**PRACTICAL LAB BOOK**

**DATA ANALYSIS using PYTHON(DAP)**

**K.N. PRAJWAL SAI**

**18BBTCS046**

**7th Sem CSE.**

**STRINGS**

**Objective – WAP to print Hello World!**

var1 = 'Hello World!'

var2 = "Python Programming"

print(var1," ",var2)

Output

Hello World!   Python Programming

Program

var1 = 'Hello World!'

print ("Updated String :- ", var1[0:6] + 'Prajwal')

Output

Updated String :- Hello Prajwal

Program

str1 = input("Please Enter Your Own String : ")

str2 = str1

str3 = str1[:]

str4 = str1[2:6]

print("The Final String : Str2  = ", str2)

print("The Final String : Str3  = = ", str3)

print("The Final String : Str4  = = ", str4)

Output

Please Enter Your Own String : hello class

The Final String : Str2 = hello class

The Final String : Str3 = = hello class

The Final String : Str4 = = llo

Program Objective

#Python String capitalize() method returns a copy of the string with only its first character capitalized.

str = "this is string example....wow!!!";

print ("str.capitalize() : ", str.capitalize())

**OUTPUT**

str.capitalize() : This is string example....wow!!!

**PROGRAM**

#center() returns centered in a string of length width. Padding is done using the specified fillchar. Default filler is a space.

str = "this is string example....wow!!!"

print ("str.center(40, 'a') : ", str.center(40, '\*'))

**OUTPUT**

str.center(40, 'a') : \*\*\*\*this is string example....wow!!!\*\*\*\*

**PROGRAM OBJECTIVE**

#join() returns a string in which the string elements of sequence have been joined by str separator.

s = "  \*  \*  ";

seq = ("abc", "bttt", "cqweqe"); # This is sequence of strings.

print (s.join( seq ))

**OUTPUT**

str = "this is string example....wow!!!";

print ("Length of the string: ", len(str))

**PROGRAM**

#lstrip() returns a copy of the string in which all chars have been stripped from the beginning of the string (default whitespace characters)

str = "     this is string example....wow!!!     ";

print (str.lstrip())

str = "88888888this is string example....wow!!!999999";

print (str.lstrip('8'))

print (str.rstrip('9'))

OUTPUT

this is string example....wow!!!

this is string example....wow!!!999999

88888888this is string example....wow!!!

**Program**

str = "this is string example....wow!!!";

print ("str.capitalize() : ", str.upper())

**Output**

str.capitalize() : THIS IS STRING EXAMPLE....WOW!!!

**PROGRAM**

str = "this is string example....wow!!!";

print (str.swapcase())

str = "THIS IS STRING EXAMPLE....WOW!!!";

print (str.swapcase())

**OUTPUT**

THIS IS STRING EXAMPLE....WOW!!!

this is string example....wow!!!

**LISTS**

PROGRAM

list1 = ['physics', 'chemistry', 1997, 2000];

list2 = [1, 2, 3, 4, 5, 6, 7 ];

print ("list1[0]: ", list1[2])

print ("list2[1:5]: ", list2[1:3])

OUTPUT

list1[0]: 1997

list2[1:5]: [2, 3]

**PROGRAM**

list = ['physics', 'chemistry', 1997, 2000];

print ("Value available at index 2 : ")

print (list[2])

list[2] = 999999;

print ("New value available at index 2 : ")

print (list[2])

**OUTPUT**

Value available at index 2 :

1997

New value available at index 2 :

999999

**PROGRAM**

list1, list2 = [123, 'xyz', 'zara'], [456, 'abc']

print ("First list length : ", len(list1))

print ("Second list length : ", len(list2))

**OUTPUT**

First list length : 3

Second list length : 2

**PROGRAM**

aList = [123, 'xyz', 'zara', 'abc'];

print(aList)

print ("List after popping last element : ", aList.pop())

print(aList)

print ("List after popping element from mentioned index: ", aList.pop(2))

print(aList)

aList.insert(3, 2009)

print(aList)

**OUTPUT**

[123, 'xyz', 'zara', 'abc']

List after popping last element : abc

[123, 'xyz', 'zara']

List after popping element from mentioned index: zara

[123, 'xyz']

[123, 'xyz', 2009]

**PROGRAM**

aList = ['naveen', 'shabnam', 'sonia', 'ali', 'nive'];

aList.sort();

print ("List : ", aList)

**OUTPUT**

List : ['ali', 'naveen', 'nive', 'shabnam', 'sonia']

**IF- ELSE in Python**

**PROGRAM**- TO print even or odd

 num = int(input("enter the number?"))

    if num%2 == 0:

        print("Number is even...")

    else:

        print("Number is odd...")

**OUTPUT**

enter the number?5

Number is odd...

**PROGRAM TO KNOW LARGEST NUMBER**

 a = int(input("Enter a- "));

    b = int(input("Enter b- "));

    c = int(input("Enter c- "));

    if a>b and a>c:

        print("a is largest");

    if b>a and b>c:

        print("b is largest");

    if c>a and c>b:

        print("c is largest");

**OUTPUT**

Enter a- 10

Enter b- 20

Enter c- 12

b is largest

**Objective:** To check for age above 18 using if-else statement

**CODE:** age = int(input("Enter age"))

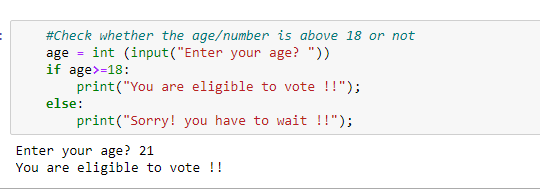
if age >= 18:

print("Your age is 18+")

print("You are eligible to vote")

print("This statement is outside the body of if statement")

**OUPUT**



**Objective:** To know whether you are eligible to vote or not based on your age

**CODE :**

 age = int (input("Enter your age? "))

    if age>=18:

        print("You are eligible to vote !!");

    else:

        print("Sorry! you have to wait !!");

**OUTPUT:**

Enter your age? 4

Sorry! you have to wait !!

PROGRAM to know GRADE based on marks

 marks = int(input("Enter the marks? "))

    if marks > 85 and marks <= 100:

       print("Congrats ! you scored grade A ...")

    elif marks > 60 and marks <= 85:

       print("You scored grade B + ...")

    elif marks > 40 and marks <= 60:

       print("You scored grade B ...")

    elif (marks > 30 and marks <= 40):

       print("You scored grade C ...")

    else:

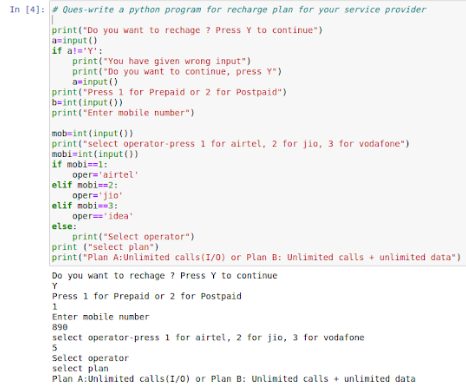
       print("Sorry you are fail ?")

OUTPUT

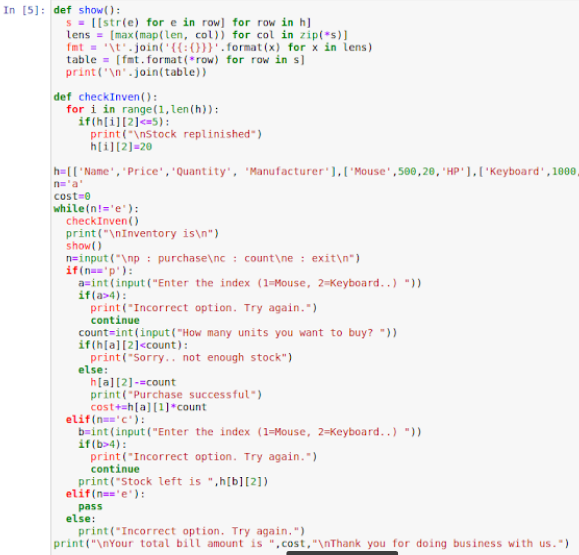
Enter the marks? 56

You scored grade B ...

**RECHARGE PLAN**



**INVENTORY SYSTEM**



**LAND LORD TENANT program**

utilities=input("Enter all bills to be shared separated with a comma\n")

l1=utilities.split(",")

l2=[]

sum=0

n=int(input("Enter no. of tenants "))

n+=1

for i in range(len(l1)):

  print("Enter total amount for",l1[i],"bill ")

  l2.append(float(input()))

for i in range(len(l1)):

  sum+=round(l2[i]/n,2)

  print("The",l1[i],"bill is split as",round(l2[i]/n,2),"per head")

print("\nThe total per head is",sum)

**OUTPUT**

Enter all bills to be shared separated with a comma

w,e,n

Enter no. of tenants 1

Enter total amount for w bill

100

Enter total amount for e bill

200

Enter total amount for n bill

30

The w bill is split as 50.0 per head

The e bill is split as 100.0 per head

The n bill is split as 15.0 per head

The total per head is 165.0

**FOOD DELIVERY program**

deli=['Zomato','uber food','Swiggy','Amazon Food']

foo=['Burger','Roll','Pizza']

pri=[120,100,200]

repeat='y'

while(repeat!='n'):

  user=int(input("1. Zomato\n2. Swiggy\n3. Amazon Food\nEnter your preferred delivery partner "))

  choice=int(input("\n1. Burger : 120/-\n2. Roll : 100/-\n3. Pizza : 200/-\nEnter your choice "))

  loc=input("\nEnter your address ")

  print("\nYour order for",foo[choice-1],"has been placed on",deli[user-1],"and will arrive at your address",loc,"\nTotal payment to be made is ",pri[choice-1])

  repeat=input("Do you wish to order again? (y/n)\n")

**OUTPUT**

1. Zomato

2. Swiggy

3. Amazon Food

Enter your preferred delivery partner 2

1. Burger : 120/-

2. Roll : 100/-

3. Pizza : 200/-

Enter your choice 1

Enter your address 3

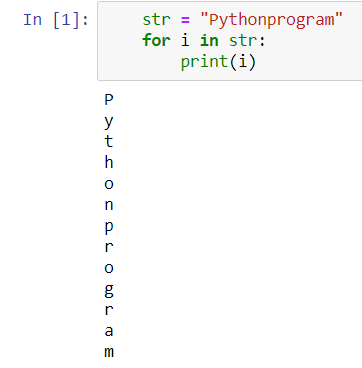
Your order for Burger has been placed on uber food and will arrive at your address 3

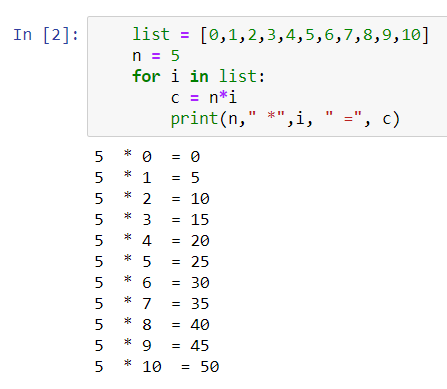
Total payment to be made is 120

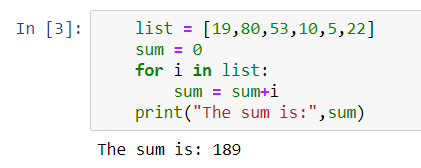
Do you wish to order again? (y/n)

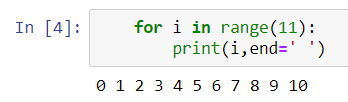
N

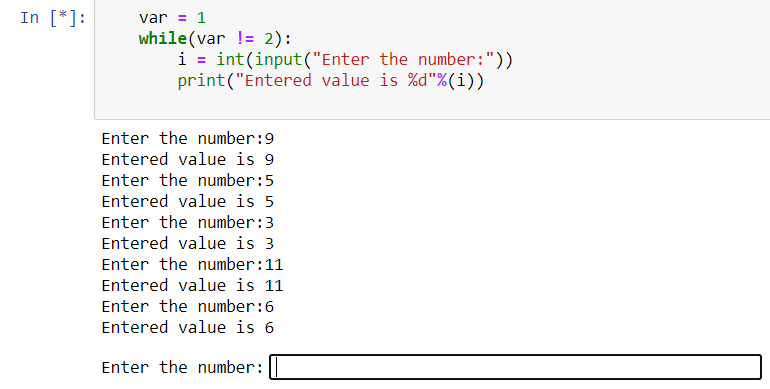
**Loops-Breaks—Continue Pass in Python**

****

****

****

****

****

**Dictionaries in Python**

**PROGRAM**

dict={'name':'PRAJWAL','designation':'AsP','qual':'phd','univ':'cmr'}

print(dict['name'])

print(dict['qual'])

OUTPUT

PRAJWAL

phd

**PROGRAM**

dict={'name':'prajwal','designation':'AsP','qual':'phd','univ':'cmr'}

dict['name']='SAI'

print(dict['name'])

dict['class']='cse7sem'

print(dict['class'])

**OUTPUT**

SAI

cse7sem

**PROGRAM**

dict={'name':'PRAJWAL','designation':'AsP','qual':'phd','univ':'cmr', 'name':'AJAY'}

dict2={'edu':'btech-mtech','students':'7sem'}

print(dict)

print(dict2)

dict.update(dict2)

print(dict)

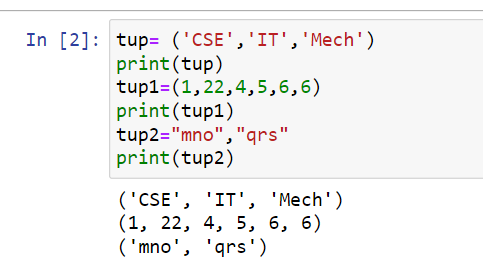
**OUTPUT**

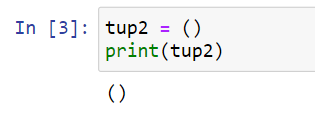
{'name': 'AJAY', 'designation': 'AsP', 'qual': 'phd', 'univ': 'cmr'}

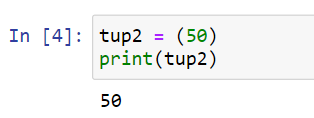
{'edu': 'btech-mtech', 'students': '7sem'}

{'name': 'AJAY', 'designation': 'AsP', 'qual': 'phd', 'univ': 'cmr', 'edu': 'btech-mtech', 'students': '7sem'}

**TUPLES in Python**

****

****

****

**PROGRAM**

tup1= ('cse','it','mech','ece','electrical')

print(tup1)

tup2= ('PRAJWAL','nishitha','roy')

print(tup2)

a=len(tup1)

b=len(tup2)

print(a)

print(b)

print(len(tup1))

**OUTPUT**

('cse', 'it', 'mech', 'ece', 'electrical')

(‘PRAJWAL', 'nishitha', 'roy')

5

3

5

**PROGRAM**

tup1= ('cse','it','mech','ece','electrical')

print(tup1)

tup2= ('prajwal','nishitha','roy')

print(tup2)

print(max(tup1))

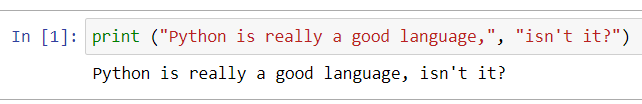
**OUTPUT**

('cse', 'it', 'mech', 'ece', 'electrical')

('prajwal', 'nishitha', 'roy')

mech

**Files in Python**

****

**PROGRAM**

# Open a file

fo = open("cse7sem.txt", "wb")

print ("Name of the file: ", fo.name)

# Close opend file

fo.close()

**OUTPUT**

Name of the file: cse7sem.txt

**PROGRAM**

f = open("cse7sem1.txt", "a")

f.write("sharma")

f.close()

#open and read the file after the appending:

f = open("cse7sem1.txt", "r")

print(f.read())

**OUTPUT**

sharmaPrajwalPrajwalsharma

**PROGRAM**

fo = open("cse7sem.txt", "wb")

print ("Name of the file: ", fo.name)

print ("Closed or not : ", fo.closed)

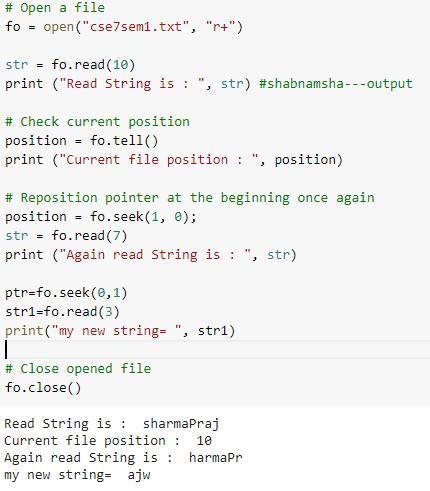
print ("Opening mode : ", fo.mode)

**OUTPUT**

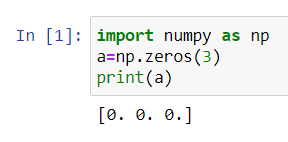
Name of the file: cse7sem.txt

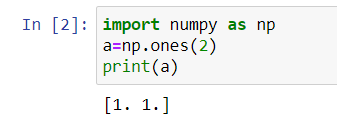
Closed or not : False

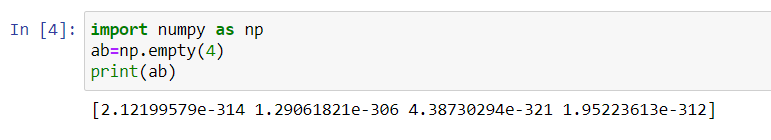
Opening mode : wb

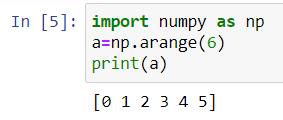
****

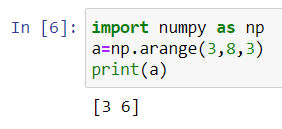
**NumPy in Python**

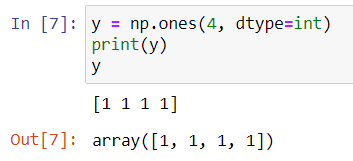
****

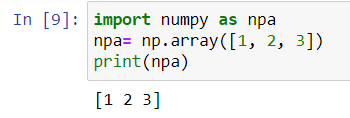
****

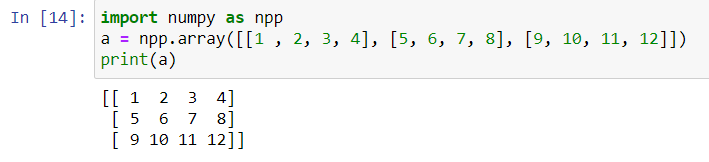
****

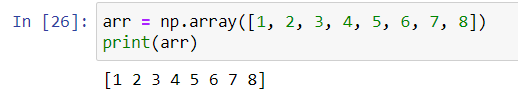
****

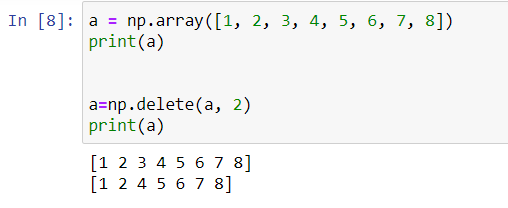
****

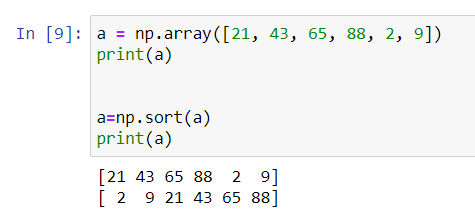
****

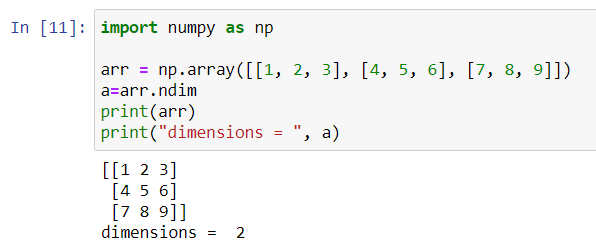
****

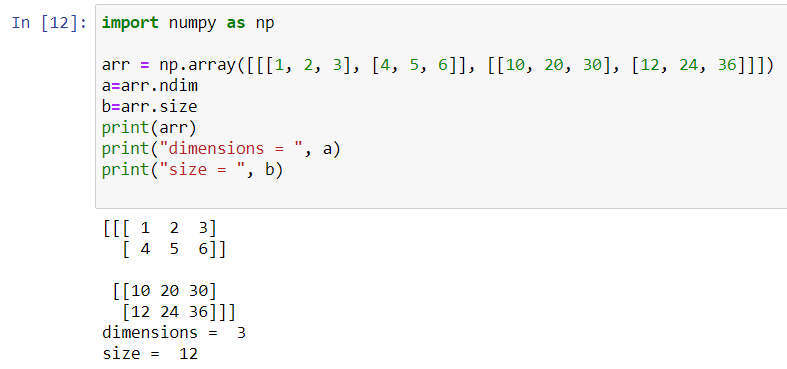
****

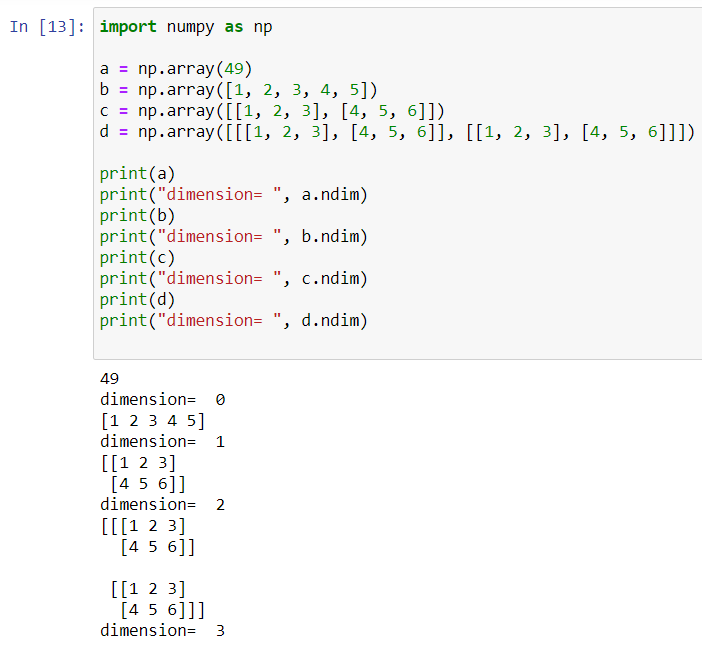
****

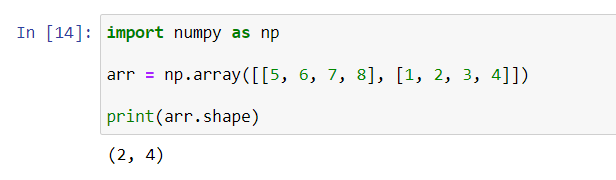
****

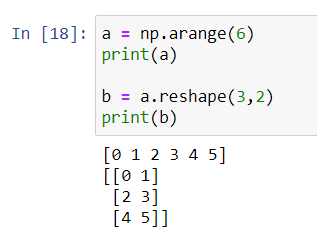
****

****

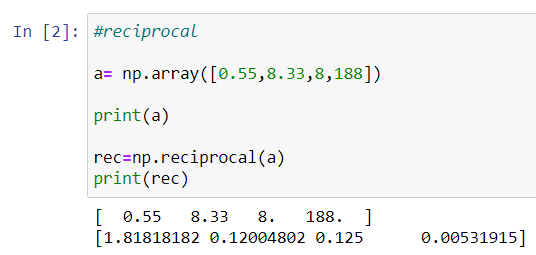
****

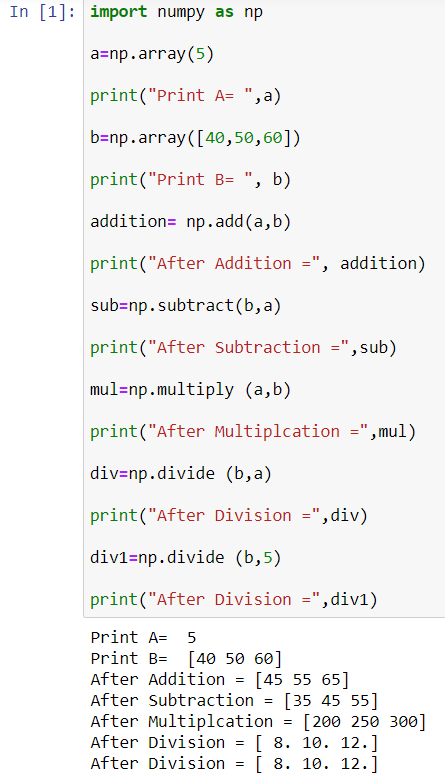
****

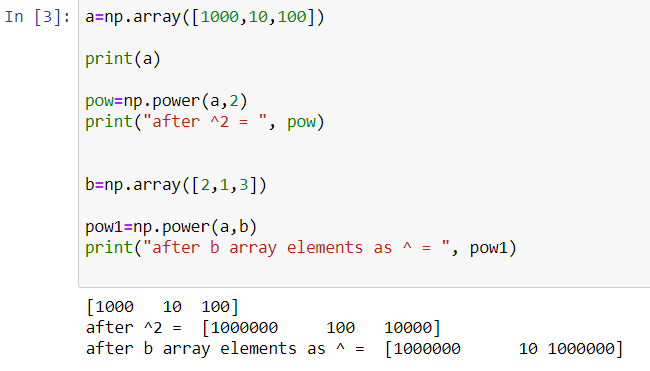
****

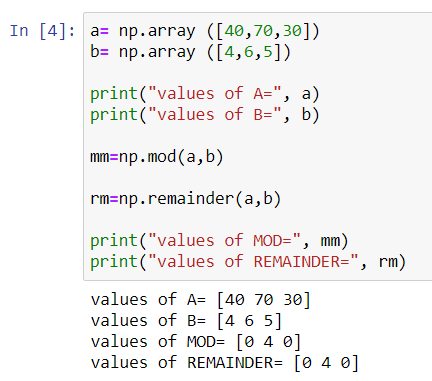
****

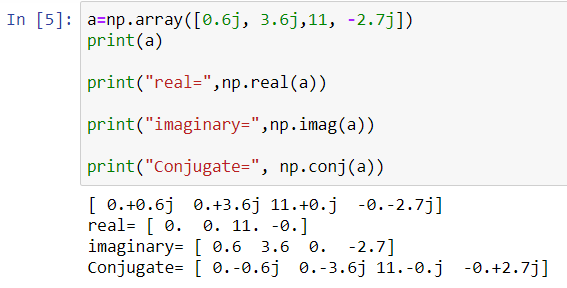
**NumPy Arithmetic OP in Python**

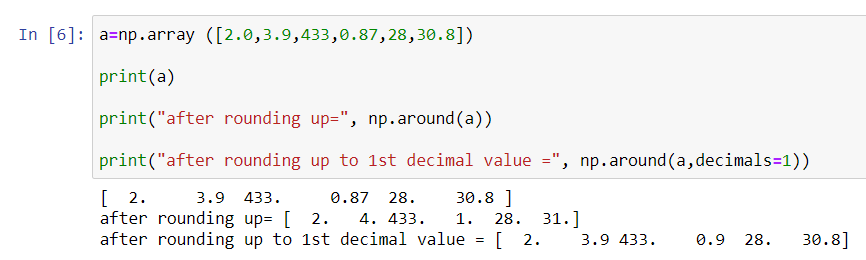
****

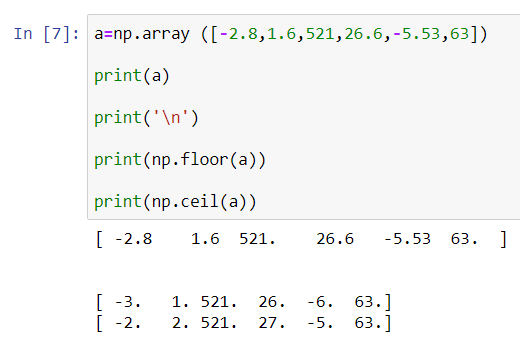
****

****

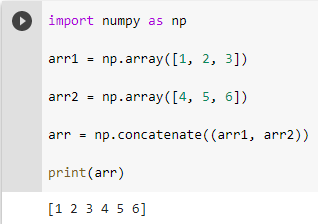
****

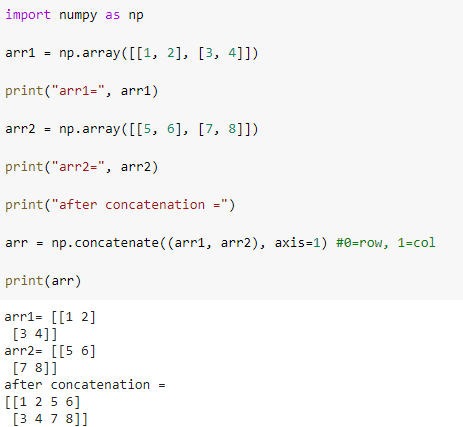
****

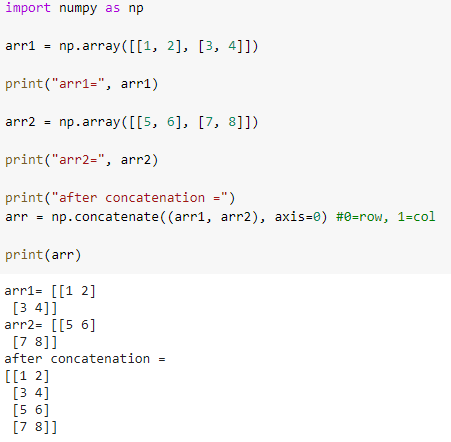
****

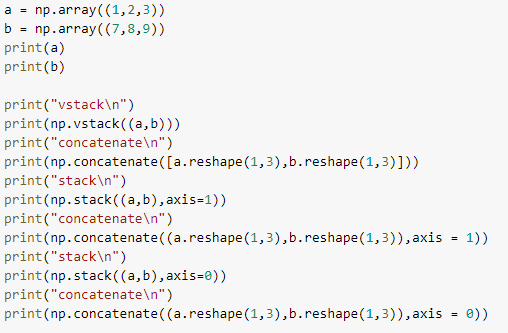
****

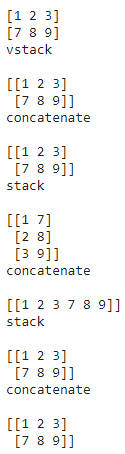
**numpy concat, stack**

****

****

****

****

****