

Name: Muskan Agrawal

MOR Sec - B

Ramjas College

Python Assignment

1. Write program to enter name and display as “Hello, Name”.

```
In [1]: name = input("Enter your name: ")  
print(" Hello, " + name)
```

```
Enter your name: Muskan Agrawal  
Hello, Muskan Agrawal
```

2. Write a menu driven program to enter two numbers and print the arithmetic operations like

a) $+ b) - c) * d) / e) // f) \%$.

```

In [6]: def sum(a,b):
        return(a+b)

def diff(a,b):
    return a-b

def prod(a,b):
    return a*b

def div(a,b):
    return a/b

def floordiv(a,b):
    return a//b

def mod(a,b):
    return a%b

x = int(input("Enter first number: "))
y = int(input("Enter second number: "))
number = 0
while number == 0 :
    z = input(" Choose an operator: \na: +   b: -   c: *   d: /   e: //   f: % \n")

    if z == "a":
        print("Sum is: ", sum(x, y))
    elif z == "b":
        print("Difference is: ", diff(x, y))
    elif z == "c":
        print("Product is: ", prod(x, y))
    elif z == "d":
        print("Division is: ", div(x, y))
    elif z == "e":
        print("Floor division is: ", floordiv(x, y))
    elif z == "f":
        print("Modulus is: ", mod(x, y))
    else:
        print("Enter Valid option")
    number = int(input("Enter 0 to continue. Enter Any Other Number to Exit "))

```

```

Enter first number: 2
Enter second number: 3
Choose an operator:
a: +   b: -   c: *   d: /   e: //   f: %
Your Choice: a
Sum is: 5
Enter 0 to continue. Enter Any Other Number to Exit 0
Choose an operator:
a: +   b: -   c: *   d: /   e: //   f: %
Your Choice: b
Difference is: -1
Enter 0 to continue. Enter Any Other Number to Exit 0
Choose an operator:
a: +   b: -   c: *   d: /   e: //   f: %
Your Choice: c

```

```
Product is: 6
Enter 0 to continue. Enter Any Other Number to Exit 0
Choose an operator:
a: +   b: -   c: *   d: /   e: //   f: %
Your Choice: d
Division is: 0.6666666666666666
Enter 0 to continue. Enter Any Other Number to Exit 0
Choose an operator:
a: +   b: -   c: *   d: /   e: //   f: %
Your Choice: e
Floor division is: 0
Enter 0 to continue. Enter Any Other Number to Exit 0
Choose an operator:
a: +   b: -   c: *   d: /   e: //   f: %
Your Choice: f
Modulus is: 2
Enter 0 to continue. Enter Any Other Number to Exit 4
```

3. To compute the roots of a quadratic equation.

```

In [7]: import math
new = 0
while new == 0:
    a = int(input("Enter coefficient of x^2: "))
    if a == 0:
        print("coefficient of x^2 cannot be 0")
    else:
        b = int(input("Enter coefficient of x: "))
        c = int(input("Enter coefficient of 1: "))

        d = b ** 2 - (4 * a * c)
        if d < 0:
            re = (- b) / 2 * a
            first_im = (math.sqrt(-d)) / 2 * a
            sec_im = (-math.sqrt(-d)) / 2 * a

            print("First root is : " + str(re) + "+" + str(first_im) + "i and Second
                  sec_im) + "i")

        else:
            first = (- b + math.sqrt(d)) / 2 * a
            sec = (- b - math.sqrt(d)) / 2 * a
            print("First root is:", first, "and second root is", sec)

        new = int(input("Enter 0 to continue. Enter any other Number to Exit: "))

```

```

Enter coefficient of x^2: 1
Enter coefficient of x: 5
Enter coefficient of 1: 6
First root is: -2.0 and second root is -3.0
Enter 0 to continue. Enter any other Number to Exit: 0
Enter coefficient of x^2: 1
Enter coefficient of x: 4
Enter coefficient of 1: 5
First root is :-2.0+1.0i and Second root is -2.0-1.0i
Enter 0 to continue. Enter any other Number to Exit: 8

```

4. Write a menu driven Program to reverse the entered numbers and print the sum of digits entered.

```

In [10]: def rev(x):
reverse = str(a) + str(b)
print("Reverse of the entered number is ",reverse)

def add(x):
sum = a+b
print("Sum of the enteres digits of number is",sum)

num = int(input("Enter the number: "))
a = num % 10
b = (num - a) // 10

new = 0
while new == 0:
    op = int(input("Choose the Opton\n1. Reverse the number      2. Print the sum

    if op == 1:
        rev(num)
    elif op == 2:
        add(num)
    else:
        print("Invalid option")
    new = int(input("Enter 0 to continue. Enter any other Number to Exit: "))

```

```

Enter the number: 73
Choose the Opton
1. Reverse the number      2. Print the sum of digits
Your Choice:1
Reverse of the entered number is  37
Enter 0 to continue. Enter any other Number to Exit: 0
Choose the Opton
1. Reverse the number      2. Print the sum of digits
Your Choice:2
Sum of the enteres digits of number is 10
Enter 0 to continue. Enter any other Number to Exit: 5

```

5. Write a menu driven Program to enter the number and print whether the number is

a) odd or even b) prime.

```
In [20]: def odd_even(x):
    if x%2 == 0:
        print(x, "is an even number")
    else:
        print(x, "is a odd number")

def prime(y):
    if y > 1:
        for i in range(2, y):
            if y % i == 0:
                print(y, "is not a prime number")
                break
            else:
                print(y, "is a prime number")
                break
    else:
        print(num, "is a prime number")

new = 0
while new == 0:
    num = int(input("Enter a number: "))

    op = int(input("Find 1. Odd or Even    2. Prime\n Your Choice:"))
    if op == 1:
        odd_even(num)
    elif op == 2:
        prime(num)
    else:
        print("Invalid option")
    new = int(input("Enter 0 to continue. Enter any other Number to Exit: "))
```

```
Enter a number: 6
Find 1. Odd or Even    2. Prime
Your Choice:1
6 is an even number
Enter 0 to continue. Enter any other Number to Exit: 0
Enter a number: 9
Find 1. Odd or Even    2. Prime
Your Choice:2
9 is a prime number
Enter 0 to continue. Enter any other Number to Exit: 8
```

6. Program to find maximum out of entered 3 numbers

```
In [21]: def max(a,b,c):
          if a >= b and a >= c :
              print(a,"is the maximum number")
          elif b>c:
              print(b,"is the maximum number")
          else:
              print(c,"is the maximum number")

          a = int(input("Enter first number: "))
          b = int(input("Enter second number: "))
          c = int(input("Enter third number: "))
          max(a,b,c)
```

```
Enter first number: 56
Enter second number: 97
Enter third number: 123
123 is the maximum number
```

7. Write a program to display ASCII code of a character and vice versa.

```
In [25]: new = 0
          while new== 0:
              word = input("Enter a character:")
              for i in word:
                  print("Ascii code of", i, "is:", ord(i))
              code = int(input("Enter an Ascii code:"))
              print("Character value of", code, "is:", chr(code))
              new = int(input("Enter 0 to continue. Enter any other Number to Exit: "))
```

```
Enter a character:Muskan
Ascii code of M is: 77
Ascii code of u is: 117
Ascii code of s is: 115
Ascii code of k is: 107
Ascii code of a is: 97
Ascii code of n is: 110
Enter an Ascii code:98
Character value of 98 is: b
Enter 0 to continue. Enter any other Number to Exit: 9
```

8. Write a Program to check if the entered number is Armstrong or not.

```
In [26]: new = 0
while new == 0:
    num = input("Enter the number to check if it is armstrong or not : ")
    a = 0
    for i in range(0, len(num)):
        a += int(num[i]) ** len(num)
    val = str(a)
    if val == num:
        print("Its an armstrong number")
    else:
        print("Its not an armstrong number")
    new = int(input("Enter 0 to check any other number. Enter any other Number to Exit: "))
```

Enter the number to check if it is armstrong or not : 4150
Its not an armstrong number
Enter 0 to check any other number. Enter any other Number to Exit: 0
Enter the number to check if it is armstrong or not : 407
Its an armstrong number
Enter 0 to check any other number. Enter any other Number to Exit: 7

9. Write a Program to find factorial of the entered number using recursion.

```
In [7]: def fact(num):
        if num == 1:
            return num
        else:
            return num*fact(num - 1)

num = int(input("Enter a number:"))
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    print("Factorial of", num, "is:", fact(num))
```

Enter a number:5
Factorial of 5 is: 120

10. Write a Program to enter the number of terms and to print the Fibonacci Series.


```
In [28]: i = int(input("Enter no. of terms: "))
a= 0
b=1
count = 0
print("Fibonacci series upto",i,"number of terms is: ",end="")
while count<i:
    if i<0:
        print("Invalid input")
    elif i == 0:
        print(0)
    elif i == 1:
        print(1)
    else:
        print(a, end=" ")
        c = a + b
        a = b
        b = c
        count+=1
```

Enter no. of terms: 13

Fibonacci series upto 13 number of terms is: 0 1 1 2 3 5 8 13 21 34 5
5 89 144

11. Write a Program to enter the numbers and to print greatest number using loop.

```
In [29]: greatest = int(input("Enter a number :"))
n = 0
while n!=0:
    num = int(input("Enter another num or Enter -1 to exit :"))
    if num > greatest:
        greatest = num
    elif num == -1:
        break
print("Maximum of given numbers is:",greatest)
```

Enter a number :45

Enter another num or Enter -1 to exit :23

Enter another num or Enter -1 to exit :67

Enter another num or Enter -1 to exit :-1

Maximum of given numbers is: 67

12. Write a Program to enter the string and to check if it's palindrome or not using loop.

```
In [31]: def palo(s):
    j = 1
    for i in range(0, (len(s))//2):
        if s[i] == s[i - j]:
            j += 2
            return True
        else:
            return False
new = 0
while new == 0:
    s = input("Enter a string: ")
    if (palo(s) == True):
        print(s, "is a palindrome word")
    else:
        print(s, "is not a palindrome word")
    new = int(input("Enter 0 to continue. Enter any other Number to Exit: "))
```

Enter a string: refer
refer is a palindrome word
Enter 0 to continue. Enter any other Number to Exit: 0
Enter a string: civic
civic is a palindrome word
Enter 0 to continue. Enter any other Number to Exit: 9

13. Write a Program to enter the 5 subjects numbers and print the grades A/B/C/D/E.

```
In [32]: def marks(n):
    if n > 0.9:
        print("Grade A")
    elif 0.90 >= n > 0.80:
        print("Grade B")
    elif 0.80 >= n > 0.70:
        print("Grade C")
    elif 0.70 >= n > 0.60:
        print("Grade D")
    else:
        print("Grade E")

a = int(input("Enter number of subject 1:" ))
b = int(input("Enter number of subject 2:" ))
c = int(input("Enter number of subject 3:" ))
d = int(input("Enter number of subject 4:" ))
e = int(input("Enter number of subject 5:" ))
sum = a+b+c+d+e
avg = sum/500
marks(avg)
```

Enter number of subject 1:89
Enter number of subject 2:67
Enter number of subject 3:93
Enter number of subject 4:100
Enter number of subject 5:65
Grade B

14. Write a program in python language to display the given pattern:

```

5
4 5
3 4 5
2 3 4 5
1 2 3 4 5

```

```

In [1]: i =5
while i >0:
    a = 1
    while a <6:
        if i > a:
            print(" ", end= " ")
        else:
            print( a,end = " ")
        a += 1
    print("\n")
    i= i - 1

```

```

5
4 5
3 4 5
2 3 4 5
1 2 3 4 5

```

15. Write a python function sin(x,n) to calculate the value of sin(x) using its Taylor series expansion up to n terms.

```

In [2]: import math

def sin(x,n):
    sum = 0
    for i in range(1, n + 1, 4):
        sum += (x ** i) / math.factorial(i)
    for i in range(3, n + 1, 4):
        sum -= (x ** i) / math.factorial(i)
    return sum
n = int(input("Enter n: "))
x = float(input("Enter x: "))
print("sin(",x,"",n,"") = ", sin(x,n) )

```

```

Enter n: 6
Enter x: 93
sin( 93.0 , 6 ) = 57840064.275

```

16. Write a Program to determine EOQ using various inventory models.

```

In [2]: import math
def EOQ(D,A,H):
    Qstar = math.ceil(math.sqrt(2*D*A/H))
    tau = math.ceil(Qstar/D*365)
    aoc = D/Qstar*A
    ahc = Qstar/2*H
    tic = aoc+ahc
    print("EOQ :",Qstar)
    print("Average Inventory :",Qstar/2)
    print("Cycle time :",tau,"Days")
    print("Total inventory cost :",tic)
def EOQS(D,A,H,S):
    Qstar = math.ceil(math.sqrt(2*D*A*(H+S)/(H*S)))
    tau = math.ceil(Qstar/D*365)
    b= math.ceil(Qstar*(H/(H+S)))
    Td=math.ceil(((Qstar-b)/D)*365)
    Tb=math.ceil((b/D)*365)
    tsc=S*(b**2/(2*Qstar))
    tcc=H*((Qstar-b)**2/(2*Qstar))
    toc=A*(D/Qstar)
    tic=tsc+tcc+toc
    print("EOQ :",Qstar)
    print("Opt Backorder :",b)
    print("Time during which demand is met :",Td)
    print("Time during which demand is Backordered:",Tb)
    print("Cycle time :",tau,"Days")
    print("Total inventory cost :",tic)
def EPQS(D,A,H,P,S):
    d=1/365
    Qstar = math.ceil(math.sqrt((2*A*D*P*(H+S))/(H*S*(P-d))))
    print("EOQ :",Qstar)
    tc = math.sqrt((2*A*D*H*S*(P-d))/(P*(H+S)))
    print("Cost :",tc)
def EPQ(D,A,H,P):
    Qstar = math.ceil(math.sqrt((2*D*A)/(H*(1-(D/P)))))
    tau = math.ceil(Qstar/D*365)
    npr = D/Qstar
    apc = npr*A
    ahc = Qstar/2*(1-(D/P))*H
    tic=apc+ahc
    Imax = math.ceil(Qstar*(1-(D/P)))
    print("EPQ :",Qstar)
    print("Max Inventory :",Imax)
    print("Average Inventory :",math.ceil(Imax/2))
    print("Cycle time :",tau,"Days")
    print("Total inventory cost :",tic)
def main():
    D=float(input("Enter Demand : "))
    A=float(input("Enter Set-Up cost : "))
    H=float(input("Enter Holding cost : "))
    n=1
    while(n!=0):
        print("MENU")
        print("1. EOQ")
        print("2. EOQ (Shortage)")

```

```

print("3. EPQ")
print("4. FINITE PRODUCTION RATE AND SHORTAGES")
print("5. CHANGE INPUTS")
print("0. Exit")
char = int(input("ENTER CHOICE : "))
if(char==1):
    EOQ(D,A,H)
elif(char==2):
    S=float(input("Enter Shortage cost : "))
    EOQS(D,A,H,S)
elif(char==3):
    P=float(input("Enter the units produced : "))
    EPQ(D,A,H,P)
elif(char==4):
    P=float(input("Enter the units produced : "))
    S=float(input("Enter Shortage cost : "))
    EPQS(D,A,H,P,S)
elif(char==5):
    main()
else :
    print("INVALID CHOICE")
    n = int(input("Enter 0 to exit. Any any number to continue. "))
if __name__=="__main__":
    main()

```

```

Enter Demand : 50
Enter Set-Up cost : 400
Enter Holding cost : 609
MENU
1. EOQ
2. EOQ (Shortage)
3. EPQ
4. FINITE PRODUCTION RATE AND SHORTAGES
5. CHANGE INPUTS
0. Exit
ENTER CHOICE : 1
EOQ : 9
Average Inventory : 4.5
Cycle time : 66 Days
Total inventory cost : 4962.72222222223
Enter 0 to exit. ANY any number to continue.0

```

17. Write a Program to determine different characteristics using various Queuing models

```

In [25]: import math
class Queue():
    def __init__(self):
        print("MENU")
        print("1. M/M/1 model")
        print("2. M/M/1/K model")
        print("3. M/M/C model")
        print("4. M/M/C/K model")
        self.choice=int(input("Enter choice : "))
        if(self.choice>4 or self.choice<1):
            print("Enter a valid choice ")
    def takeinput(self):
        self.l=float(input("Enter arrival rate (Lambda) : "))
        self.u=float(input("Enter departure rate (Mu) : "))
        self.p=self.l/self.u
    def model1(self):
        self.Ls=self.p/(1-self.p)
        self.Lq=self.p**2/(1-self.p)
        self.Ws=self.Ls/self.l
        self.Wq=self.Lq/self.l
    def model2(self,k):
        self.po=((1-self.p)/(1-self.p**(k+1)))
        self.pn=self.po*(self.p**k)
        if self.p==1:
            self.Ls=k/2
        else:
            self.Ls=((self.p/(1-self.p))-((k+1)(self.p(k+1))/(1-(self.p**(k+1)))))
        self.Lq=self.Ls-(self.l*(1-self.pn)/self.u)
        self.Ws=self.Ls/(self.l*(1-self.pn))
        self.Wq=self.Lq/(self.l*(1-self.pn))
    def model3(self,c):
        self.pod=0
        for i in range(1,c):
            self.pod=self.pod+((self.p**i)/math.factorial(i))
        self.pod=self.pod+((self.p*c)*c*self.u/(math.factorial(c)((c*self.u)-self.pod))
        self.po=1/self.pod
        self.Lq=((self.p*c)*self.l*self.u*self.po)/(math.factorial(c-1)((c*self.u)-self.pod))
        self.Ls= self.Lq+self.p
        self.Wq=((self.p*c)*self.u*self.po)/math.factorial(c-1)((self.u*c)-self.pod)
        self.Ws=self.Wq+(1/self.u)
    def model4(self,c,k):
        self.pod=0
        for i in range(1,c):
            self.pod=self.pod+((self.p**i)/math.factorial(i))
        self.pod=self.pod+((self.p*c)(k-c+1)/(math.factorial(c)))
        self.po=1/self.pod
        self.np=self.p/c
        self.ps=(self.p**c)*self.po/math.factorial(c)
        self.Lq((((c*self.np)*self.np)*self.np*self.po(1-(self.np*(k-c+1)))-((1-self.ps)))
        self.Ls=self.Lq+(self.p*(1-self.ps))
        self.Ws=self.Ls/(self.l*(1-self.ps))
        self.Wq=self.Ws-(1/self.u)
    def result(self):
        print("Ls = ",self.Ls)
        print("Lq = ",self.Lq)
        print("Ws = ", self.Ws)

```

```
        print("Wq = ",self.Wq)
ob=Queue()
if ob.choice==1:
    ob.takeinput()
    ob.model1()
    ob.result()
if ob.choice==2:
    ob.takeinput()
    k=int(input("Enter system capacity (K) : "))
    ob.model2(k)
    ob.result()
if ob.choice==3:
    ob.takeinput()
    c=int(input("Enter number of servers (C) : "))
    ob.model3(c)
    ob.result()
if ob.choice==4:
    ob.takeinput()
    k=int(input("Enter system capacity (K) : "))
    c=int(input("Enter number of servers (C) : "))
    ob.model4(c,k)
    ob.result()
```

MENU

```
1. M/M/1 model
2. M/M/1/K model
3. M/M/C model
4. M/M/C/K model
Enter choice : 1
Enter arrival rate (Lambda) : 6
Enter departure rate (Mu) : 9
Ls = 1.9999999999999998
Lq = 1.3333333333333333
Ws = 0.3333333333333333
Wq = 0.22222222222222218
```

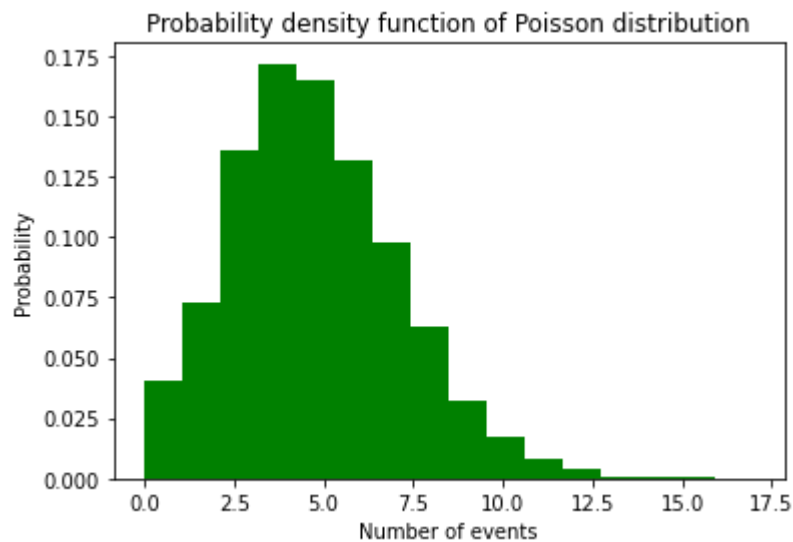
18. Write a Program to implement Inheritance. Create a class Employee inherit two classes Manager and Clerk from Employee

```
In [3]: class Employee:
    def __init__(self, name, age, salary):
        self.name = name
        self.age = age
        self.salary=salary
    def display(self):
        print("NAME : ", self.name)
        print("AGE : ", self.age)
        print("SALARY : ", self.salary)
class Clerk(Employee):
    def __init__(self, name, age, salary,typingspeed):
        Employee.__init__(self, name, age, salary)
        self.typingspeed = typingspeed
    def overtime(self):
        Employee.display(self)
        print("TYPING SPEED : ", self.typingspeed)
        print("OVERTIME WAGE : ", self.typingspeed*100)
class Manager(Employee):
    def __init__(self, name, age, salary):
        Employee.__init__(self, name, age, salary)
        self.t=1
    def tax(self):
        Employee.display(self)
        print("TAX : ", self.salary*self.t/100)
c=Clerk('John Doe',28,22000,40)
m=Manager('Jane Doe',30,35000)
print('PARENT'.center(20,'-'))
c.display()
print('OWN'.center(20,'-'))
c.overtime()
print('PARENT'.center(20,'-'))
m.display()
print('OWN'.center(20,'-'))
m.tax()
```

```
-----PARENT-----
NAME :  John Doe
AGE :  28
SALARY :  22000
-----OWN-----
NAME :  John Doe
AGE :  28
SALARY :  22000
TYPING SPEED :  40
OVERTIME WAGE :  4000
-----PARENT-----
NAME :  Jane Doe
AGE :  30
SALARY :  35000
-----OWN-----
NAME :  Jane Doe
AGE :  30
SALARY :  35000
TAX :  350.0
```


19. Program to fit Poisson distribution on a given data.

```
In [5]: import numpy as np
import matplotlib.pyplot as plt
sn = np.random.poisson(5,10000)
plt.hist(sn, 16, density=True,color='Green')
plt.title("Probability density function of Poisson distribution")
plt.ylabel("Probability")
plt.xlabel("Number of events")
plt.show()
```

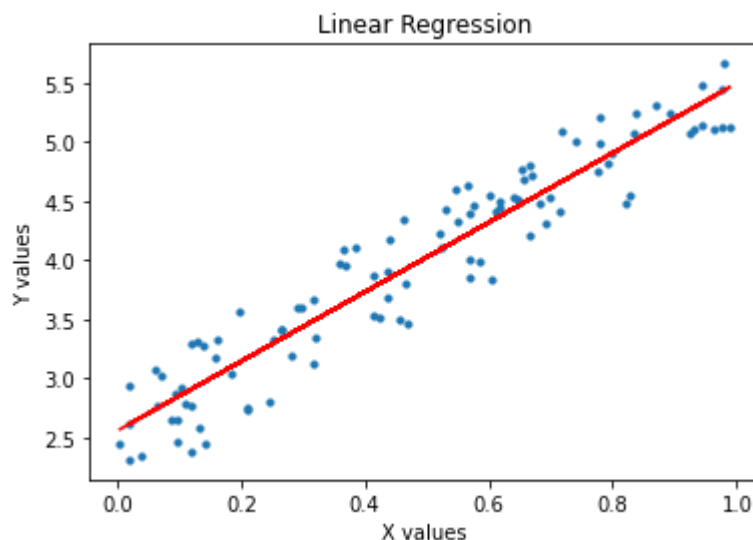
**20. Write a program to implement linear regression using python.**

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
np.random.seed(0)
x = np.random.rand(100, 1)
y = 2 + 3 * x + np.random.rand(100, 1)
regression_model = LinearRegression()
regression_model.fit(x, y)
y_predicted = regression_model.predict(x)
rmse = mean_squared_error(y, y_predicted)
print('Slope:', regression_model.coef_)
print('Intercept:', regression_model.intercept_)
print('Root mean squared error: ', rmse)
plt.scatter(x, y, s=10)
plt.xlabel('x')
plt.ylabel('y')
plt.plot(x, y_predicted, color='r')
plt.title("Linear Regression")
plt.xlabel("X values")
plt.ylabel("Y values")
plt.show()
```

Slope: [[2.93655106]]

Intercept: [2.55808002]

Root mean squared error: 0.07623324582875007



21. Write a program to perform read and write operation with .csv file.

```
In [13]: import pandas as pd
import random
import csv

with open('C:\\Users\\Me\\NewData.csv','r') as file:
    reader=csv.reader(file)
    for row in reader:
        print(row)
```

```
['1', '2', '3', '4']
['3', '3', '4', '7']
['5', '4', '5', '10']
['7', '5', '6', '13']
['9', '6', '7', '16']
['11', '7', '8', '19']
['13', '8', '9', '22']
```

```
In [16]: with open('C:\\Users\\Me\\NewData.csv', 'w', newline='') as file:
    writer=csv.writer(file)
    writer.writerow(['Attribute1', 'Attribute2', 'Class'])
    writer.writerow([11, 3, 0])
    writer.writerow([32, 3, 1])

with open('C:\\Users\\Me\\NewData.csv','r') as file:
    reader=csv.reader(file)
    for row in reader:
        print(row)
```

```
['Attribute1', 'Attribute2', 'Class']
['11', '3', '0']
['32', '3', '1']
```

22. Write a Program to enter multiple values-based data in multiple columns/rows and show that data in python using DataFrames and pandas.

```
In [19]: import pandas as pd
import random
print("Enter attribute 1")
att1 = [float(x) for x in input().split()]
print("Enter attribute 2")
att2 = [float(x) for x in input().split()]
print("Enter class")
cls = [float(x) for x in input().split()]
data = {'Attribute1':att1,'Attribute2':att2,'Class':cls}
df = pd.DataFrame(data)
df
```

```
Enter attribute 1
1 2 3
Enter attribute 2
5 6 7
Enter class
8 9 3
```

Out[19]:

	Attribute1	Attribute2	Class
0	1.0	5.0	8.0
1	2.0	6.0	9.0
2	3.0	7.0	3.0

23. WAP in python to perform various statistical measures using pandas.

```
In [22]: import pandas as pd
import numpy as np
df=pd.read_csv('C:\\Users\\Me\\RawData.csv')
df.head()
```

Out[22]:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	1	1	22	22	22	19	18	14	49.895756	17.775994	5.270920	0.771761	0.018632	0.006864
1	1	1	24	24	22	18	16	13	57.709936	23.799994	3.325423	0.234185	0.003903	0.003903
2	1	1	62	60	59	54	47	33	55.831441	27.993933	12.687485	4.852282	1.393889	0.373252
3	1	1	55	53	53	50	43	31	40.467228	18.445954	9.118901	3.079428	0.840261	0.272434
4	1	1	44	44	44	41	39	27	18.026254	8.570709	0.410381	0.000000	0.000000	0.000000

```

In [23]: def statisticalmeasures(df):
        data=[]
        for i in df.columns:
            data.append(df[i].mean())
            data.append(df[i].median())
            data.append(df[i].mode()[0])
            data.append(df[i].count())
            data.append(df[i].std())
            data.append(df[i].max())
            data.append(df[i].min())
            data.append(df[i].quantile(0.75)-df[i].quantile(0.25))
        return data

data=np.array(statisticalmeasures(df))

StatsMeasure = pd.DataFrame(np.array_split(data, 20),columns=['Mean', 'Median', 'Mo
StatsMeasure

```

Out[23]:

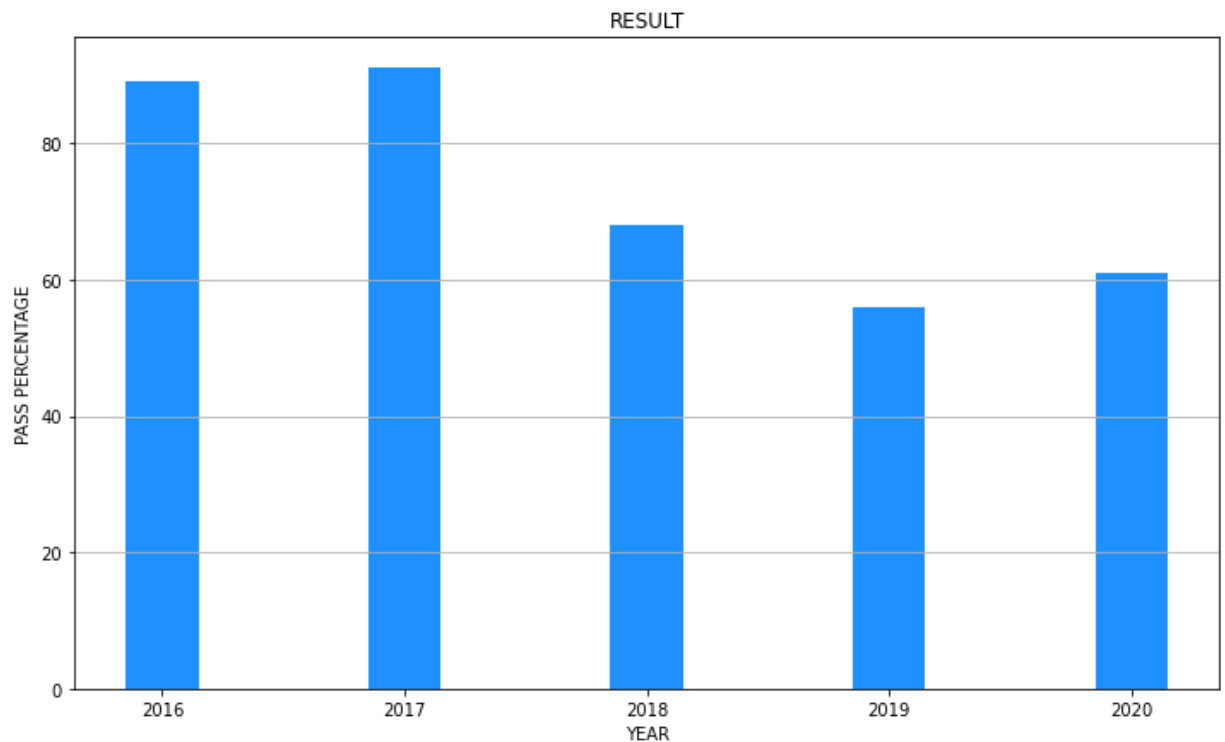
	Mean	Median	Mode	Count	Std	Max	Min	IQR
1	0.996230	1.000000	1.000000	1061.0	0.061314	1.000000	0.000000	0.000000
2	0.916117	1.000000	1.000000	1061.0	0.277343	1.000000	0.000000	0.000000
3	38.658812	36.000000	16.000000	1061.0	25.675524	151.000000	1.000000	40.000000
4	37.137606	35.000000	15.000000	1061.0	24.183066	132.000000	1.000000	38.000000
5	35.361923	32.000000	13.000000	1061.0	22.885503	120.000000	1.000000	37.000000
6	32.497644	30.000000	11.000000	1061.0	21.188935	105.000000	1.000000	35.000000
7	28.943450	25.000000	10.000000	1061.0	19.559696	97.000000	1.000000	33.000000
8	21.270500	18.000000	9.000000	1061.0	15.084584	89.000000	1.000000	24.000000
9	64.350017	45.003816	6.193941	1061.0	58.479527	403.939108	0.349274	65.334377
10	23.163337	17.293493	1.625616	1061.0	21.600927	167.131427	0.000000	23.437195
11	8.751308	4.563607	0.000000	1061.0	11.566846	106.070092	0.000000	10.549984
12	1.856494	0.513135	0.000000	1061.0	3.990004	59.766121	0.000000	1.829457
13	0.566788	0.023200	0.000000	1061.0	2.544188	51.423208	0.000000	0.195595
14	0.210432	0.001914	0.000000	1061.0	1.070305	20.098605	0.000000	0.041503
15	0.085072	0.000000	0.000000	1061.0	0.402848	5.937799	0.000000	0.004810
16	0.036054	0.000000	0.000000	1061.0	0.177188	3.086753	0.000000	0.003848
17	0.522824	0.523010	0.486570	1061.0	0.028021	0.592217	0.367762	0.040455
18	0.108430	0.106623	0.107603	1061.0	0.018137	0.219199	0.057906	0.023941
19	0.333648	0.000000	0.000000	1061.0	0.471738	1.000000	0.000000	1.000000
20	0.534402	1.000000	1.000000	1061.0	0.499050	1.000000	0.000000	1.000000

24. Write a program to plot a bar chart in python to display the result of a school for five consecutive years.

```
In [7]: import matplotlib.pyplot as plt

import numpy as np
import random

def barplot(label,value,width=0.3):
    fig = plt.figure(figsize=(12,7))
    plt.bar(label,value,color='dodgerblue',width=width)
    plt.xlabel('YEAR')
    plt.ylabel('PASS PERCENTAGE')
    plt.title('RESULT' )
    plt.grid(axis='y')
year = range(2016,2021)
result = np.array(random.sample(range(50, 100), 5))
barplot(year,result)
```

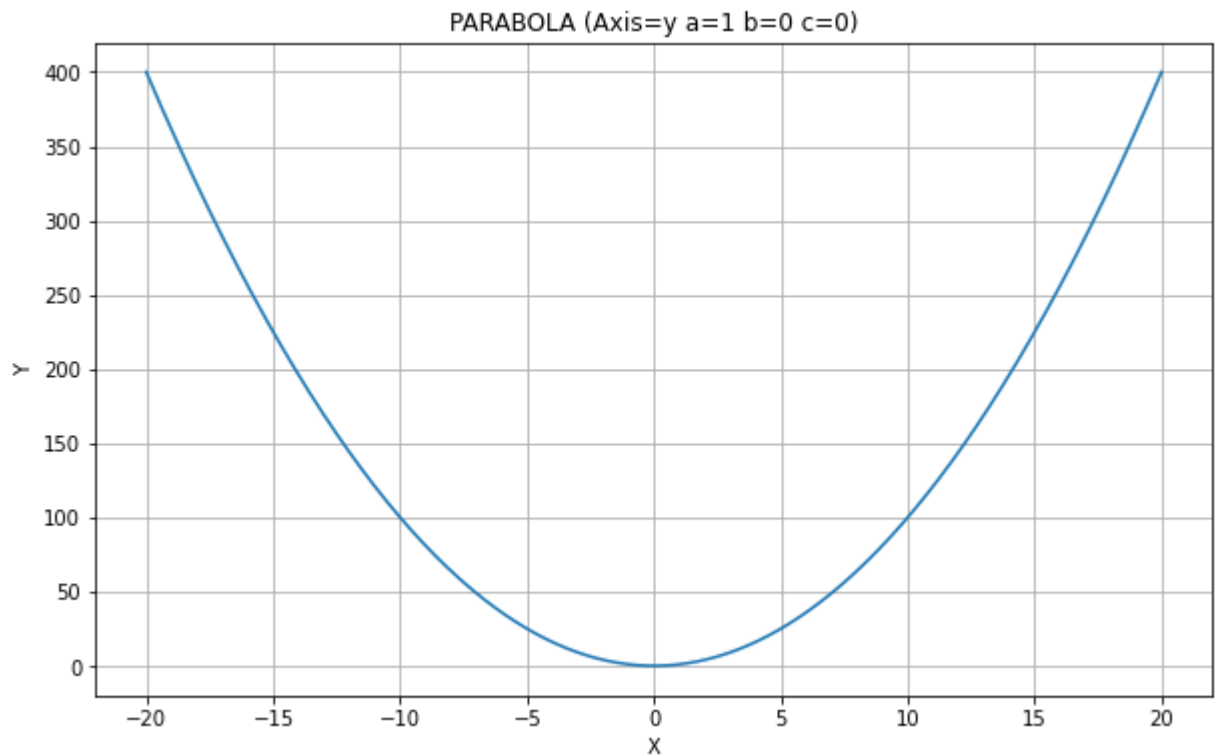


25. Write a program in python to plot a graph for the function $y = x^2$

In [6]:

```
import matplotlib.pyplot as plt
import numpy as np
def parabola(min,max,a,b,c,ax_symmetry='y'):
    if (ax_symmetry=='y'):
        x=np.linspace(min, max, 500)
        y=a*(x**2)+b*x+c
    elif (ax_symmetry=='x'):
        y=np.linspace(min, max, 500)
        x=a*(y**2)+b*y+c
    else:
        print("WRONG INPUT")
        return
    fig = plt.figure(figsize=(10,6))
    plt.plot(x,y)
    plt.xlabel('X')
    plt.ylabel('Y')
    plt.title('PARABOLA (Axis={} a={} b={} c={})'.format(ax_symmetry,a,b,c))
    plt.grid(True)

parabola(-20,20,1,0,0,'y')
```



26. Write a program in python to plot a pie chart on consumption of water in daily life.

```
In [27]: import matplotlib.pyplot as plt
import numpy as np
import random
def pie(label,val,sa=90):
    fig = plt.figure(figsize=(16,9))
    plt.pie(val, labels = label ,startangle=sa, autopct=lambda x: '{:.2f}% ({:.0f})'.format(x/100, x))
    plt.title('Consumption of water in daily life')
purpose = ['Drinking','Cooking','Laundering','Cleaning','Miscellaneous']
usage = np.array(random.sample(range(10, 50), 5))
pie(purpose,usage)
```

