users/S KRISHNA/Desktop'**)

import salary\_increment as i

import saldecreament

salary= i.increment(45000,10)

print(salary)

i.nothing()

salary= saldecreament.decrement(45000, 10)

print(salary)

#switch, case,default,do while

#iterators, generators

a=[1,2,3,4,5]

#for i in a: print(i)

b=iter(a)

print(b.\_\_next\_\_())

print(b.\_\_next\_\_())

print(next(b))

**'''print(next(b))**

**print(next(b))**

**print(next(b))**

**print(next(b))**

**print(next(b))**

**print(next(b))'''**

#switch, case,default,do while

#iterators, generators

def ten():

n=1

while n<=10:

sq=n\*n#1,4,9,16

yield sq

n+=1

v=ten()

#print(v)

for i in v: print(i)

#calendar module

import calendar

for i in range(1,13):

a=calendar.monthcalendar(2022,i)#jan

#print(a)

week1=a[0]

week2=a[1]

if week1[calendar.FRIDAY]!=0:

salary=week1[calendar.FRIDAY]

else: salary=week2[calendar.FRIDAY]

print(calendar.month\_name[i],salary)