Program: 1
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a program to input roll no, student name, marks of physics, chemistry and maths out of 100. (0-100). Calculate total, percentage, calculate STATUS (pass, fail) if students scores above 40 in all the 3 subjects the STATUS should be pass otherwise fail. Calculate GRADE if STATUS is pass. Grade must be based on percentage value.
if percentage is above 70, then grade must be DISTINCTION
if percentage is above 60, then grade must be FIRST CLASS
if percentage is above 50, then grade must be SECOND CLASS
if percentage is above 40, then grade must be PASS CLASS

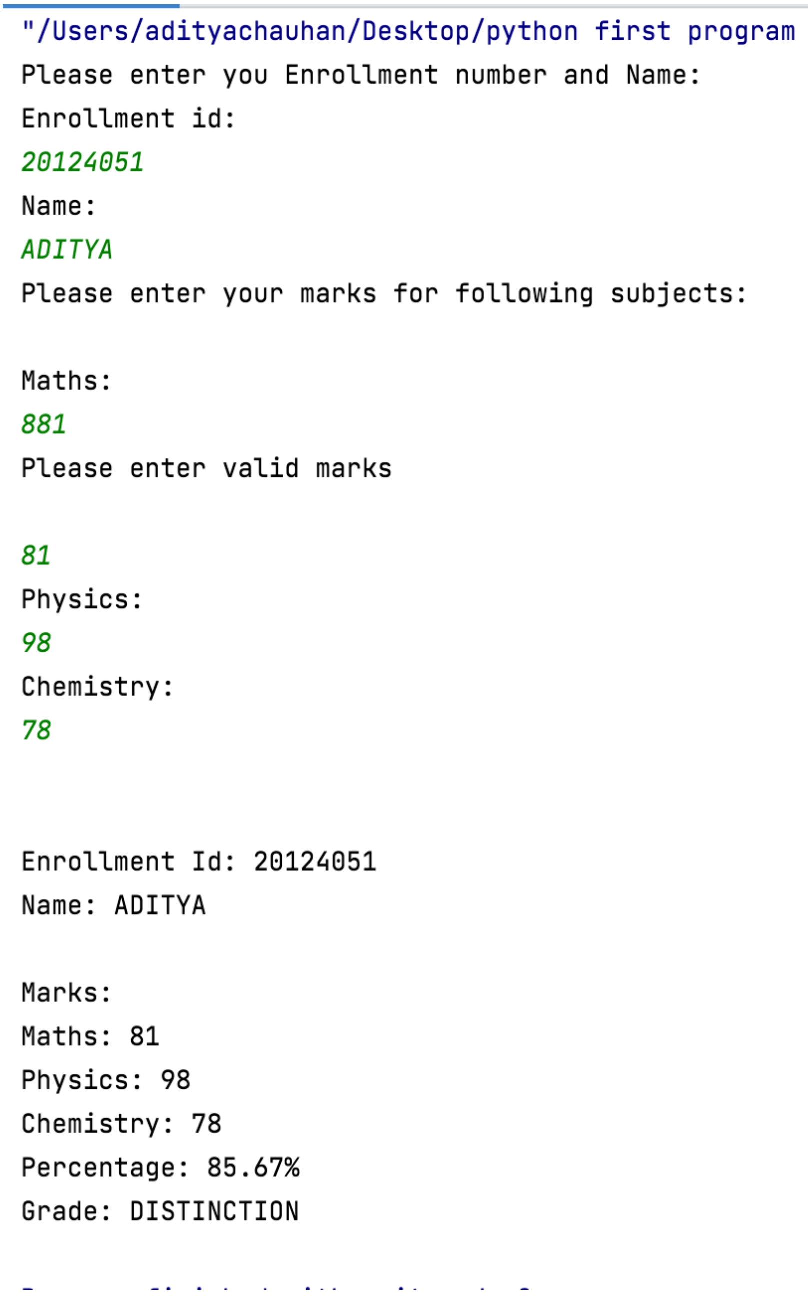
Input :-

```
print('Please enter you Enrollment number and Name: ')
while True:
    try:
        id = int(input('Enrollment id: \n'))
    except:
        print("Alphabets or Special characters are not 
allowed in Enrollment Id.")
    else:
        break


while (len(str(id)) != 8):
    print('Please enter a valid Enrolment Id: ')
    id = int(input('Enrollment id: \n'))


name = input('Name: \n')




def validate(subject):
    while (subject not in range(0, 101)):
        print("Please enter valid marks\n")
        subject = int(input())
    validate.condition = True
    validate.export = subject
    return subject


print("Please enter your marks for following subjects: \n")

try:
    maths = int(input('Maths: \n'))
except:
    print("Alphabets or Special characters are not allowed.")
    maths = int(input('Maths: \n'))

validate(maths)
if validate.condition == True:
    maths = validate.export

try:
    physics = int(input('Physics: \n'))
except:
    print("Alphabets or Special characters are not allowed.")
    physics = int(input('Physics: \n'))

validate(physics)
```

```
if validate.condition == True:
    physics = validate.export

try:
    chemistry = int(input('Chemistry: \n'))
except:
    print("Alphabets or Special characters are not allowed.")
    chemistry = int(input('Chemistry: \n'))

validate(chemistry)
if validate.condition == True:
    chemistry = validate.export
total = maths + physics + chemistry
percentage = (total * 100) / 300



def calstatus():
    if (maths > 40 and physics > 40 and chemistry > 40):
        calstatus.status = 'Pass'
    else:
        calstatus.status = 'Fail'

    return calstatus.status



def calgrade():
    if calstatus.status == 'Pass':
```

```python
        if percentage > 40:
            calgrade.grade = 'PASS CLASS'
            if percentage > 50:
                calgrade.grade = 'SECOND CLASS'
                if percentage > 60:
                    calgrade.grade = 'FIRST CLASS'
                    if percentage > 70:
                        calgrade.grade = 'DISTINCTION'
        return calgrade.grade
    else:
        calgrade.grade = 'You have failed in one or more than one subject(s).\nTherefore your grade cannot be calculated.'
        return calgrade.grade



calstatus()
calgrade()


print()
print()
print("Enrollment Id:", id)
print("Name:", name)
print()
print('Marks:')
print('Maths:', maths)
print('Physics:', physics)
print('Chemistry:', chemistry)
print('Percentage: {0:.2f}%'.format(percentage))

print('Grade:', calgrade.grade)
```

Output :-



```
"/Users/adityachauhan/Desktop/python first program
Please enter you Enrollment number and Name:
Enrollment id:
20124051
Name:
ADITYA
Please enter your marks for following subjects:

Maths:
881
Please enter valid marks

81
Physics:
98
Chemistry:
78


Enrollment Id: 20124051
Name: ADITYA

Marks:
Maths: 81
Physics: 98
Chemistry: 78
Percentage: 85.67%
Grade: DISTINCTION
```

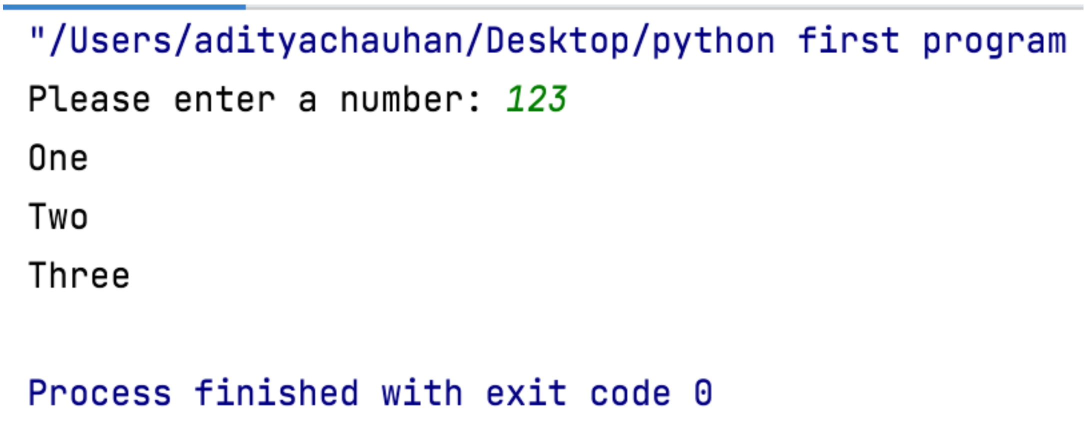
Program: 2
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a program which inputs a number. Display that number in word format.
Eg.
459 – Four Five Nine
7091 – Seven Zero Nine One
26 - Two Six

Input :-

```
class Word_Format:


    def __init__(self):

        self.__digit = input('Please enter a number: ')




    def printValue(self, number):

        if number == '0':

            print("Zero ")


        elif number == '1':

            print("One ")


        elif number == '2':

            print("Two ")


        elif number == '3':

            print("Three")
        elif number == '4':
            print("Four ")

        elif number == '5':
            print("Five ")

        elif number == '6':
            print("Six ")

        elif number == '7':
            print("Seven")

        elif number == '8':
            print("Eight")

        elif number == '9':
            print("Nine ")

    def printWord(self):
        i = 0
        num = self.__digit
        length = len(num)


        while i < length:
            self.printValue(num[i])
            i += 1
```

```
run = Word_Format()

run.printWord()
```

Output :-



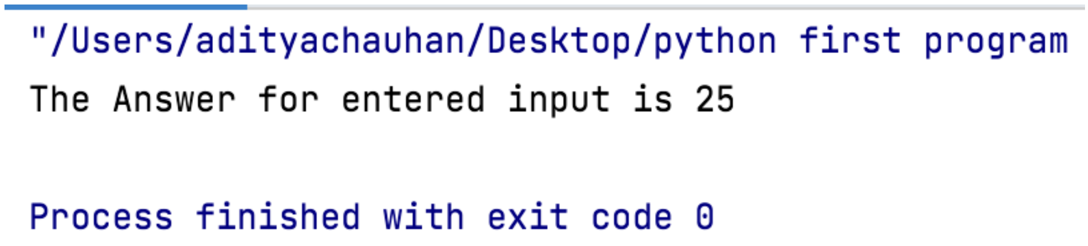
```
"/Users/adityachauhan/Desktop/python first program
Please enter a number: 123
One
Two
Three

Process finished with exit code 0
```

Program: 3
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write an OOP to calculate exponent from inputted base and power value.
Eg. Enter a base value : 3
Enter a power value : 4
For base 3 and power 4, the answer is 81

Input :-

```
class exponent:


    def init(self):

        self.__num = 0


    def get_number(self, value):

        self.__num = value
    def display(self):
        print("The Answer for entered input is
{0}".format(self.__num))

    def calculate(self, object):
        solution = exponent()
        solution.__num = self.__num ** object.__num
        return solution


base = exponent()
base.get_number(5)
power = exponent()
power.get_number(2)
solution = base.calculate(power)
solution.display()
```

Output :-



```
"/Users/adityachauhan/Desktop/python first program
The Answer for entered input is 25

Process finished with exit code 0
```

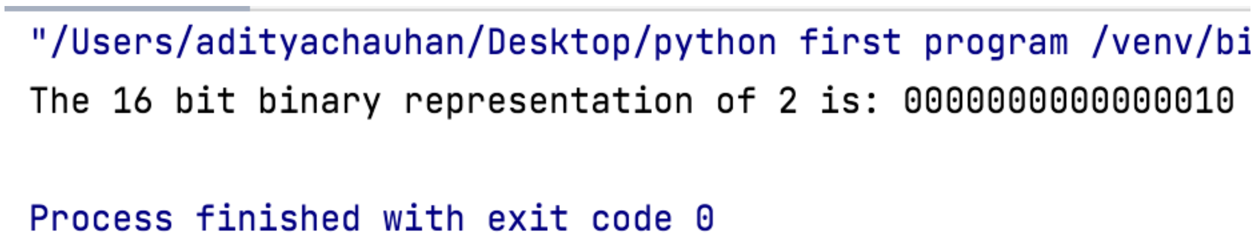
Program:-4
@author: 20124051 Aditya Chauhan
@description: Program No. -
Program to print binary form a any number using 16 bit representation. (without library function) (You can use list for 16 bit representation)
Eg. Enter any number : 20
0000000000010100
Enter any number -5
1000000000000101

Input :-

```
class bin_representation:

    def __init__(self,num):

        self.__string = ''

        self.__num = num


    def convert(self):

        i = 1 << 15

        while (i > 0):


            if ((self.__num & i) != 0):


                self.__string += '1'


            else:

                self.__string += '0'


            i = i // 2

    def display(self):

        print('The 16 bit binary representation of {0} is:
{1}'.format(self.__num,self.__string))
```

```
run = bin_representation(2)

run.convert()

run.display()
```

Output :-



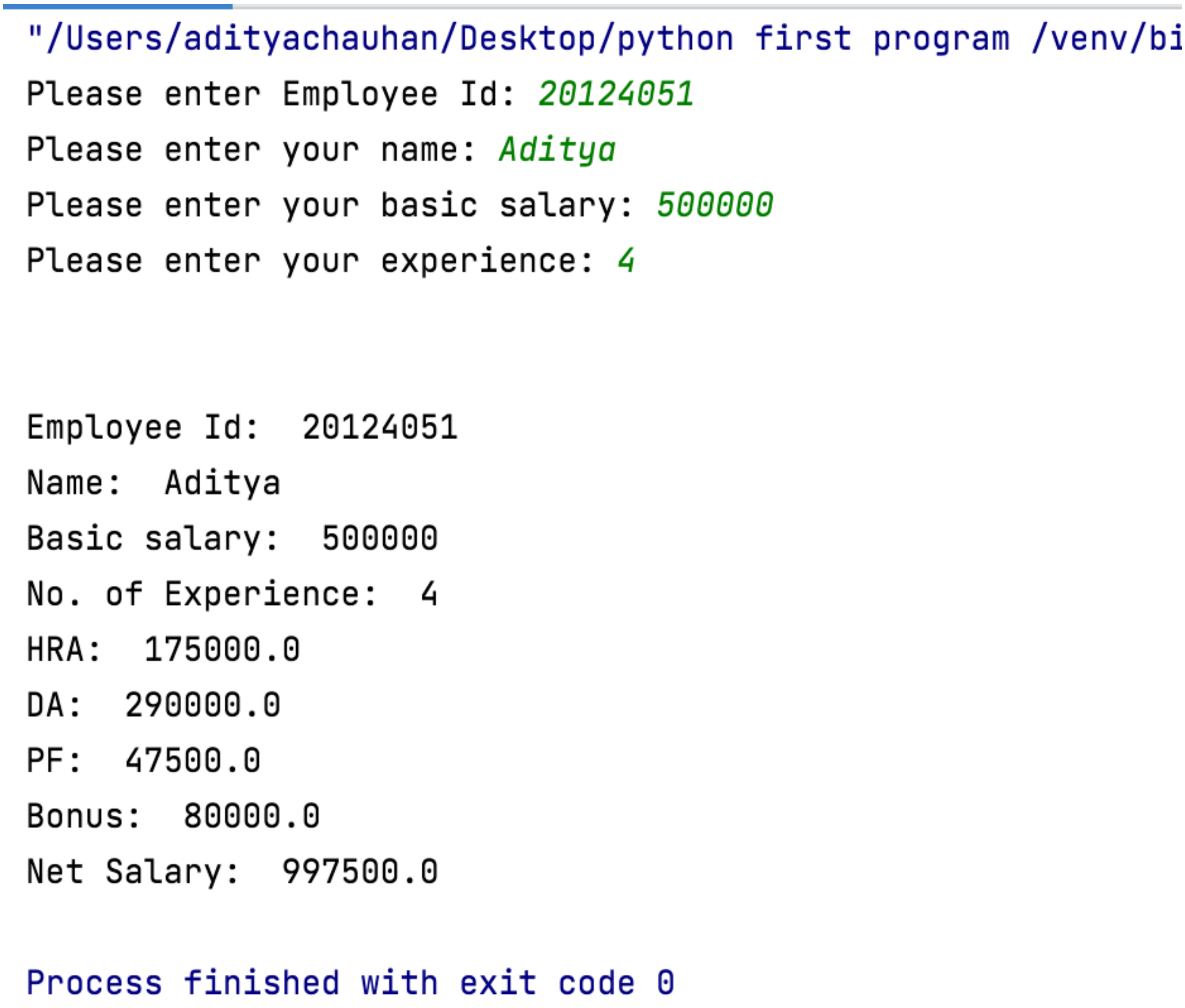
```
"/Users/adityachauhan/Desktop/python first program /venv/bi
The 16 bit binary representation of 2 is: 0000000000000010

Process finished with exit code 0
```

Program:-5
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a OOP in python to input empid, name, basic salary, no. of experience in yrs. Calculate hra(35% of basic), da (58% of basic) and pf (9.5% of basic).
Also calculate bonus based on experience in years.
If experience in years is >= 30, bonus must be 59% of basic,
If experience in years is >=23, bonus must be 51% of basic,
If experience in years is >=15, bonus must be 45% of basic,
If experience in years is >=7, bonus must be 33% of basic,
If experience in years is <7, bonus must be 16% of basic
Calculate netsalary as basic+da+hra-pf+bonus.

Input :-

```
class Employee:

    def __init__(self):

        self.__empid = 0

        self.__name = ''

        self.__basic_sal = 0

        self.__experience = 0
    def get_input(self):


        while True:
            try:
                self.__empid = int(input('Please enter Employee Id: '))
                if (len(str(self.__empid)) != 8):
                    print('Please enter a valid Employee Id: ')
                    self.__empid = int(input('Please enter Employee Id: '))
                else:
                    pass
            except:
                print("Alphabets or Special characters are not allowed in Employee Id.")
            else:
                break


        self.__name = input('Please enter your name: ')

        while True:
            try:
                self.__basic_sal = int(input('Please enter your basic salary: '))
            except:
                print("Alphabets or Special characters are not allowed in Salary.")
            else:
                break

        while True:
            try:
                self.__experience = int(input('Please enter your experience: '))
            except:
                print("Alphabets or Special characters are not allowed in Salary.")
            else:
                break

    def calculate(self):
        self.__hra = (self.__basic_sal * 35) /100
        self.__da = (self.__basic_sal * 58) /100
        self.__pf = (self.__basic_sal * 9.5) /100

    def cal_bonus(self):
        exp = self.__experience
        self.__bonus = 0

        if exp >= 30:
            self.__bonus = 59
        elif exp >= 23:
            self.__bonus = 51
        elif exp >= 15:
            self.__bonus = 45
        elif exp >= 7:
            self.__bonus = 33
        elif exp < 7:
            self.__bonus = 16

        self.__calculated_bonus = (self.__basic_sal *
self.__bonus) /100

    def net_salary(self):
        self.__netsalary = self.__basic_sal + self.__da +
self.__hra - self.__pf + self.__calculated_bonus

    def display(self):
        print("\n")
        print("Employee Id: ",self.__empid)
        print("Name: ",self.__name)
        print("Basic salary: ",self.__basic_sal)
        print("No. of Experience: ",self.__experience)
        print("HRA: ",self.__hra)
        print("DA: ",self.__da)
        print("PF: ",self.__pf)
        print("Bonus: ",self.__calculated_bonus)
        print("Net Salary: ",self.__netsalary)
run = Employee()
run.get_input()
run.calculate()

run.cal_bonus()

run.net_salary()

run.display()
```

Output :-



```
"/Users/adityachauhan/Desktop/python first program /venv/bi
Please enter Employee Id: 20124051
Please enter your name: Aditya
Please enter your basic salary: 500000
Please enter your experience: 4



Employee Id:  20124051
Name:  Aditya
Basic salary:  500000
No. of Experience:  4
HRA:  175000.0
DA:  290000.0
PF:  47500.0
Bonus:  80000.0
Net Salary:  997500.0

Process finished with exit code 0
```

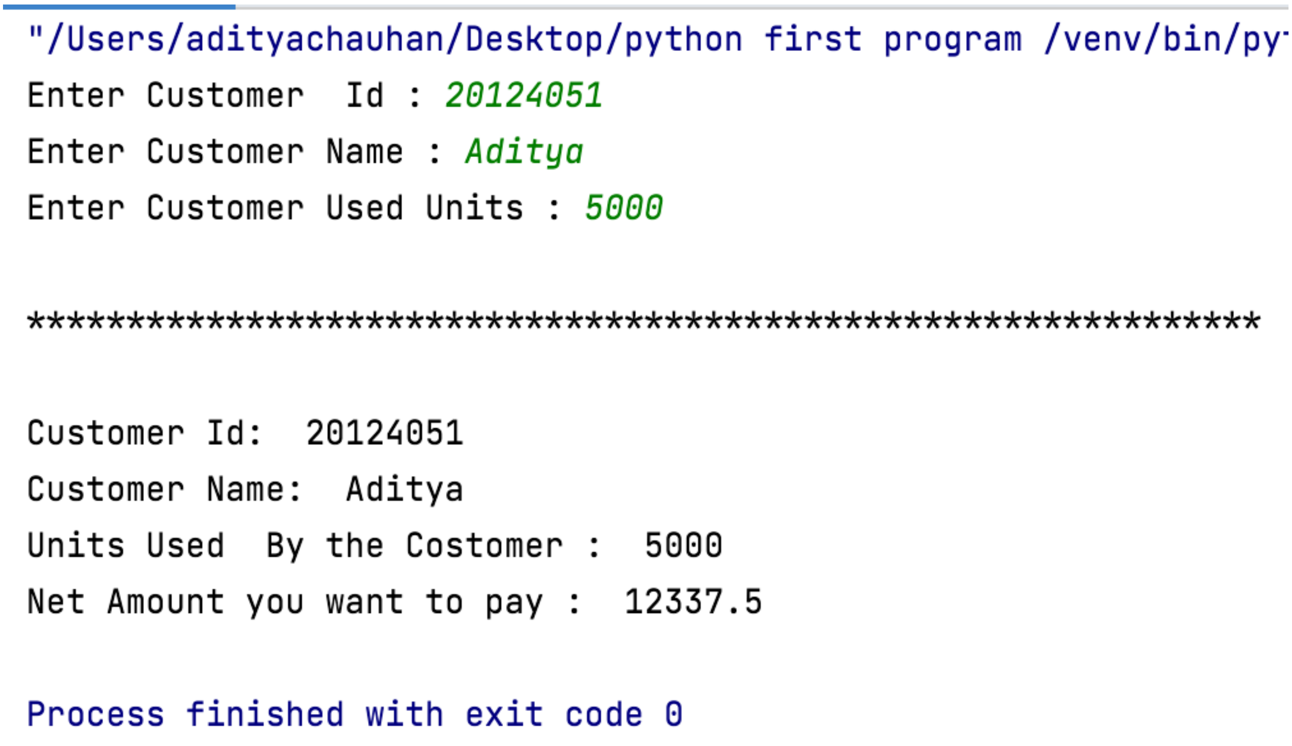
Program:-6
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a OOP program to input Customer id , Customer name, electricity unit charges used.
Calculate electricity bill according to the given condition:
For first 50 units Rs. 0.50/unit
For next 100 units Rs. 0.75/unit
For next 100 units Rs. 1.20/unit
For unit above 250 Rs. 1.50/unit
An additional surcharge of 20% is added to the bill

Input :-

```
class ElectricityBill:


    def __init__(self):

        self.__cusid = 0

        self.__name = ''

        self.__unit_used = 0

        self.__tunit = 0


    def get_input(self):

        self.__cusid = int(input("Enter Customer  Id : "))

        self.__name = input("Enter Customer Name : ")

        self.__unit_used = int(input("Enter Customer Used 
Units : "))


    def calculate(self):

        self._extra = (self.__unit_used * 0.2 )


    def condition(self):

        unit = self.__unit_used
        if(unit <= 50):
            self.__tunit = unit*0.5
        elif(unit <= 100):
            self.__tunit = (50*0.5) + ((unit-50)*0.75)
        elif (unit <= 150):
            self.__tunit = (50*0.5) + (50*0.75) + ((unit-100)
* 1.2)
        elif (unit <= 250):
            self.__tunit = (50*0.5) +(50*0.75)+(100*1.2)+
((unit-150) * 1.5)
        elif (unit >= 250):
            self.__tunit = (50*0.5) + (50*0.75)+(100*1.2)+
((unit - 150) * 2.3)


    def net_units(self):
        self._net_units = (self.__tunit + self._extra)


    def display(self):

print("\n****************************************************
**********\n")
        print("Customer Id: ", self.__cusid)
        print("Customer Name: ", self.__name)
        print("Units Used  By the Costomer : ",
self.__unit_used)
```

```
        print("Net Amount you want to pay : ",
self._net_units)




run=ElectricityBill()

run.get_input()

run.calculate()

run.condition()

run.net_units()

run.display()
```

Output :-



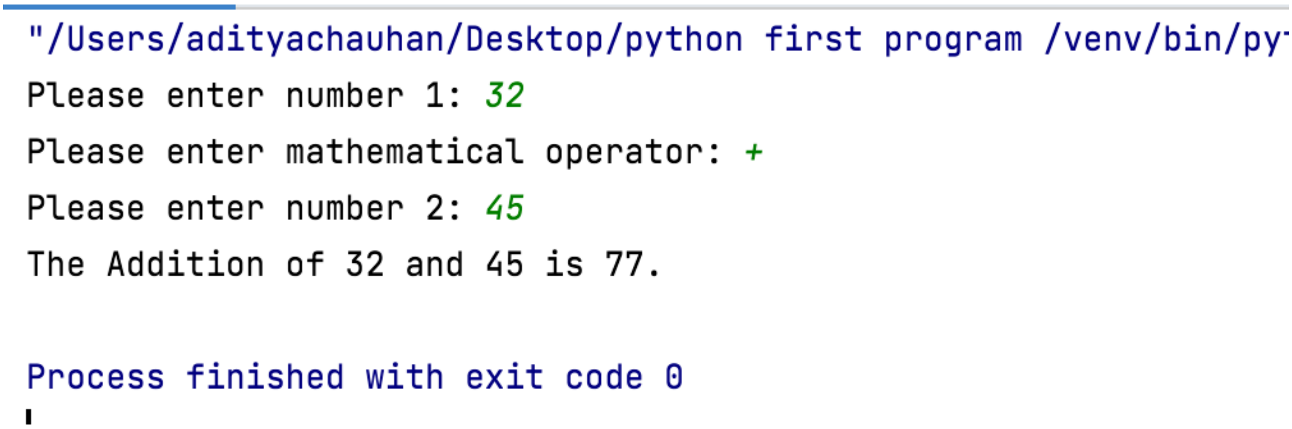
```
"/Users/adityachauhan/Desktop/python first program /venv/bin/py
Enter Customer  Id : 20124051
Enter Customer Name : Aditya
Enter Customer Used Units : 5000

**************************************************************

Customer Id:  20124051
Customer Name:  Aditya
Units Used  By the Costomer :  5000
Net Amount you want to pay :  12337.5

Process finished with exit code 0
```

Program:-7
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write an OOP program to accept two numbers and one mathematical operator. Calculate and display appropriate answer.
Eg output
Enter first number : 45
Enter mathematical operator : +
Enter second number : 60
45 + 60 = 105

Input :-

```
class Calculator:
    def __init__(self):
        self.__num_1 = int(input('Please enter number 1: '))
        self.__operator = input('Please enter mathematical
operator: ')
        self.__num_2 = int(input('Please enter number 2: '))


    def calculate(self):
        self.__calculation = 0
        self.__opName = ''
        if self.__operator == '+':
            self.__calculation = self.__num_1 + self.__num_2
            self.__opName = 'Addition'
        elif self.__operator == '-':
            self.__calculation = self.__num_1 - self.__num_2
            self.__opName = 'Subtraction'
        elif self.__operator == '*':
            self.__calculation = self.__num_1 * self.__num_2
            self.__opName = 'Multiplication'
        elif self.__operator == '/':
            self.__calculation = self.__num_1 / self.__num_2
            self.__opName = 'Division'

    def display(self):
        print('The {0} of {1} and {2} is {3}.'.format(self.__opName,self.__num_1,self.__num_2,self.__calculation))


run = Calculator()
run.calculate()
run.display()
```

Output :-



```
"/Users/adityachauhan/Desktop/python first program /venv/bin/py
Please enter number 1: 32
Please enter mathematical operator: +
Please enter number 2: 45
The Addition of 32 and 45 is 77.

Process finished with exit code 0
```

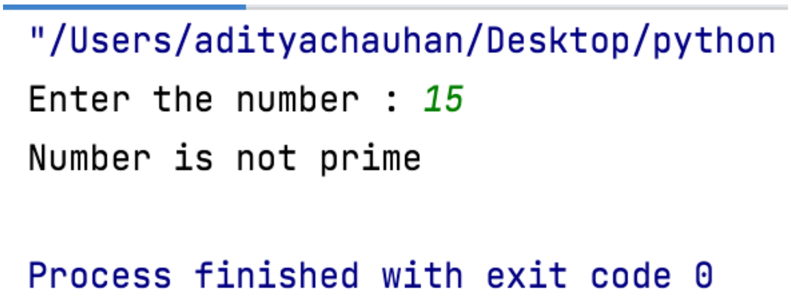
Program:-8
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a program to check whether number is prime or not.
Enter a number : 13
13 is prime
Enter a number : 45
45 is not a prime number

Input :-

```
class PrimeNumber:

    def __int__(self):

        self.Starting_Number=0

        self.Ending_Number=0

        self.num=0


    def display_Condition(self):

        self.Starting_Number=int(input("Enter Starting Number: "))

        self.Ending_Number=int(input("Enter Ending Number: "))

        lower_number = int(self.Starting_Number)

        upper_number = int(self.Ending_Number)

        print("\nPrime Numbers between the given range:")

        for self.num in range(lower_number, upper_number+1):

            if self.num>1:

                for i in range(2, self.num):

                    if(self.num%i)==0:

                        break

                else:

                    print(self.num)
```

```
r=PrimeNumber()

r.display_Condition()
```

Output :-



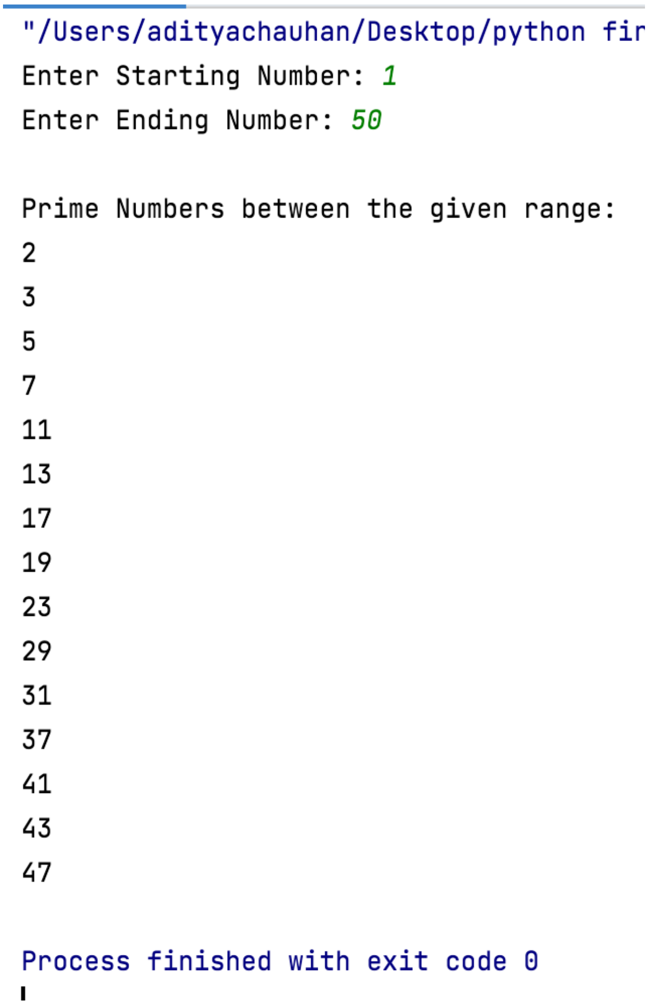
```
"/Users/adityachauhan/Desktop/python
Enter the number : 15
Number is not prime

Process finished with exit code 0
```

Program:-9
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a program to display set of prime numbers between the given input range from user.
Enter start number : 10
Enter end number : 30
11,13,17,19,23,29

Input :-

```
class PrimeNumber:

    def __int__(self):

        self.Starting_Number=0

        self.Ending_Number=0

        self.num=0


    def display_Condition(self):

        self.Starting_Number=int(input("Enter Starting Number: "))

        self.Ending_Number=int(input("Enter Ending Number: "))
        lower_number = int(self.Starting_Number)
        upper_number = int(self.Ending_Number)
        print("\nPrime Numbers between the given range:")
        for self.num in range(lower_number, upper_number+1):
            if self.num>1:
                for i in range(2, self.num):
                    if(self.num%i)==0:
                        break
                else:
                    print(self.num)


r=PrimeNumber()
r.display_Condition()
```

Output :-



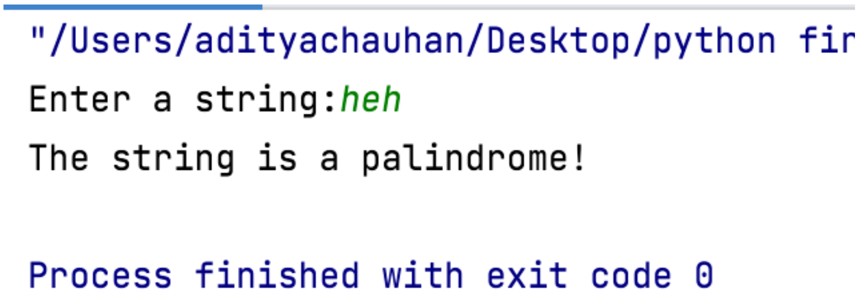
```
"/Users/adityachauhan/Desktop/python fir
Enter Starting Number: 1
Enter Ending Number: 50

Prime Numbers between the given range:
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47

Process finished with exit code 0
```

Program:-10
@author: 20124051 Aditya Chauhan
@description: Program No. -
A program to check whether inputted string is palindrome or not.
Eg Enter a name : liril
Liril is a palindrom

Input :-

```
class Palindrome:
    def __int__(self):
        self.String = ''


    def display_Condition(self):


        self.String = input("Enter a string:")


        if (self.String == self.String[::-1]):
            print("The string is a palindrome!")
        else:
            print("The string isn't a palindrome!")



r = Palindrome()
r.display_Condition()
```

Output :-



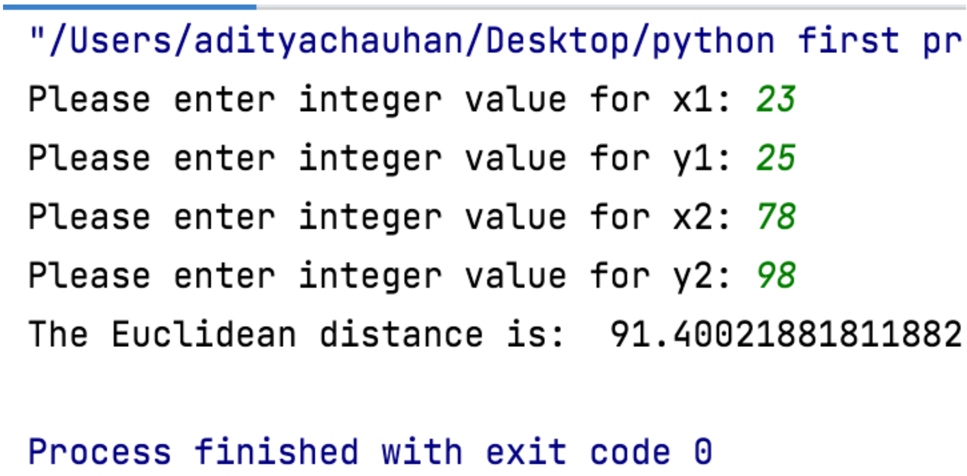
```
"/Users/adityachauhan/Desktop/python fir
Enter a string:heh
The string is a palindrome!

Process finished with exit code 0
```

Program:-11
@author: 20124051 Aditya Chauhan
@description: Program No. -
Write a OO program to find Euclidean Distance.

Input :-

```
class Point:


    def __init__(self):
        self.__x1 = 0
        self.__y1 = 0
        self.__x2 = 0
        self.__y2 = 0
        self.__delta_x = 0
        self.__delta_y =  0
        self.__euDist = 0


    def get_input(self):
        self.__x1 = int(input('Please enter integer value for x1: '))
        self.__y1 = int(input('Please enter integer value for y1: '))
        self.__x2 = int(input('Please enter integer value for x2: '))
        self.__y2 = int(input('Please enter integer value for y2: '))


    def dist_to_point(self):
        self.__delta_x = self.__x2 - self.__x1
        self.__delta_y = self.__y2 - self.__y1
        self.__euDist = (self.__delta_x ** 2 + self.__delta_y
** 2) ** 0.5

        return self.__euDist


    def display(self):

        print("The Euclidean distance is: ",self.__euDist)


run = Point()

run.get_input()

run.dist_to_point()

run.display()
```

Output :-



```
"/Users/adityachauhan/Desktop/python first pr
Please enter integer value for x1: 23
Please enter integer value for y1: 25
Please enter integer value for x2: 78
Please enter integer value for y2: 98
The Euclidean distance is:  91.40021881811882

Process finished with exit code 0
```