Django Internship Report

Personal Details

Name	:	Nishant Movaliya
College Name	:	Government Engineering College,Modasa
Degree	:	BE
Semester	:	7
Github URL	:	https://github.com/nishantmovaliya

Company Details

Company Name	:	Akash Technolabs
External Guide	:	Akash Padhiyar
Training Duration	:	26-05-2021 to 14-06-2021

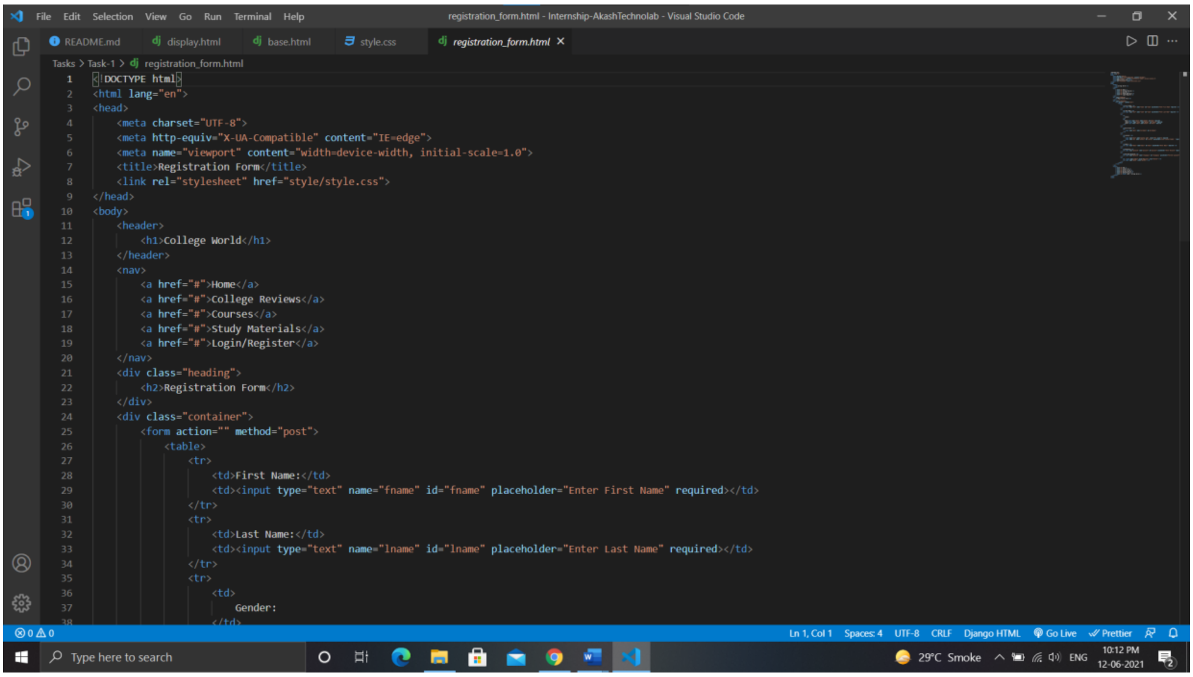
Index

Index	Task Detail	Page Name
1	Basic HTML based : To Create Registration form using div/table.	3
2	Write Basic Python Programs which is run in lecture.	5
3	Complete the Task Sent by Ma'am (15 Examples)	8
4	1.List All the Operators 2.Functions and Modules	11
5	Solve the Tasks Sent by Ma'am: (OOP-Python)	15
6	Install Django, Run and Display Browser Window	19
7	Display Hello World Text in Browser in Django	21
8	Template Integration and URL Routing	23
9	Using POST method to get User Values and returning back on page.	26
10	Implementing Models and Fetching Values using Function Based Views and Class Based Views	29

Task-1

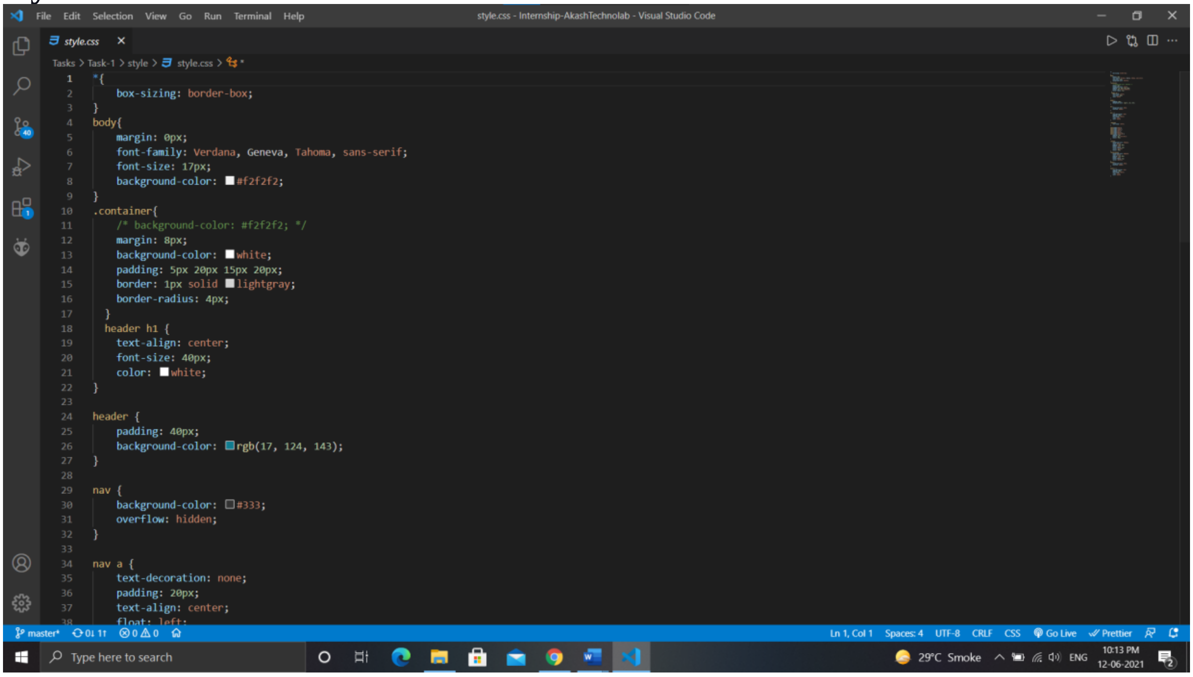
Basic HTML based : To Create Registration form using div/table.

registration_form.html


File Edit Selection View Go Run Terminal Help
registration_form.html - Internship-AkashTechnolab - Visual Studio Code
README.md
display.html
base.html
style.css
registration_form.html
Tasks > Task-1 > registration_form.html
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Django HTML Go Live Prettier
Type here to search
29°C Smoke ENG 10:12 PM 12-06-2021

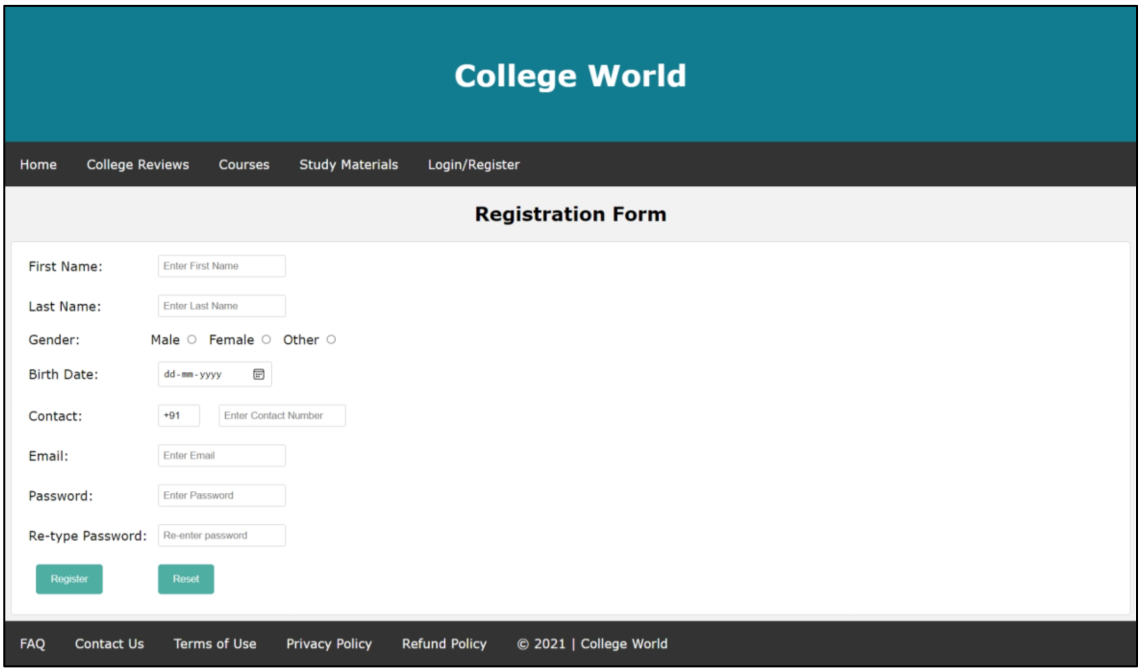
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form</title>
    <link rel="stylesheet" href="style/style.css">
</head>
<body>
    <header>
        <h1>College World</h1>
    </header>
    <nav>
        <a href="#">Home</a>
        <a href="#">College Reviews</a>
        <a href="#">Courses</a>
        <a href="#">Study Materials</a>
        <a href="#">Login/Register</a>
    </nav>
    <div class="heading">
        <h2>Registration Form</h2>
    </div>
    <div class="container">
        <form action="" method="post">
            <table>
                <tr>
                    <td>First Name:</td>
                    <td><input type="text" name="fname" id="fname" placeholder="Enter First Name" required></td>
                </tr>
                <tr>
                    <td>Last Name:</td>
                    <td><input type="text" name="lname" id="lname" placeholder="Enter Last Name" required></td>
                </tr>
                <tr>
                    <td>
                        Gender:
                    </td>
```

style.css


File Edit Selection View Go Run Terminal Help
style.css - Internship-AkashTechnolab - Visual Studio Code
style.css
Tasks > Task-1 > style > style.css > *
master* 0↓ 1↑ 0 0
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF CSS Go Live Prettier
Type here to search
29°C Smoke ENG 10:13 PM 12-06-2021

```css
*{
    box-sizing: border-box;
}
body{
    margin: 0px;
    font-family: Verdana, Geneva, Tahoma, sans-serif;
    font-size: 17px;
    background-color: #f2f2f2;
}
.container{
    /* background-color: #f2f2f2; */
    margin: 8px;
    background-color: white;
    padding: 5px 20px 15px 20px;
    border: 1px solid lightgray;
    border-radius: 4px;
  }
  header h1 {
    text-align: center;
    font-size: 40px;
    color: white;
}

header {
    padding: 40px;
    background-color: rgb(17, 124, 143);
}

nav {
    background-color: #333;
    overflow: hidden;
}

nav a {
    text-decoration: none;
    padding: 20px;
    text-align: center;
    float: left;
```

Output:



College World

Registration Form

First Name: Enter First Name

Last Name: Enter Last Name

Gender: Male ○ Female ○ Other ○

Birth Date: dd-mm-yyyy

Contact: +91 Enter Contact Number

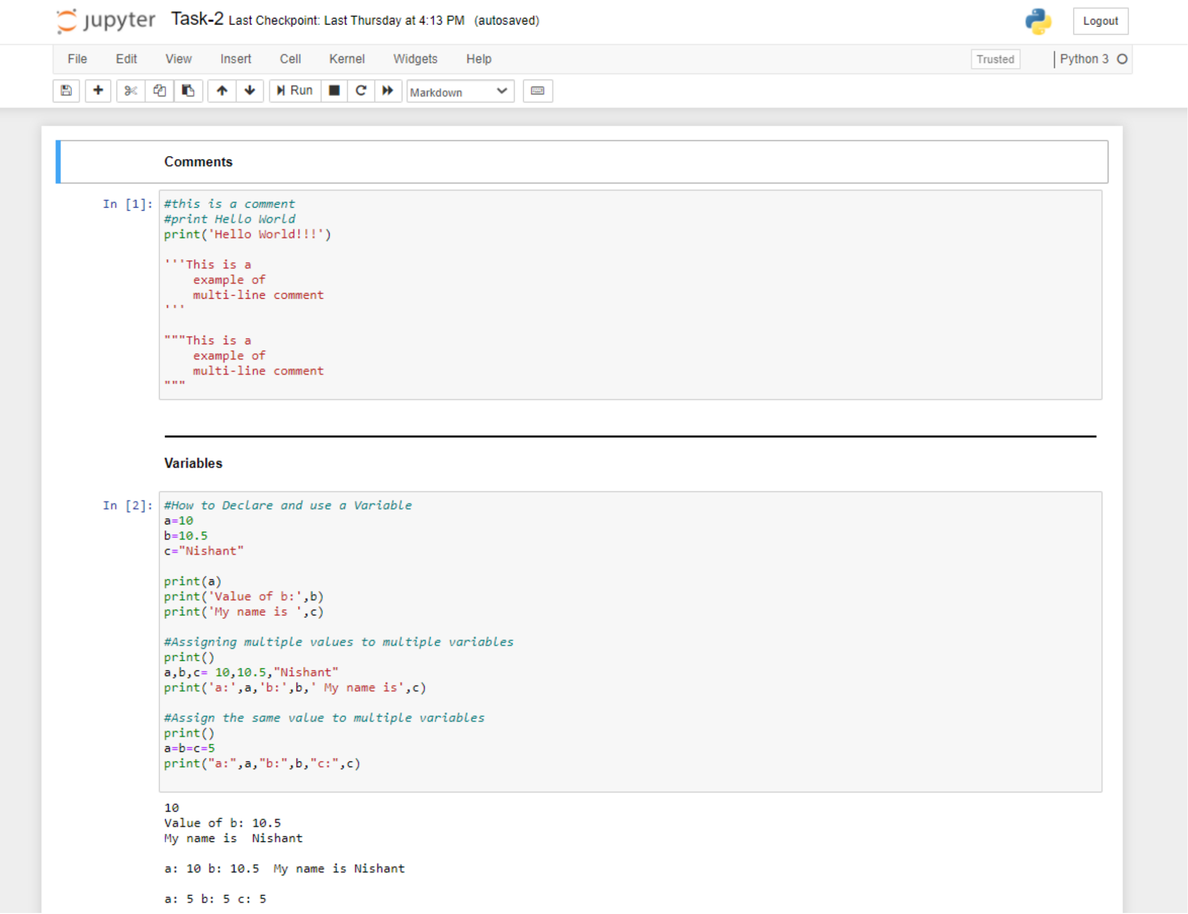
Email: Enter Email

Password: Enter Password

Re-type Password: Re-enter password

Task-2

Task: Write Basic Python Programs which is run in lecture.



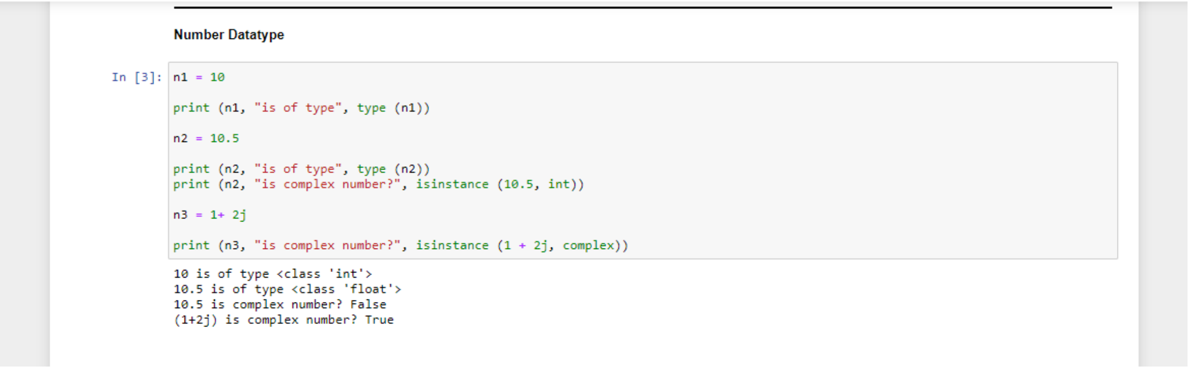
Comments

In [1]:
```
#this is a comment
#print Hello World
print('Hello World!!!')

'''This is a
    example of
    multi-line comment
'''

"""This is a
    example of
    multi-line comment
"""
```

Variables

In [2]:
```
#How to Declare and use a Variable
a=10
b=10.5
c="Nishant"

print(a)
print('Value of b:',b)
print('My name is ',c)

#Assigning multiple values to multiple variables
print()
a,b,c= 10,10.5,"Nishant"
print('a:',a,'b:',b,' My name is',c)

#Assign the same value to multiple variables
print()
a=b=c=5
print("a:",a,"b:",b,"c:",c)
```

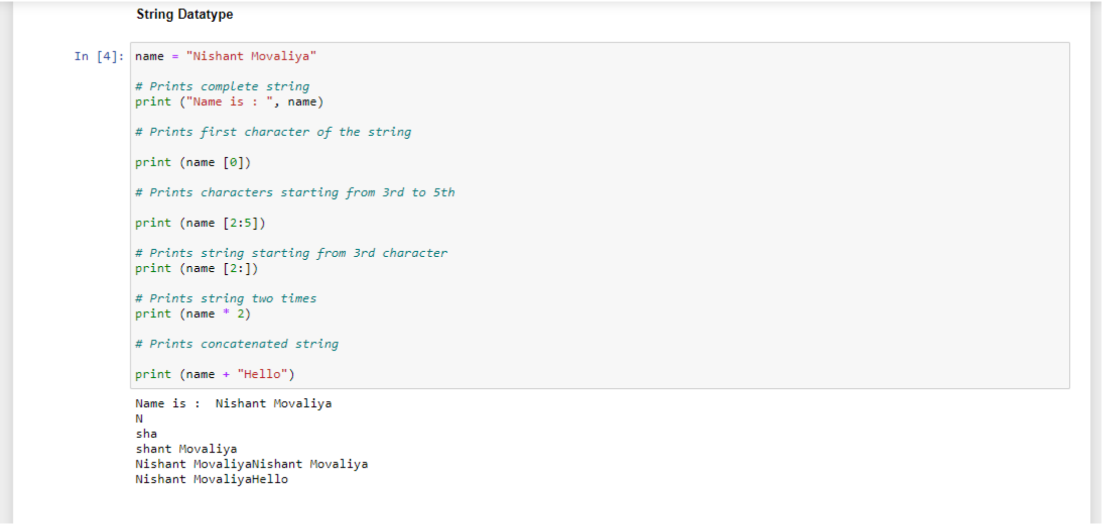
```
10
Value of b: 10.5
My name is  Nishant

a: 10 b: 10.5  My name is Nishant

a: 5 b: 5 c: 5
```



Number Datatype

In [3]:
```
n1 = 10

print (n1, "is of type", type (n1))

n2 = 10.5

print (n2, "is of type", type (n2))
print (n2, "is complex number?", isinstance (10.5, int))

n3 = 1+ 2j

print (n3, "is complex number?", isinstance (1 + 2j, complex))
```

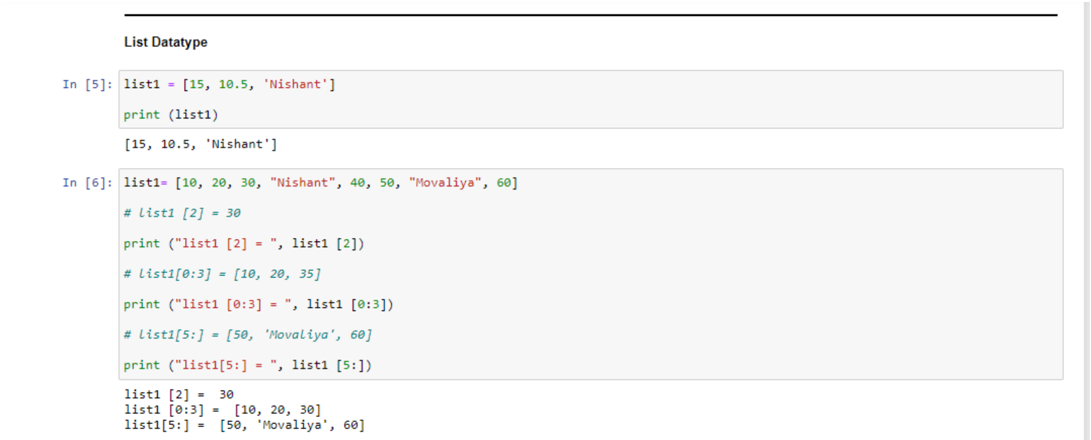
```
10 is of type <class 'int'>
10.5 is of type <class 'float'>
10.5 is complex number? False
(1+2j) is complex number? True
```



String Datatype

In [4]:
```
name = "Nishant Movaliya"

# Prints complete string
print ("Name is : ", name)

# Prints first character of the string

print (name [0])

# Prints characters starting from 3rd to 5th

print (name [2:5])

# Prints string starting from 3rd character
print (name [2:])

# Prints string two times
print (name * 2)

# Prints concatenated string

print (name + "Hello")
```

```
Name is :  Nishant Movaliya
N
sha
shant Movaliya
Nishant MovaliyaNishant Movaliya
Nishant MovaliyaHello
```



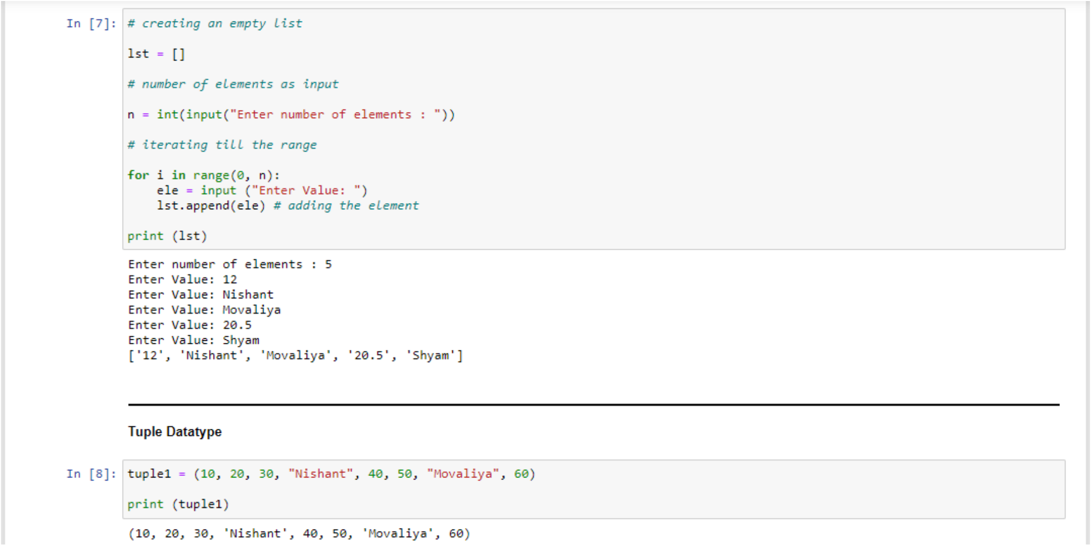
List Datatype

In [5]:
```
list1 = [15, 10.5, 'Nishant']

print (list1)
```

```
[15, 10.5, 'Nishant']
```

In [6]:
```
list1= [10, 20, 30, "Nishant", 40, 50, "Movaliya", 60]

# list1 [2] = 30

print ("list1 [2] = ", list1 [2])

# list1[0:3] = [10, 20, 35]

print ("list1 [0:3] = ", list1 [0:3])

# list1[5:] = [50, 'Movaliya', 60]

print ("list1[5:] = ", list1 [5:])
```

```
list1 [2] =  30
list1 [0:3] =  [10, 20, 30]
list1[5:] =  [50, 'Movaliya', 60]
```



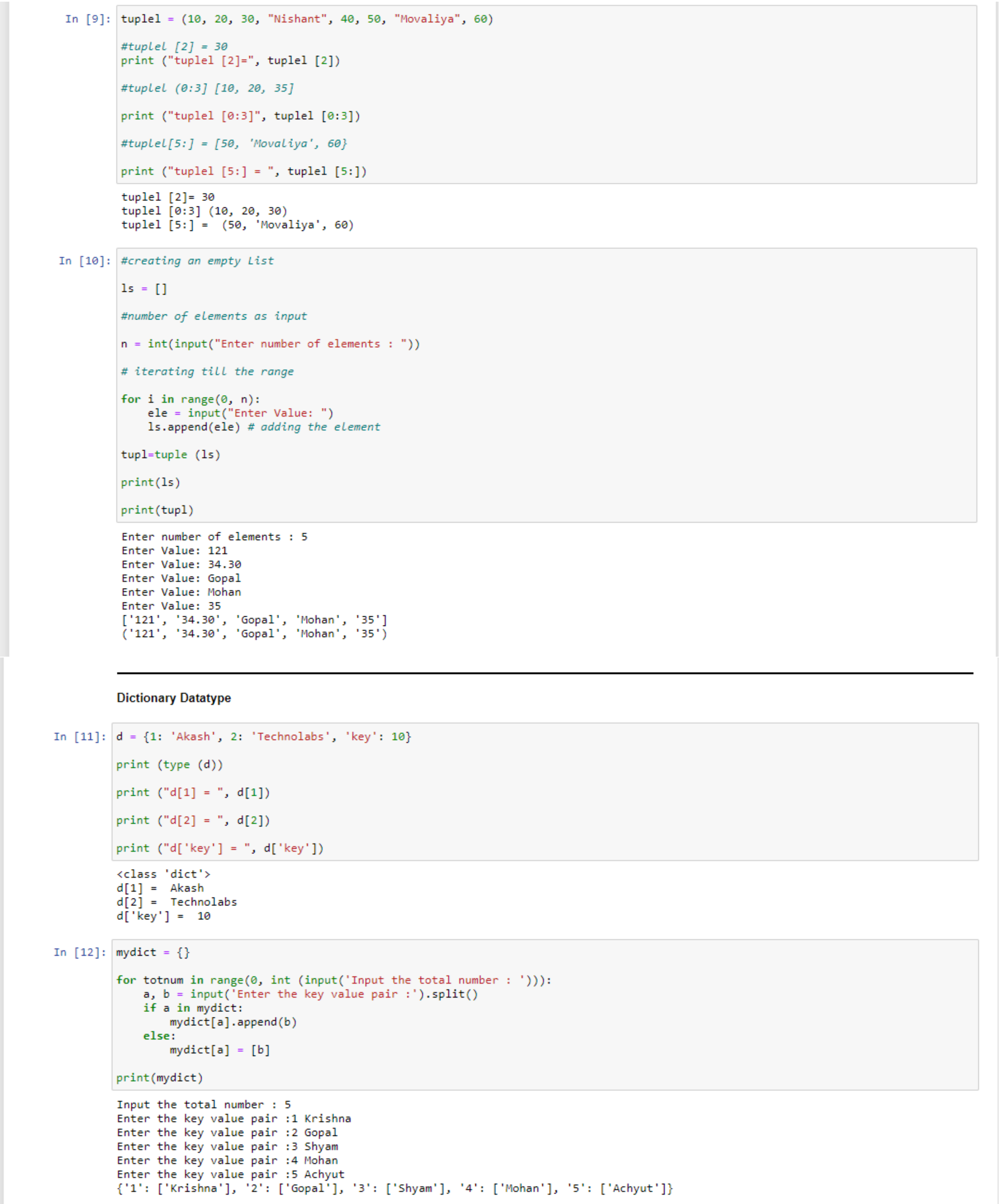
In [7]:
```
# creating an empty list

lst = []

# number of elements as input

n = int(input("Enter number of elements : "))

# iterating till the range

for i in range(0, n):
    ele = input ("Enter Value: ")
    lst.append(ele) # adding the element

print (lst)
```

```
Enter number of elements : 5
Enter Value: 12
Enter Value: Nishant
Enter Value: Movaliya
Enter Value: 20.5
Enter Value: Shyam
['12', 'Nishant', 'Movaliya', '20.5', 'Shyam']
```

Tuple Datatype

In [8]:
```
tuple1 = (10, 20, 30, "Nishant", 40, 50, "Movaliya", 60)

print (tuple1)
```

```
(10, 20, 30, 'Nishant', 40, 50, 'Movaliya', 60)
```



```
In [9]: tuplel = (10, 20, 30, "Nishant", 40, 50, "Movaliya", 60)

        #tuplel [2] = 30
        print ("tuplel [2]=", tuplel [2])

        #tuplel (0:3] [10, 20, 35]

        print ("tuplel [0:3]", tuplel [0:3])

        #tuplel[5:] = [50, 'Movaliya', 60}

        print ("tuplel [5:] = ", tuplel [5:])
```

```
tuplel [2]= 30
tuplel [0:3] (10, 20, 30)
tuplel [5:] =  (50, 'Movaliya', 60)
```

```
In [10]: #creating an empty list

         ls = []

         #number of elements as input

         n = int(input("Enter number of elements : "))

         # iterating till the range

         for i in range(0, n):
             ele = input("Enter Value: ")
             ls.append(ele) # adding the element

         tupl=tuple (ls)

         print(ls)

         print(tupl)
```

```
Enter number of elements : 5
Enter Value: 121
Enter Value: 34.30
Enter Value: Gopal
Enter Value: Mohan
Enter Value: 35
['121', '34.30', 'Gopal', 'Mohan', '35']
('121', '34.30', 'Gopal', 'Mohan', '35')
```

Dictionary Datatype

```
In [11]: d = {1: 'Akash', 2: 'Technolabs', 'key': 10}

         print (type (d))

         print ("d[1] = ", d[1])

         print ("d[2] = ", d[2])

         print ("d['key'] = ", d['key'])
```

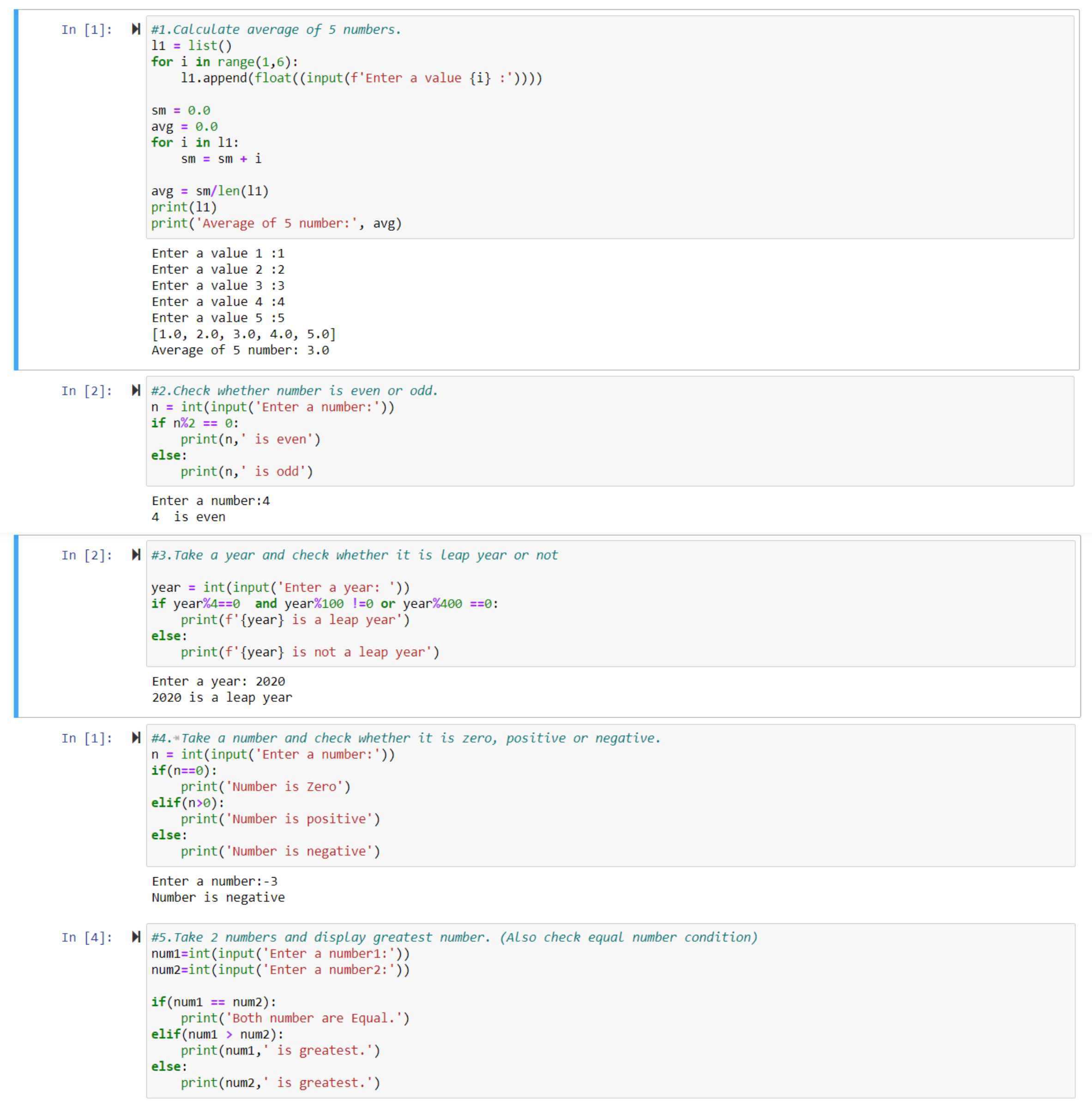
```
<class 'dict'>
d[1] =  Akash
d[2] =  Technolabs
d['key'] =  10
```

```
In [12]: mydict = {}

         for totnum in range(0, int (input('Input the total number : '))):
             a, b = input('Enter the key value pair :').split()
             if a in mydict:
                 mydict[a].append(b)
             else:
                 mydict[a] = [b]

         print(mydict)
```

```
Input the total number : 5
Enter the key value pair :1 Krishna
Enter the key value pair :2 Gopal
Enter the key value pair :3 Shyam
Enter the key value pair :4 Mohan
Enter the key value pair :5 Achyut
{'1': ['Krishna'], '2': ['Gopal'], '3': ['Shyam'], '4': ['Mohan'], '5': ['Achyut']}
```

Task-3:

Task: Complete the Task Sent by Ma'am (15 Examples)



```
In [1]:
#1.Calculate average of 5 numbers.
l1 = list()
for i in range(1,6):
    l1.append(float((input(f'Enter a value {i} :'))))

sm = 0.0
avg = 0.0
for i in l1:
    sm = sm + i

avg = sm/len(l1)
print(l1)
print('Average of 5 number:', avg)
```

```
Enter a value 1 :1
Enter a value 2 :2
Enter a value 3 :3
Enter a value 4 :4
Enter a value 5 :5
[1.0, 2.0, 3.0, 4.0, 5.0]
Average of 5 number: 3.0
```

```
In [2]:
#2.Check whether number is even or odd.
n = int(input('Enter a number:'))
if n%2 == 0:
    print(n,' is even')
else:
    print(n,' is odd')
```

```
Enter a number:4
4  is even
```

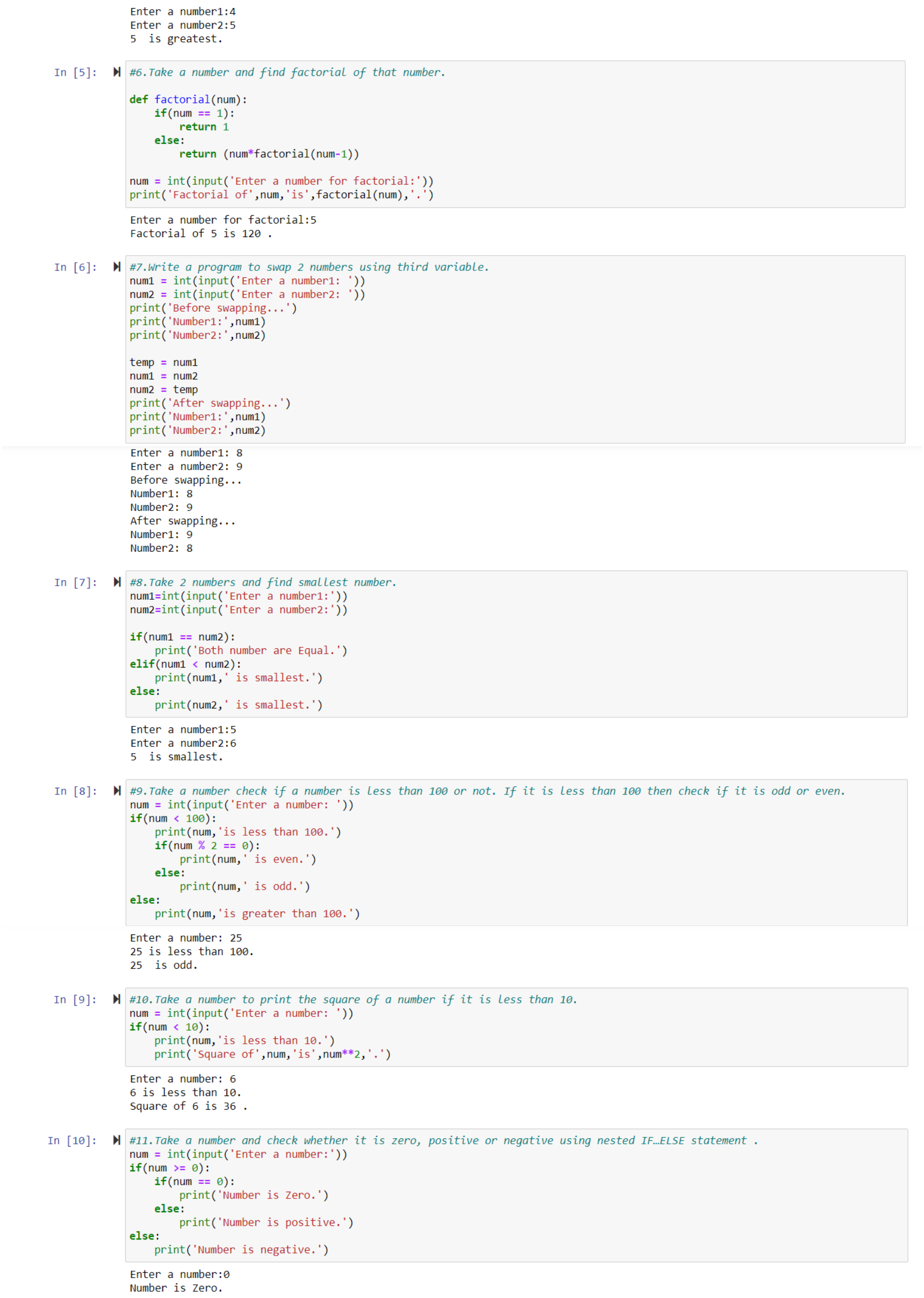
```
In [2]:
#3.Take a year and check whether it is leap year or not

year = int(input('Enter a year: '))
if year%4==0  and year%100 !=0 or year%400 ==0:
    print(f'{year} is a leap year')
else:
    print(f'{year} is not a leap year')
```

```
Enter a year: 2020
2020 is a leap year
```

```
In [1]:
#4.Take a number and check whether it is zero, positive or negative.
n = int(input('Enter a number:'))
if(n==0):
    print('Number is Zero')
elif(n>0):
    print('Number is positive')
else:
    print('Number is negative')
```

```
Enter a number:-3
Number is negative
```

```
In [4]:
#5.Take 2 numbers and display greatest number. (Also check equal number condition)
num1=int(input('Enter a number1:'))
num2=int(input('Enter a number2:'))

if(num1 == num2):
    print('Both number are Equal.')
elif(num1 > num2):
    print(num1,' is greatest.')
else:
    print(num2,' is greatest.')
```



```
Enter a number1:4
Enter a number2:5
5  is greatest.
```

In [5]:

```python
#6.Take a number and find factorial of that number.

def factorial(num):
    if(num == 1):
        return 1
    else:
        return (num*factorial(num-1))

num = int(input('Enter a number for factorial:'))
print('Factorial of',num,'is',factorial(num),'.')
```

```
Enter a number for factorial:5
Factorial of 5 is 120 .
```

In [6]:

```python
#7.Write a program to swap 2 numbers using third variable.
num1 = int(input('Enter a number1: '))
num2 = int(input('Enter a number2: '))
print('Before swapping...')
print('Number1:',num1)
print('Number2:',num2)

temp = num1
num1 = num2
num2 = temp
print('After swapping...')
print('Number1:',num1)
print('Number2:',num2)
```

```
Enter a number1: 8
Enter a number2: 9
Before swapping...
Number1: 8
Number2: 9
After swapping...
Number1: 9
Number2: 8
```

In [7]:

```python
#8.Take 2 numbers and find smallest number.
num1=int(input('Enter a number1:'))
num2=int(input('Enter a number2:'))

if(num1 == num2):
    print('Both number are Equal.')
elif(num1 < num2):
    print(num1,' is smallest.')
else:
    print(num2,' is smallest.')
```

```
Enter a number1:5
Enter a number2:6
5  is smallest.
```

In [8]:

```python
#9.Take a number check if a number is less than 100 or not. If it is less than 100 then check if it is odd or even.
num = int(input('Enter a number: '))
if(num < 100):
    print(num,'is less than 100.')
    if(num % 2 == 0):
        print(num,' is even.')
    else:
        print(num,' is odd.')
else:
    print(num,'is greater than 100.')
```

```
Enter a number: 25
25 is less than 100.
25  is odd.
```

In [9]:

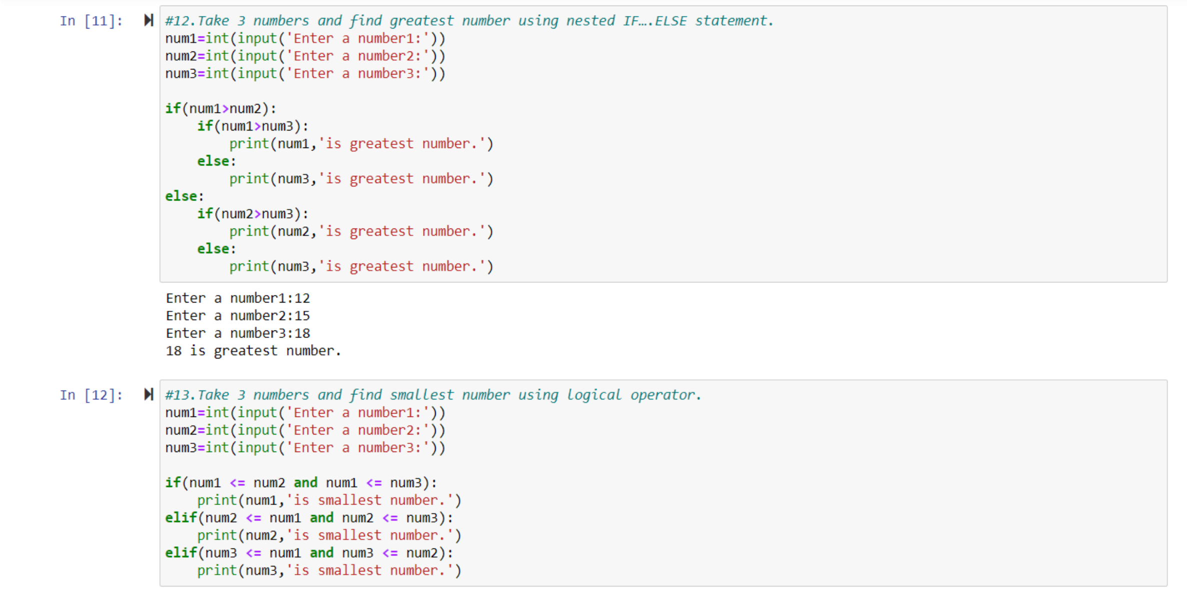
```python
#10.Take a number to print the square of a number if it is less than 10.
num = int(input('Enter a number: '))
if(num < 10):
    print(num,'is less than 10.')
    print('Square of',num,'is',num**2,'.')
```

```
Enter a number: 6
6 is less than 10.
Square of 6 is 36 .
```

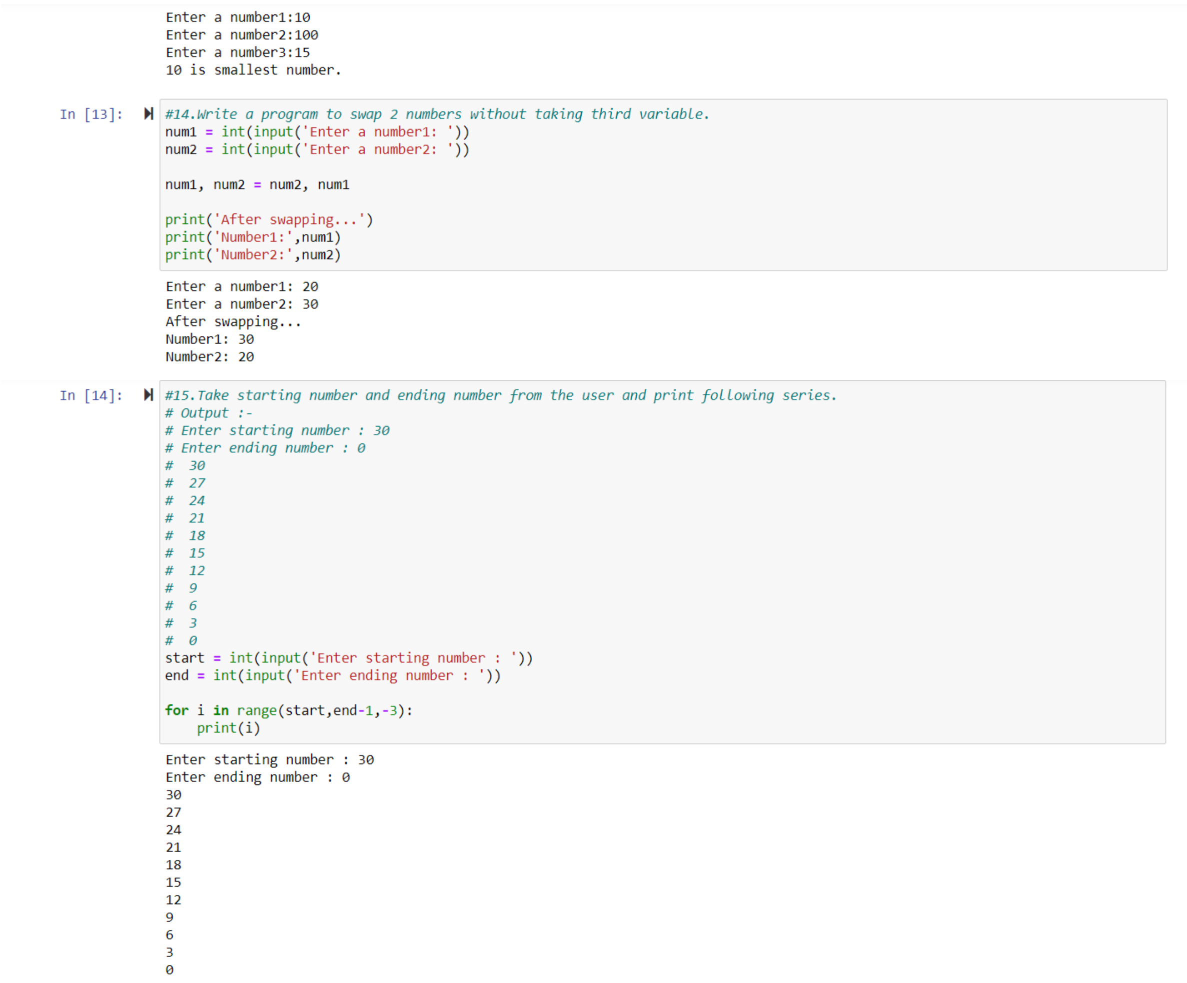
In [10]:

```python
#11.Take a number and check whether it is zero, positive or negative using nested IF…ELSE statement .
num = int(input('Enter a number:'))
if(num >= 0):
    if(num == 0):
        print('Number is Zero.')
    else:
        print('Number is positive.')
else:
    print('Number is negative.')
```

```
Enter a number:0
Number is Zero.
```



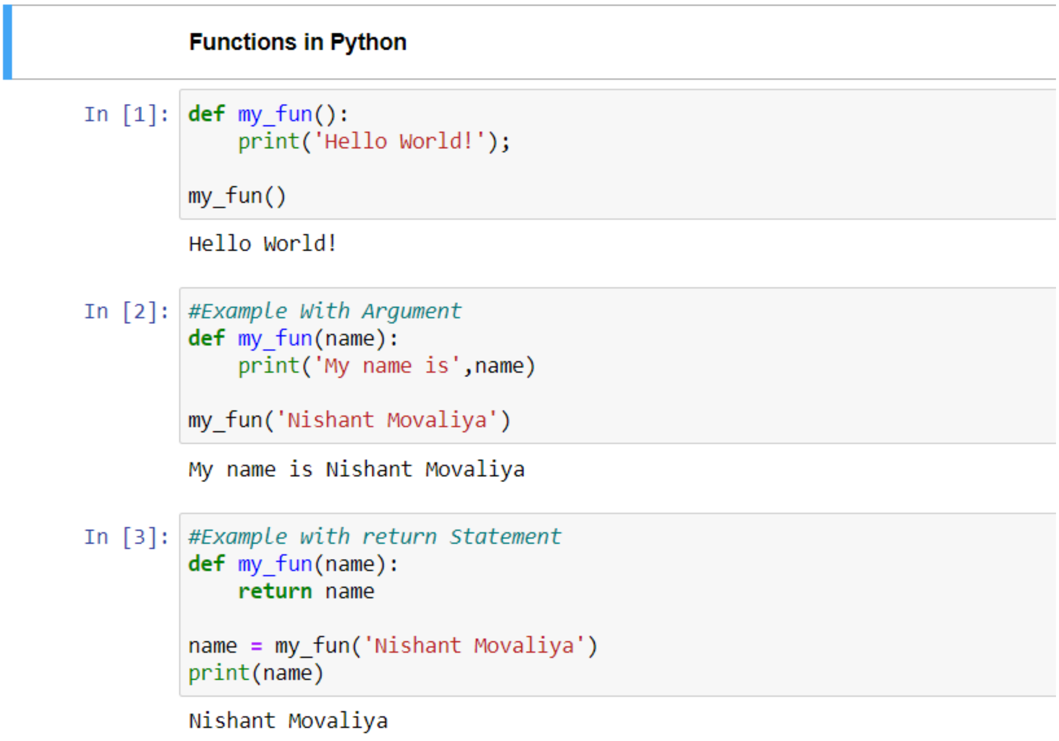
```
In [11]:  #12.Take 3 numbers and find greatest number using nested IF….ELSE statement.
          num1=int(input('Enter a number1:'))
          num2=int(input('Enter a number2:'))
          num3=int(input('Enter a number3:'))

          if(num1>num2):
              if(num1>num3):
                  print(num1,'is greatest number.')
              else:
                  print(num3,'is greatest number.')
          else:
              if(num2>num3):
                  print(num2,'is greatest number.')
              else:
                  print(num3,'is greatest number.')

          Enter a number1:12
          Enter a number2:15
          Enter a number3:18
          18 is greatest number.

In [12]:  #13.Take 3 numbers and find smallest number using logical operator.
          num1=int(input('Enter a number1:'))
          num2=int(input('Enter a number2:'))
          num3=int(input('Enter a number3:'))

          if(num1 <= num2 and num1 <= num3):
              print(num1,'is smallest number.')
          elif(num2 <= num1 and num2 <= num3):
              print(num2,'is smallest number.')
          elif(num3 <= num1 and num3 <= num2):
              print(num3,'is smallest number.')
```



```
          Enter a number1:10
          Enter a number2:100
          Enter a number3:15
          10 is smallest number.

In [13]:  #14.Write a program to swap 2 numbers without taking third variable.
          num1 = int(input('Enter a number1: '))
          num2 = int(input('Enter a number2: '))

          num1, num2 = num2, num1

          print('After swapping...')
          print('Number1:',num1)
          print('Number2:',num2)

          Enter a number1: 20
          Enter a number2: 30
          After swapping...
          Number1: 30
          Number2: 20

In [14]:  #15.Take starting number and ending number from the user and print following series.
          # Output :-
          # Enter starting number : 30
          # Enter ending number : 0
          #  30
          #  27
          #  24
          #  21
          #  18
          #  15
          #  12
          #  9
          #  6
          #  3
          #  0
          start = int(input('Enter starting number : '))
          end = int(input('Enter ending number : '))

          for i in range(start,end-1,-3):
              print(i)

          Enter starting number : 30
          Enter ending number : 0
          30
          27
          24
          21
          18
          15
          12
          9
          6
          3
          0
```

Task-4

Task: 1.List All the Operators 2.Functions and Modules

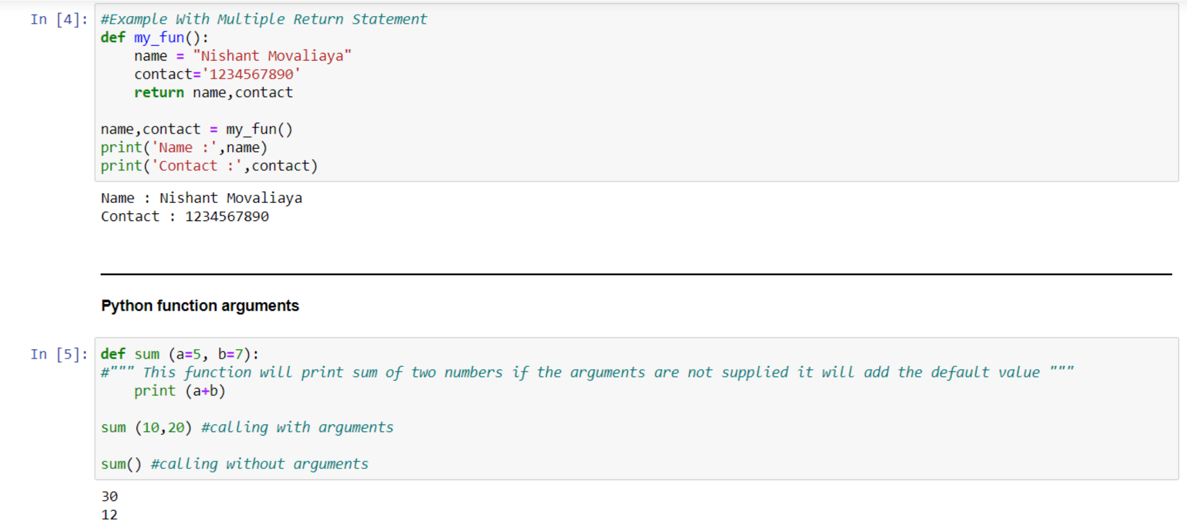


Functions in Python

```
In [1]: def my_fun():
            print('Hello World!');

        my_fun()

Hello World!
```

```
In [2]: #Example With Argument
        def my_fun(name):
            print('My name is',name)

        my_fun('Nishant Movaliya')

My name is Nishant Movaliya
```

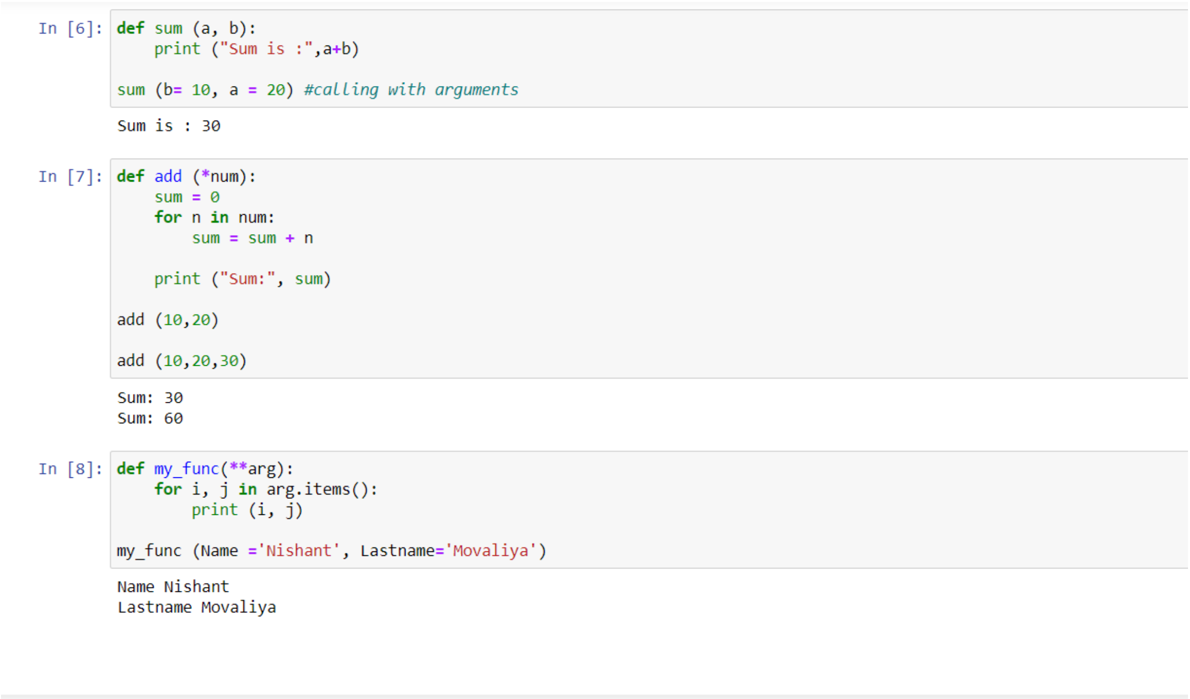
```
In [3]: #Example with return Statement
        def my_fun(name):
            return name

        name = my_fun('Nishant Movaliya')
        print(name)

Nishant Movaliya
```



```
In [4]: #Example With Multiple Return Statement
        def my_fun():
            name = "Nishant Movaliaya"
            contact='1234567890'
            return name,contact

        name,contact = my_fun()
        print('Name :',name)
        print('Contact :',contact)

Name : Nishant Movaliaya
Contact : 1234567890
```

Python function arguments

```
In [5]: def sum (a=5, b=7):
        #""" This function will print sum of two numbers if the arguments are not supplied it will add the default value """
            print (a+b)

        sum (10,20) #calling with arguments

        sum() #calling without arguments

30
12
```



In [6]:

```
def sum (a, b):
    print ("Sum is :",a+b)

sum (b= 10, a = 20) #calling with arguments
```

```
Sum is : 30
```

In [7]:

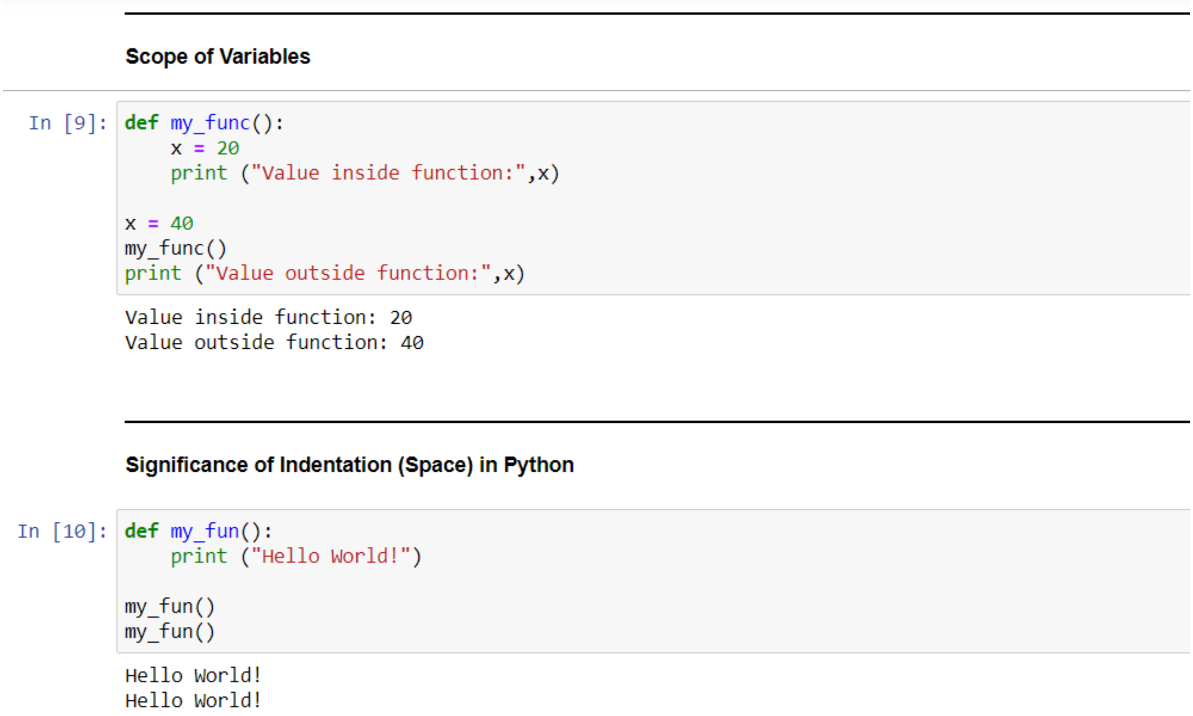
```
def add (*num):
    sum = 0
    for n in num:
        sum = sum + n

    print ("Sum:", sum)

add (10,20)

add (10,20,30)
```

```
Sum: 30
Sum: 60
```

In [8]:

```
def my_func(**arg):
    for i, j in arg.items():
        print (i, j)

my_func (Name ='Nishant', Lastname='Movaliya')
```

```
Name Nishant
Lastname Movaliya
```



Scope of Variables

In [9]:

```
def my_func():
    x = 20
    print ("Value inside function:",x)

x = 40
my_func()
print ("Value outside function:",x)
```

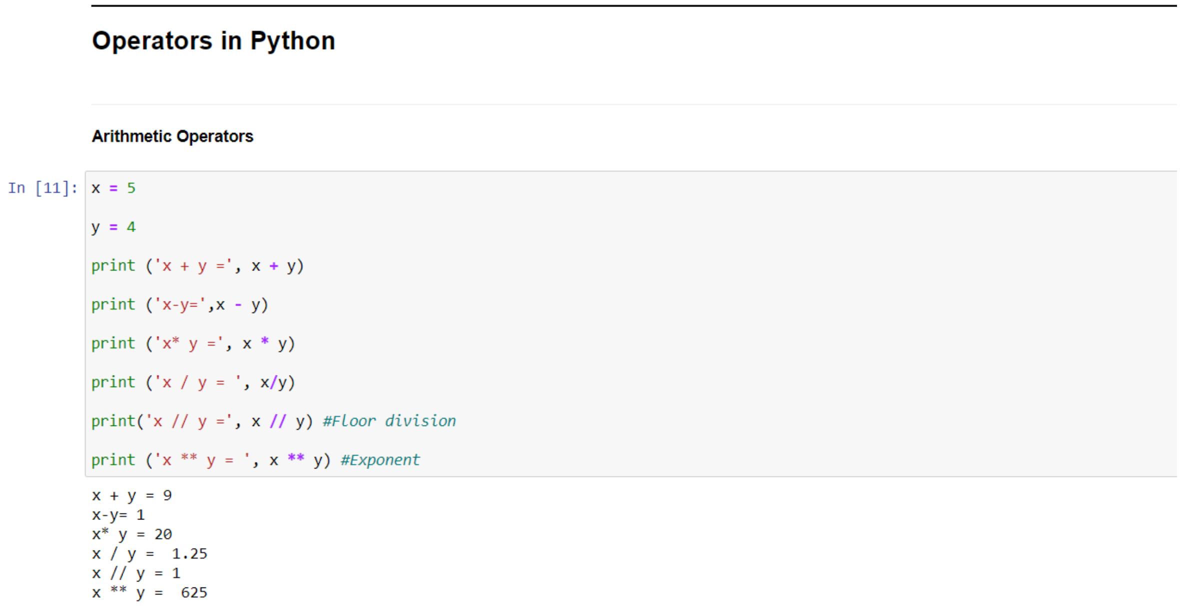
```
Value inside function: 20
Value outside function: 40
```

Significance of Indentation (Space) in Python

In [10]:

```
def my_fun():
    print ("Hello World!")

my_fun()
my_fun()
```

```
Hello World!
Hello World!
```



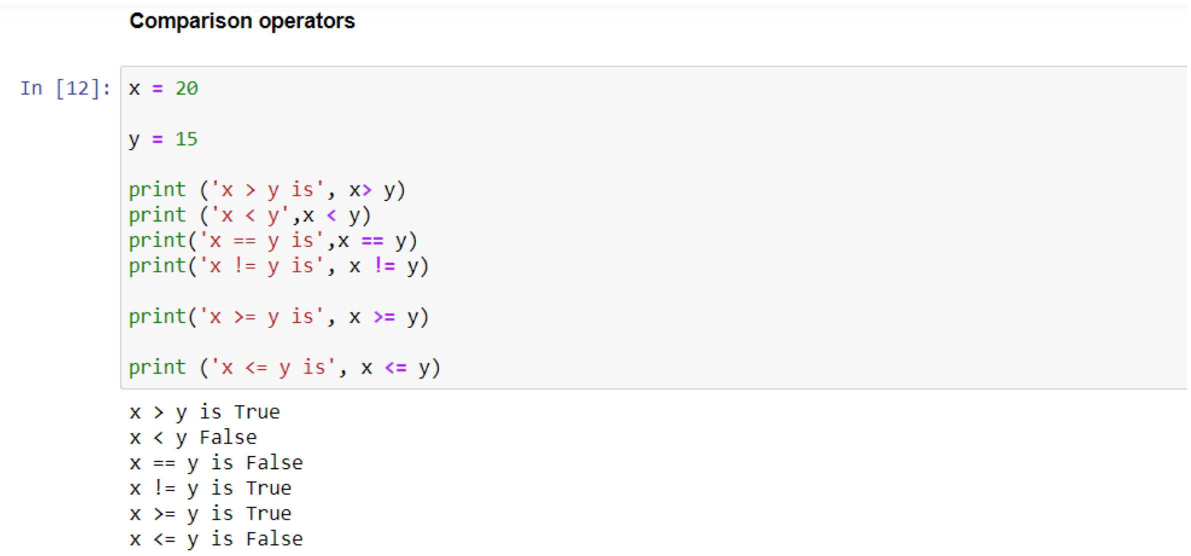
Operators in Python

Arithmetic Operators

In [11]:

```
x = 5

y = 4

print ('x + y =', x + y)

print ('x-y=',x - y)

print ('x* y =', x * y)

print ('x / y = ', x/y)

print('x // y =', x // y) #Floor division

print ('x ** y = ', x ** y) #Exponent
```

```
x + y = 9
x-y= 1
x* y = 20
x / y =  1.25
x // y = 1
x ** y =  625
```

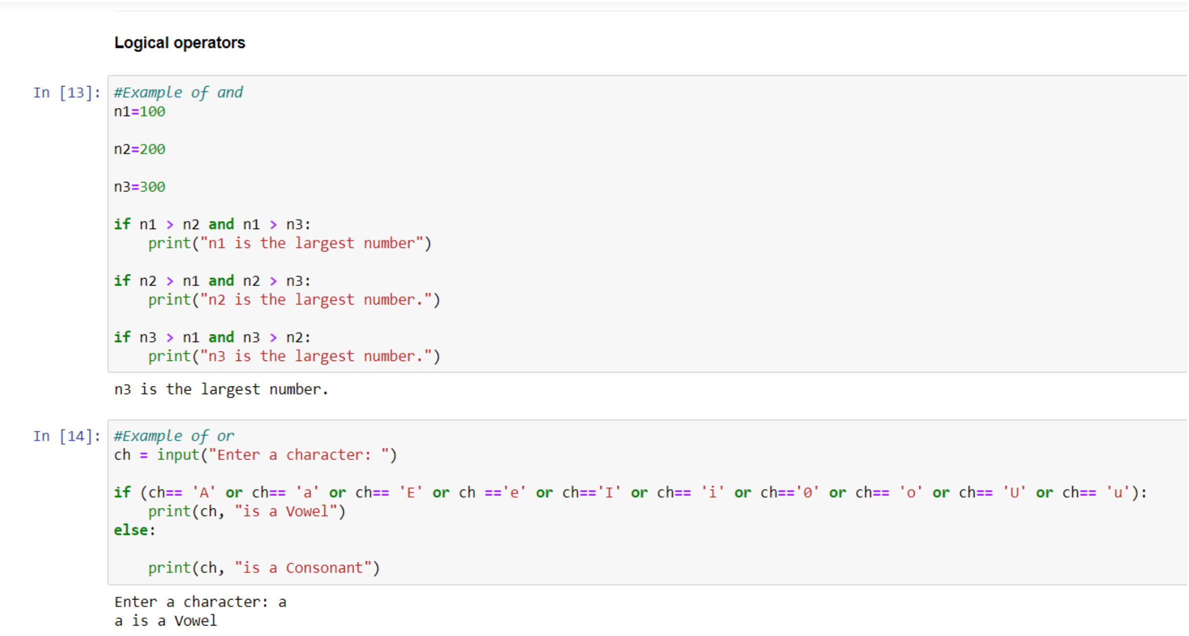


Comparison operators

In [12]:

```
x = 20

y = 15

print ('x > y is', x> y)
print ('x < y',x < y)
print('x == y is',x == y)
print('x != y is', x != y)

print('x >= y is', x >= y)

print ('x <= y is', x <= y)
```

```
x > y is True
x < y False
x == y is False
x != y is True
x >= y is True
x <= y is False
```



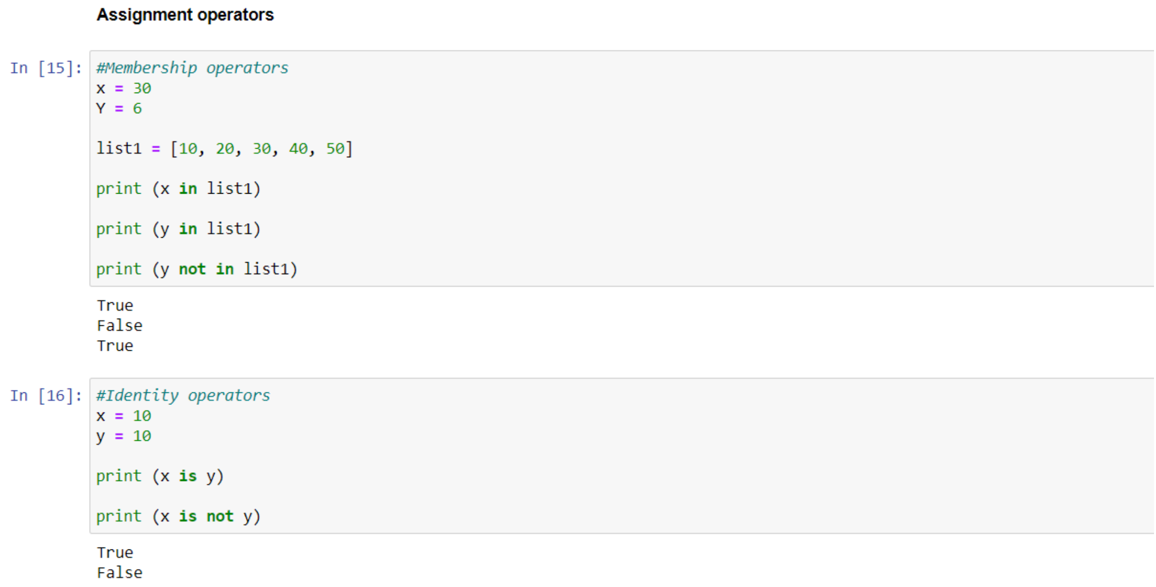
Logical operators

```
In [13]: #Example of and
         n1=100

         n2=200

         n3=300

         if n1 > n2 and n1 > n3:
             print("n1 is the largest number")

         if n2 > n1 and n2 > n3:
             print("n2 is the largest number.")

         if n3 > n1 and n3 > n2:
             print("n3 is the largest number.")
```

```
n3 is the largest number.
```

```
In [14]: #Example of or
         ch = input("Enter a character: ")

         if (ch== 'A' or ch== 'a' or ch== 'E' or ch =='e' or ch=='I' or ch== 'i' or ch=='0' or ch== 'o' or ch== 'U' or ch== 'u'):
             print(ch, "is a Vowel")
         else:

             print(ch, "is a Consonant")
```

```
Enter a character: a
a is a Vowel
```



Assignment operators

```
In [15]: #Membership operators
         x = 30
         Y = 6

         list1 = [10, 20, 30, 40, 50]

         print (x in list1)

         print (y in list1)

         print (y not in list1)
```

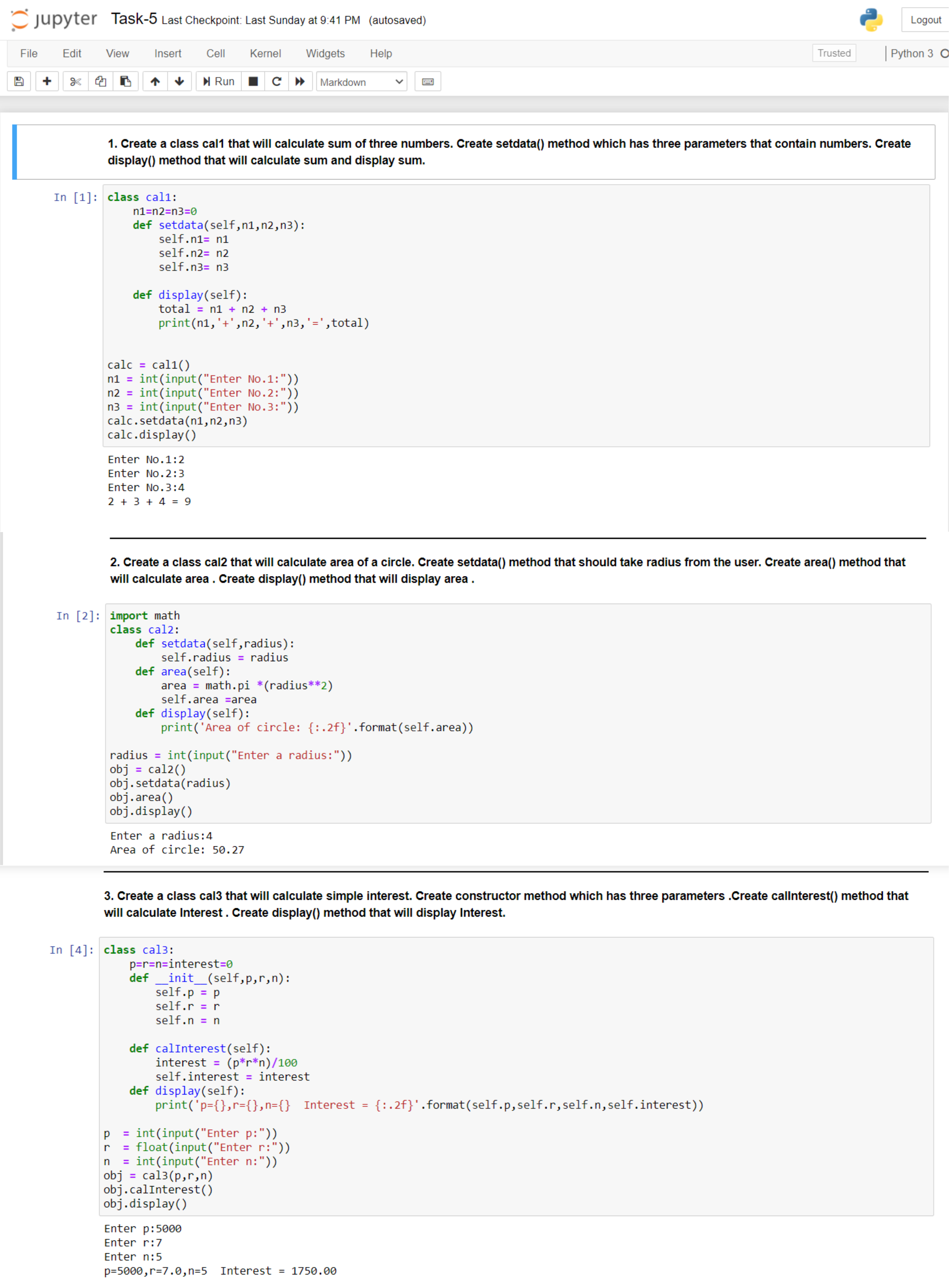
```
True
False
True
```

```
In [16]: #Identity operators
         x = 10
         y = 10

         print (x is y)

         print (x is not y)
```

```
True
False
```

Task-5

Task- Solve the Tasks Sent by Ma'am: (OOP-Python)


jupyter
Task-5 Last Checkpoint: Last Sunday at 9:41 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown

1. Create a class cal1 that will calculate sum of three numbers. Create setdata() method which has three parameters that contain numbers. Create display() method that will calculate sum and display sum.

In [1]:

```
class cal1:
    n1=n2=n3=0
    def setdata(self,n1,n2,n3):
        self.n1= n1
        self.n2= n2
        self.n3= n3

    def display(self):
        total = n1 + n2 + n3
        print(n1,'+',n2,'+',n3,'=',total)


calc = cal1()
n1 = int(input("Enter No.1:"))
n2 = int(input("Enter No.2:"))
n3 = int(input("Enter No.3:"))
calc.setdata(n1,n2,n3)
calc.display()
```

```
Enter No.1:2
Enter No.2:3
Enter No.3:4
2 + 3 + 4 = 9
```

2. Create a class cal2 that will calculate area of a circle. Create setdata() method that should take radius from the user. Create area() method that will calculate area . Create display() method that will display area .

In [2]:

```
import math
class cal2:
    def setdata(self,radius):
        self.radius = radius
    def area(self):
        area = math.pi *(radius**2)
        self.area =area
    def display(self):
        print('Area of circle: {:.2f}'.format(self.area))

radius = int(input("Enter a radius:"))
obj = cal2()
obj.setdata(radius)
obj.area()
obj.display()
```

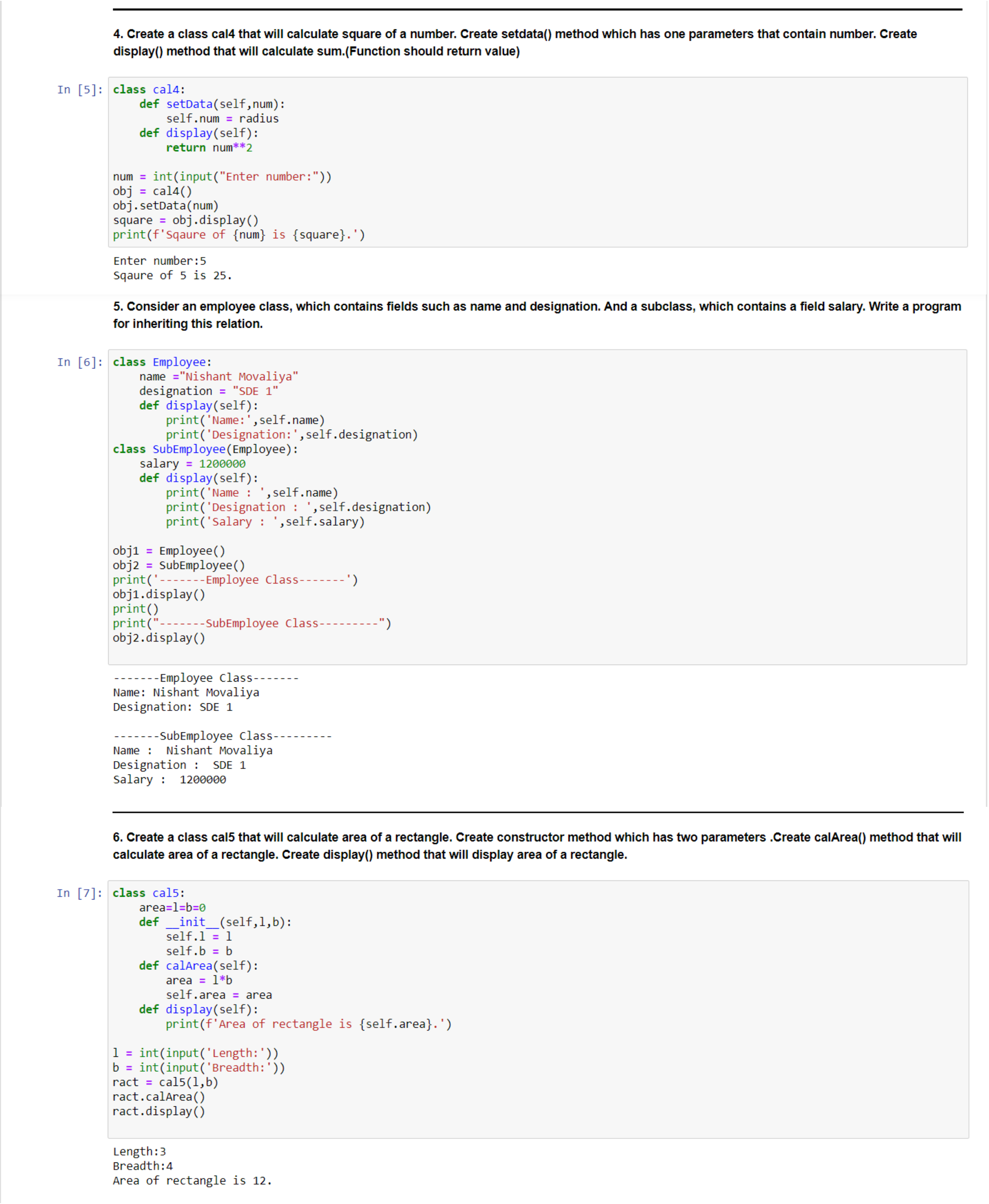
```
Enter a radius:4
Area of circle: 50.27
```

3. Create a class cal3 that will calculate simple interest. Create constructor method which has three parameters .Create callnterest() method that will calculate Interest . Create display() method that will display Interest.

In [4]:

```
class cal3:
    p=r=n=interest=0
    def __init__(self,p,r,n):
        self.p = p
        self.r = r
        self.n = n

    def calInterest(self):
        interest = (p*r*n)/100
        self.interest = interest
    def display(self):
        print('p={},r={},n={}  Interest = {:.2f}'.format(self.p,self.r,self.n,self.interest))

p  = int(input("Enter p:"))
r  = float(input("Enter r:"))
n  = int(input("Enter n:"))
obj = cal3(p,r,n)
obj.calInterest()
obj.display()
```

```
Enter p:5000
Enter r:7
Enter n:5
p=5000,r=7.0,n=5  Interest = 1750.00
```



4. Create a class cal4 that will calculate square of a number. Create setdata() method which has one parameters that contain number. Create display() method that will calculate sum.(Function should return value)

In [5]:

```python
class cal4:
    def setData(self,num):
        self.num = radius
    def display(self):
        return num**2

num = int(input("Enter number:"))
obj = cal4()
obj.setData(num)
square = obj.display()
print(f'Sqaure of {num} is {square}.')
```

```
Enter number:5
Sqaure of 5 is 25.
```

5. Consider an employee class, which contains fields such as name and designation. And a subclass, which contains a field salary. Write a program for inheriting this relation.

In [6]:

```python
class Employee:
    name ="Nishant Movaliya"
    designation = "SDE 1"
    def display(self):
        print('Name:',self.name)
        print('Designation:',self.designation)
class SubEmployee(Employee):
    salary = 1200000
    def display(self):
        print('Name : ',self.name)
        print('Designation : ',self.designation)
        print('Salary : ',self.salary)

obj1 = Employee()
obj2 = SubEmployee()
print('-------Employee Class-------')
obj1.display()
print()
print("-------SubEmployee Class---------")
obj2.display()
```

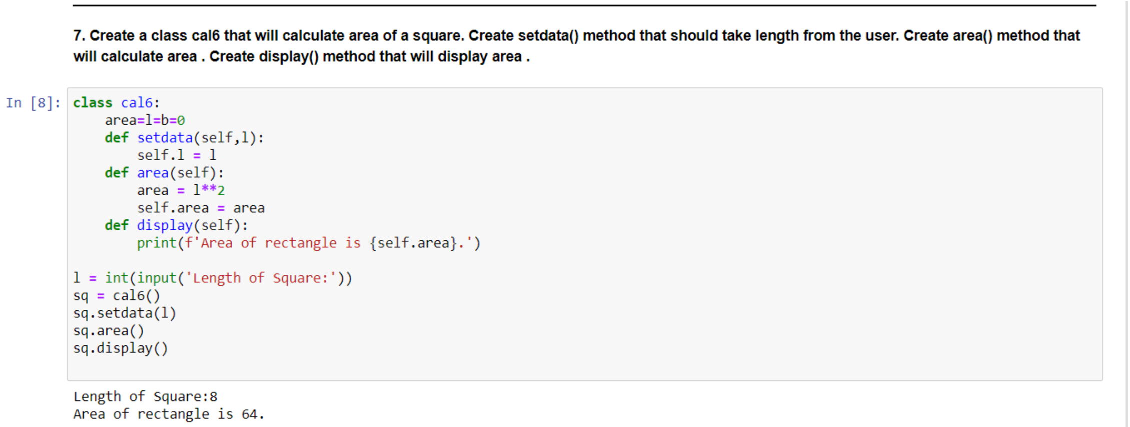
```
-------Employee Class-------
Name: Nishant Movaliya
Designation: SDE 1

-------SubEmployee Class---------
Name :  Nishant Movaliya
Designation :  SDE 1
Salary :  1200000
```

6. Create a class cal5 that will calculate area of a rectangle. Create constructor method which has two parameters .Create calArea() method that will calculate area of a rectangle. Create display() method that will display area of a rectangle.

In [7]:

```python
class cal5:
    area=l=b=0
    def __init__(self,l,b):
        self.l = l
        self.b = b
    def calArea(self):
        area = l*b
        self.area = area
    def display(self):
        print(f'Area of rectangle is {self.area}.')

l = int(input('Length:'))
b = int(input('Breadth:'))
ract = cal5(l,b)
ract.calArea()
ract.display()
```

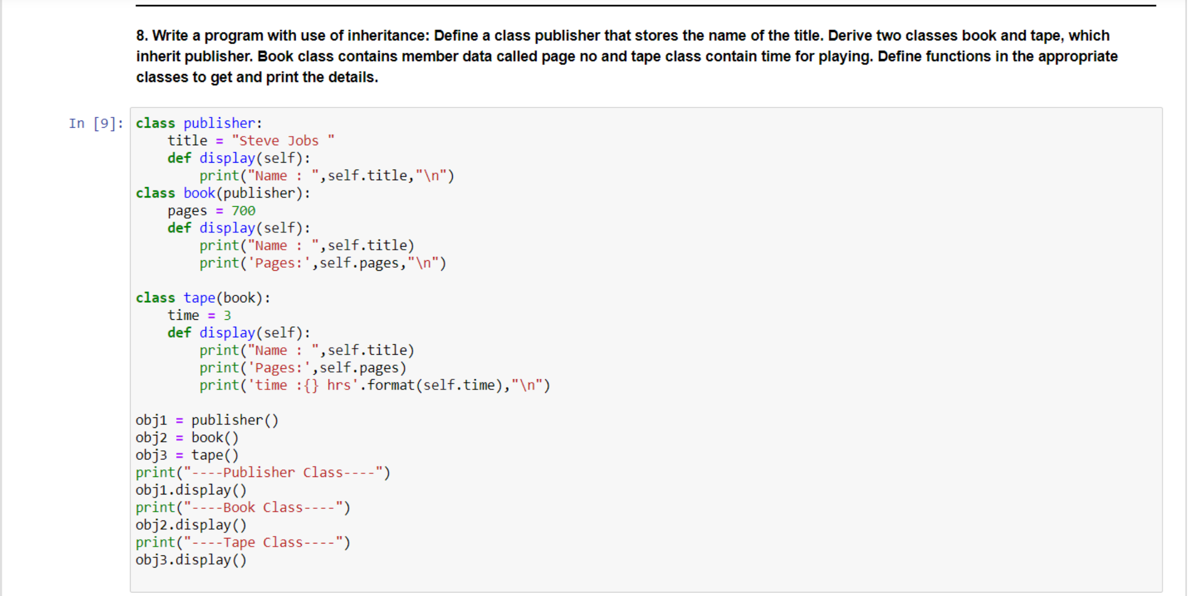
```
Length:3
Breadth:4
Area of rectangle is 12.
```



7. Create a class cal6 that will calculate area of a square. Create setdata() method that should take length from the user. Create area() method that will calculate area . Create display() method that will display area .

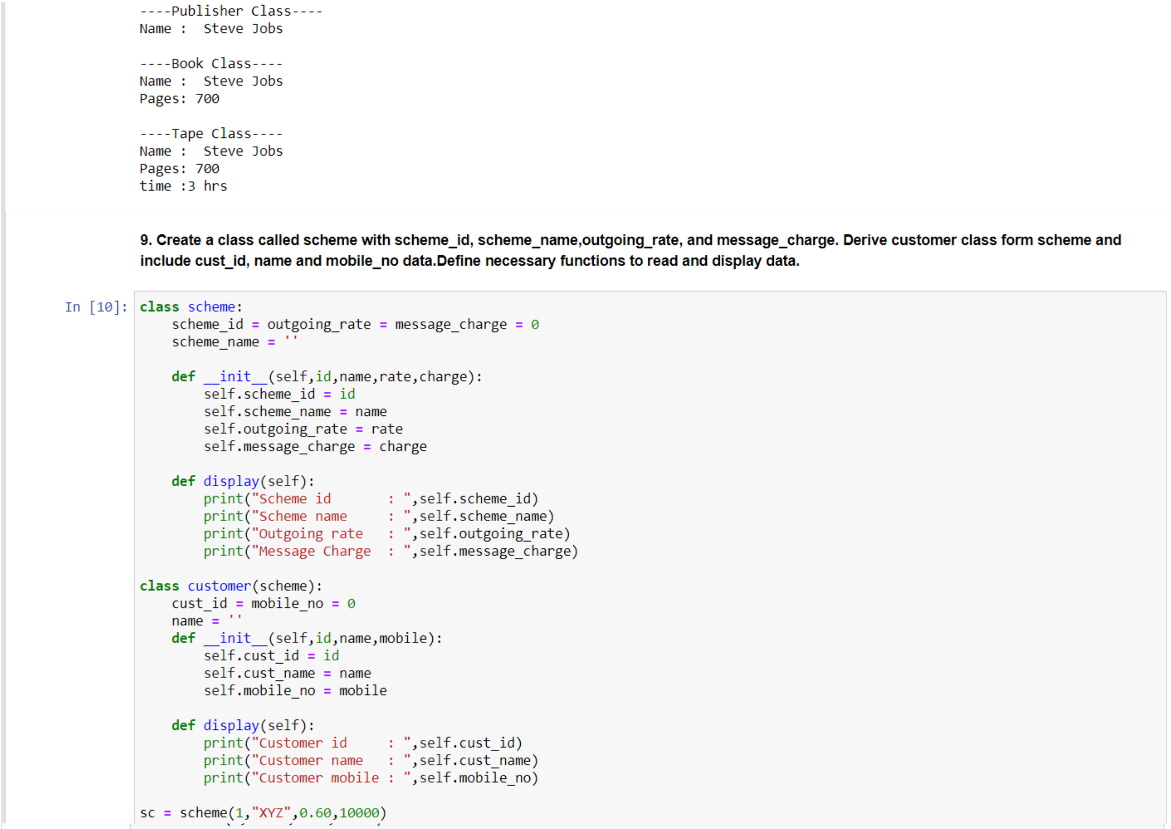
In [8]:
```
class cal6:
    area=l=b=0
    def setdata(self,l):
        self.l = l
    def area(self):
        area = l**2
        self.area = area
    def display(self):
        print(f'Area of rectangle is {self.area}.')

l = int(input('Length of Square:'))
sq = cal6()
sq.setdata(l)
sq.area()
sq.display()
```

```
Length of Square:8
Area of rectangle is 64.
```



8. Write a program with use of inheritance: Define a class publisher that stores the name of the title. Derive two classes book and tape, which inherit publisher. Book class contains member data called page no and tape class contain time for playing. Define functions in the appropriate classes to get and print the details.

In [9]:
```
class publisher:
    title = "Steve Jobs "
    def display(self):
        print("Name : ",self.title,"\n")
class book(publisher):
    pages = 700
    def display(self):
        print("Name : ",self.title)
        print('Pages:',self.pages,"\n")

class tape(book):
    time = 3
    def display(self):
        print("Name : ",self.title)
        print('Pages:',self.pages)
        print('time :{} hrs'.format(self.time),"\n")

obj1 = publisher()
obj2 = book()
obj3 = tape()
print("----Publisher Class----")
obj1.display()
print("----Book Class----")
obj2.display()
print("----Tape Class----")
obj3.display()
```

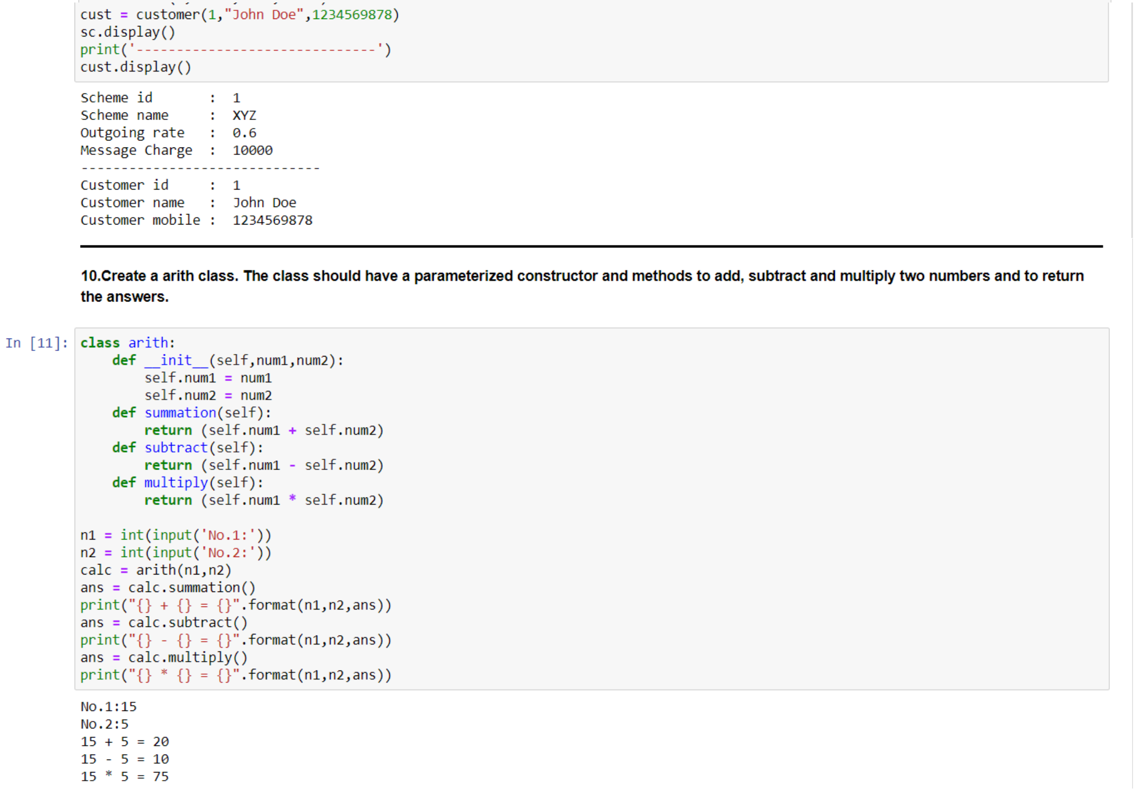


```
----Publisher Class----
Name :  Steve Jobs

----Book Class----
Name :  Steve Jobs
Pages: 700

----Tape Class----
Name :  Steve Jobs
Pages: 700
time :3 hrs
```

9. Create a class called scheme with scheme_id, scheme_name,outgoing_rate, and message_charge. Derive customer class form scheme and include cust_id, name and mobile_no data.Define necessary functions to read and display data.

In [10]:

```
class scheme:
    scheme_id = outgoing_rate = message_charge = 0
    scheme_name = ''

    def __init__(self,id,name,rate,charge):
        self.scheme_id = id
        self.scheme_name = name
        self.outgoing_rate = rate
        self.message_charge = charge

    def display(self):
        print("Scheme id       : ",self.scheme_id)
        print("Scheme name     : ",self.scheme_name)
        print("Outgoing rate   : ",self.outgoing_rate)
        print("Message Charge  : ",self.message_charge)

class customer(scheme):
    cust_id = mobile_no = 0
    name = ''
    def __init__(self,id,name,mobile):
        self.cust_id = id
        self.cust_name = name
        self.mobile_no = mobile

    def display(self):
        print("Customer id     : ",self.cust_id)
        print("Customer name   : ",self.cust_name)
        print("Customer mobile : ",self.mobile_no)

sc = scheme(1,"XYZ",0.60,10000)
```



```
cust = customer(1,"John Doe",1234569878)
sc.display()
print('------------------------------')
cust.display()
```

```
Scheme id        :  1
Scheme name      :  XYZ
Outgoing rate    :  0.6
Message Charge   :  10000
------------------------------
Customer id      :  1
Customer name    :  John Doe
Customer mobile  :  1234569878
```

10.Create a arith class. The class should have a parameterized constructor and methods to add, subtract and multiply two numbers and to return the answers.

In [11]:

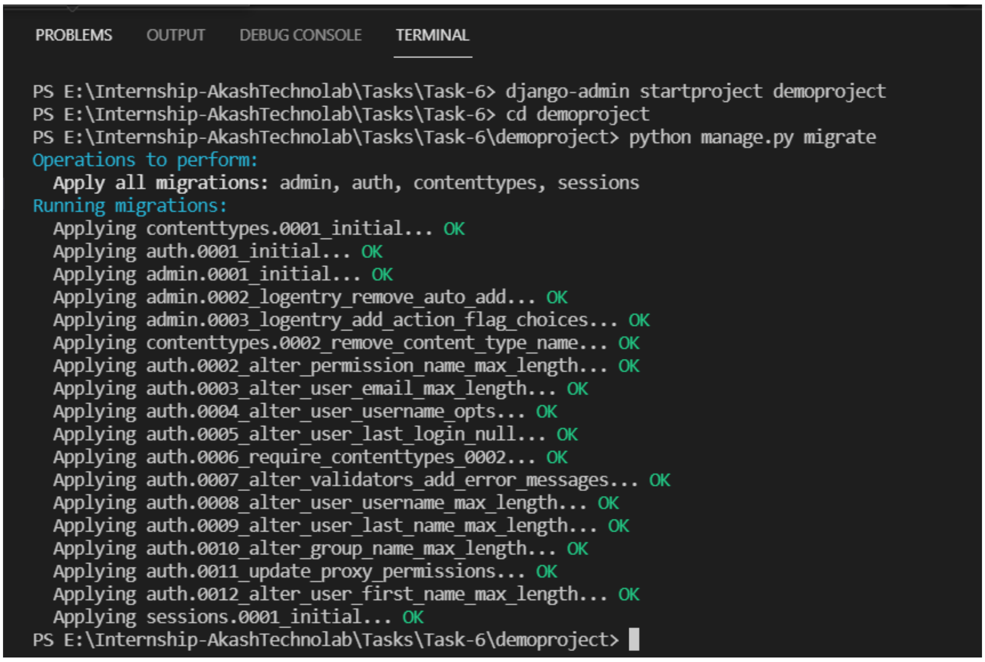
```
class arith:
    def __init__(self,num1,num2):
        self.num1 = num1
        self.num2 = num2
    def summation(self):
        return (self.num1 + self.num2)
    def subtract(self):
        return (self.num1 - self.num2)
    def multiply(self):
        return (self.num1 * self.num2)

n1 = int(input('No.1:'))
n2 = int(input('No.2:'))
calc = arith(n1,n2)
ans = calc.summation()
print("{} + {} = {}".format(n1,n2,ans))
ans = calc.subtract()
print("{} - {} = {}".format(n1,n2,ans))
ans = calc.multiply()
print("{} * {} = {}".format(n1,n2,ans))
```

```
No.1:15
No.2:5
15 + 5 = 20
15 - 5 = 10
15 * 5 = 75
```

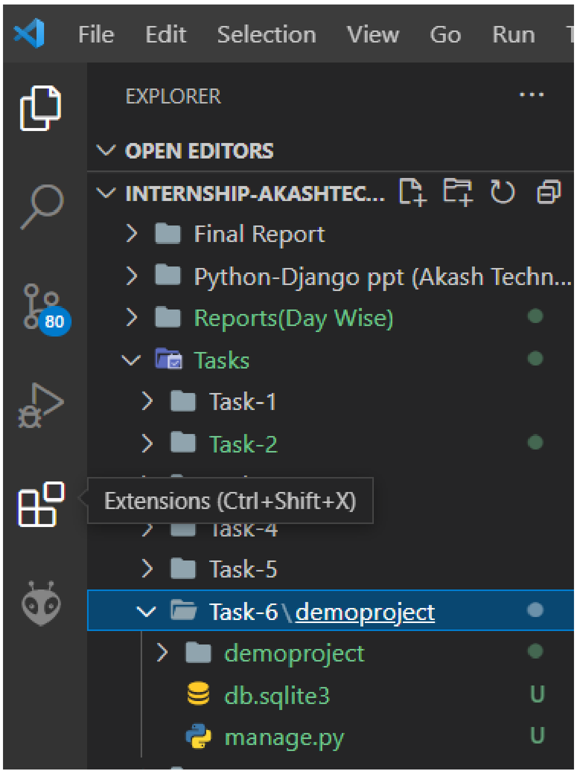
Task-6

Django Installation

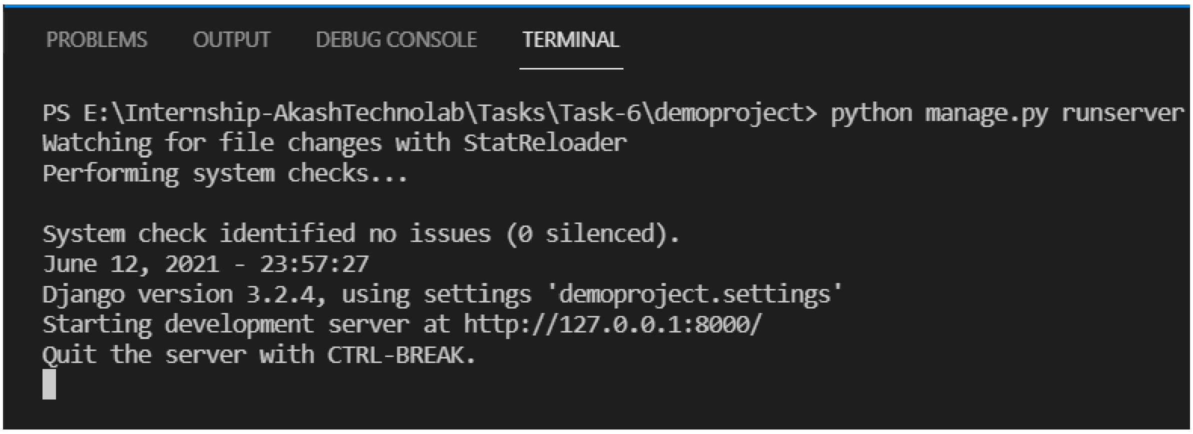
Then create new project



```
PS E:\Internship-AkashTechnolab\Tasks\Task-6> django-admin startproject demoproject
PS E:\Internship-AkashTechnolab\Tasks\Task-6> cd demoproject
PS E:\Internship-AkashTechnolab\Tasks\Task-6\demoproject> python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
PS E:\Internship-AkashTechnolab\Tasks\Task-6\demoproject> █
```

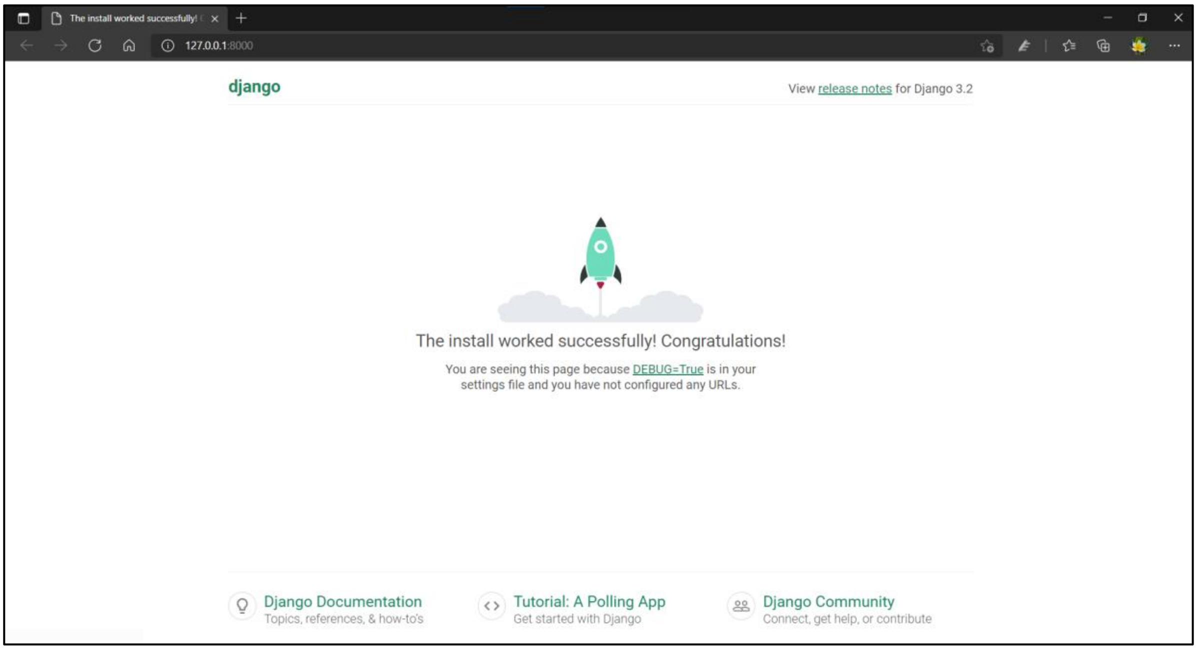


Necessary Command before running:
Python manage.py runserver



```
PS E:\Internship-AkashTechnolab\Tasks\Task-6\demoproject> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
June 12, 2021 - 23:57:27
Django version 3.2.4, using settings 'demoproject.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Browser Window:


The install worked successfully!
127.0.0.1:8000
django
View release notes for Django 3.2

The install worked successfully! Congratulations!

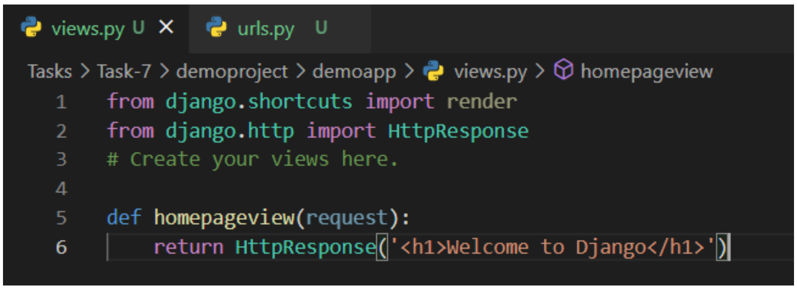
You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

Django Documentation
Topics, references, & how-to's
Tutorial: A Polling App
Get started with Django
Django Community
Connect, get help, or contribute

Task-7

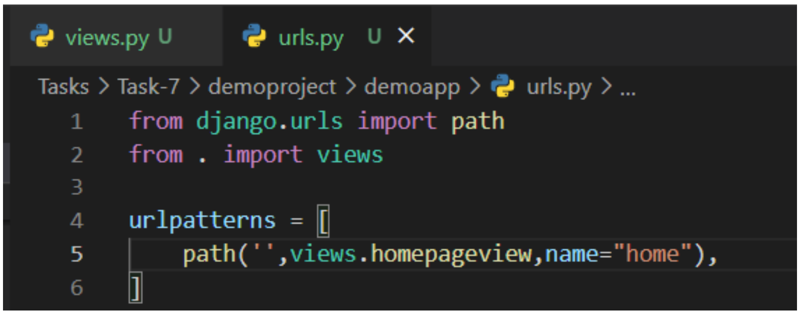
After Creating Project and Creating an App, Make the following changes: INSTALLED_APPS in settings.py of Project1:

views.py


views.py U ×
urls.py U
Tasks > Task-7 > demoproject > demoapp > views.py > homepageview

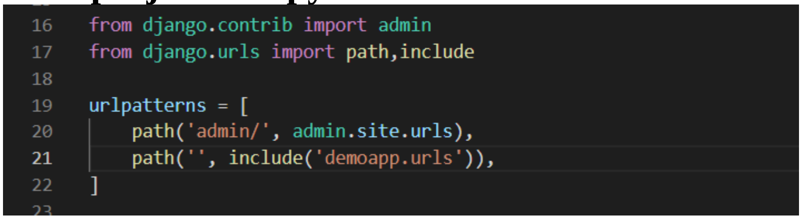
```python
from django.shortcuts import render
from django.http import HttpResponse
# Create your views here.

def homepageview(request):
    return HttpResponse('<h1>Welcome to Django</h1>')
```

urls.py



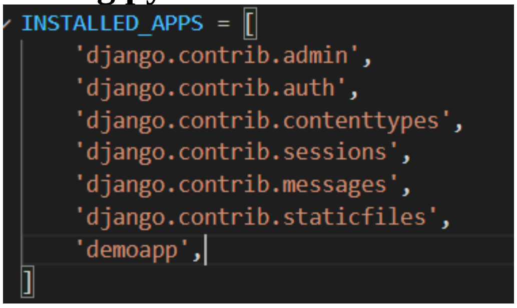
```python
from django.urls import path
from . import views

urlpatterns = [
    path('',views.homepageview,name="home"),
]
```

demoproject/urls.py



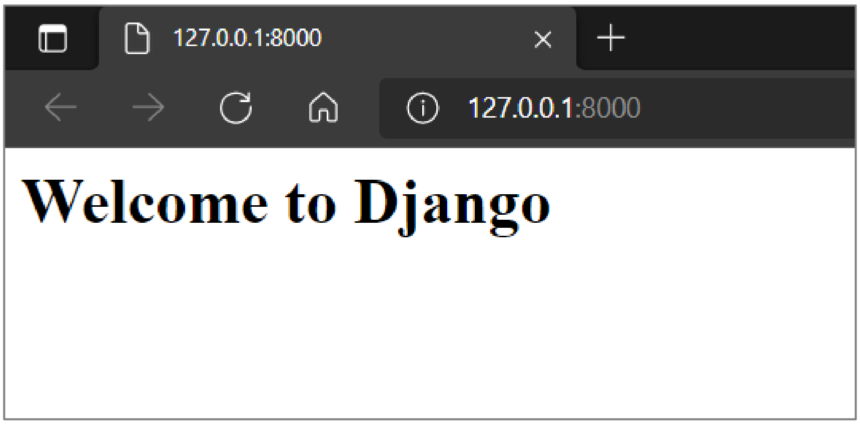
```python
from django.contrib import admin
from django.urls import path,include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('demoapp.urls')),
]
```

Setting.py



```python
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'demoapp',
]
```

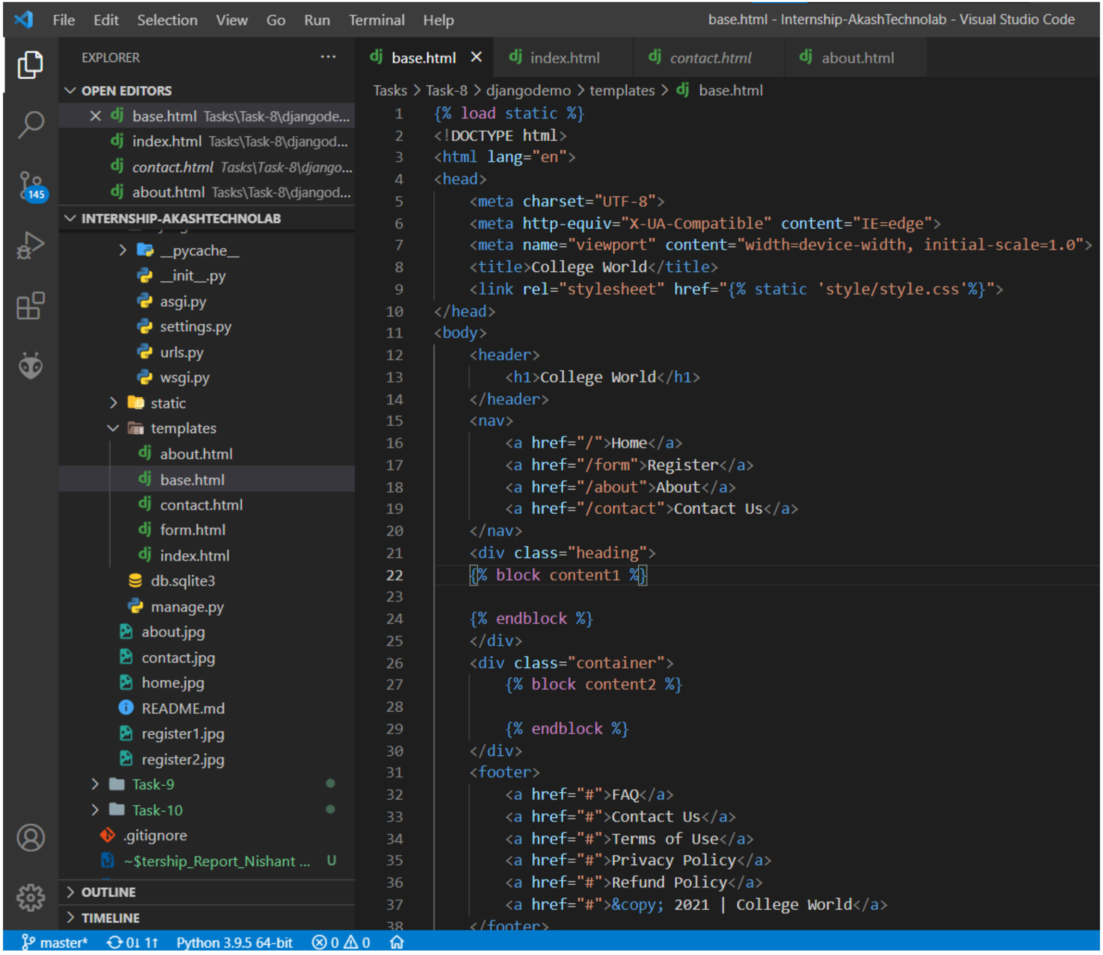
Output:



Welcome to Django

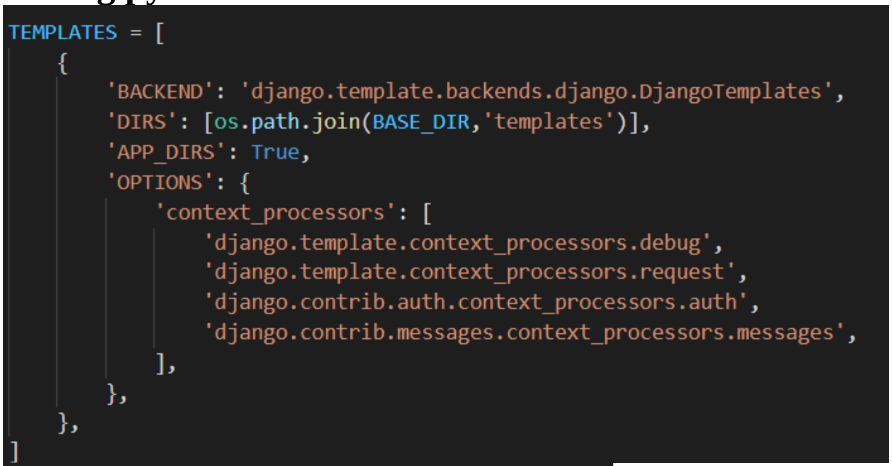
Task-8

Code File:base.html, index.html, about.html, contact.html

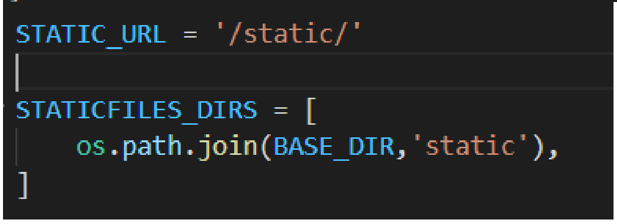

File Edit Selection View Go Run Terminal Help
base.html - Internship-AkashTechnolab - Visual Studio Code
EXPLORER
OPEN EDITORS
base.html Tasks\Task-8\djangode...
index.html Tasks\Task-8\djangod...
contact.html Tasks\Task-8\django...
about.html Tasks\Task-8\djangod...
INTERNSHIP-AKASHTECHNOLAB
__pycache__
__init__.py
asgi.py
settings.py
urls.py
wsgi.py
static
templates
about.html
base.html
contact.html
form.html
index.html
db.sqlite3
manage.py
about.jpg
contact.jpg
home.jpg
README.md
register1.jpg
register2.jpg
Task-9
Task-10
.gitignore
~$tership_Report_Nishant ... U
OUTLINE
TIMELINE
base.html index.html contact.html about.html
Tasks > Task-8 > djangodemo > templates > base.html
master* 0↓ 1↑ Python 3.9.5 64-bit ⊗ 0 ⚠ 0

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>College World</title>
    <link rel="stylesheet" href="{% static 'style/style.css'%}">
</head>
<body>
    <header>
        <h1>College World</h1>
    </header>
    <nav>
        <a href="/">Home</a>
        <a href="/form">Register</a>
        <a href="/about">About</a>
        <a href="/contact">Contact Us</a>
    </nav>
    <div class="heading">
    {% block content1 %}

    {% endblock %}
    </div>
    <div class="container">
        {% block content2 %}

        {% endblock %}
    </div>
    <footer>
        <a href="#">FAQ</a>
        <a href="#">Contact Us</a>
        <a href="#">Terms of Use</a>
        <a href="#">Privacy Policy</a>
        <a href="#">Refund Policy</a>
        <a href="#">&copy; 2021 | College World</a>
    </footer>
```

Setting.py

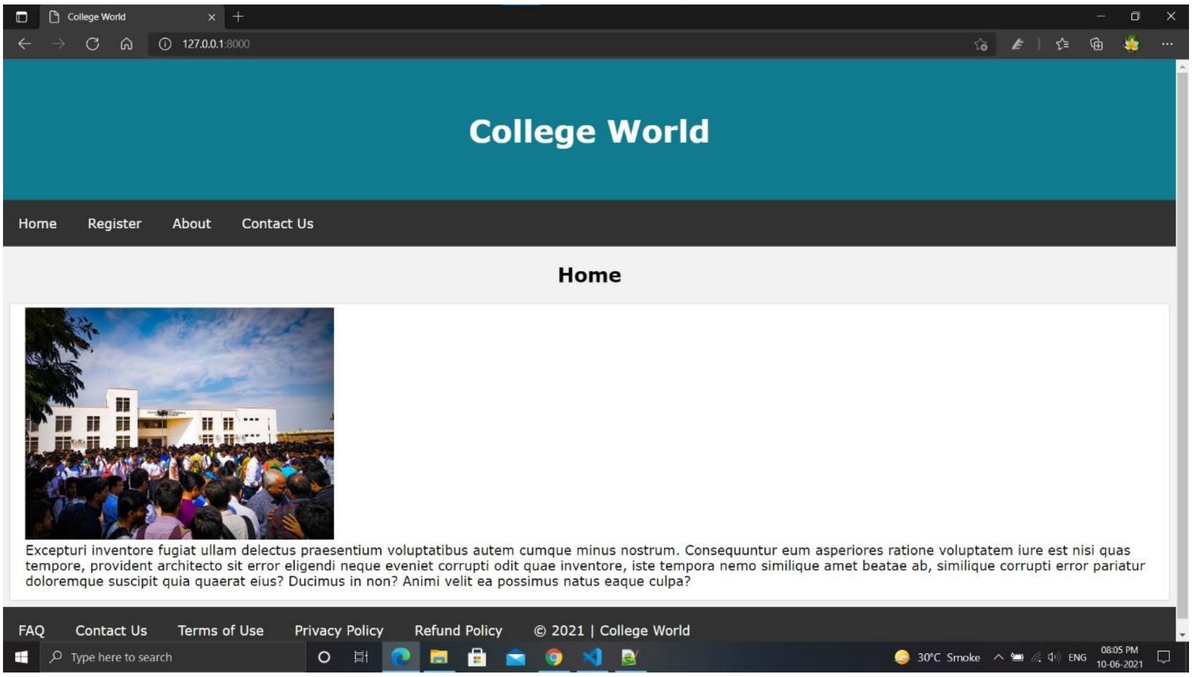


```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [os.path.join(BASE_DIR,'templates')],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]
```



```
STATIC_URL = '/static/'

STATICFILES_DIRS = [
    os.path.join(BASE_DIR,'static'),
]
```

index.html



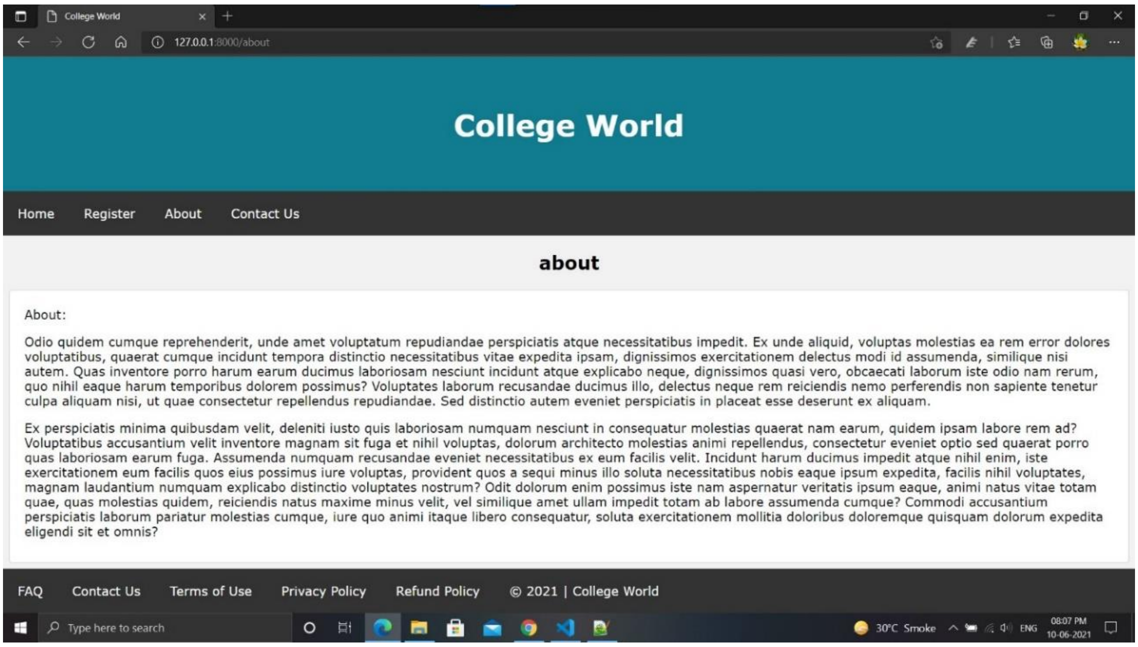
College World

Home

Excepturi inventore fugiat ullam delectus praesentium voluptatibus autem cumque minus nostrum. Consequuntur eum asperiores ratione voluptatem iure est nisi quas tempore, provident architecto sit error eligendi neque eveniet corrupti odit quae inventore, iste tempora nemo similique amet beatae ab, similique corrupti error pariatur doloremque suscipit quia quaerat eius? Ducimus in non? Animi velit ea possimus natus eaque culpa?

© 2021 | College World

about.html



College World

about

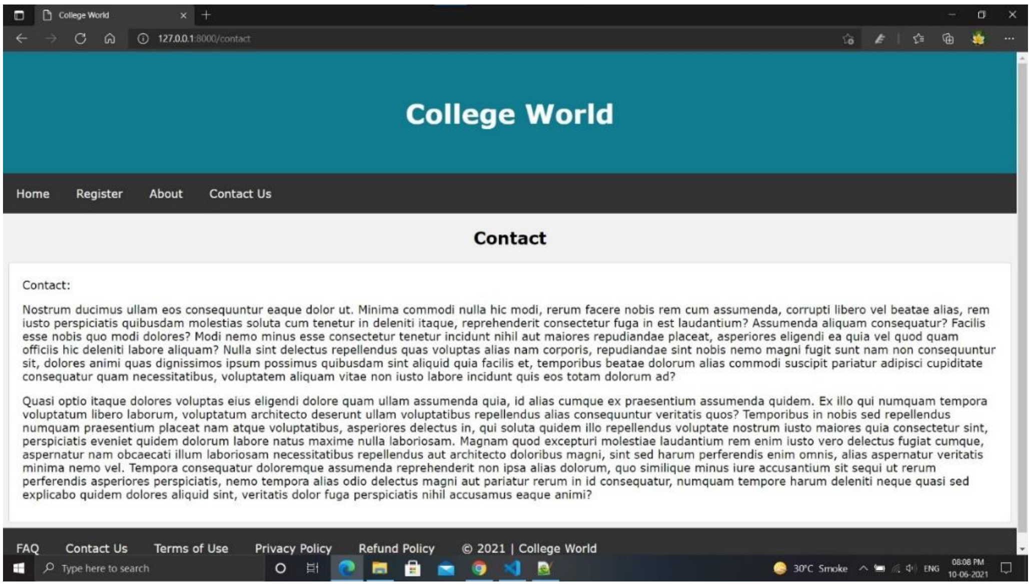
About:

Odio quidem cumque reprehenderit, unde amet voluptatum repudiandae perspiciatis atque necessitatibus impedit. Ex unde aliquid, voluptas molestias ea rem error dolores voluptatibus, quaerat cumque incidunt tempora distinctio necessitatibus vitae expedita ipsam, dignissimos exercitationem delectus modi id assumenda, similique nisi autem. Quas inventore porro harum earum ducimus laboriosam nesciunt incidunt atque explicabo neque, dignissimos quasi vero, obcaecati laborum iste odio nam rerum, quo nihil eaque harum temporibus dolorem possimus? Voluptates laborum recusandae ducimus illo, delectus neque rem reiciendis nemo perferendis non sapiente tenetur culpa aliquam nisi, ut quae consectetur repellendus repudiandae. Sed distinctio autem eveniet perspiciatis in placeat esse deserunt ex aliquam.

Ex perspiciatis minima quibusdam velit, deleniti iusto quis laboriosam numquam nesciunt in consequatur molestias quaerat nam earum, quidem ipsam labore rem ad? Voluptatibus accusantium velit inventore magnam sit fuga et nihil voluptas, dolorum architecto molestias animi repellendus, consectetur eveniet optio sed quaerat porro quas laboriosam earum fuga. Assumenda numquam recusandae eveniet necessitatibus ex eum facilis velit. Incidunt harum ducimus impedit atque nihil enim, iste exercitationem eum facilis quos eius possimus iure voluptas, provident quos a sequi minus illo soluta necessitatibus nobis eaque ipsum expedita, facilis nihil voluptates, magnam laudantium numquam explicabo distinctio voluptates nostrum? Odit dolorum enim possimus iste nam aspernatur veritatis ipsum eaque, animi natus vitae totam quae, quas molestias quidem, reiciendis natus maxime minus velit, vel similique amet ullam impedit totam ab labore assumenda cumque? Commodi accusantium perspiciatis laborum pariatur molestias cumque, iure quo animi itaque libero consequatur, soluta exercitationem mollitia doloribus doloremque quisquam dolorum expedita eligendi sit et omnis?

© 2021 | College World

contact.html



College World

Contact

Contact:

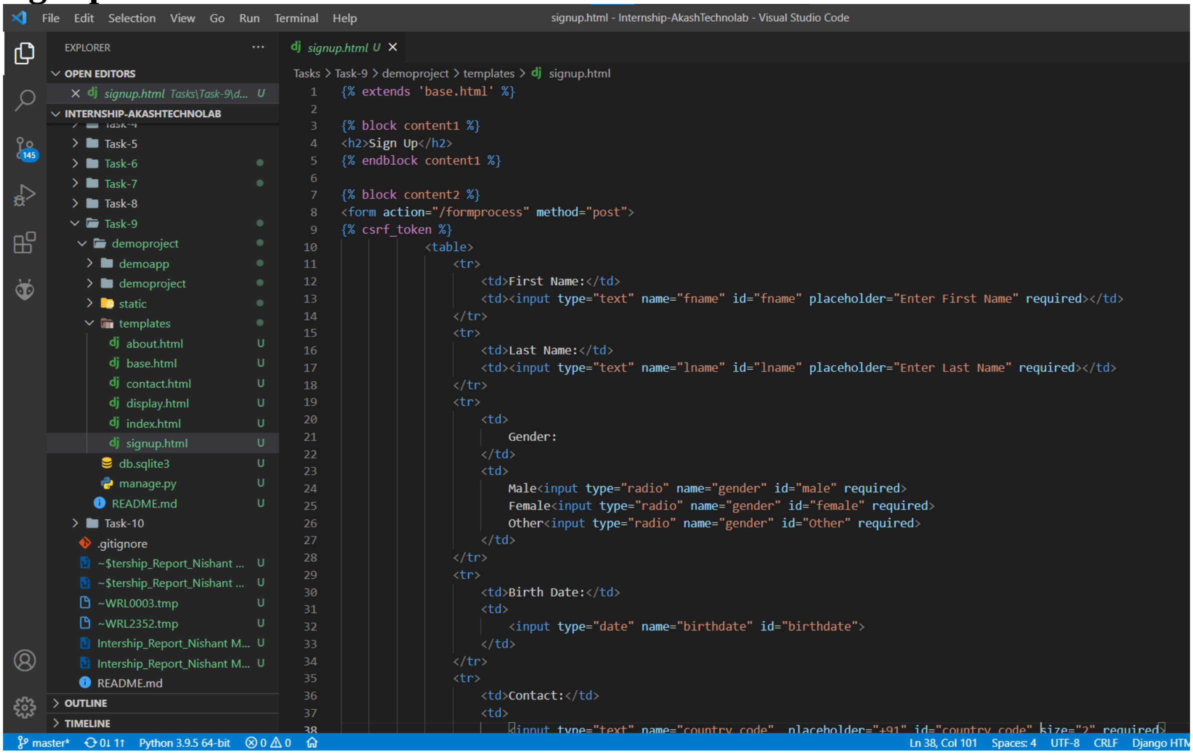
Nostrum ducimus ullam eos consequuntur eaque dolor ut. Minima commodi nulla hic modi, rerum facere nobis rem cum assumenda, corrupti libero vel beatae alias, rem iusto perspiciatis quibusdam molestias soluta cum tenetur in deleniti itaque, reprehenderit consectetur fuga in est laudantium? Assumenda aliquam consequatur? Facilis esse nobis quo modi dolores? Modi nemo minus esse consectetur tenetur incidunt nihil aut maiores repudiandae placeat, asperiores eligendi ea quia vel quod quam officiis hic deleniti labore aliquam? Nulla sint delectus repellendus quas voluptas alias nam corporis, repudiandae sint nobis nemo magni fugit sunt nam non consequuntur sit, dolores animi quas dignissimos ipsum possimus quibusdam sint aliquid quia facilis et, temporibus beatae dolorum alias commodi suscipit pariatur adipisci cupiditate consequatur quam necessitatibus, voluptatem aliquam vitae non iusto labore incidunt quis eos totam dolorum ad?

Quasi optio itaque dolores voluptas eius eligendi dolore quam ullam assumenda quia, id alias cumque ex praesentium assumenda quidem. Ex illo qui numquam tempora voluptatum libero laborum, voluptatum architecto deserunt ullam voluptatibus repellendus alias consequuntur veritatis quos? Temporibus in nobis sed repellendus numquam praesentium placeat nam atque voluptatibus, asperiores delectus in, qui soluta quidem illo repellendus voluptate nostrum iusto maiores quia consectetur sint, perspiciatis eveniet quidem dolorum labore natus maxime nulla laboriosam. Magnam quod excepturi molestiae laudantium rem enim iusto vero delectus fugiat cumque, aspernatur nam obcaecati illum laboriosam necessitatibus repellendus aut architecto doloribus magni, sint sed harum perferendis enim omnis, alias aspernatur veritatis minima nemo vel. Tempora consequatur doloremque assumenda reprehenderit non ipsa alias dolorum, quo similique minus iure accusantium sit sequi ut rerum perferendis asperiores perspiciatis, nemo tempora alias odio delectus magni aut pariatur rerum in id consequatur, numquam tempore harum deleniti neque quasi sed explicabo quidem dolores aliquid sint, veritatis dolor fuga perspiciatis nihil accusamus eaque animi?

Task-9

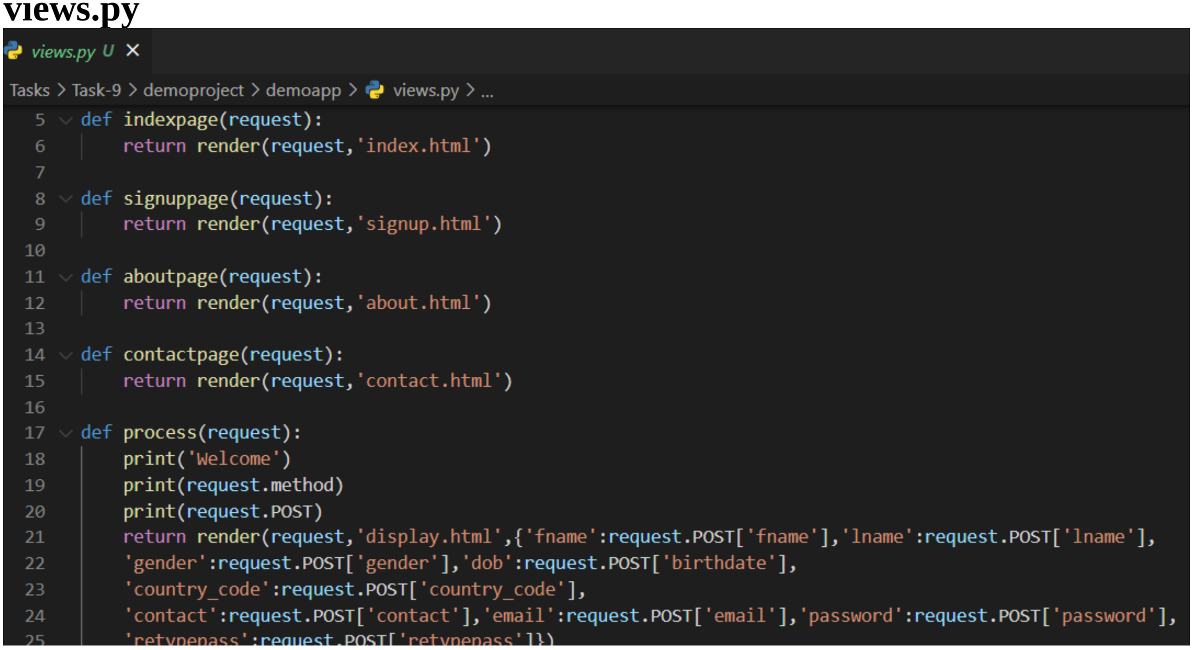
Task: Using POST method to get User Values and returning back on page. Here I'm going to extend the Project I did on Day 8

signup.html



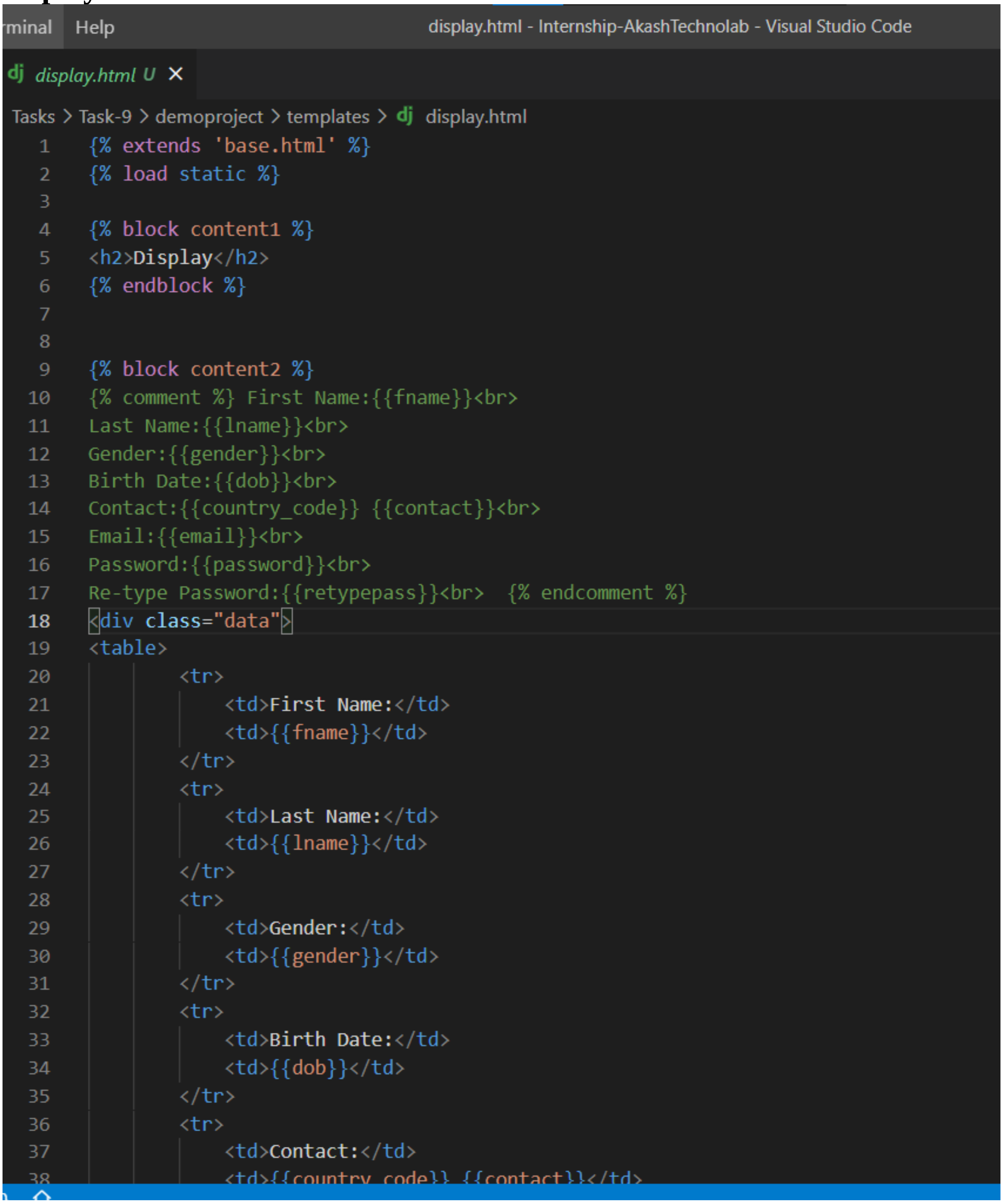
```
{% extends 'base.html' %}

{% block content1 %}
<h2>Sign Up</h2>
{% endblock content1 %}

{% block content2 %}
<form action="/formprocess" method="post">
{% csrf_token %}
            <table>
                <tr>
                    <td>First Name:</td>
                    <td><input type="text" name="fname" id="fname" placeholder="Enter First Name" required></td>
                </tr>
                <tr>
                    <td>Last Name:</td>
                    <td><input type="text" name="lname" id="lname" placeholder="Enter Last Name" required></td>
                </tr>
                <tr>
                    <td>
                        Gender:
                    </td>
                    <td>
                        Male<input type="radio" name="gender" id="male" required>
                        Female<input type="radio" name="gender" id="female" required>
                        Other<input type="radio" name="gender" id="Other" required>
                    </td>
                </tr>
                <tr>
                    <td>Birth Date:</td>
                    <td>
                        <input type="date" name="birthdate" id="birthdate">
                    </td>
                </tr>
                <tr>
                    <td>Contact:</td>
                    <td>
                        <input type="text" name="country_code"  placeholder="+91" id="country_code" size="2" required>
```

views.py



```python
def indexpage(request):
    return render(request,'index.html')

def signuppage(request):
    return render(request,'signup.html')

def aboutpage(request):
    return render(request,'about.html')

def contactpage(request):
    return render(request,'contact.html')

def process(request):
    print('Welcome')
    print(request.method)
    print(request.POST)
    return render(request,'display.html',{'fname':request.POST['fname'],'lname':request.POST['lname'],
    'gender':request.POST['gender'],'dob':request.POST['birthdate'],
    'country_code':request.POST['country_code'],
    'contact':request.POST['contact'],'email':request.POST['email'],'password':request.POST['password'],
    'retypepass':request.POST['retypepass']})
```

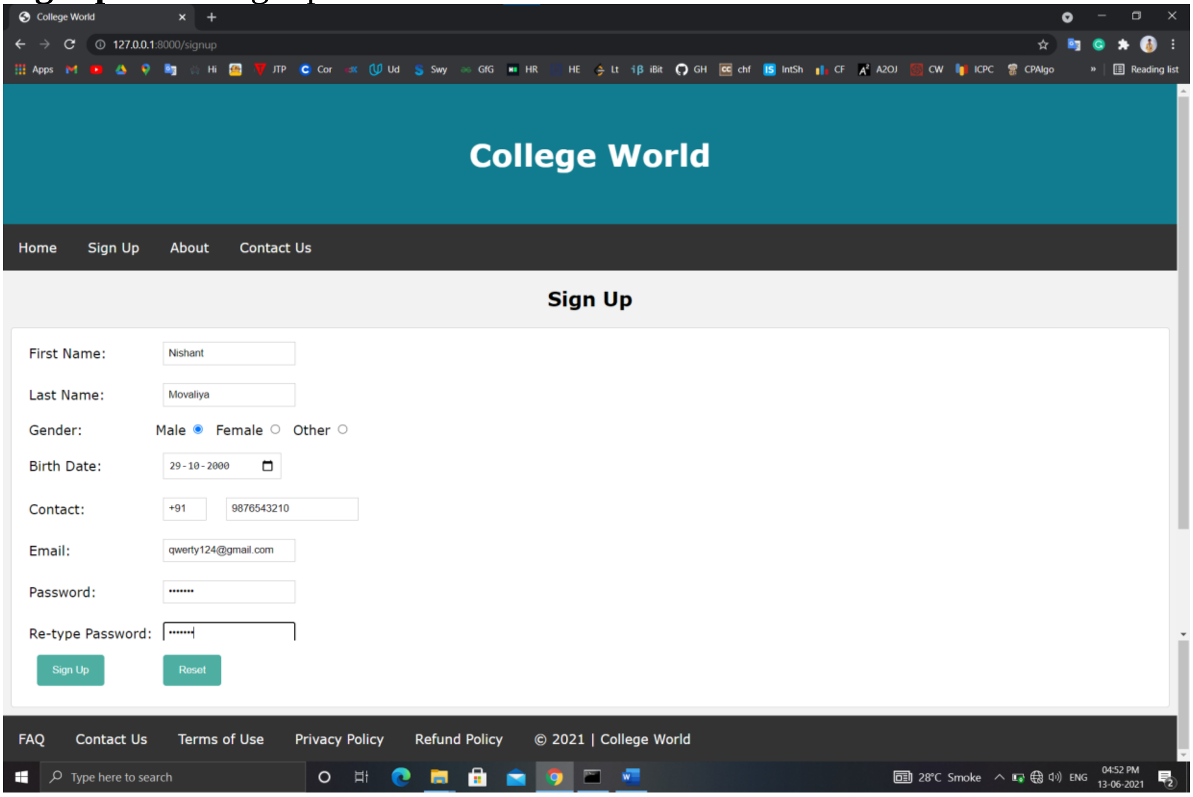
display.html


rminal
Help
display.html - Internship-AkashTechnolab - Visual Studio Code
display.html U ×
Tasks > Task-9 > demoproject > templates > display.html

```
{% extends 'base.html' %}
{% load static %}

{% block content1 %}
<h2>Display</h2>
{% endblock %}


{% block content2 %}
{% comment %} First Name:{{fname}}<br>
Last Name:{{lname}}<br>
Gender:{{gender}}<br>
Birth Date:{{dob}}<br>
Contact:{{country_code}} {{contact}}<br>
Email:{{email}}<br>
Password:{{password}}<br>
Re-type Password:{{retypepass}}<br>  {% endcomment %}
<div class="data">
<table>
        <tr>
            <td>First Name:</td>
            <td>{{fname}}</td>
        </tr>
        <tr>
            <td>Last Name:</td>
            <td>{{lname}}</td>
        </tr>
        <tr>
            <td>Gender:</td>
            <td>{{gender}}</td>
        </tr>
        <tr>
            <td>Birth Date:</td>
            <td>{{dob}}</td>
        </tr>
        <tr>
            <td>Contact:</td>
            <td>{{country_code}} {{contact}}</td>
```

Output:

Signup form: signup.html



College World

Home Sign Up About Contact Us

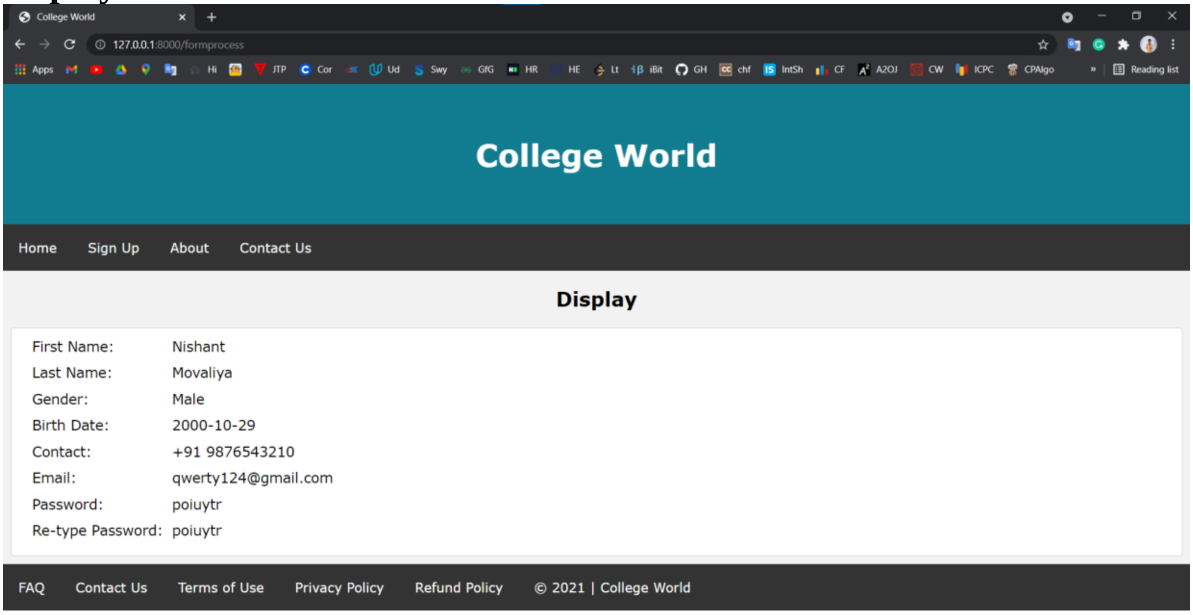
Sign Up

First Name:	Nishant
Last Name:	Movaliya
Gender:	Male ● Female ○ Other ○
Birth Date:	29-10-2000
Contact:	+91 9876543210
Email:	qwerty124@gmail.com
Password:	•••••••
Re-type Password:	•••••••

Sign Up Reset

FAQ Contact Us Terms of Use Privacy Policy Refund Policy © 2021 | College World

display.html



College World

Home Sign Up About Contact Us

Display

First Name:	Nishant
Last Name:	Movaliya
Gender:	Male
Birth Date:	2000-10-29
Contact:	+91 9876543210
Email:	qwerty124@gmail.com
Password:	poiuytr
Re-type Password:	poiuytr

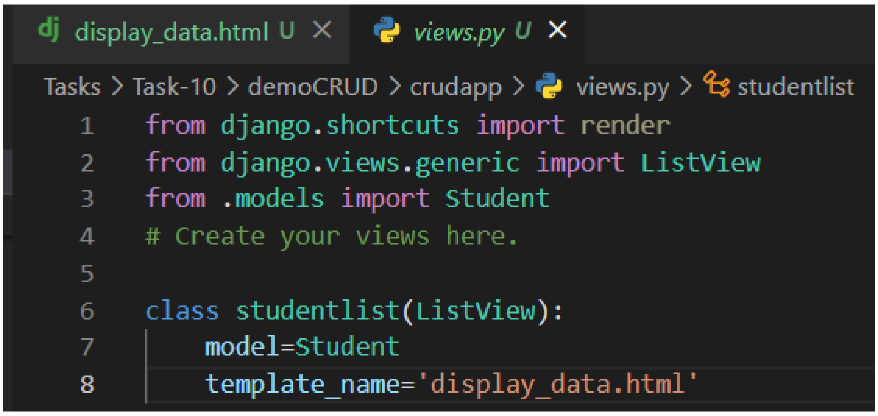
FAQ Contact Us Terms of Use Privacy Policy Refund Policy © 2021 | College World

Task-10

Task: Implementing Models and Fetching Values using Function Based Views and Class Based Views

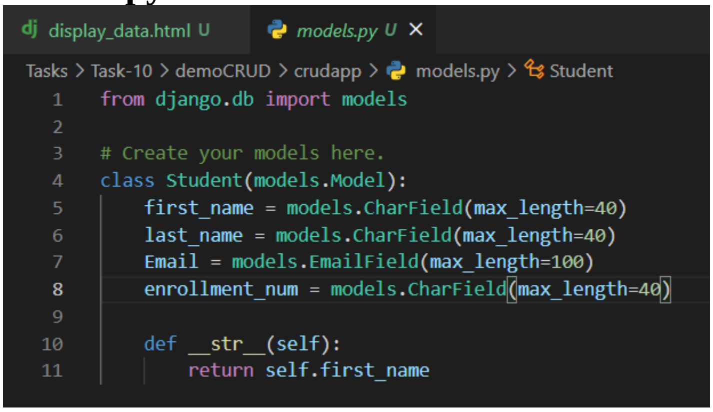
views.py

class based views:


dj display_data.html U ×
views.py U ×
Tasks > Task-10 > demoCRUD > crudapp > views.py > studentlist

```python
from django.shortcuts import render
from django.views.generic import ListView
from .models import Student
# Create your views here.

class studentlist(ListView):
    model=Student
    template_name='display_data.html'
```

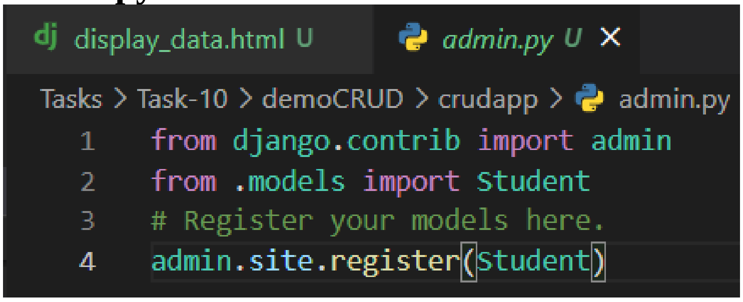
models.py



```python
from django.db import models

# Create your models here.
class Student(models.Model):
    first_name = models.CharField(max_length=40)
    last_name = models.CharField(max_length=40)
    Email = models.EmailField(max_length=100)
    enrollment_num = models.CharField(max_length=40)

    def __str__(self):
        return self.first_name
```

Registering Models in admin:

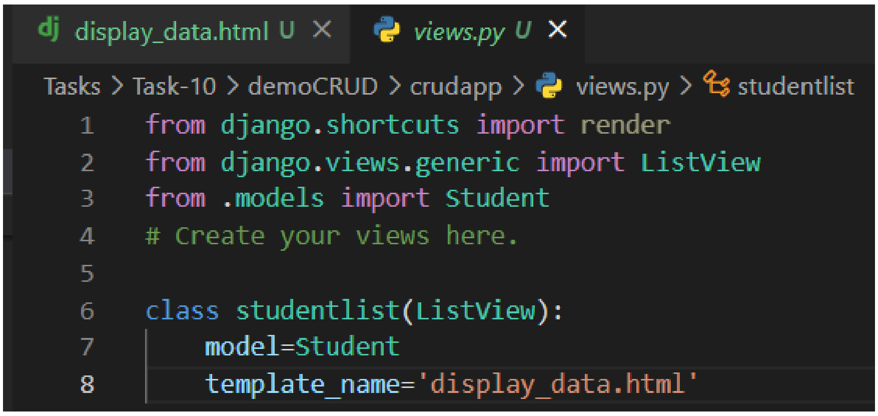
admin.py



```python
from django.contrib import admin
from .models import Student
# Register your models here.
admin.site.register(Student)
```

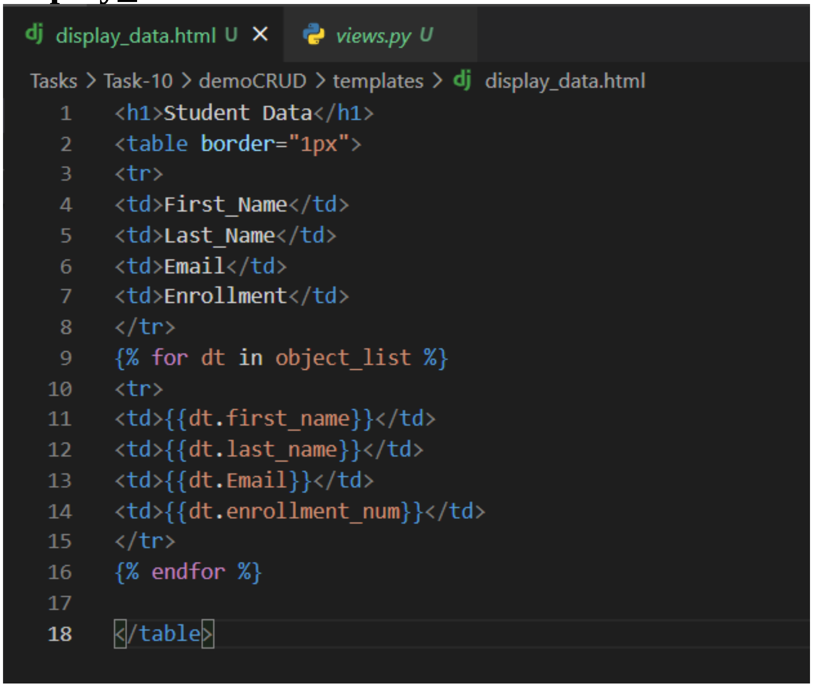
views.py

class based views:


display_data.html U ×
views.py U ×
Tasks > Task-10 > demoCRUD > crudapp > views.py > studentlist

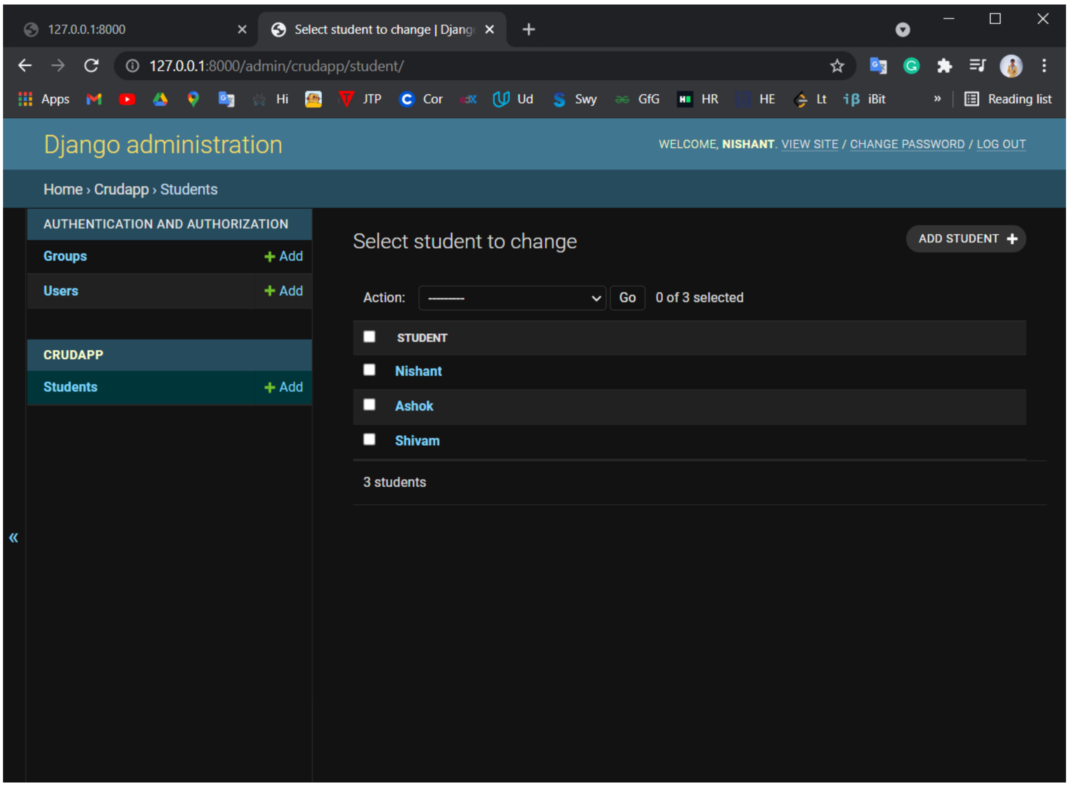
```python
from django.shortcuts import render
from django.views.generic import ListView
from .models import Student
# Create your views here.

class studentlist(ListView):
    model=Student
    template_name='display_data.html'
```

display_data.html



```html
<h1>Student Data</h1>
<table border="1px">
<tr>
<td>First_Name</td>
<td>Last_Name</td>
<td>Email</td>
<td>Enrollment</td>
</tr>
{% for dt in object_list %}
<tr>
<td>{{dt.first_name}}</td>
<td>{{dt.last_name}}</td>
<td>{{dt.Email}}</td>
<td>{{dt.enrollment_num}}</td>
</tr>
{% endfor %}

</table>
```

Output:


127.0.0.1:8000
Select student to change | Django
127.0.0.1:8000/admin/crudapp/student/
Apps Hi JTP Cor Ud Swy GfG HR HE Lt iBit Reading list

Django administration

WELCOME, NISHANT. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Crudapp › Students

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

CRUDAPP

Students + Add

Select student to change

ADD STUDENT +

Action: --------- Go 0 of 3 selected

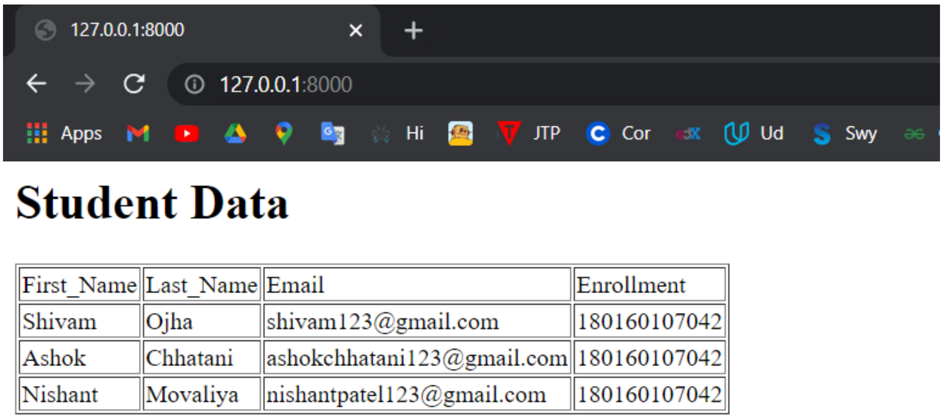
□ STUDENT

□ Nishant

□ Ashok

□ Shivam

3 students



Student Data

First_Name	Last_Name	Email	Enrollment
Shivam	Ojha	shivam123@gmail.com	180160107042
Ashok	Chhatani	ashokchhatani123@gmail.com	180160107042
Nishant	Movaliya	nishantpatel123@gmail.com	180160107042