Experiment No : 08

Aim : Write python programs to understand implementing User defined and Anonymous Functions .

Description :

A function that you define yourself in a program is known as user defined function. You can give any name to a user defined function, however you cannot use the Python keywords as function name. In python, we define the user-defined function using def keyword, followed by the function name.

Types of Functions in Python

Python Function with no argument and no return value.

Function with no argument and with a Return value.

Python Function with argument and No Return value.

Function with argument and return value.

Implementation :

Code :

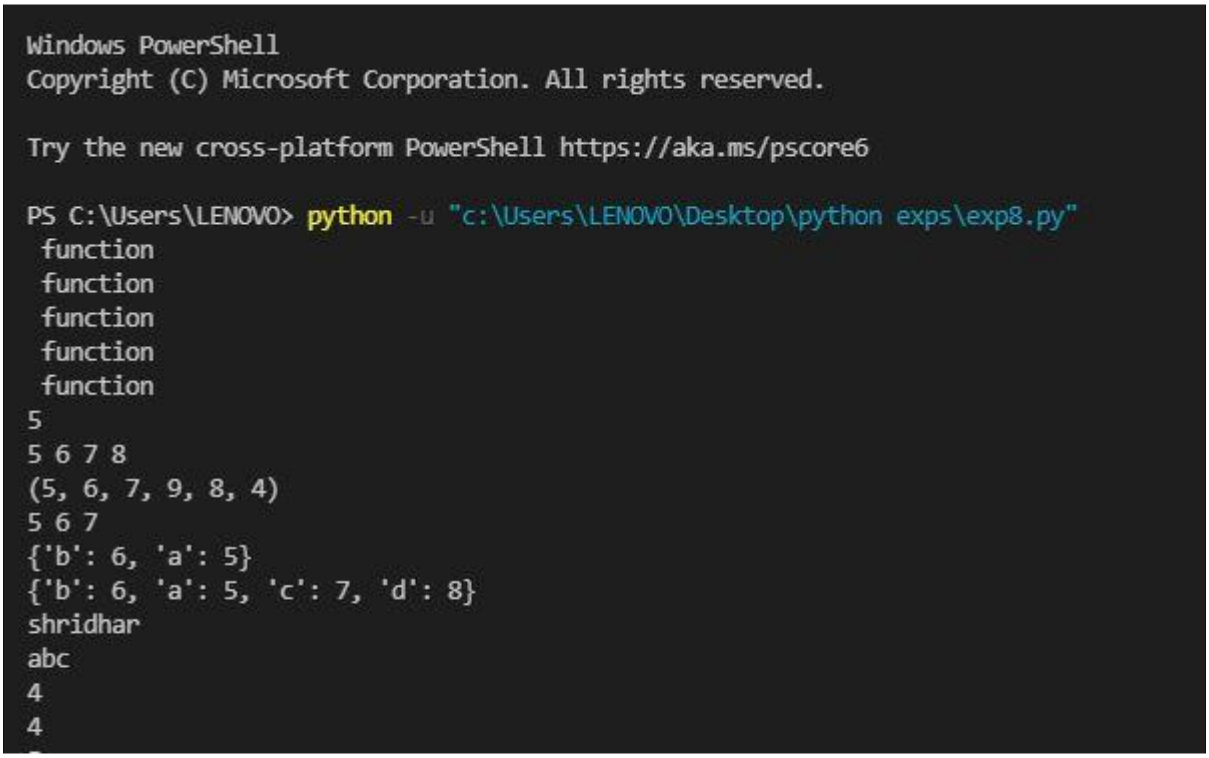
```
#USER DEFINED FUNCTION
# Creating function use def keyword.
def school():
  print(" function")


#calling function . so many times can run ( call)
school()
school()
school()
school()
school()
#parameter and arguments
# parameter(mention in round bracket, variable mentioned (a) in the definition of particular function)and
#arguments (when you call function ,mention value  and pass to the function)
#variable a is parameter
def school(a):
  print(a)
#calling function
#value 5 is argument
school(5)
```

```
#More than one parameter and arguments(no limitation).school function ,expecting four arguments

# error message(four positional arguments. but you have given 5 )

def school(a,b,c,d):

  print(a,b,c,d)

#calling function

school(5,6,7,8)


"""

# during coding and logic development, donot know how many arguments have to be pass at particular situation.

#error function

def school(a,b,c,d):

  print("a,b,c,d")

#calling function

school(5,6,7,8,9)


#error function message ( school missing one required argument )

def school(a,b,c,d):

  print("a,b,c,d")

#calling function

school(5,6,7)


"""
```

```
# treated as tuple of arguments. pass  no. of arguments  (no error will come , use *)
def school(*a):
  print(a)
#explicitly giving value to parameter
#calling function
school(5,6,7,9,8,4)

def school(a,b,c):
  print(a,b,c)
#school(5,6,7)
#school(6,7,5)


#calling function
school(b=6, a=5, c=7)
#school(b=6, a=5)   error

# keywords arguments
def school(**a):
  print(a)
#calling function
school(b=6, a=5)
# trying to recieve dictionnary of arguments
```

```
def school(**a):
  print(a)
#calling function
school(b=6, a=5, c=7,d=8)
# dictionary style output will show
#default value to parameter
def school(a="shridhar"):
  print(a)
#calling function .with no argument pass. it will consider default value.
school()

#explicit value consider like abc
def school(a="shridhar"):
  print(a)
#calling function .output shows abc value (not shridhar value)
school("abc")

# result return (value 4) ( functions return type nature)
def double(a):
 return a*2
print(double(2))

# taking in varable (b) (value taking in variable)
def double(a):
 return a*2

b=double(2)

print(b)
```

Output :



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\LENOVO> python -u "c:\Users\LENOVO\Desktop\python exps\exp8.py"
 function
 function
 function
 function
 function
5
5 6 7 8
(5, 6, 7, 9, 8, 4)
5 6 7
{'b': 6, 'a': 5}
{'b': 6, 'a': 5, 'c': 7, 'd': 8}
shridhar
abc
4
4
```

#ANONYMOUS FUNCTION OR LAMBDA FUNCTION

```
   # Anonymous function or lambda function
 # print value
show = lambda x : print(x)
show(5)
# perform addition
add = lambda x,y : x+y
print(add(5,2))
# perform addition and subtraction
add_sub = lambda x,y : (x+y , x-y)
a, s = add_sub(5,2)


print(a)
print(s)
"""
# generate error(if write like x=10 or put , )it will give outpout like
#( cannot asign to lambda)
add_sub = lambda x,y : x=10 : (x+y , x-y)
a, s = add_sub(5,2)


print(a)
print(s)
"""
```

# it will take default argument (y=2). output is 7

add = lambda x,y=3 : x+y

print(add(5,2))

# it will take assign (y=3) value .Here argument value is missing.output is 8

add = lambda x,y=3 : x+y

print(add(5))

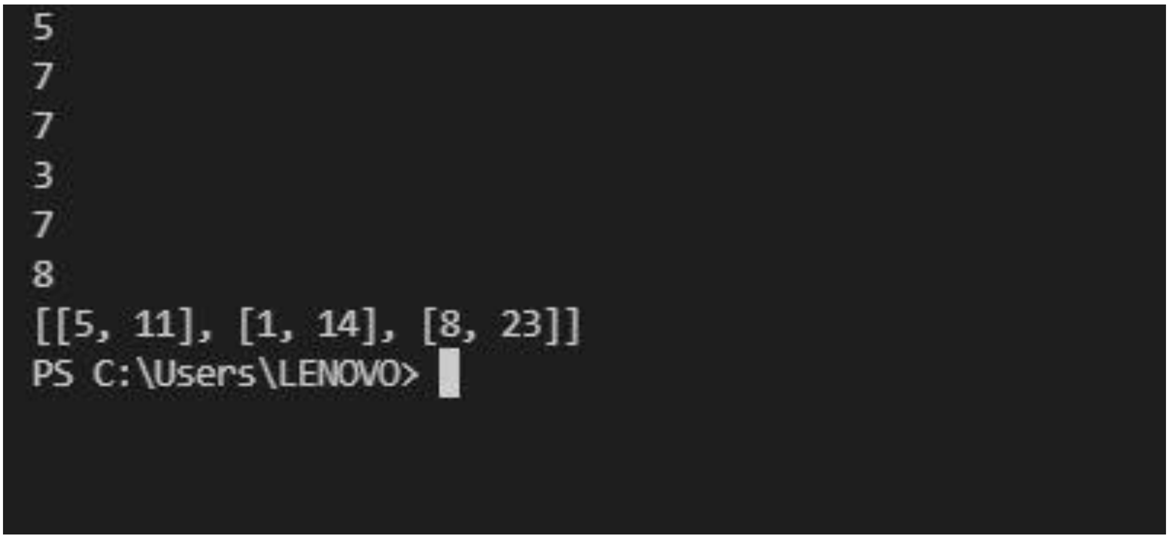
# SORT

a=[[1,14], [5,11], [8,23]]

a.sort(key=lambda x:x[1])

print(a)

Output :



```
5
7
7
3
7
8
[[5, 11], [1, 14], [8, 23]]
PS C:\Users\LENOVO> 
```

Conclusion : Therefore we have successfully implemented python programs to understand implementing User defined and Anonymous Functions