PYTHON-DJANGO INTERNSHIP REPORT

Personal Details

Name	:	BHAYALA ASHIFA M.
College Name	:	Government Engineering College, Modasa
Enrollment Number	:	180160116006
Degree	:	BE
Semester	:	7
Github URL	:	https://github.com/Ashifa7/Ashifa-Internship

Company Details

Company Name	:	Akash Technolabs
External Guide	:	Mr. Akash Padhiyar
Training Duration	:	25-05-2021 to 14-06-2021

Index

TASK FOR DAY-1

Aim: Create a student registration form using html and css:

Code:

```
<html>

<head>

<center><h1 style="border:Light SkyBlue; border-witdth:5px;">STUDENT REGISTRATION FORM</h1></center>

<style>



table{

 font-family: verdana;

 color:Black;

 font-size: 16px;

 font-style: normal;

 font-weight: bold;

 background: lightpink;

 border-collapse: seperate;

 border: 4px normal #ADD8E6;

 border-style: dotted;
}


table.inner{
 border: 10px
}


input[type=text],
input[type=email],
input[type=number]
{
 width: 50%;
 padding: 6px 12px;
 margin: 5px 0;
 box-sizing: border-box;
}
input[type=submit],
input[type=reset]{
 width: 15%;
 padding: 8px 12px;
 margin: 5px 0;
 box-sizing: border-box;
}
</style>
</head>
```

```
<body>

<table
align="center"
cellpadding = "10">



<tr>

<td>First
Name</td>

<td><input
type="text"
name="FirstName"
maxlength="50"/>

</td>

</tr>



<tr>

<td>Middle
Name</td>

<td><input
type="text"
name="MiddleName
" maxlength="50"/>

</td>

</tr>



<tr>

<td>Last Name</td>

<td><input
type="text"
```

```
name="LastName"
maxlength="50"/>

</td>

</tr>



<tr>

<td>Email ID</td>

<td><input
type="email"
name="EmailID"
maxlength="50"/></
td>

</tr>



<tr>

<td>Mobile
number</td>

<td>

<input type="text"
name="MobileNum
ber"
maxlength="10"/>

(10 Digits Only)

</td>

</tr>



<tr>

<td>Gender</td>
```

```
<td>

<input type="radio" name="Gender" value="Male" />

Male

<input type="radio" name="Gender" value="Female" />

Female

<input type="radio" name="Gender" value="Other" />

Other

</td>

</tr>



<tr>

<td>Date of Birth</td>

<td>

<select name="Day" id="Day">

<option value="-1">Day:</option>

<option value="1">1</option>

<option value="2">2</option>
```

```
<option
value="3">3</optio
n>

<option
value="4">4</optio
n>

<option
value="5">5</optio
n>

<option
value="6">6</optio
n>

<option
value="7">7</optio
n>

<option
value="8">8</optio
n>

<option
value="9">9</optio
n>

<option
value="10">10</opt
ion>

<option
value="11">11</opt
ion>

<option
value="12">12</opt
ion>

<option
value="13">13</opt
ion>
```

```
<option value="14">14</option>

<option value="15">15</option>

<option value="16">16</option>

<option value="17">17</option>

<option value="18">18</option>

<option value="19">19</option>

<option value="20">20</option>

<option value="21">21</option>

<option value="22">22</option>

<option value="23">23</option>

<option value="24">24</option>
```

```
<option
value="25">25</opt
ion>

<option
value="26">26</opt
ion>

<option
value="27">27</opt
ion>

<option
value="28">28</opt
ion>

<option
value="29">29</opt
ion>

<option
value="30">30</opt
ion>

<option
value="31">31</opt
ion>

</select>

<select
name="Month"
id="Month">

<option value="-
1">Month:</option>

<option
value="January">Ja
n</option>

<option
value="February">F
eb</option>
```

```
<option
value="March">Mar
</option>

<option
value="April">Apr<
/option>

<option
value="May">May<
/option>

<option
value="June">Jun</
option>

<option
value="July">Jul</o
ption>

<option
value="August">Au
g</option>

<option
value="September">
Sep</option>

<option
value="October">O
ct</option>

<option
value="November">
Nov</option>

<option
value="December">
Dec</option>

</select>

<select
name="Year"
```

```
id="Year">

<option value="-
1">Year:</option>

<option
value="2016">2016
</option>

<option
value="2015">2015
</option>

<option
value="2014">2014
</option>

<option
value="2013">2013
</option>

<option
value="2012">2012
</option>

<option
value="2011">2011
</option>

<option
value="2010">2010
</option>

<option
value="2009">2009
</option>

<option
value="2008">2008
</option>

<option
value="2007">2007
</option>
```

```
<option
value="2006">2006
</option>

<option
value="2005">2005
</option>

<option
value="2004">2004
</option>

<option
value="2003">2003
</option>

<option
value="2002">2002
</option>

<option
value="2001">2001
</option>

<option
value="2000">2000
</option>

<option
value="1999">1999
</option>

<option
value="1998">1998
</option>

<option
value="1997">1997
</option>

<option
value="1996">1996
</option>
```

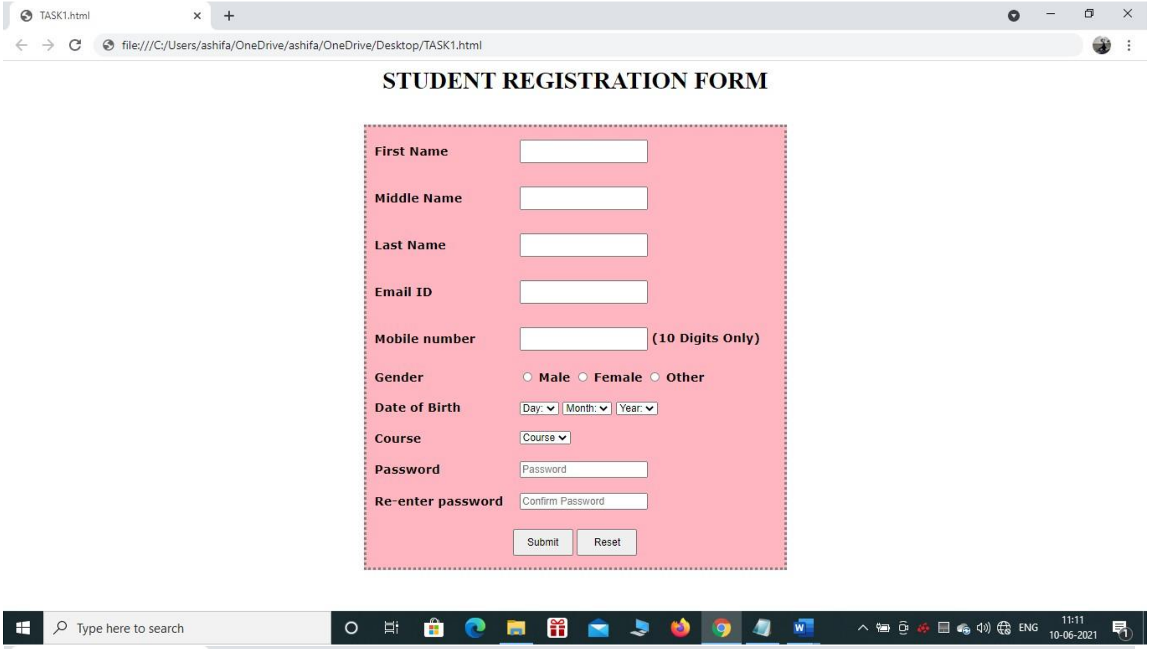
```
<option
value="1995">1995
</option>

<option
value="1994">1994
</option>

<option
value="1993">1993
</option>

<option
value="1992">1992
</option>

<option
value="1991">1991
</option>

<option
value="1990">1990
</option>

</select>

</td>

</tr>



<tr>

<td>Course<br
/></td>

<td>

<select>

<option
value="Course">Co
urse</option>
```

```
<option
value="BE">BE</o
ption>

<option
value="ME">ME</o
ption>

<option
value="B.Tech">B.
Tech</option>

<option
value="M.Tech">M.
Tech</option>

<option
value="MBA">MB
A</option>

<option
value="MCA">MC
A</option>

</select>

</td>

</tr>

<br>



<tr>

<td> Password </td>

<td>

<input
type="password"
placeholder="Passw
ord" id="password"
required>
```

```
</td>
</tr>
<tr>
<td> Re-enter password </td>
<td>
<input type="password" placeholder="Confirm Password" id="re-enter password" required>
</td>
</tr>

<tr>
<td colspan="2" align="center">
<input type="submit" value="Submit">
<input type="reset" value="Reset">
</td>
</tr>
</table>
</form>
</body>
```

```
</html>
```

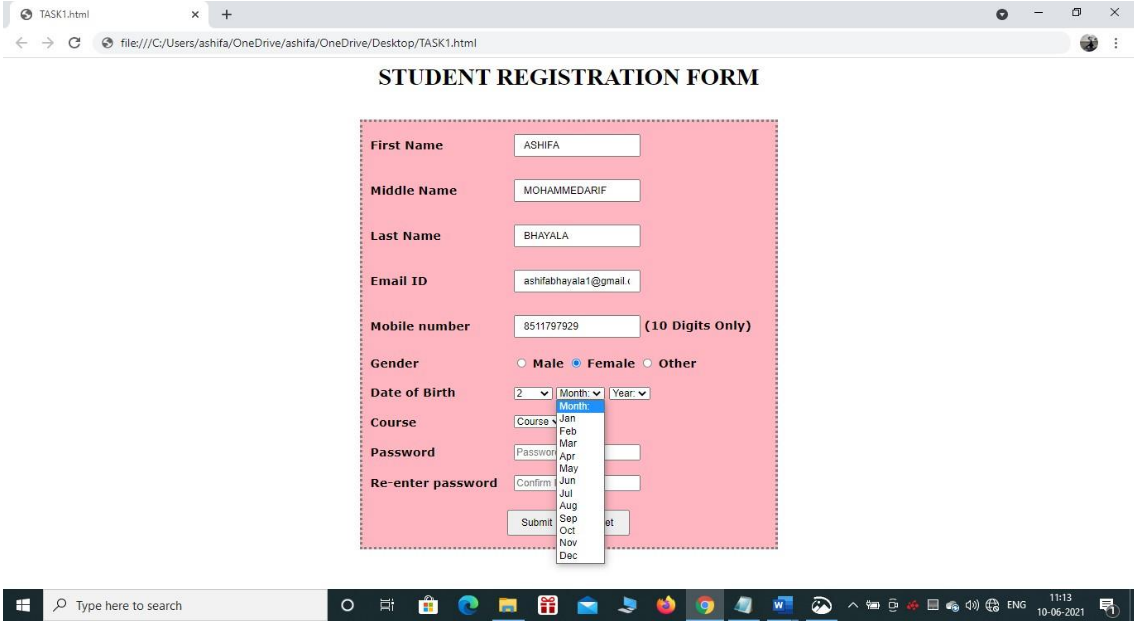
Output:



TASK1.html
file:///C:/Users/ashifa/OneDrive/ashifa/OneDrive/Desktop/TASK1.html
Type here to search
ENG
11:11
10-06-2021

STUDENT REGISTRATION FORM

First Name	
Middle Name	
Last Name	
Email ID	
Mobile number	(10 Digits Only)
Gender	○ Male ○ Female ○ Other
Date of Birth	Day: Month: Year:
Course	Course
Password	Password
Re-enter password	Confirm Password



STUDENT REGISTRATION FORM

First Name	ASHIFA
Middle Name	MOHAMMEDARIF
Last Name	BHAYALA
Email ID	ashifabhayala1@gmail.c
Mobile number	8511797929 (10 Digits Only)
Gender	○ Male ● Female ○ Other
Date of Birth	2 Month: Year:
Course	Course
Password	Passwor
Re-enter password	Confirm

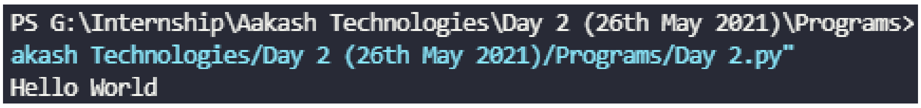
TASK FOR DAY-2

Aim: Write Basic Python Programs:

1) Hello world program
   Code:

```
# This Prints Hello World
print("Hello World")
```

Output:

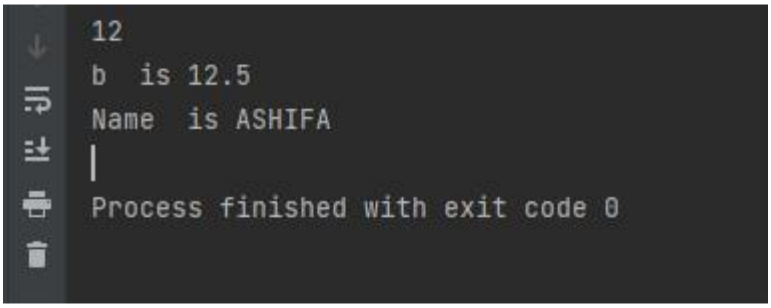


```
PS G:\Internship\Aakash Technologies\Day 2 (26th May 2021)\Programs>
akash Technologies/Day 2 (26th May 2021)/Programs/Day 2.py"
Hello World
```

2) Declaration and use of variable
   Code:

```
# 2. Declaration and use of variable
a=12
b=12.5
name="ASHIFA"

print(a)
print("b is", b)
print("Name is" ,name)
```

Output:

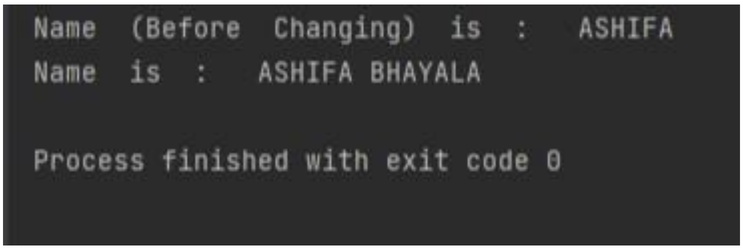


```
12
b  is 12.5
Name  is ASHIFA

Process finished with exit code 0
```

3) Changing the value of variable
   Code:

```
# 3.Changing the value of variable
name="ASHIFA"
print("Name (Before Changing) is : " ,name)
# Assigning new value to same variable
name="ASHIFA BHAYALA"
print("Name is : " ,name)
```

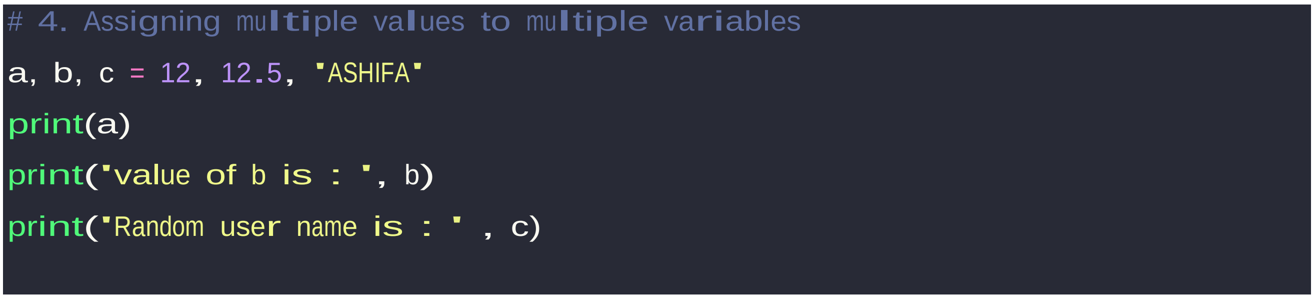
Output:



```
Name  (Before  Changing)  is  :   ASHIFA
Name  is  :   ASHIFA BHAYALA

Process finished with exit code 0
```

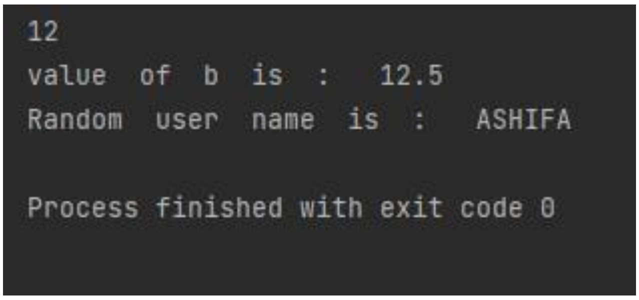
4) Assigning multiple values to multiple variables

Code:



```
# 4. Assigning multiple values to multiple variables
a, b, c = 12, 12.5, "ASHIFA"
print(a)
print("value of b is : ", b)
print("Random user name is : " , c)
```

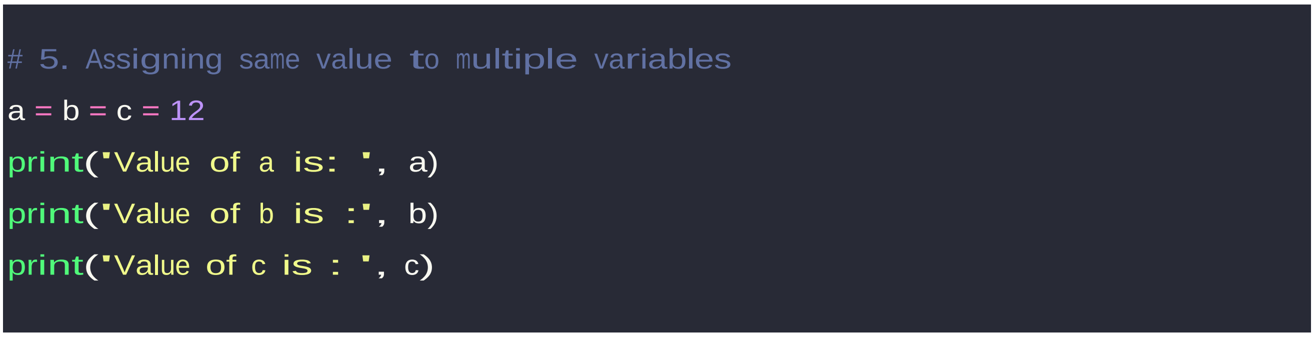
Output:



```
12
value  of  b  is  :   12.5
Random  user  name  is  :   ASHIFA

Process finished with exit code 0
```

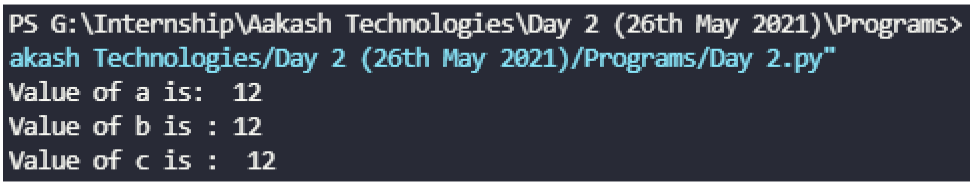
5) Assigning same value to multiple variables

Code:



```
# 5. Assigning same value to multiple variables
a = b = c = 12
print("Value of a is: ", a)
print("Value of b is :", b)
print("Value of c is : ", c)
```

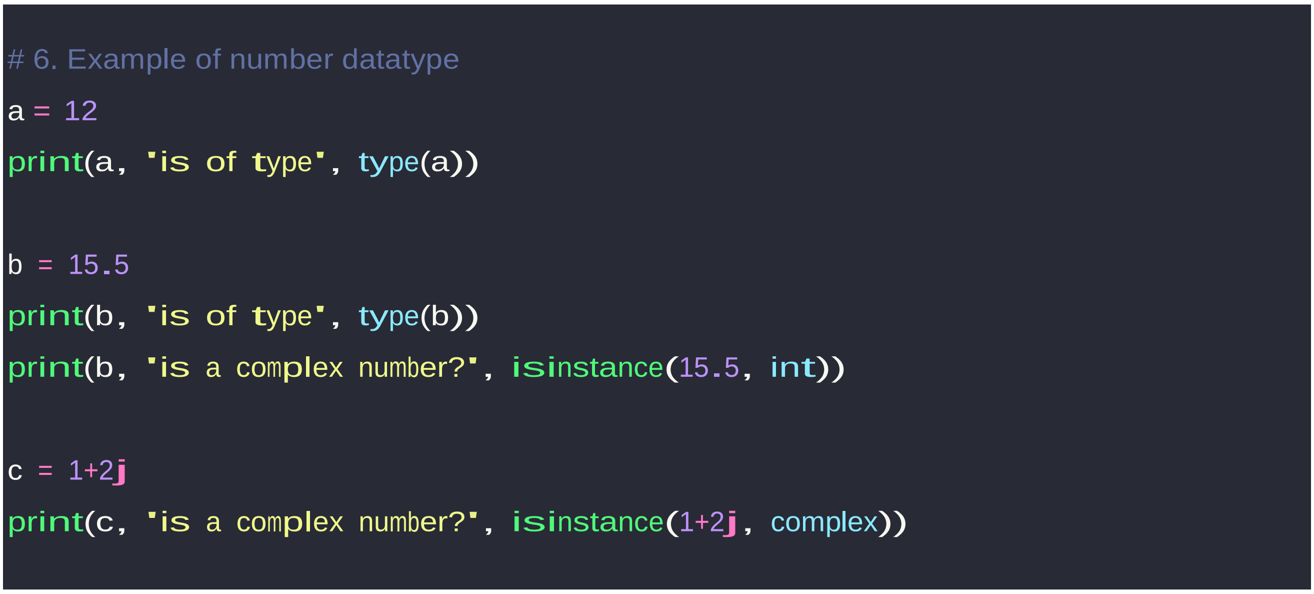
Output:



```
PS G:\Internship\Aakash Technologies\Day 2 (26th May 2021)\Programs>
akash Technologies/Day 2 (26th May 2021)/Programs/Day 2.py"
Value of a is:  12
Value of b is : 12
Value of c is :  12
```

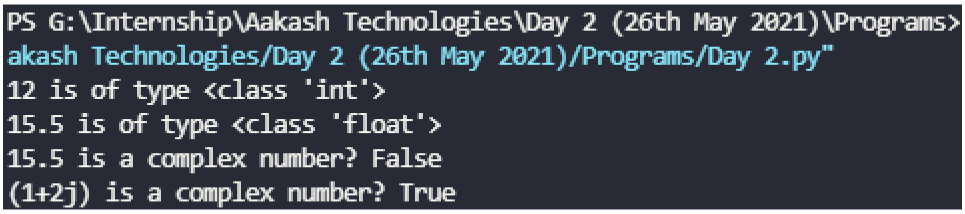
6) Example of number datatype

Code:



```
# 6. Example of number datatype
a = 12
print(a, 'is of type', type(a))

b = 15.5
print(b, 'is of type', type(b))
print(b, 'is a complex number?', isinstance(15.5, int))

c = 1+2j
print(c, 'is a complex number?', isinstance(1+2j, complex))
```

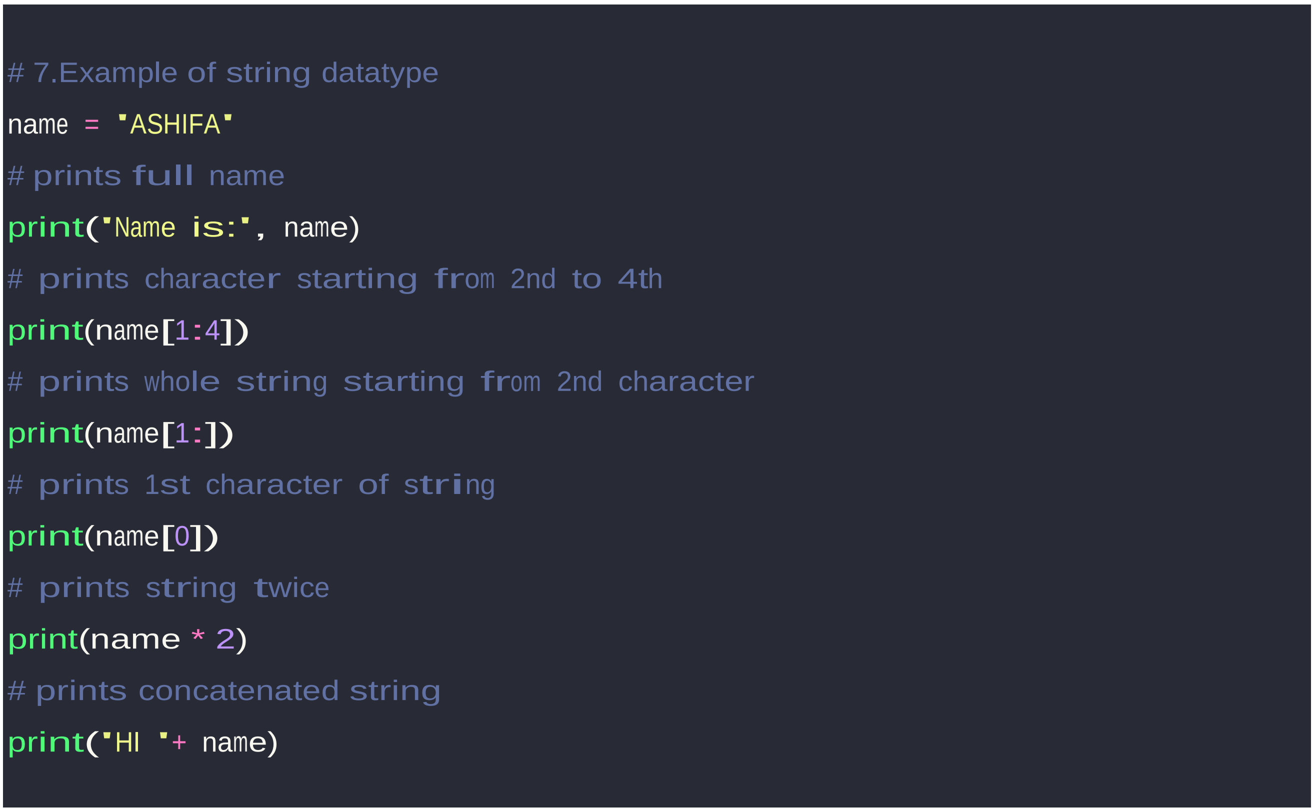
Output:



```
PS G:\Internship\Aakash Technologies\Day 2 (26th May 2021)\Programs>
akash Technologies/Day 2 (26th May 2021)/Programs/Day 2.py"
12 is of type <class 'int'>
15.5 is of type <class 'float'>
15.5 is a complex number? False
(1+2j) is a complex number? True
```

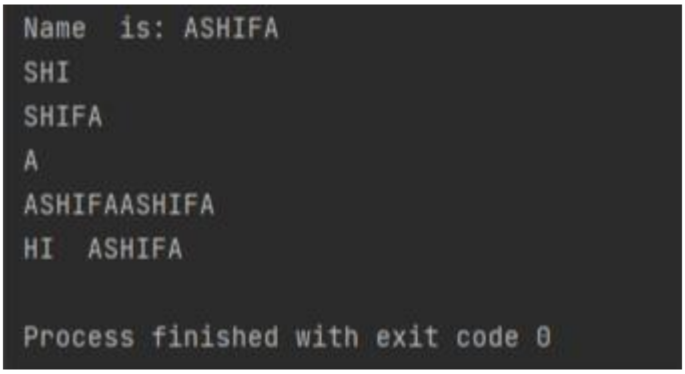
7) Example of string datatype

Code:



```
# 7.Example of string datatype
name = 'ASHIFA'
# prints full name
print('Name is:', name)
# prints character starting from 2nd to 4th
print(name[1:4])
# prints whole string starting from 2nd character
print(name[1:])
# prints 1st character of string
print(name[0])
# prints string twice
print(name * 2)
# prints concatenated string
print('HI '+ name)
```

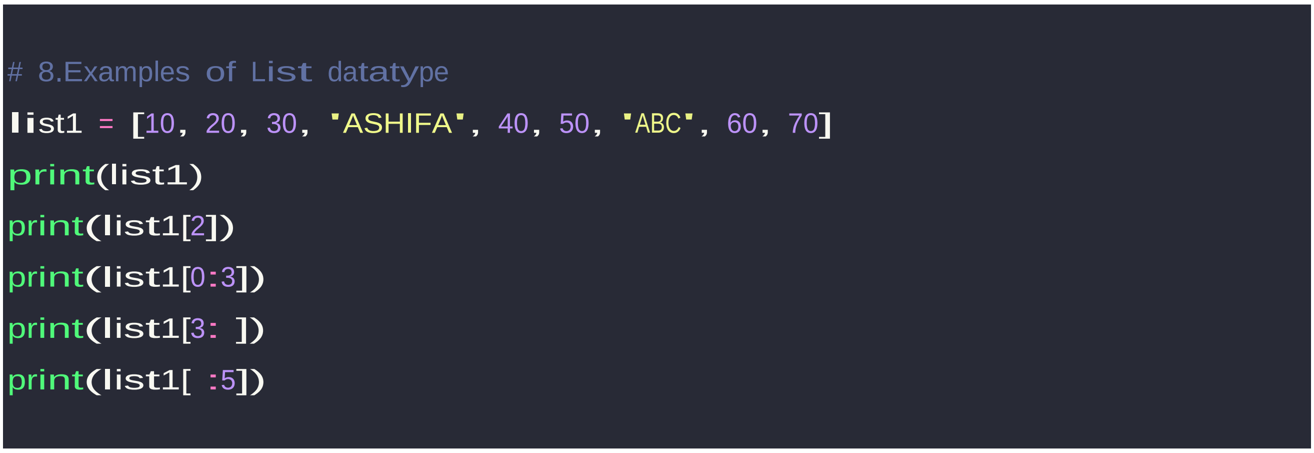
Output:



```
Name  is: ASHIFA
SHI
SHIFA
A
ASHIFAASHIFA
HI  ASHIFA

Process finished with exit code 0
```

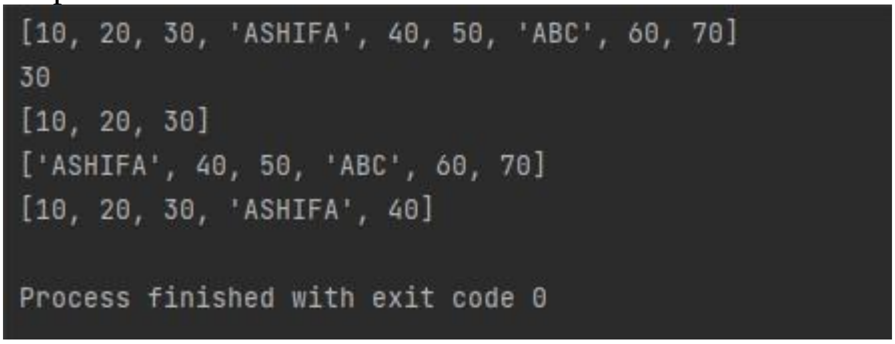
8) Examples of List datatype

Code:



```
# 8.Examples of List datatype
list1 = [10, 20, 30, "ASHIFA", 40, 50, "ABC", 60, 70]
print(list1)
print(list1[2])
print(list1[0:3])
print(list1[3: ])
print(list1[ :5])
```

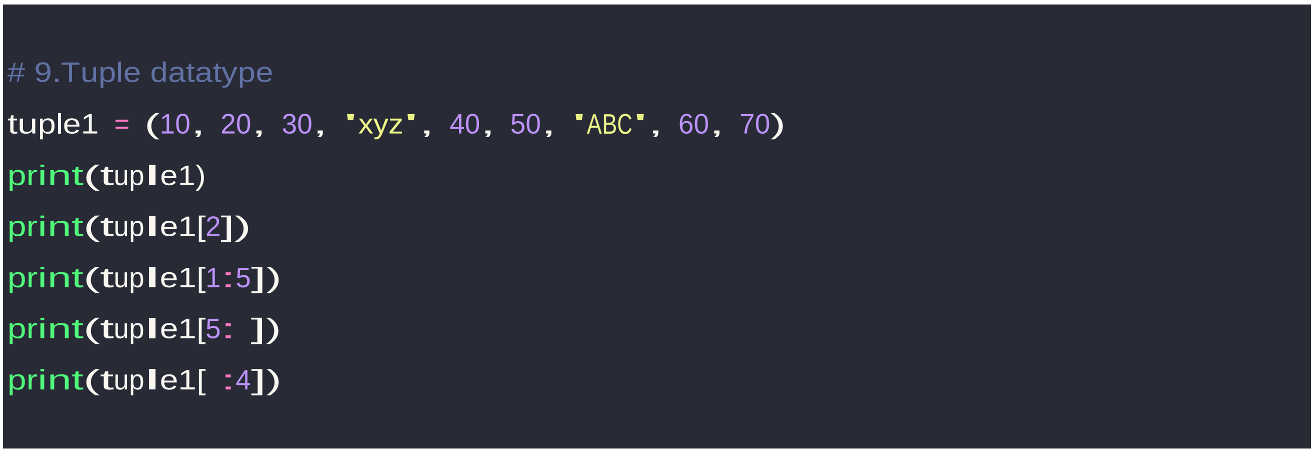
output:



```
[10, 20, 30, 'ASHIFA', 40, 50, 'ABC', 60, 70]
30
[10, 20, 30]
['ASHIFA', 40, 50, 'ABC', 60, 70]
[10, 20, 30, 'ASHIFA', 40]

Process finished with exit code 0
```

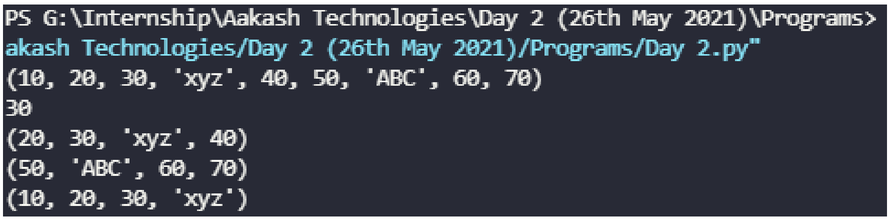
9) Tuple Datatype:

Code:



```
# 9.Tuple datatype
tuple1 = (10, 20, 30, "xyz", 40, 50, "ABC", 60, 70)
print(tuple1)
print(tuple1[2])
print(tuple1[1:5])
print(tuple1[5: ])
print(tuple1[ :4])
```

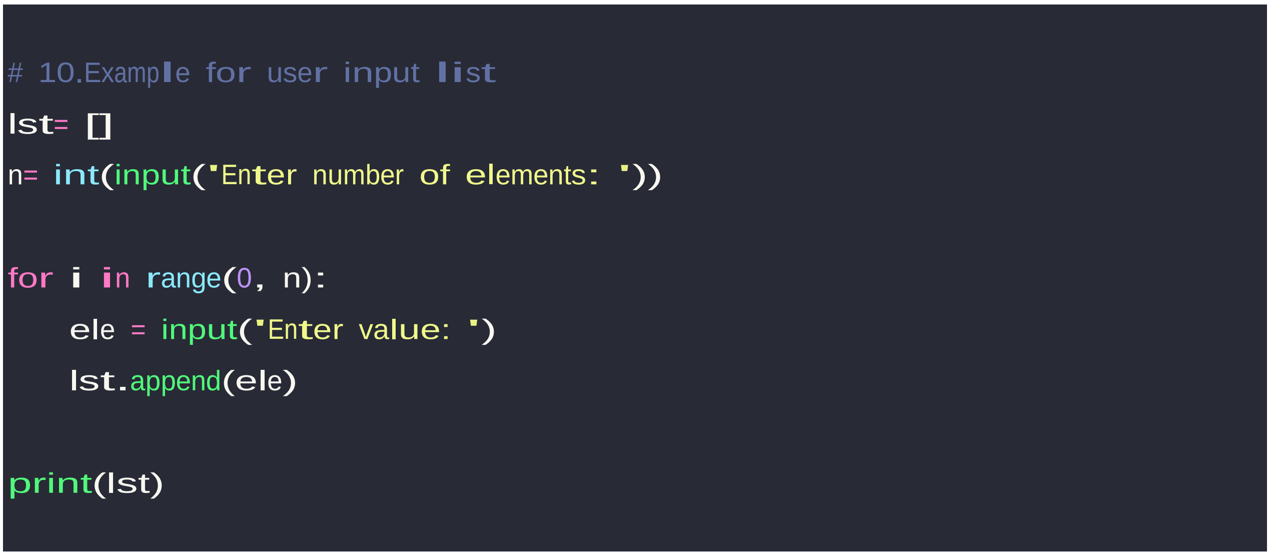
Output:



```
PS G:\Internship\Aakash Technologies\Day 2 (26th May 2021)\Programs>
akash Technologies/Day 2 (26th May 2021)/Programs/Day 2.py"
(10, 20, 30, 'xyz', 40, 50, 'ABC', 60, 70)
30
(20, 30, 'xyz', 40)
(50, 'ABC', 60, 70)
(10, 20, 30, 'xyz')
```

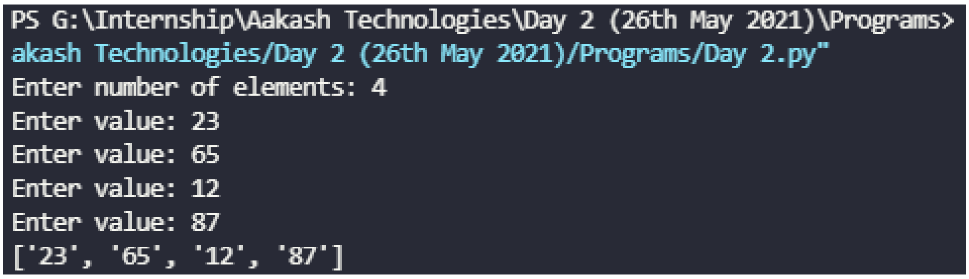
10) Example for user input list

Code:



```
# 10.Example for user input list
lst= []
n= int(input("Enter number of elements: "))

for i in range(0, n):
    ele = input("Enter value: ")
    lst.append(ele)

print(lst)
```

Output:



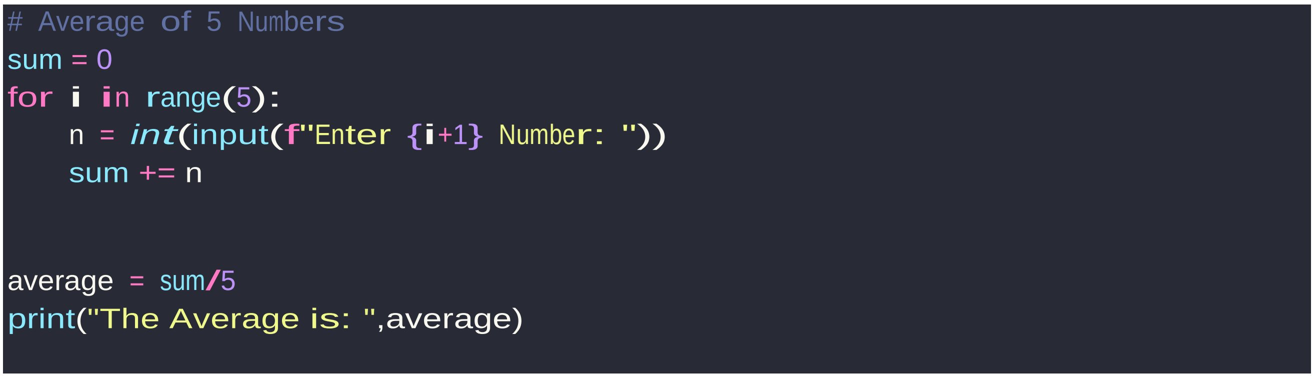
```
PS G:\Internship\Aakash Technologies\Day 2 (26th May 2021)\Programs>
akash Technologies/Day 2 (26th May 2021)/Programs/Day 2.py"
Enter number of elements: 4
Enter value: 23
Enter value: 65
Enter value: 12
Enter value: 87
['23', '65', '12', '87']
```

TASK FOR DAY-3

Aim: Complete the Following Tasks:

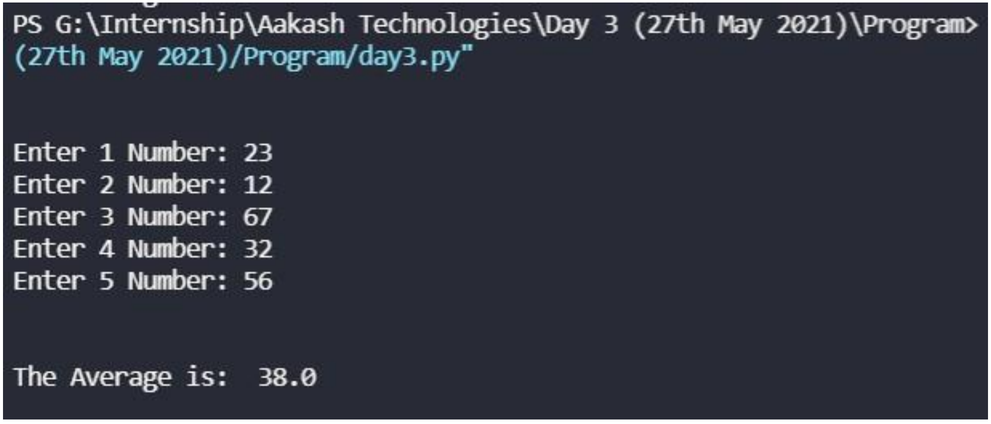
1. Calculate average of 5 numbers.

   Code:



```
# Average of 5 Numbers
sum = 0
for i in range(5):
    n = int(input(f"Enter {i+1} Number: "))
    sum += n


average = sum/5
print("The Average is: ",average)
```

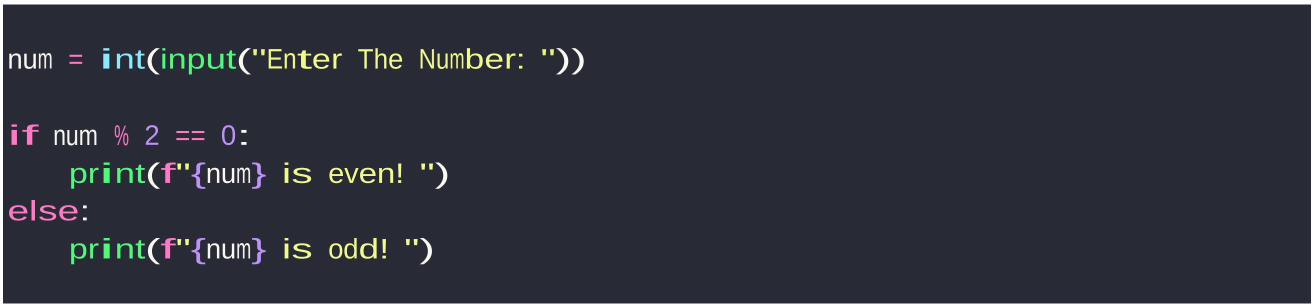
   Output:



```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/day3.py"


Enter 1 Number: 23
Enter 2 Number: 12
Enter 3 Number: 67
Enter 4 Number: 32
Enter 5 Number: 56


The Average is:  38.0
```

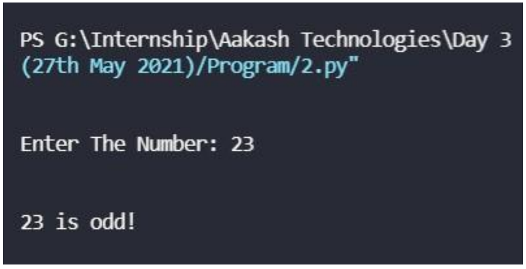
2. Check whether number is even or odd.

   Code:

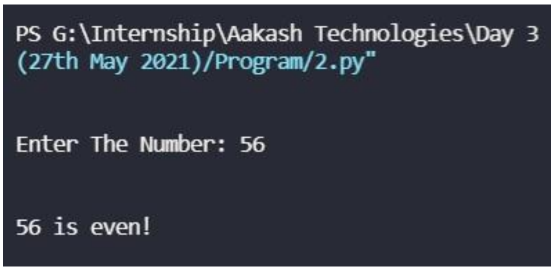


```
num = int(input("Enter The Number: "))

if num % 2 == 0:
    print(f"{num} is even! ")
else:
    print(f"{num} is odd! ")
```

   Output:

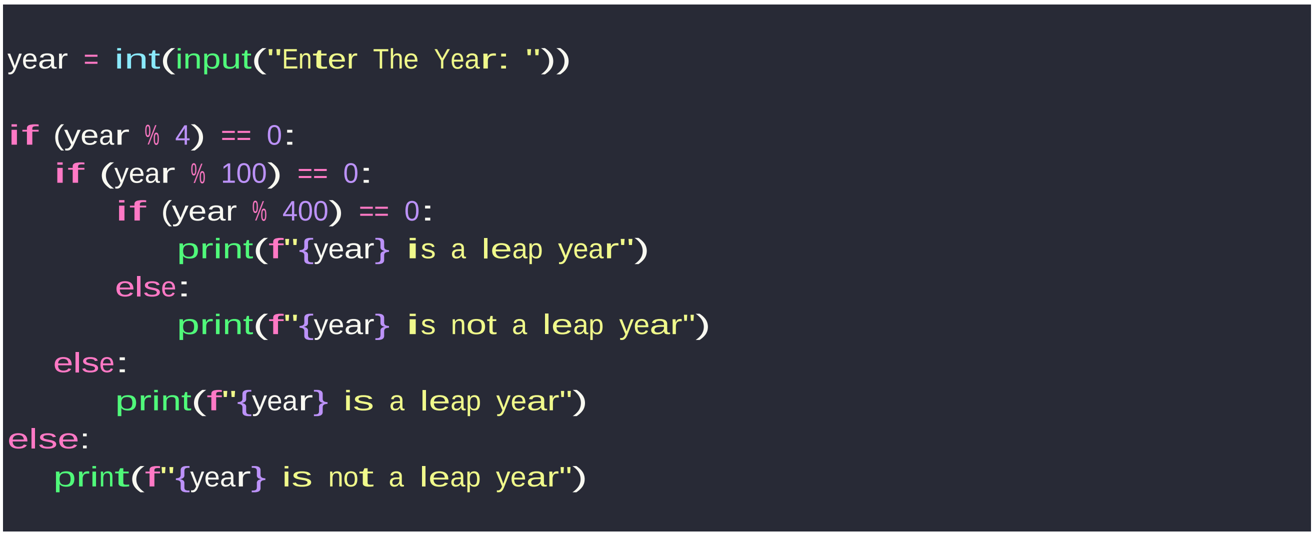


```
PS G:\Internship\Aakash Technologies\Day 3
(27th May 2021)/Program/2.py"


Enter The Number: 23


23 is odd!
```



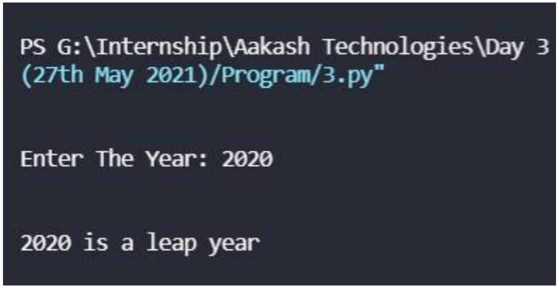
```
PS G:\Internship\Aakash Technologies\Day 3
(27th May 2021)/Program/2.py"


Enter The Number: 56


56 is even!
```

3. Take a year and check whether it is leap year or not.

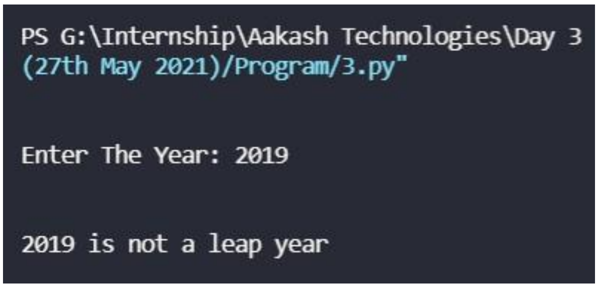


```
year = int(input("Enter The Year: "))

if (year % 4) == 0:
   if (year % 100) == 0:
       if (year % 400) == 0:
           print(f"{year} is a leap year")
       else:
           print(f"{year} is not a leap year")
   else:
       print(f"{year} is a leap year")
else:
   print(f"{year} is not a leap year")
```

Output:



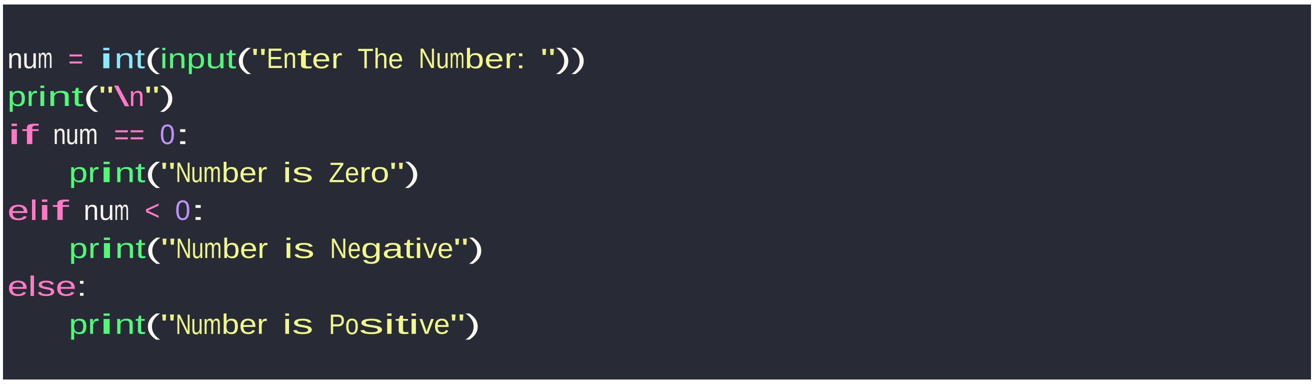
```
PS G:\Internship\Aakash Technologies\Day 3
(27th May 2021)/Program/3.py"

Enter The Year: 2020

2020 is a leap year
```



```
PS G:\Internship\Aakash Technologies\Day 3
(27th May 2021)/Program/3.py"

Enter The Year: 2019

2019 is not a leap year
```

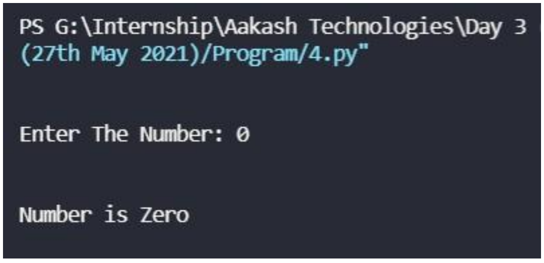
4. Take a number and check whether it is zero, positive or negative.

Code:

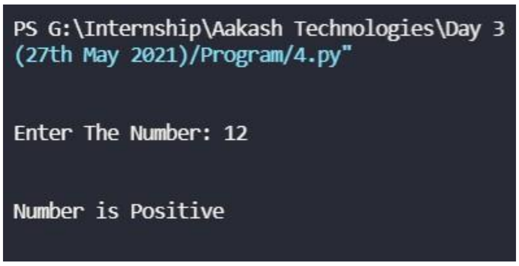


```
num = int(input("Enter The Number: "))
print("\n")
if num == 0:
    print("Number is Zero")
elif num < 0:
    print("Number is Negative")
else:
    print("Number is Positive")
```

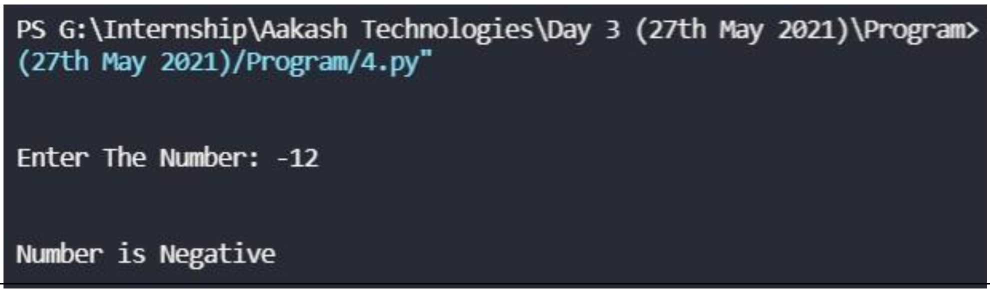
Output:



```
PS G:\Internship\Aakash Technologies\Day 3
(27th May 2021)/Program/4.py"

Enter The Number: 0

Number is Zero
```

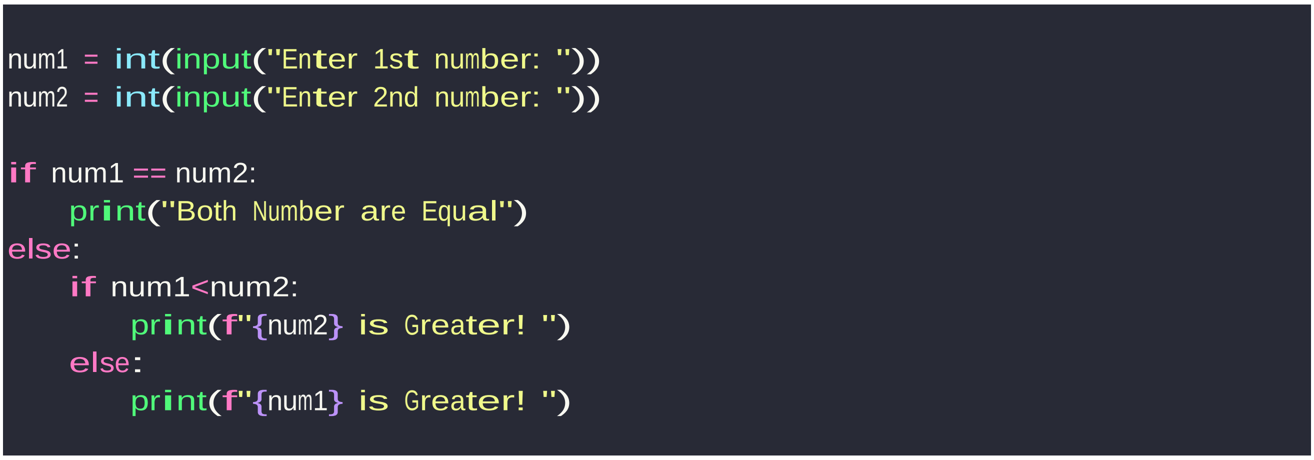


```
PS G:\Internship\Aakash Technologies\Day 3
(27th May 2021)/Program/4.py"

Enter The Number: 12

Number is Positive
```



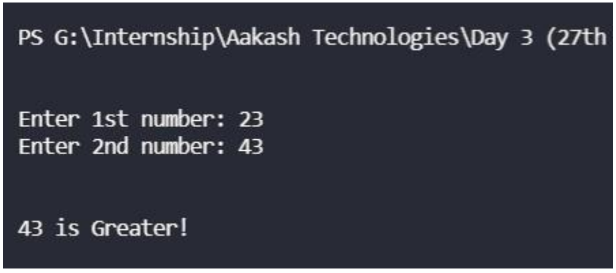
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/4.py"

Enter The Number: -12

Number is Negative
```

5. Take 2 numbers and display greatest number. (Also check equal number condition)

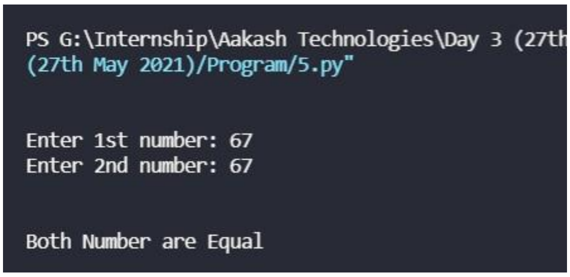


```
num1 = int(input("Enter 1st number: "))
num2 = int(input("Enter 2nd number: "))

if num1 == num2:
    print("Both Number are Equal")
else:
    if num1<num2:
        print(f"{num2} is Greater! ")
    else:
        print(f"{num1} is Greater! ")
```

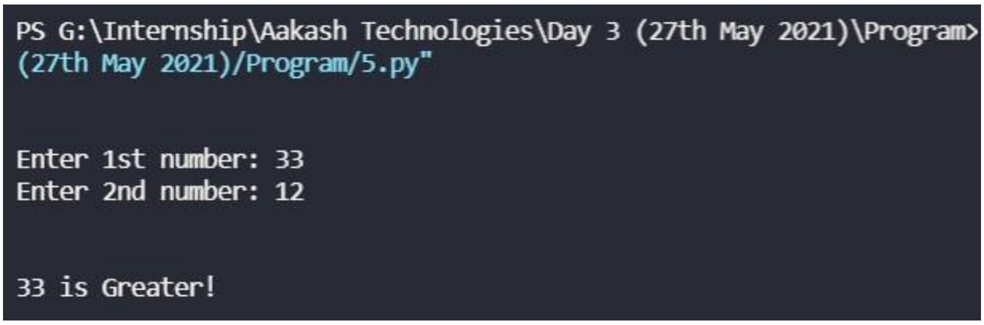
Output:



```
PS G:\Internship\Aakash Technologies\Day 3 (27th


Enter 1st number: 23
Enter 2nd number: 43


43 is Greater!
```

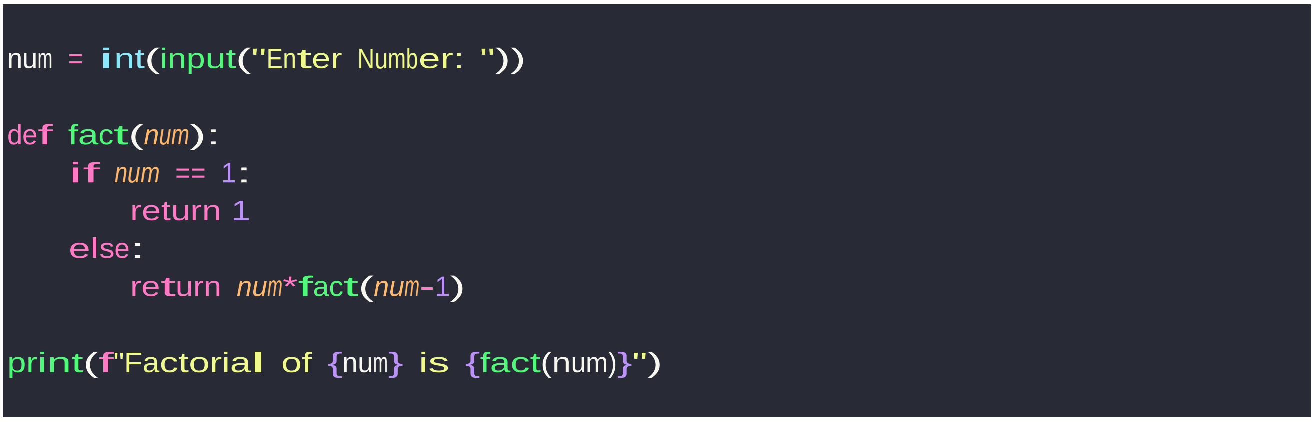


```
PS G:\Internship\Aakash Technologies\Day 3 (27th
(27th May 2021)/Program/5.py"


Enter 1st number: 67
Enter 2nd number: 67


Both Number are Equal
```



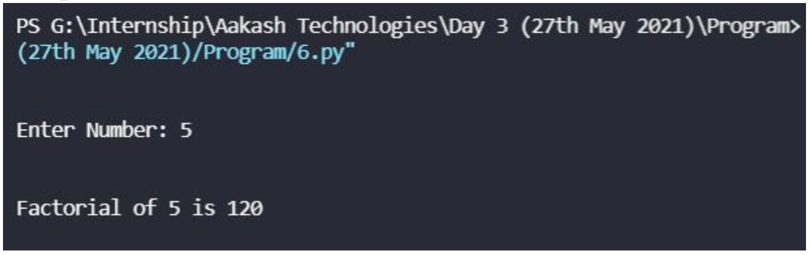
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/5.py"


Enter 1st number: 33
Enter 2nd number: 12


33 is Greater!
```

6. Take a number and find factorial of that number.



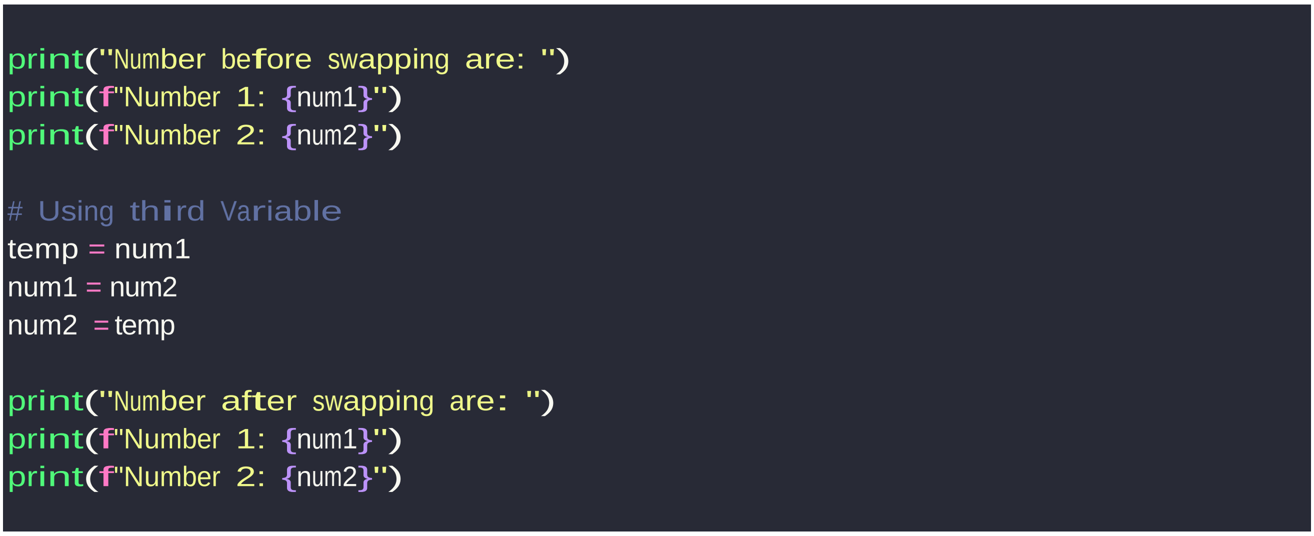
```
num = int(input("Enter Number: "))

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

print(f"Factorial of {num} is {fact(num)}")
```

Output:



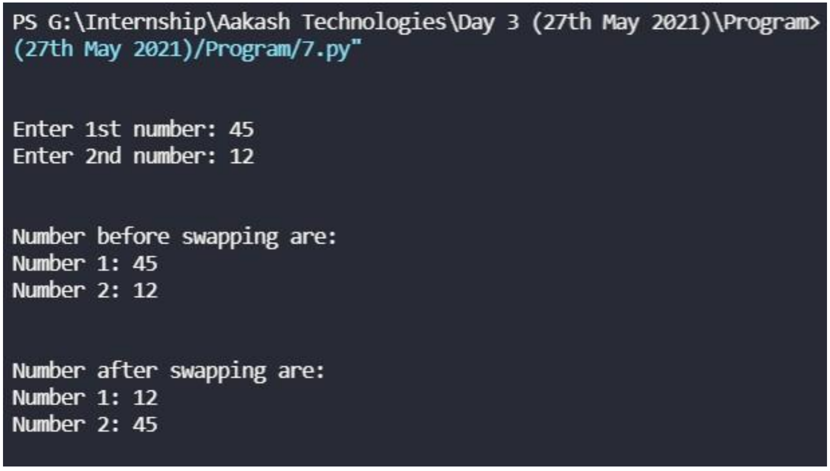
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/6.py"


Enter Number: 5


Factorial of 5 is 120
```

7. Write a program to swap 2 numbers using third variable.



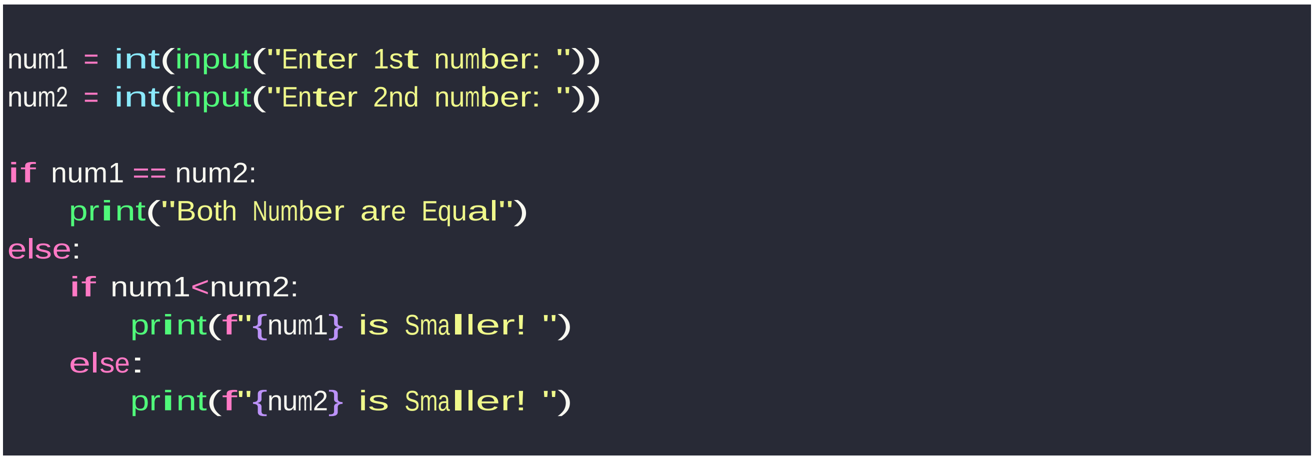
```
print("Number before swapping are: ")
print(f"Number 1: {num1}")
print(f"Number 2: {num2}")

# Using third Variable
temp = num1
num1 = num2
num2  = temp

print("Number after swapping are: ")
print(f"Number 1: {num1}")
print(f"Number 2: {num2}")
```

Output:



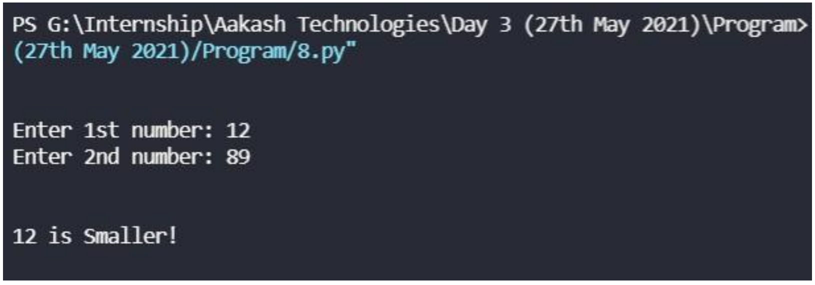
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/7.py"


Enter 1st number: 45
Enter 2nd number: 12


Number before swapping are:
Number 1: 45
Number 2: 12


Number after swapping are:
Number 1: 12
Number 2: 45
```

8. Take 2 numbers and find smallest number.



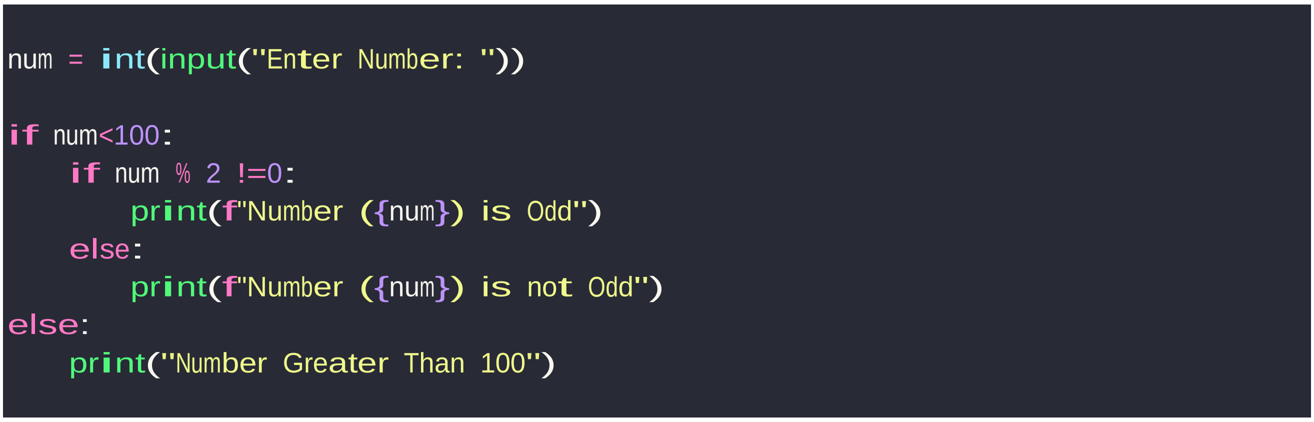
```
num1 = int(input("Enter 1st number: "))
num2 = int(input("Enter 2nd number: "))

if num1 == num2:
    print("Both Number are Equal")
else:
    if num1<num2:
        print(f"{num1} is Smaller! ")
    else:
        print(f"{num2} is Smaller! ")
```

Output:



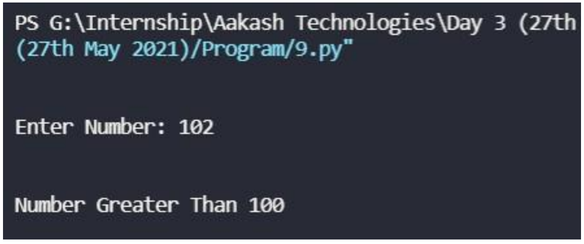
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/8.py"


Enter 1st number: 12
Enter 2nd number: 89


12 is Smaller!
```

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.

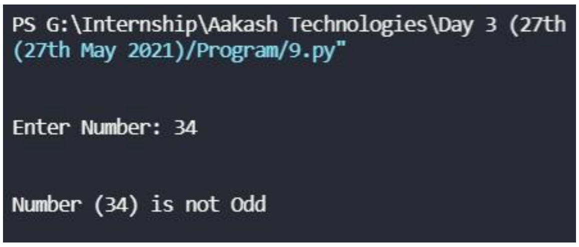


```
num = int(input("Enter Number: "))

if num<100:
    if num % 2 !=0:
        print(f"Number ({num}) is Odd")
    else:
        print(f"Number ({num}) is not Odd")
else:
    print("Number Greater Than 100")
```

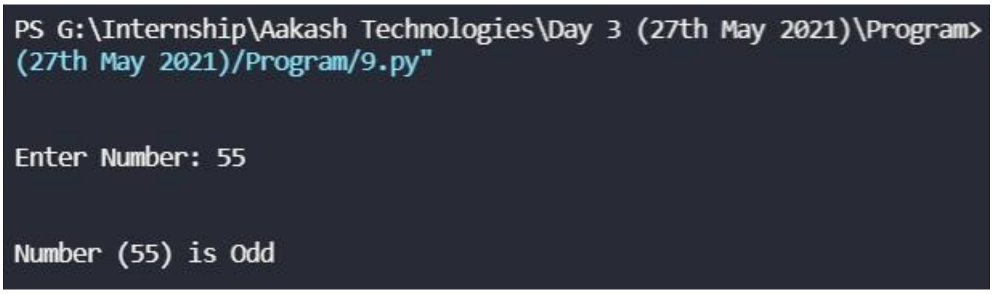
Output:



```
PS G:\Internship\Aakash Technologies\Day 3 (27th
(27th May 2021)/Program/9.py"


Enter Number: 102


Number Greater Than 100
```

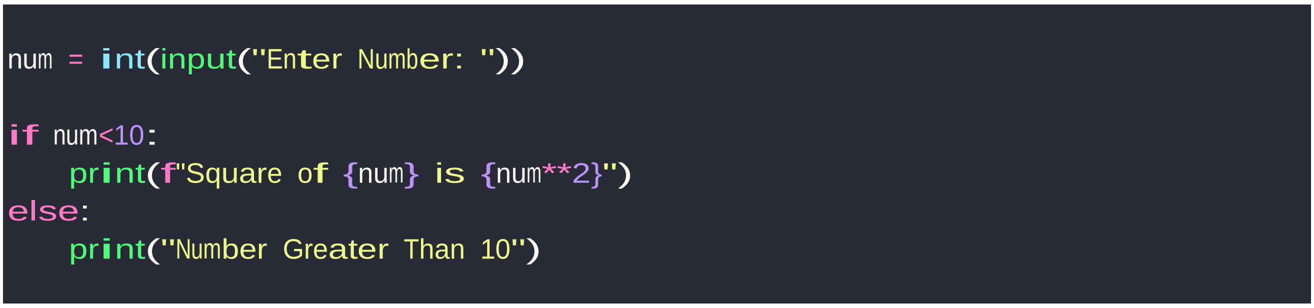


```
PS G:\Internship\Aakash Technologies\Day 3 (27th
(27th May 2021)/Program/9.py"


Enter Number: 34


Number (34) is not Odd
```



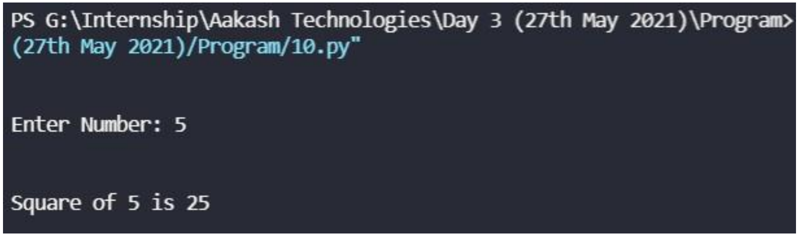
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/9.py"


Enter Number: 55


Number (55) is Odd
```

10. Take a number to print the square of a number if it is less than 10.



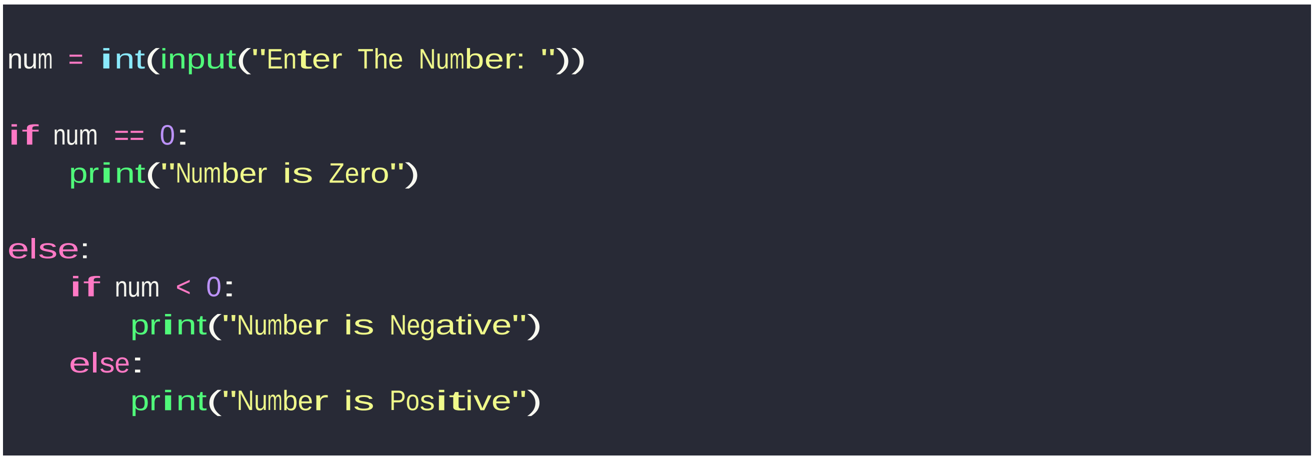
```
num = int(input("Enter Number: "))

if num<10:
    print(f"Square of {num} is {num**2}")
else:
    print("Number Greater Than 10")
```

Output:



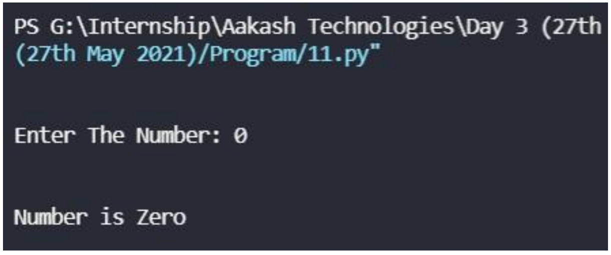
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/10.py"


Enter Number: 5


Square of 5 is 25
```

11. Take a number and check whether it is zero, positive or negative using nested IF…ELSE statement.

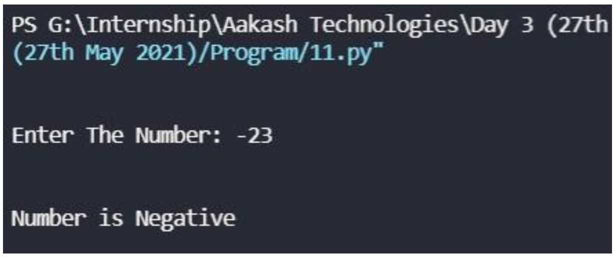


```
num = int(input("Enter The Number: "))

if num == 0:
    print("Number is Zero")

else:
    if num < 0:
        print("Number is Negative")
    else:
        print("Number is Positive")
```

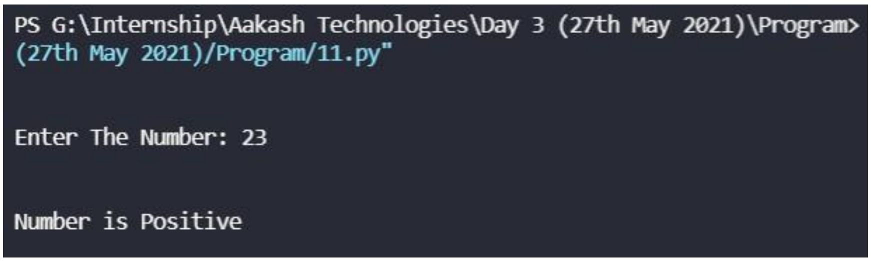
Output:



```
PS G:\Internship\Aakash Technologies\Day 3 (27th
(27th May 2021)/Program/11.py"

Enter The Number: 0

Number is Zero
```

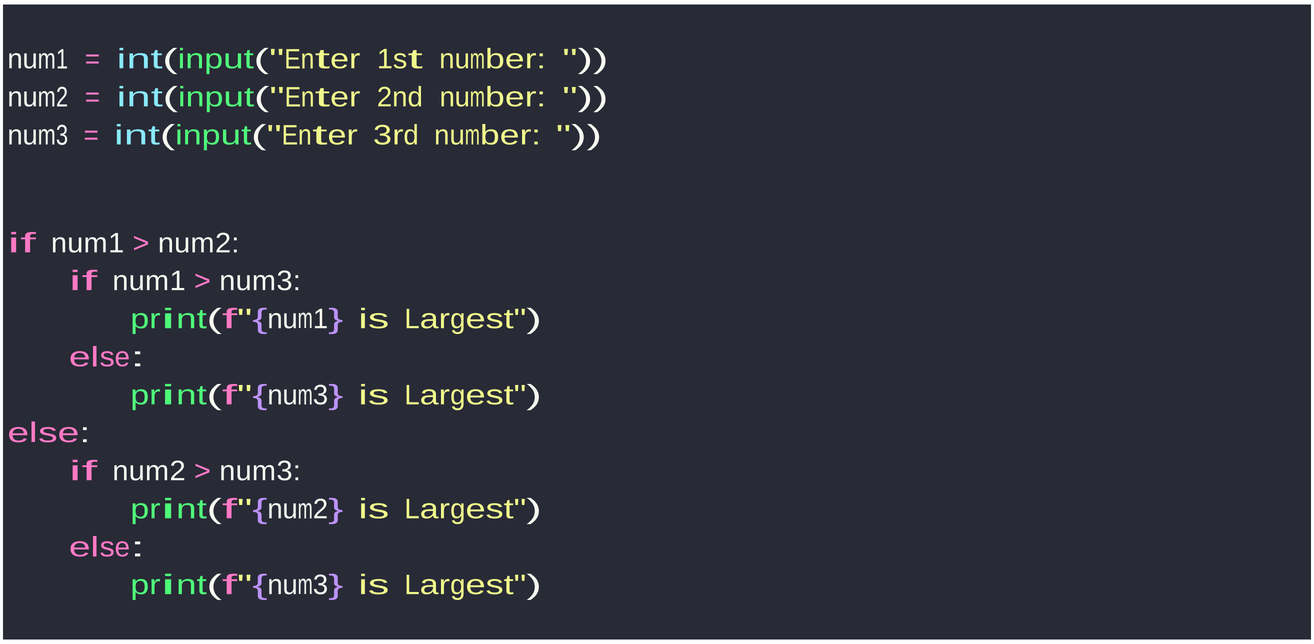


```
PS G:\Internship\Aakash Technologies\Day 3 (27th
(27th May 2021)/Program/11.py"

Enter The Number: -23

Number is Negative
```



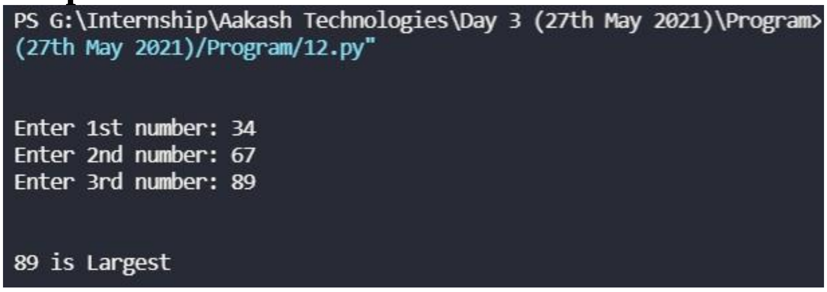
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/11.py"

Enter The Number: 23

Number is Positive
```

12. Take 3 numbers and find greatest number using nested IF….ELSE statement.



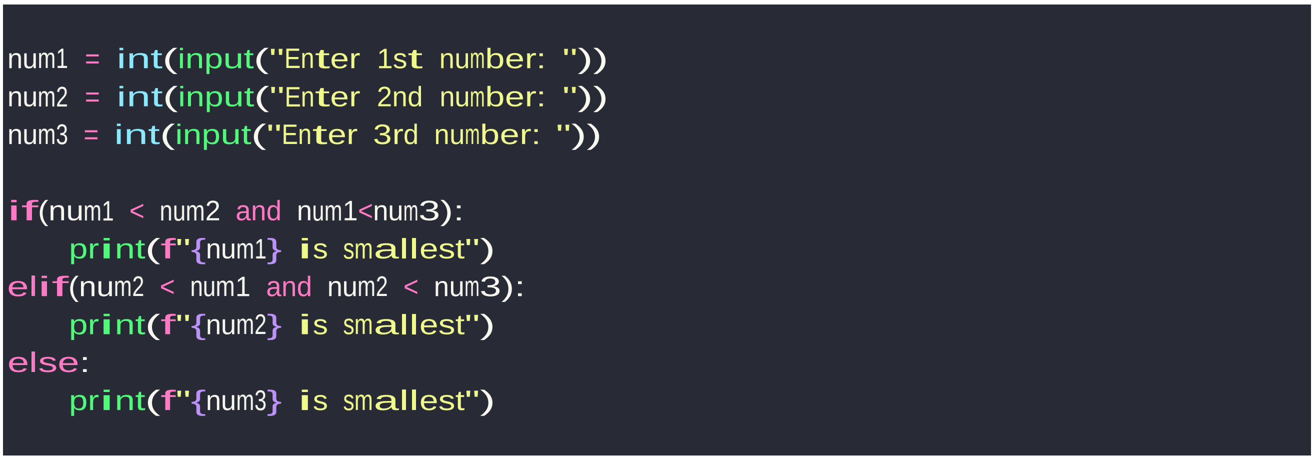
```
num1 = int(input("Enter 1st number: "))
num2 = int(input("Enter 2nd number: "))
num3 = int(input("Enter 3rd number: "))


if num1 > num2:
    if num1 > num3:
        print(f"{num1} is Largest")
    else:
        print(f"{num3} is Largest")
else:
    if num2 > num3:
        print(f"{num2} is Largest")
    else:
        print(f"{num3} is Largest")
```

Output:



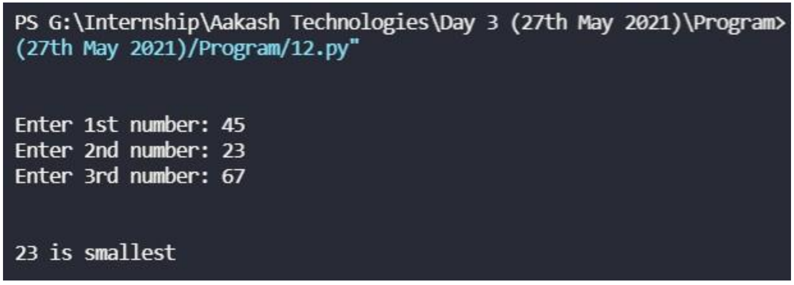
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/12.py"


Enter 1st number: 34
Enter 2nd number: 67
Enter 3rd number: 89


89 is Largest
```

13. Take 3 numbers and find smallest number using logical operator.



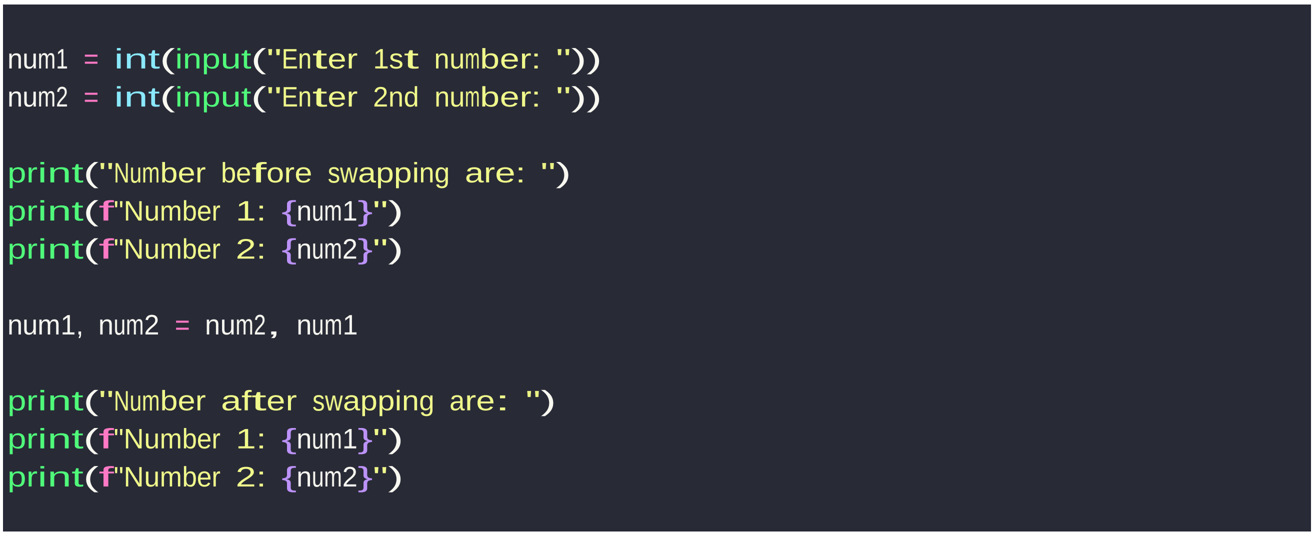
```
num1 = int(input("Enter 1st number: "))
num2 = int(input("Enter 2nd number: "))
num3 = int(input("Enter 3rd number: "))

if(num1 < num2 and num1<num3):
    print(f"{num1} is smallest")
elif(num2 < num1 and num2 < num3):
    print(f"{num2} is smallest")
else:
    print(f"{num3} is smallest")
```

Output:



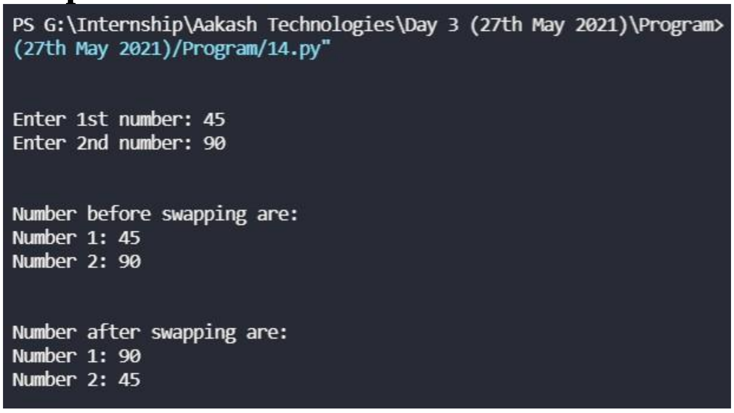
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/12.py"


Enter 1st number: 45
Enter 2nd number: 23
Enter 3rd number: 67


23 is smallest
```

14. Write a program to swap 2 numbers without taking third variable.



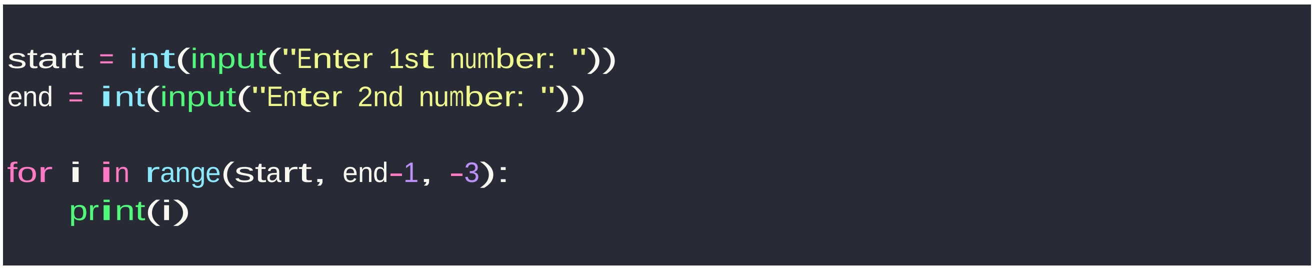
```
num1 = int(input("Enter 1st number: "))
num2 = int(input("Enter 2nd number: "))

print("Number before swapping are: ")
print(f"Number 1: {num1}")
print(f"Number 2: {num2}")

num1, num2 = num2, num1

print("Number after swapping are: ")
print(f"Number 1: {num1}")
print(f"Number 2: {num2}")
```

Output:



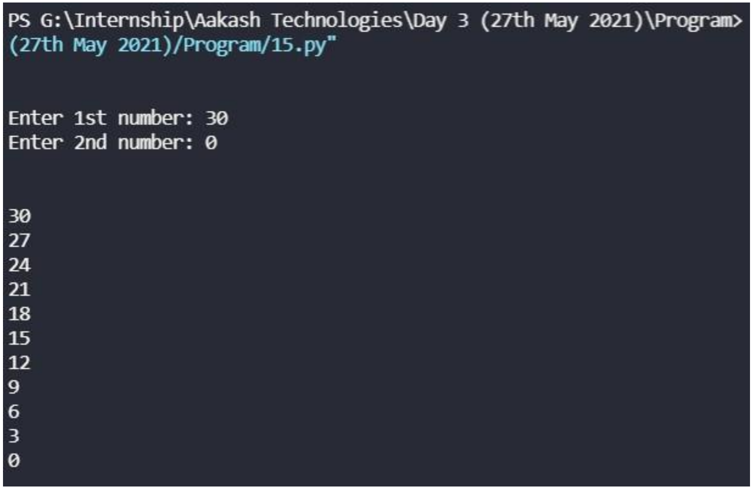
```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/14.py"


Enter 1st number: 45
Enter 2nd number: 90


Number before swapping are:
Number 1: 45
Number 2: 90


Number after swapping are:
Number 1: 90
Number 2: 45
```

15. Take starting number and ending number from the user and print following series.



```
start = int(input("Enter 1st number: "))
end = int(input("Enter 2nd number: "))

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

Output:



```
PS G:\Internship\Aakash Technologies\Day 3 (27th May 2021)\Program>
(27th May 2021)/Program/15.py"


Enter 1st number: 30
Enter 2nd number: 0


30
27
24
21
18
15
12
9
6
3
0
```

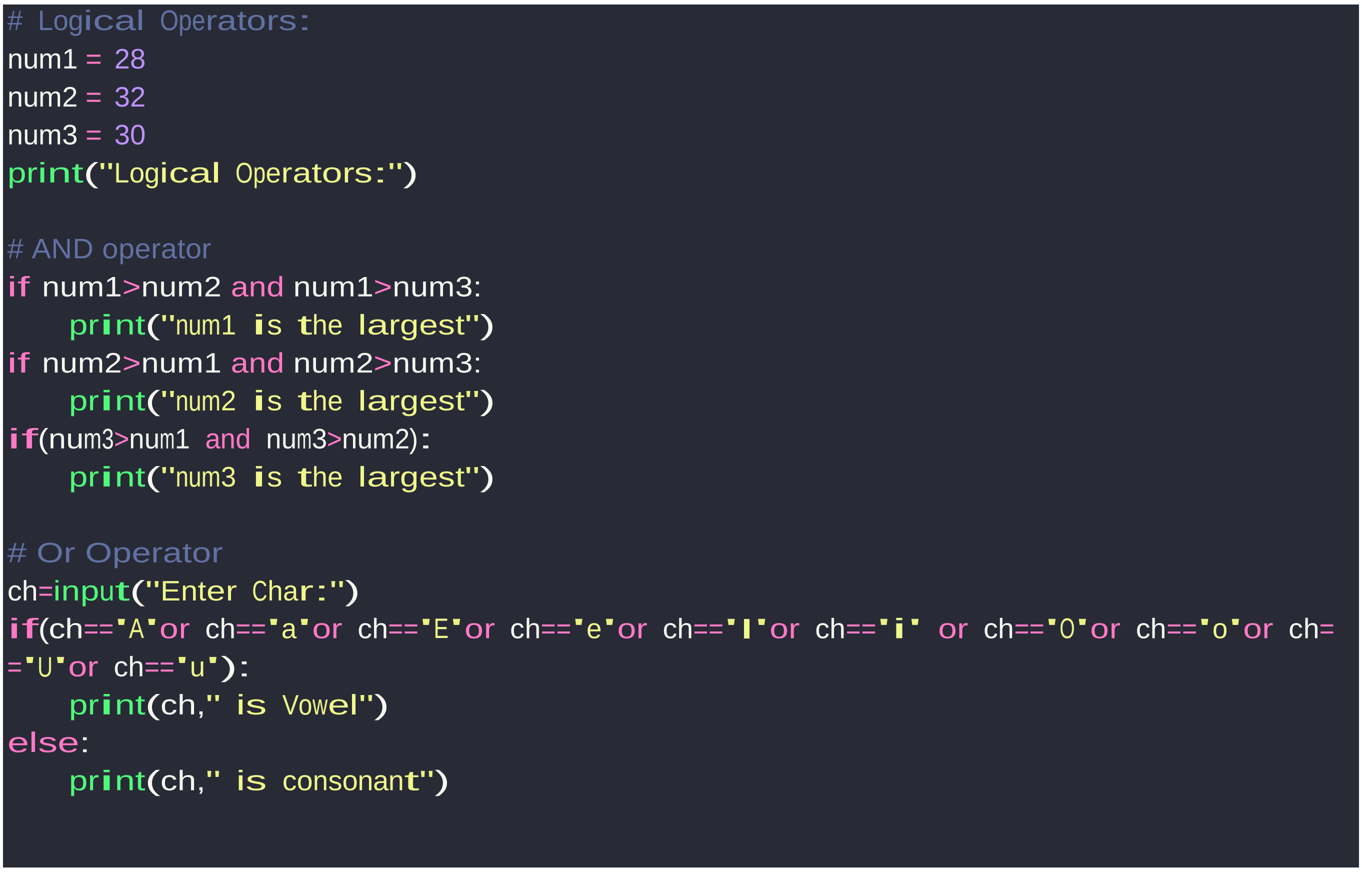
TASK FOR DAY-4

Aim: Understanding operators and modules:

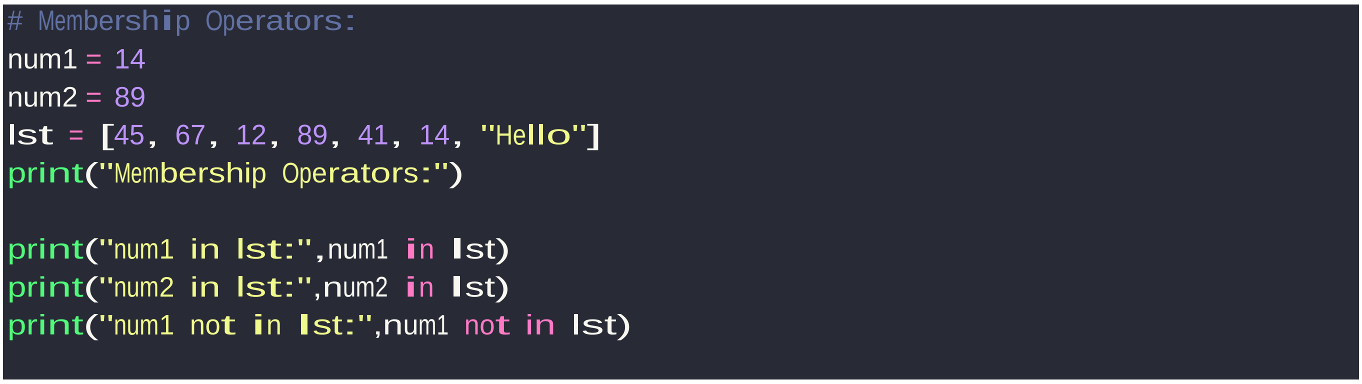
1. List All the Operators:

```
# Arithmetic Operators:
num1 = 20
num2 = 10

print("Arithmetic Operators:")

result = num1 + num2
print("Addition  is:  ", result)

result = num1 - num2
print("Substraction is: ", result)

result = num1 * num2
print("Multiplication is: ",  result)

result = num1 / num2
print("Division is: ", result)

result = num1 // num2
print("Integer Division is: ",  result)

result = num1 % num2
print("Modulo  is: ", result)
```

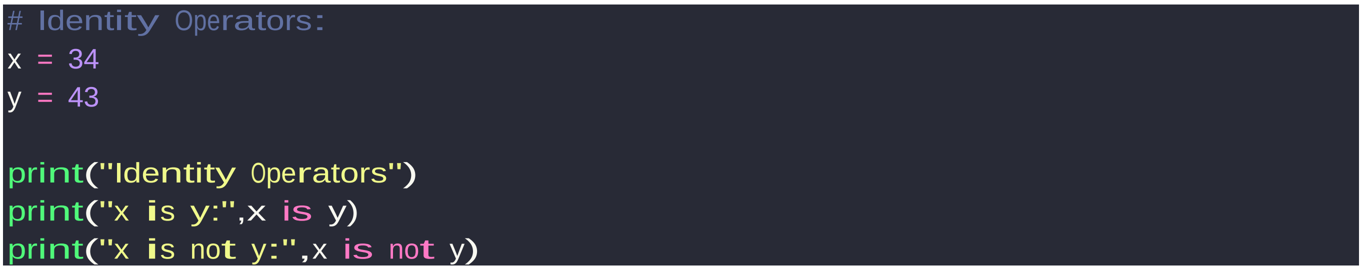
```
# Comparison Operators:
num1 = 25
num2 = 17
print("Comparison Operators:")

result = num1 > num2
print("Num1 is Greater than Num 2: ", result)

result = num1 < num2
print("Num1 is Less than Num 2: ", result)

result = num1 == num2
print("Num1 is Equal to Num 2: ", result)

result = num1 >= num2
print("Num1 is Greater than or equal to Num 2: ", result)

result = num1 <= num2
print("Num1 is Less than or equal to Num 2: ", result)

result = num1 != num2
print("Num1 is not euqal to Num 2: ", result)
```



```
# Logical Operators:
num1 = 28
num2 = 32
num3 = 30
print("Logical Operators:")

# AND operator
if num1>num2 and num1>num3:
    print("num1 is the largest")
if num2>num1 and num2>num3:
    print("num2 is the largest")
if(num3>num1 and num3>num2):
    print("num3 is the largest")

# Or Operator
ch=input("Enter Char:")
if(ch=='A'or ch=='a'or ch=='E'or ch=='e'or ch=='I'or ch=='i' or ch=='O'or ch=='o'or ch=
='U'or ch=='u'):
    print(ch," is Vowel")
else:
    print(ch," is consonant")
```

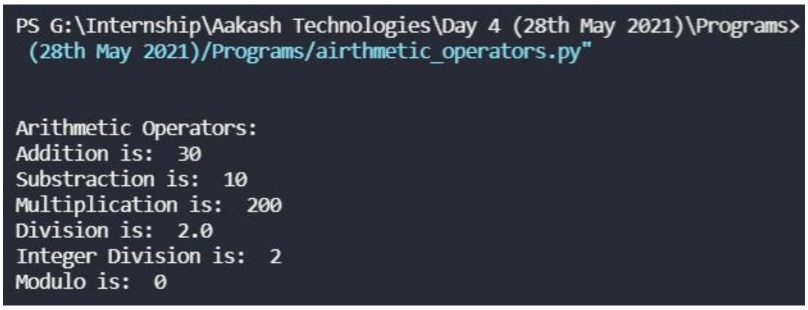


```
# Membership Operators:
num1 = 14
num2 = 89
lst = [45, 67, 12, 89, 41, 14, "Hello"]
print("Membership Operators:")

print("num1 in lst:",num1 in lst)
print("num2 in lst:",num2 in lst)
print("num1 not in lst:",num1 not in lst)
```

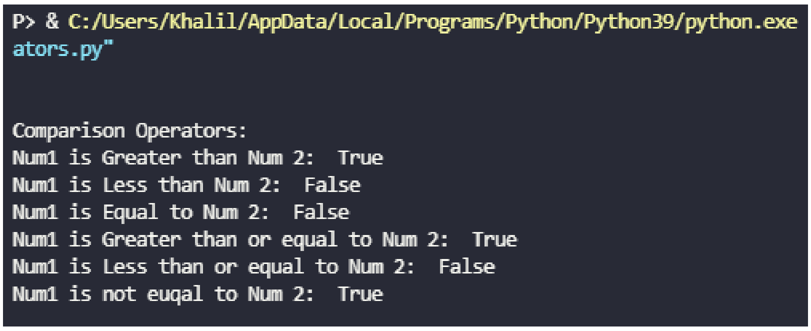


```
# Identity Operators:
x = 34
y = 43

print("Identity Operators")
print("x is y:",x is y)
print("x is not y:",x is not y)
```

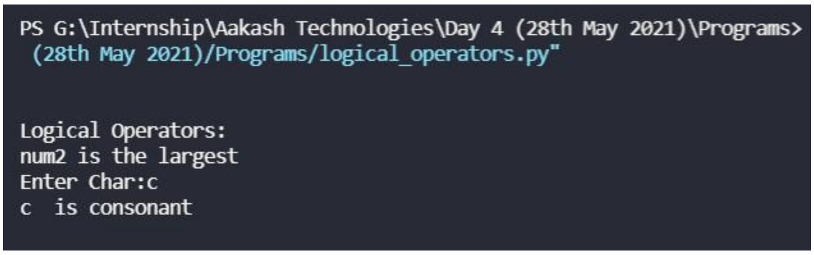
Outputs:



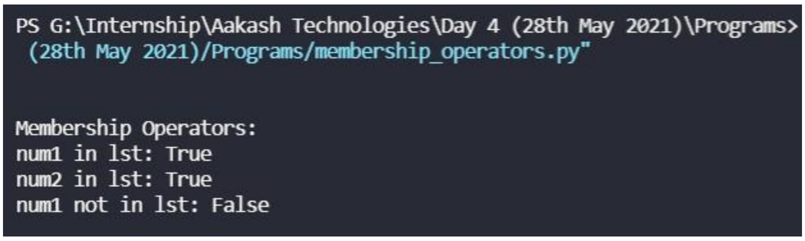
```
PS G:\Internship\Aakash Technologies\Day 4 (28th May 2021)\Programs>
 (28th May 2021)/Programs/airthmetic_operators.py"


Arithmetic Operators:
Addition is:  30
Substraction is:  10
Multiplication is:  200
Division is:  2.0
Integer Division is:  2
Modulo is:  0
```



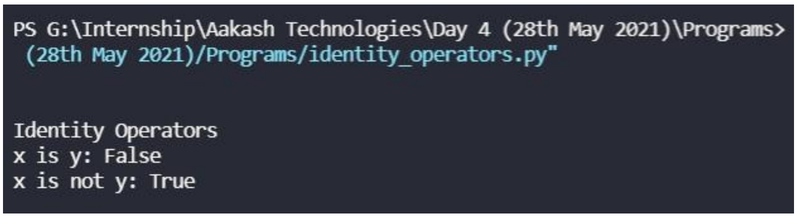
```
P> & C:/Users/Khalil/AppData/Local/Programs/Python/Python39/python.exe
ators.py"


Comparison Operators:
Num1 is Greater than Num 2:  True
Num1 is Less than Num 2:  False
Num1 is Equal to Num 2:  False
Num1 is Greater than or equal to Num 2:  True
Num1 is Less than or equal to Num 2:  False
Num1 is not euqal to Num 2:  True
```



```
PS G:\Internship\Aakash Technologies\Day 4 (28th May 2021)\Programs>
 (28th May 2021)/Programs/logical_operators.py"


Logical Operators:
num2 is the largest
Enter Char:c
c  is consonant
```

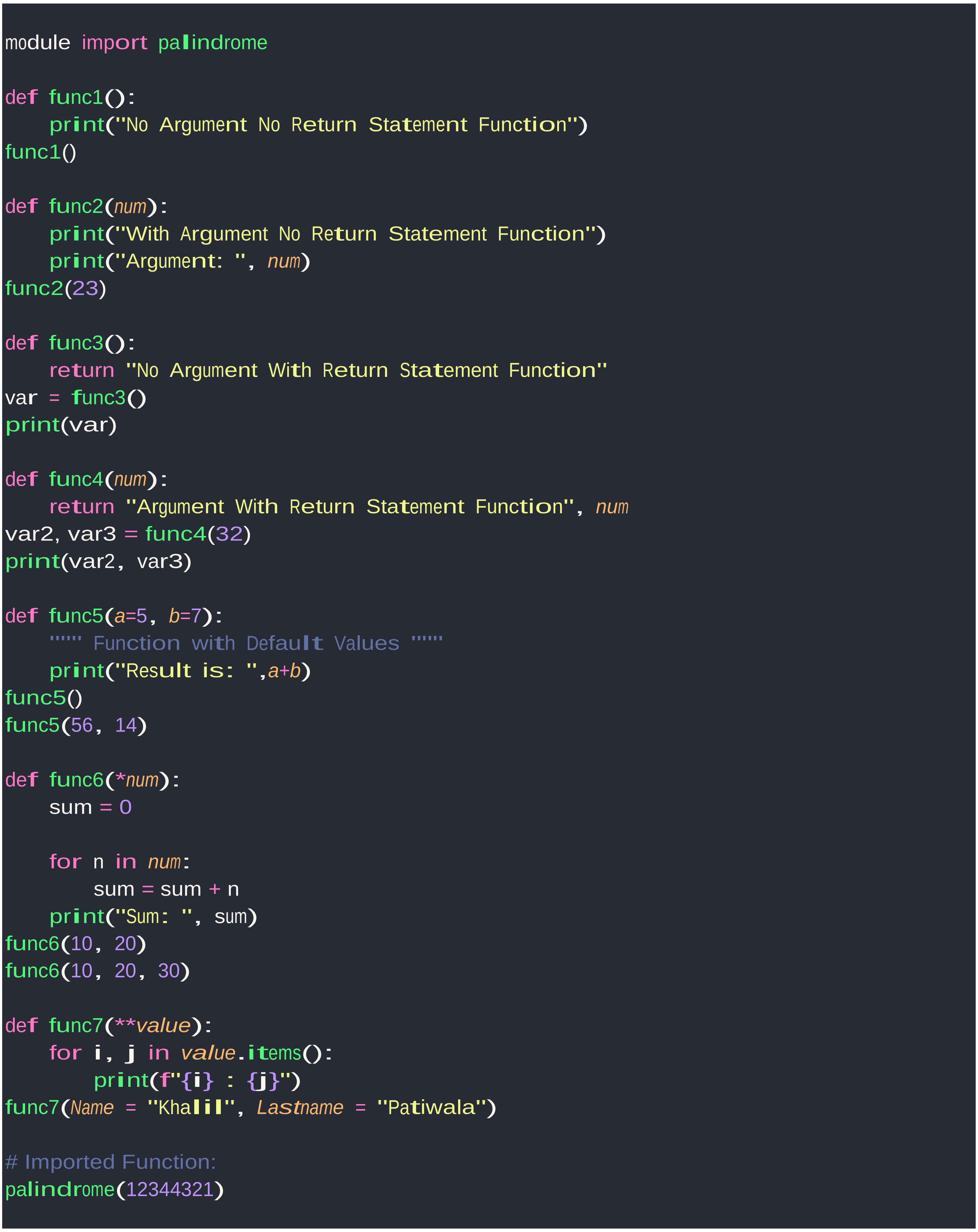


```
PS G:\Internship\Aakash Technologies\Day 4 (28th May 2021)\Programs>
 (28th May 2021)/Programs/membership_operators.py"


Membership Operators:
num1 in lst: True
num2 in lst: True
num1 not in lst: False
```



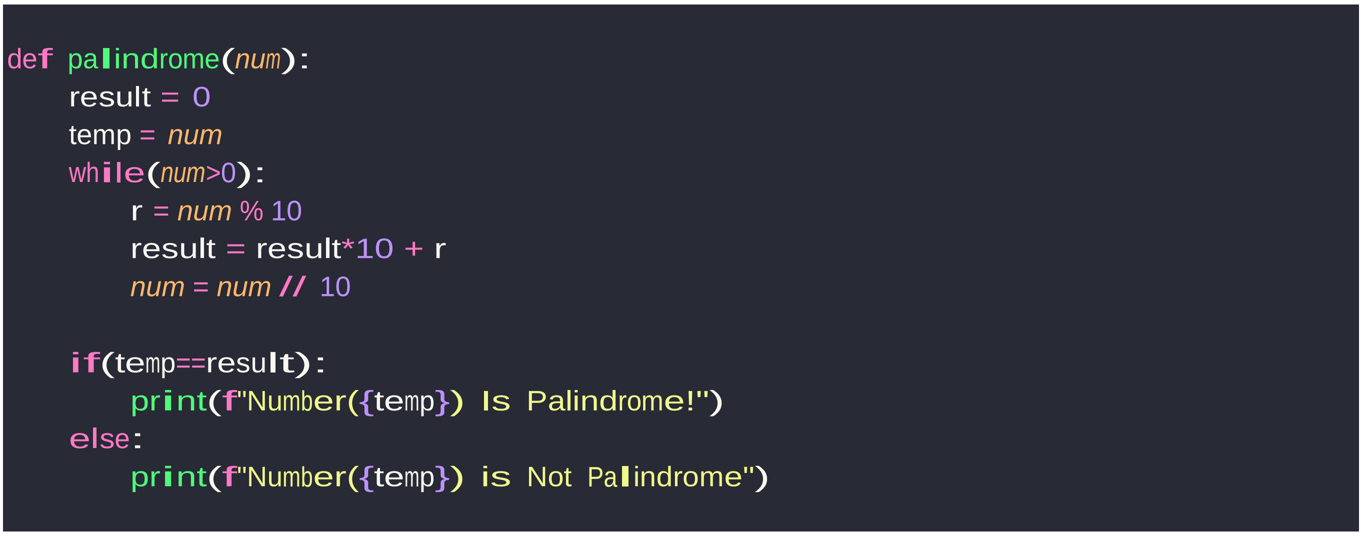
```
PS G:\Internship\Aakash Technologies\Day 4 (28th May 2021)\Programs>
 (28th May 2021)/Programs/identity_operators.py"


Identity Operators
x is y: False
x is not y: True
```

2. Functions and Modules:



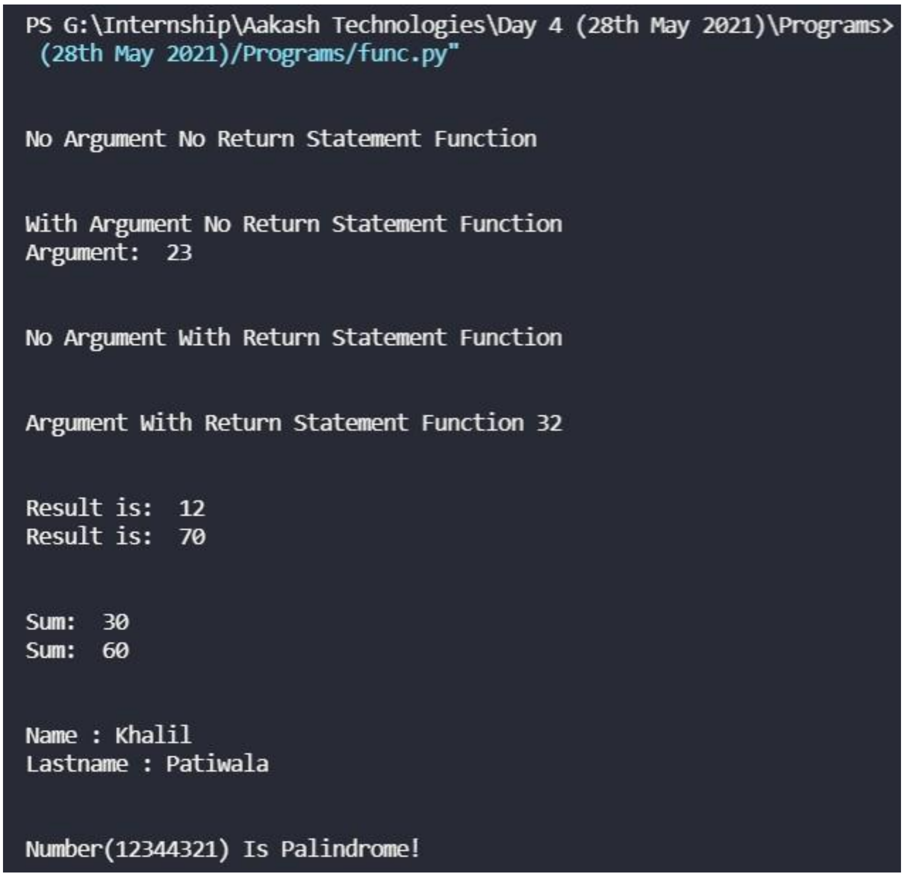
```
module import palindrome

def func1():
    print("No Argument No Return Statement Function")
func1()

def func2(num):
    print("With Argument No Return Statement Function")
    print("Argument: ", num)
func2(23)

def func3():
    return "No Argument With Return Statement Function"
var = func3()
print(var)

def func4(num):
    return "Argument With Return Statement Function", num
var2, var3 = func4(32)
print(var2, var3)

def func5(a=5, b=7):
    """ Function with Default Values """
    print("Result is: ",a+b)
func5()
func5(56, 14)

def func6(*num):
    sum = 0

    for n in num:
        sum = sum + n
    print("Sum: ", sum)
func6(10, 20)
func6(10, 20, 30)

def func7(**value):
    for i, j in value.items():
        print(f"{i} : {j}")
func7(Name = "Khalil", Lastname = "Patiwala")

# Imported Function:
palindrome(12344321)
```

Palindrome Function Present in module:



```
def palindrome(num):
    result = 0
    temp = num
    while(num>0):
        r = num % 10
        result = result*10 + r
        num = num // 10

    if(temp==result):
        print(f"Number({temp}) Is Palindrome!")
    else:
        print(f"Number({temp}) is Not Palindrome")
```

Output:

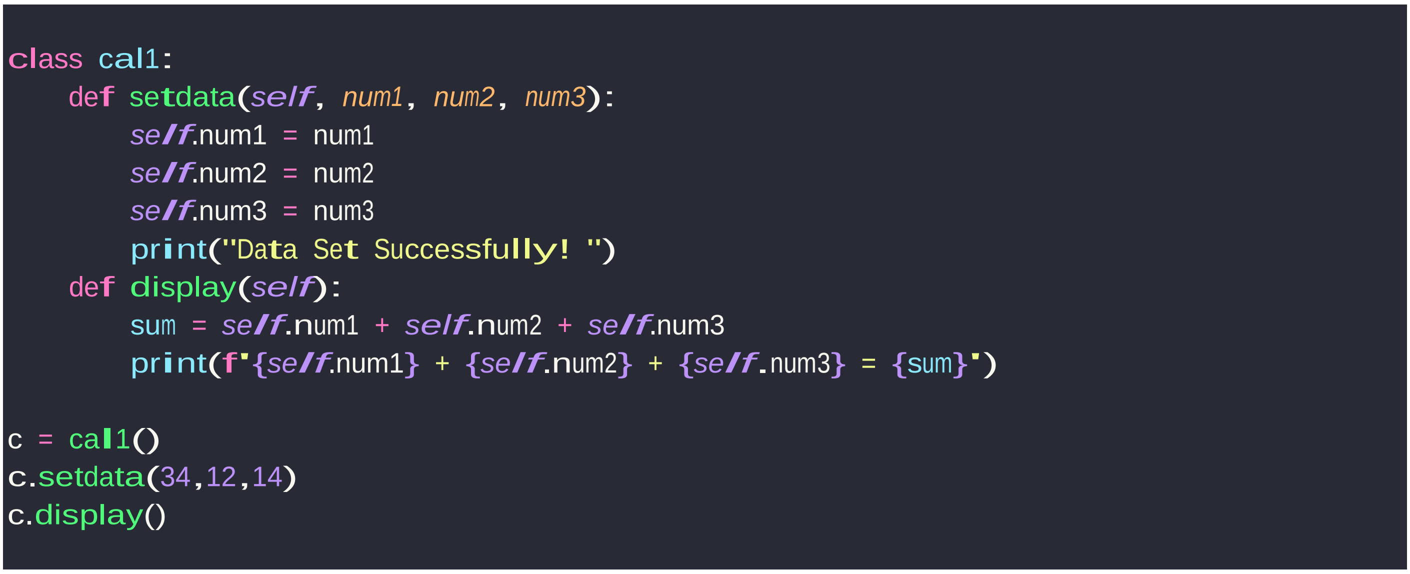


```
PS G:\Internship\Aakash Technologies\Day 4 (28th May 2021)\Programs>
 (28th May 2021)/Programs/func.py"


No Argument No Return Statement Function


With Argument No Return Statement Function
Argument:  23


No Argument With Return Statement Function


Argument With Return Statement Function 32


Result is:  12
Result is:  70


Sum:  30
Sum:  60


Name : Khalil
Lastname : Patiwala


Number(12344321) Is Palindrome!
```

TASK FOR DAY-5

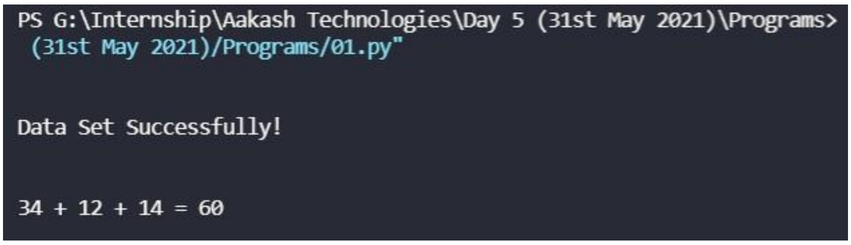
AIM: Solve the Following: (OOP-Python)

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.



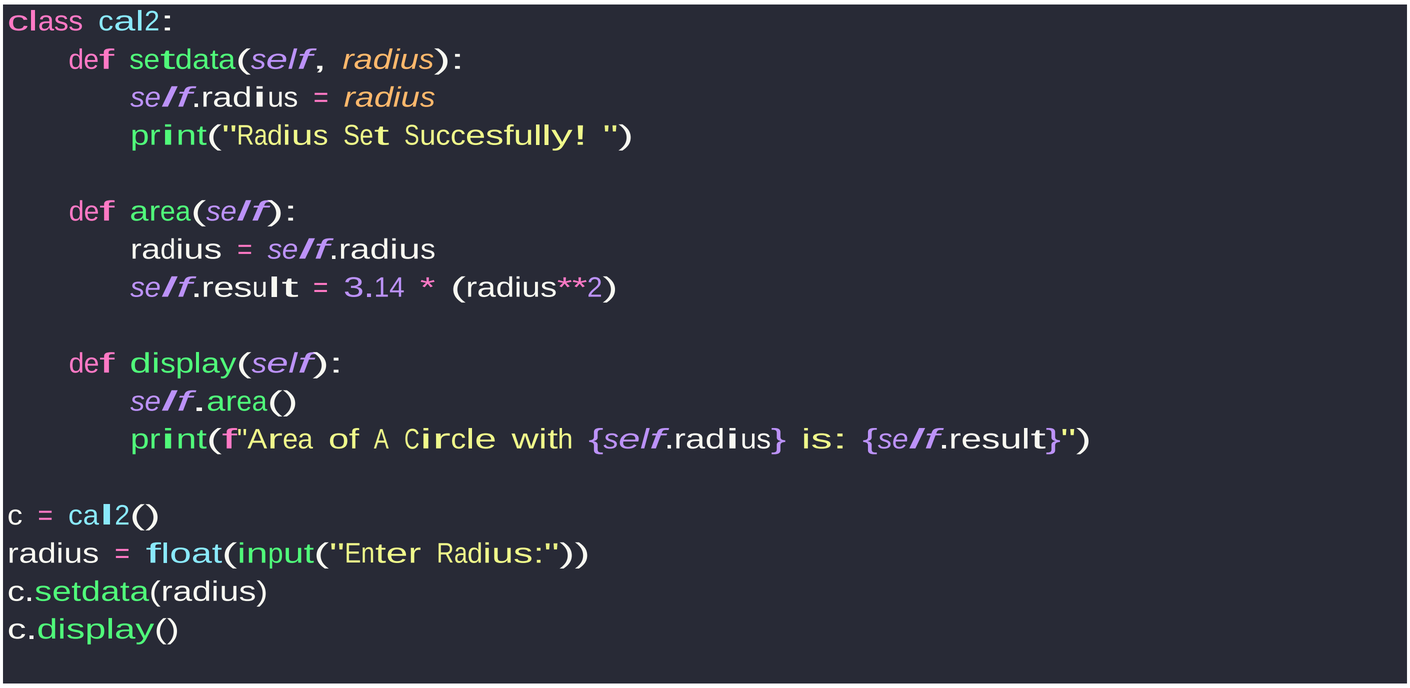
```
class cal1:
    def setdata(self, num1, num2, num3):
        self.num1 = num1
        self.num2 = num2
        self.num3 = num3
        print("Data Set Successfully! ")
    def display(self):
        sum = self.num1 + self.num2 + self.num3
        print(f"{self.num1} + {self.num2} + {self.num3} = {sum}")

c = cal1()
c.setdata(34,12,14)
c.display()
```

Output:



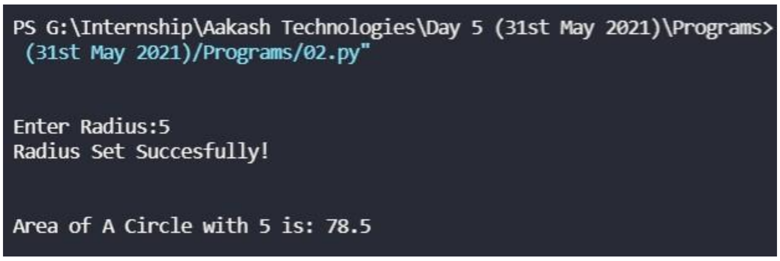
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/01.py"


Data Set Successfully!


34 + 12 + 14 = 60
```

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 .



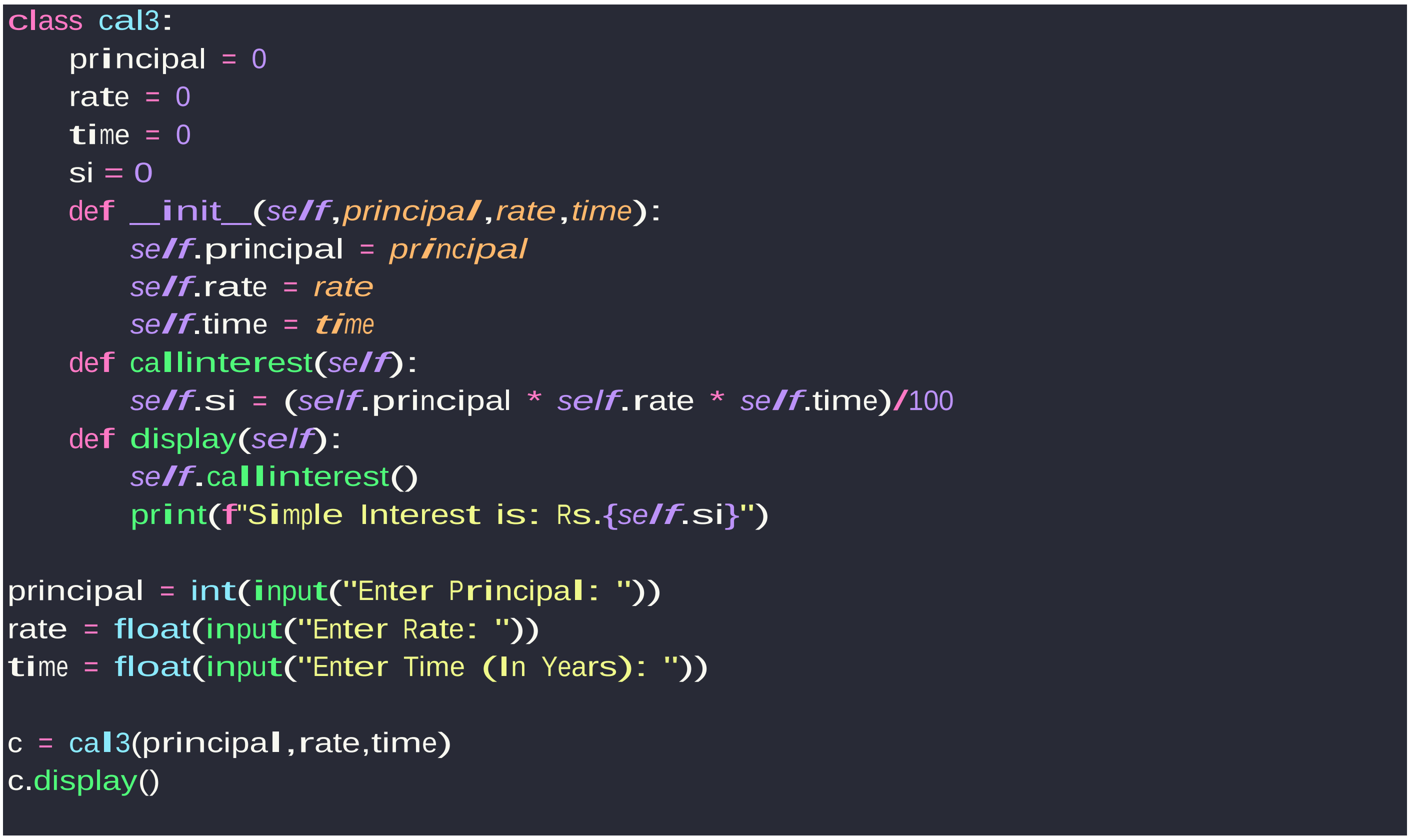
```
class cal2:
    def setdata(self, radius):
        self.radius = radius
        print("Radius Set Succesfully! ")

    def area(self):
        radius = self.radius
        self.result = 3.14 * (radius**2)

    def display(self):
        self.area()
        print(f"Area of A Circle with {self.radius} is: {self.result}")

c = cal2()
radius = float(input("Enter Radius:"))
c.setdata(radius)
c.display()
```

Output:



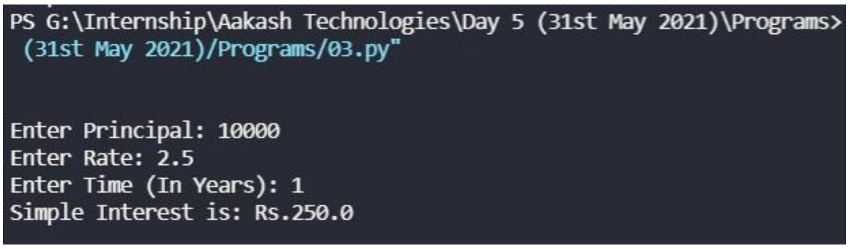
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/02.py"


Enter Radius:5
Radius Set Succesfully!


Area of A Circle with 5 is: 78.5
```

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



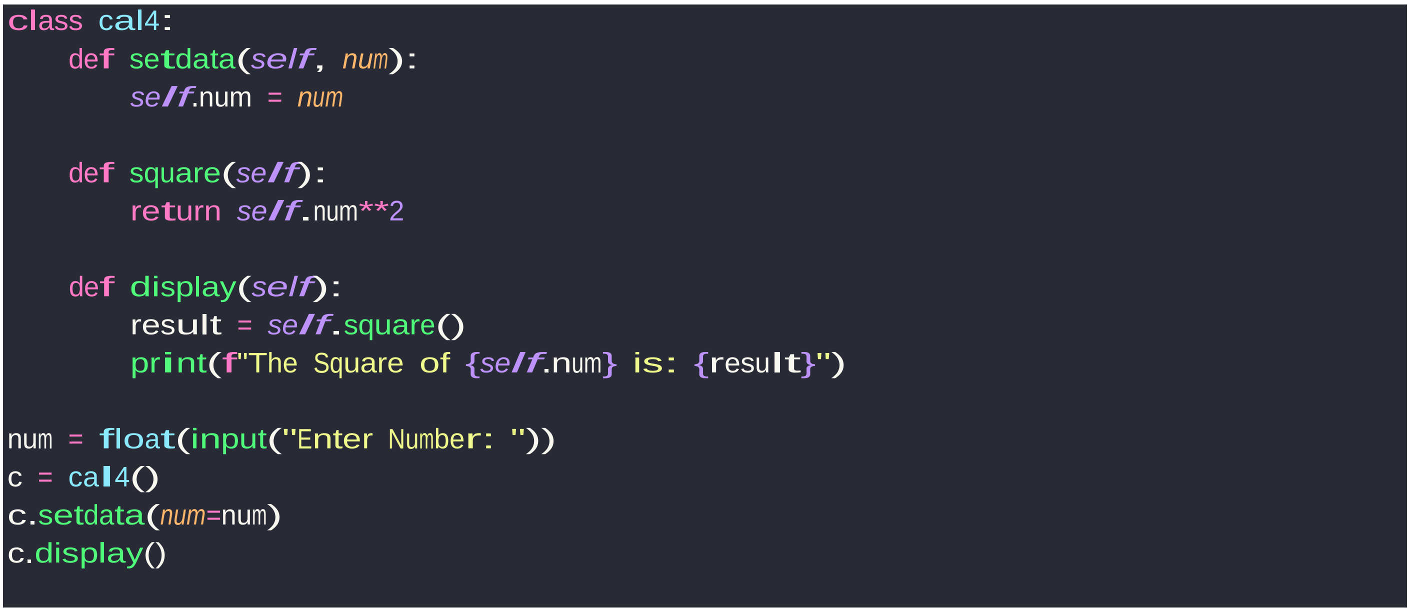
```
class cal3:
    principal = 0
    rate = 0
    time = 0
    si = 0
    def __init__(self,principal,rate,time):
        self.principal = principal
        self.rate = rate
        self.time = time
    def calInterest(self):
        self.si = (self.principal * self.rate * self.time)/100
    def display(self):
        self.calInterest()
        print(f"Simple Interest is: Rs.{self.si}")

principal = int(input("Enter Principal: "))
rate = float(input("Enter Rate: "))
time = float(input("Enter Time (In Years): "))

c = cal3(principal,rate,time)
c.display()
```

Output:



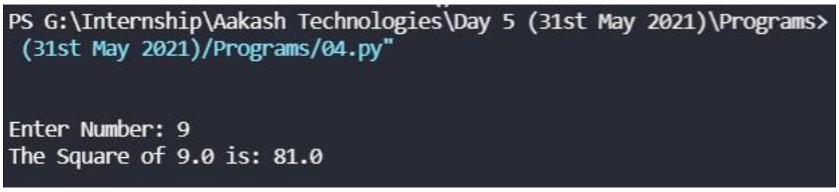
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/03.py"


Enter Principal: 10000
Enter Rate: 2.5
Enter Time (In Years): 1
Simple Interest is: Rs.250.0
```

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)



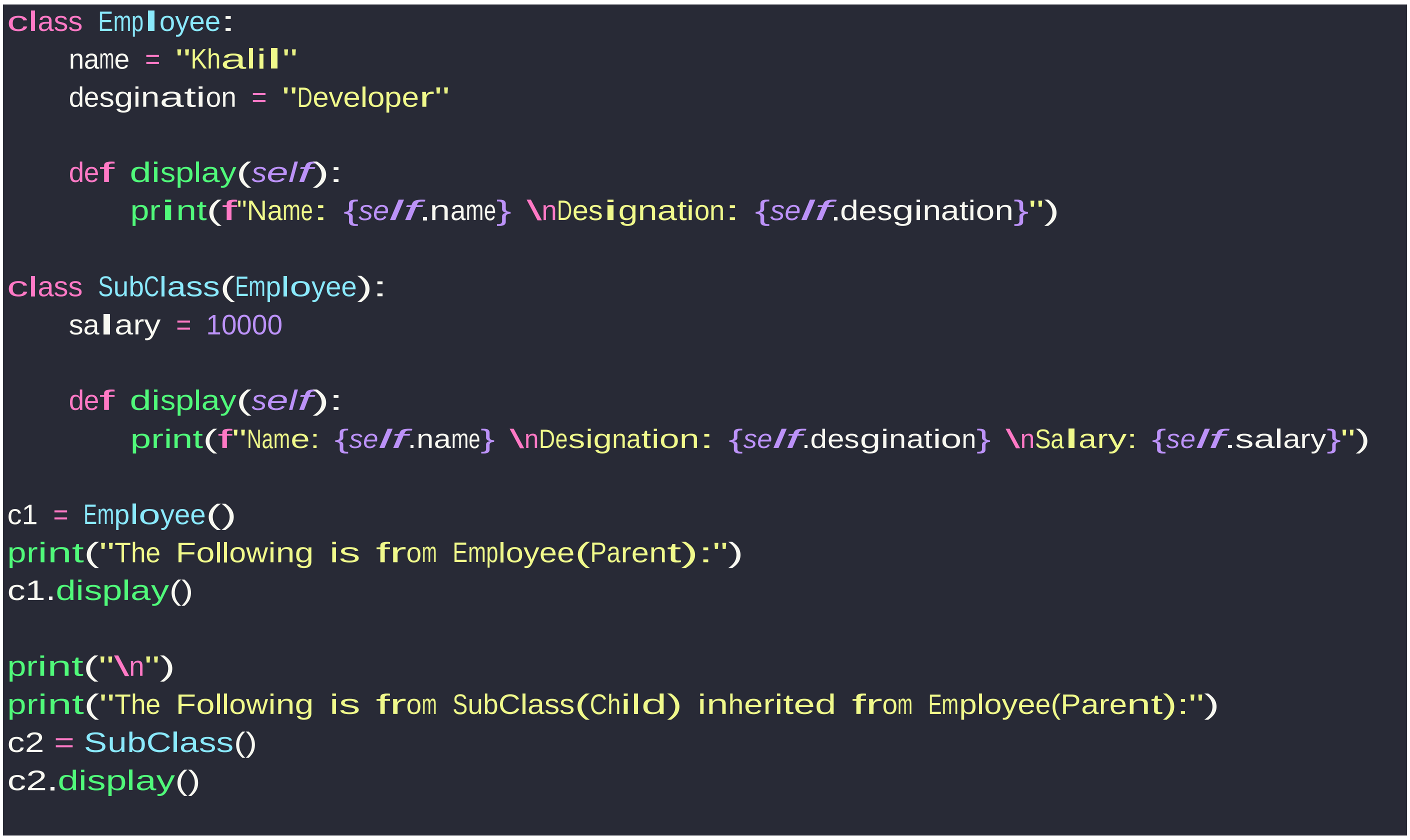
```
class cal4:
    def setdata(self, num):
        self.num = num

    def square(self):
        return self.num**2

    def display(self):
        result = self.square()
        print(f"The Square of {self.num} is: {result}")

num = float(input("Enter Number: "))
c = cal4()
c.setdata(num=num)
c.display()
```

Output:



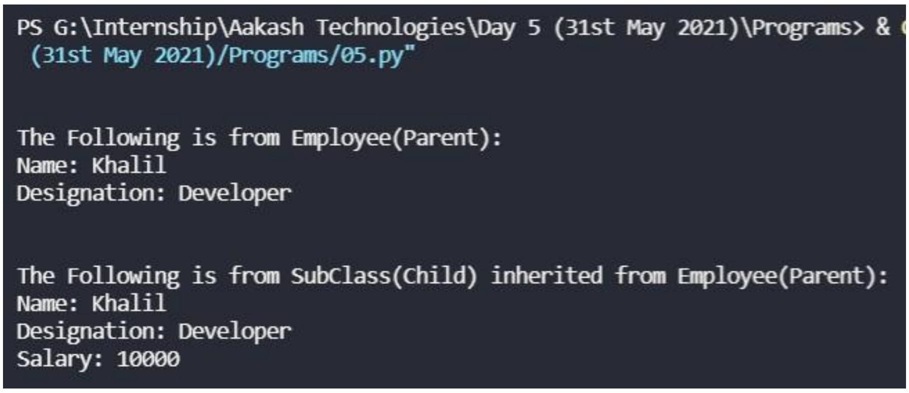
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/04.py"


Enter Number: 9
The Square of 9.0 is: 81.0
```

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.



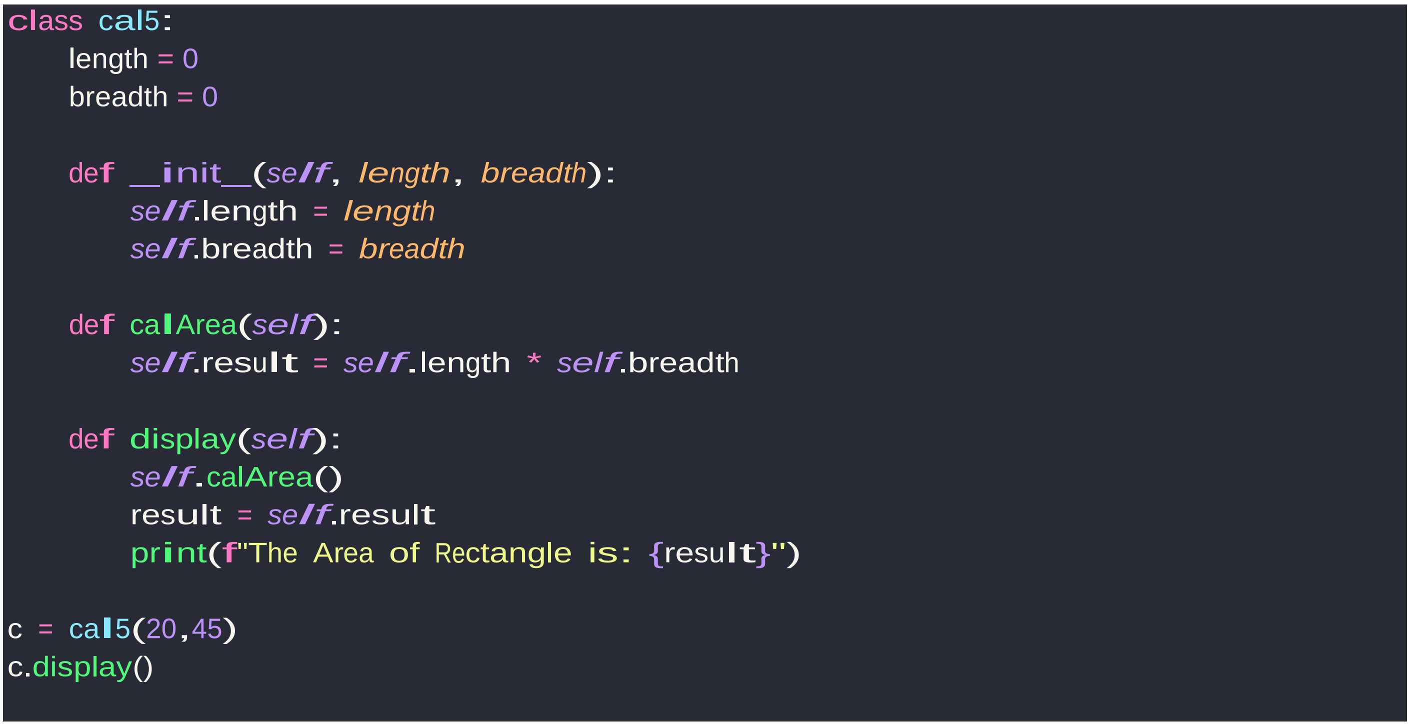
```
class Employee:
    name = "Khalil"
    desgination = "Developer"

    def display(self):
        print(f"Name: {self.name} \nDesignation: {self.desgination}")

class SubClass(Employee):
    salary = 10000

    def display(self):
        print(f"Name: {self.name} \nDesignation: {self.desgination} \nSalary: {self.salary}")

c1 = Employee()
print("The Following is from Employee(Parent):")
c1.display()

print("\n")
print("The Following is from SubClass(Child) inherited from Employee(Parent):")
c2 = SubClass()
c2.display()
```

Output:



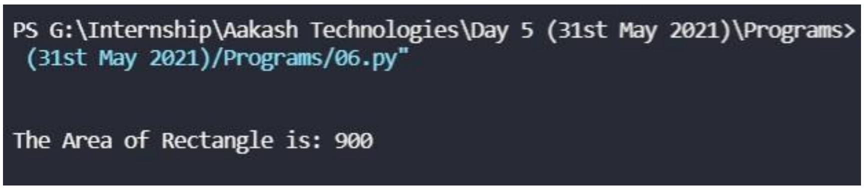
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs> &
 (31st May 2021)/Programs/05.py"


The Following is from Employee(Parent):
Name: Khalil
Designation: Developer


The Following is from SubClass(Child) inherited from Employee(Parent):
Name: Khalil
Designation: Developer
Salary: 10000
```

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.



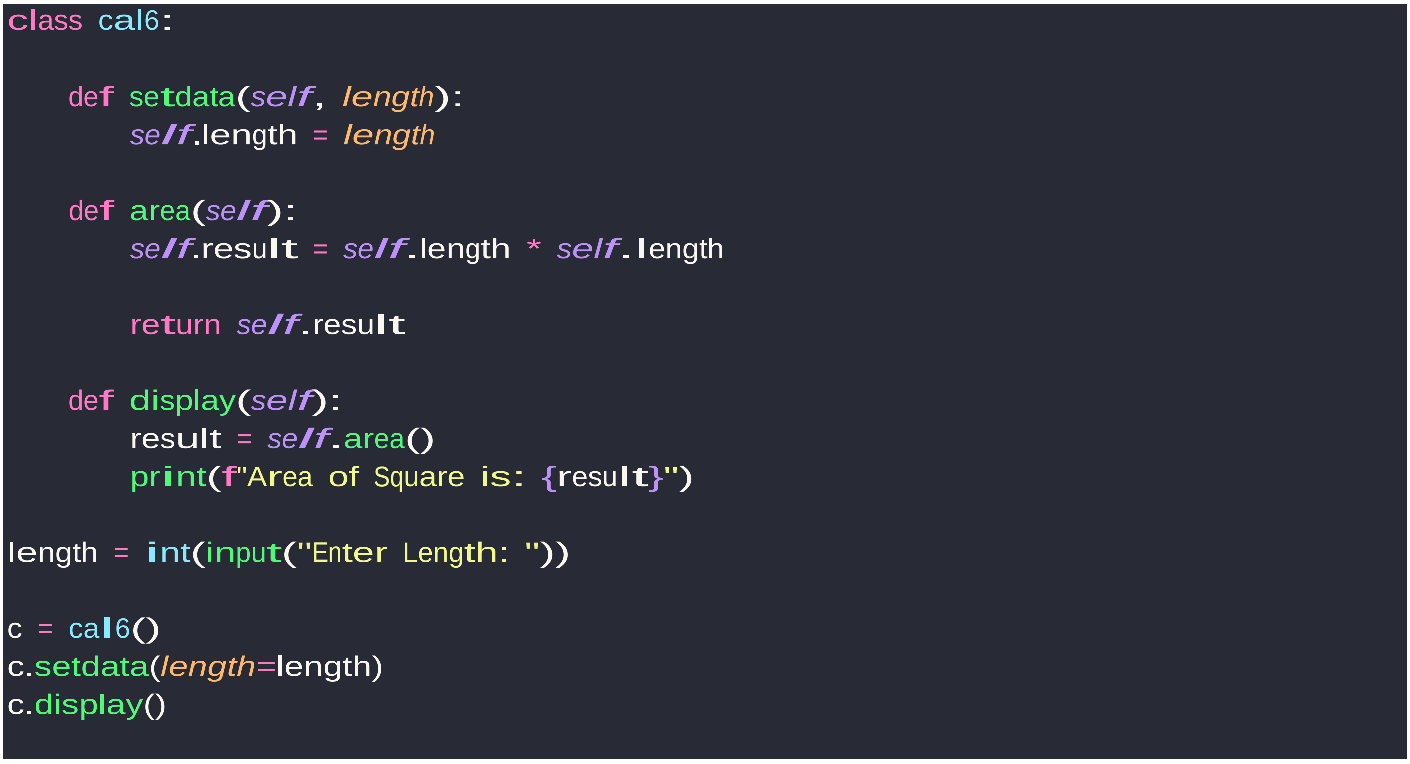
```
class cal5:
    length = 0
    breadth = 0

    def __init__(self, length, breadth):
        self.length = length
        self.breadth = breadth

    def calArea(self):
        self.result = self.length * self.breadth

    def display(self):
        self.calArea()
        result = self.result
        print(f"The Area of Rectangle is: {result}")

c = cal5(20,45)
c.display()
```

Output:



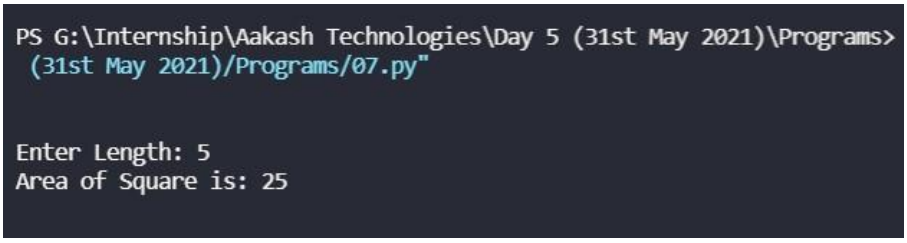
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/06.py"


The Area of Rectangle is: 900
```

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 .



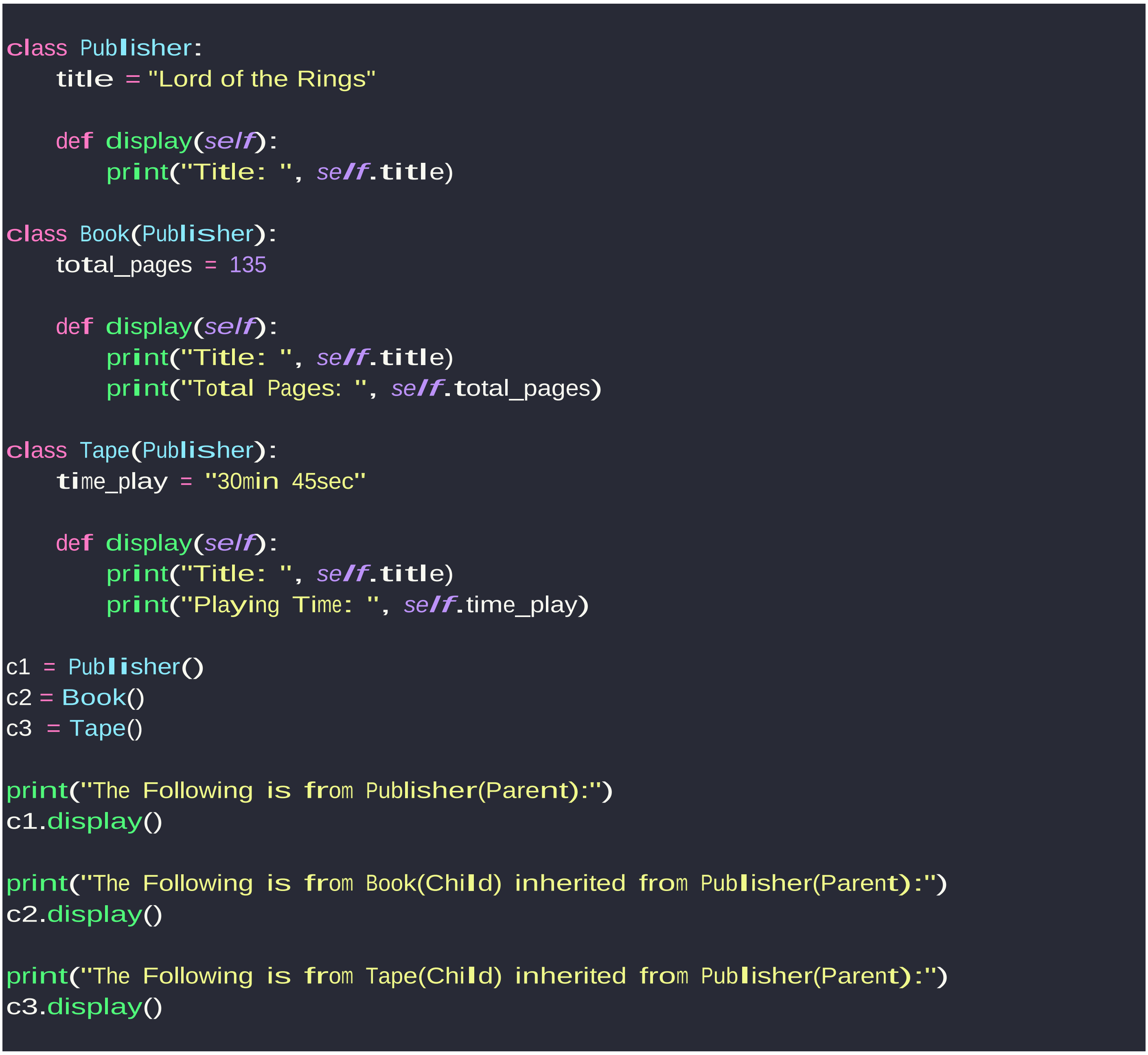
```
class cal6:

    def setdata(self, length):
        self.length = length

    def area(self):
        self.result = self.length * self.length

        return self.result

    def display(self):
        result = self.area()
        print(f"Area of Square is: {result}")

length = int(input("Enter Length: "))

c = cal6()
c.setdata(length=length)
c.display()
```

Output:



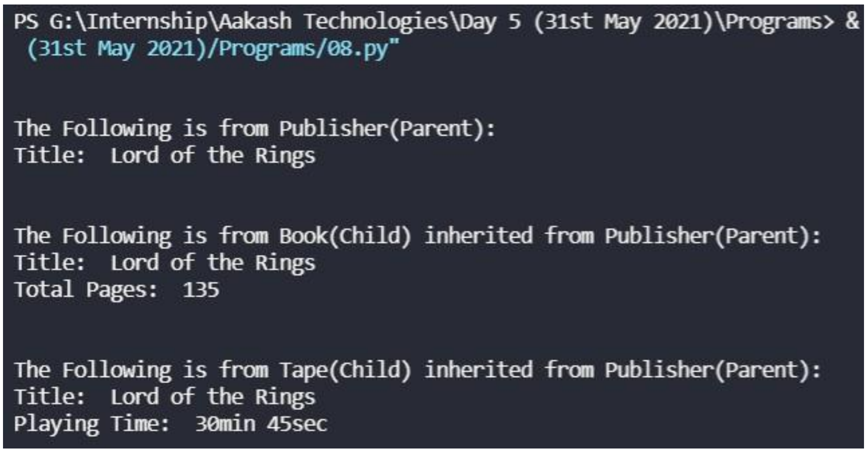
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/07.py"


Enter Length: 5
Area of Square is: 25
```

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.



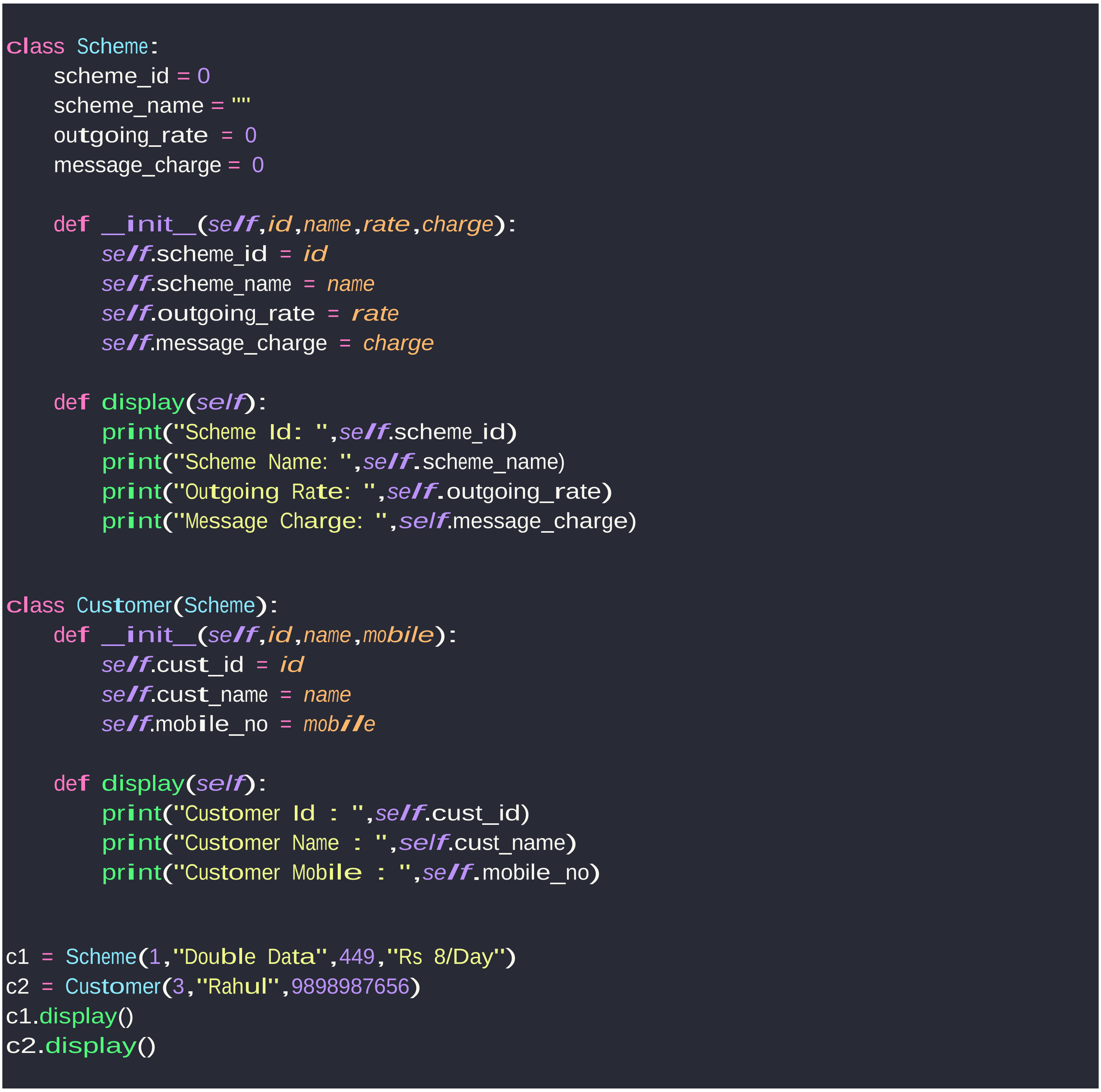
```
class Publisher:
    title = "Lord of the Rings"

    def display(self):
        print("Title: ", self.title)

class Book(Publisher):
    total_pages = 135

    def display(self):
        print("Title: ", self.title)
        print("Total Pages: ", self.total_pages)

class Tape(Publisher):
    time_play = "30min 45sec"

    def display(self):
        print("Title: ", self.title)
        print("Playing Time: ", self.time_play)

c1  = Publisher()
c2 = Book()
c3  = Tape()

print("The Following is from Publisher(Parent):")
c1.display()

print("The Following is from Book(Child) inherited from Publisher(Parent):")
c2.display()

print("The Following is from Tape(Child) inherited from Publisher(Parent):")
c3.display()
```

Output:



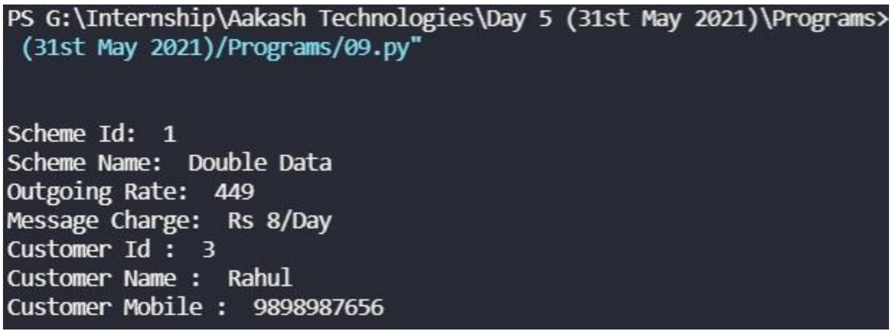
```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs> &
 (31st May 2021)/Programs/08.py"


The Following is from Publisher(Parent):
Title:  Lord of the Rings


The Following is from Book(Child) inherited from Publisher(Parent):
Title:  Lord of the Rings
Total Pages:  135


The Following is from Tape(Child) inherited from Publisher(Parent):
Title:  Lord of the Rings
Playing Time:  30min 45sec
```

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.



```
class Scheme:
    scheme_id = 0
    scheme_name = ""
    outgoing_rate = 0
    message_charge = 0

    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):
    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)


c1 = Scheme(1,"Double Data",449,"Rs 8/Day")
c2 = Customer(3,"Rahul",9898987656)
c1.display()
c2.display()
```

Output:

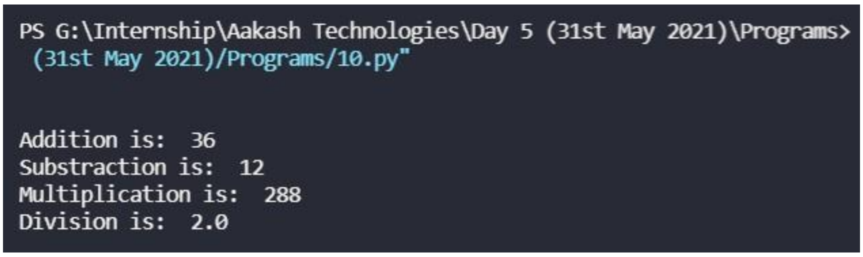


```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/09.py"


Scheme Id:  1
Scheme Name:  Double Data
Outgoing Rate:  449
Message Charge:  Rs 8/Day
Customer Id :  3
Customer Name :  Rahul
Customer Mobile :  9898987656
```

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.

```
class arith:

    def __init__(self, num1, num2):
        self.num1 = num1
        self.num2 = num2

    def add(self):
        result = self.num1 + self.num2
        return result

    def subtract(self):
        result = self.num1 - self.num2
        return result

    def multiply(self):
        result = self.num1 * self.num2
        return  result

    def divide(self):
        result = self.num1 / self.num2
        return  result

c = arith(24,12)

print("Addition is: ",c.add())
print("Substraction is: ", c.subtract())
print("Multiplication is: ", c.multiply())
print("Division is: ", c.divide())
```

Output:

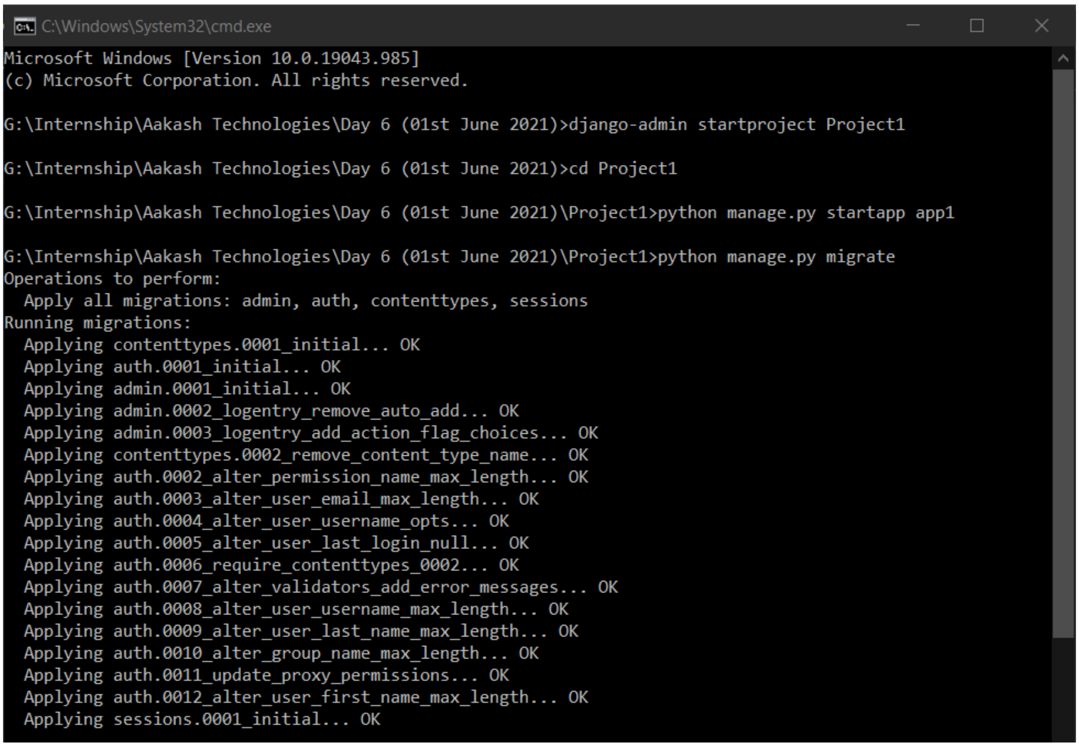


```
PS G:\Internship\Aakash Technologies\Day 5 (31st May 2021)\Programs>
 (31st May 2021)/Programs/10.py"


Addition is:  36
Substraction is:  12
Multiplication is:  288
Division is:  2.0
```

TASK FOR DAY-6

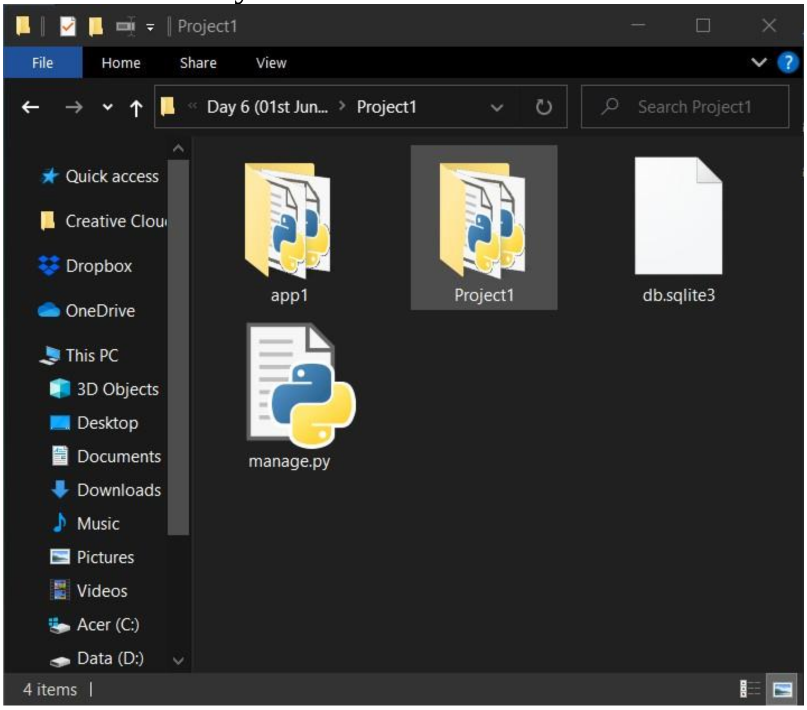
AIM: Install Django, Run and Display Browser Window.

Necessary Command before running:



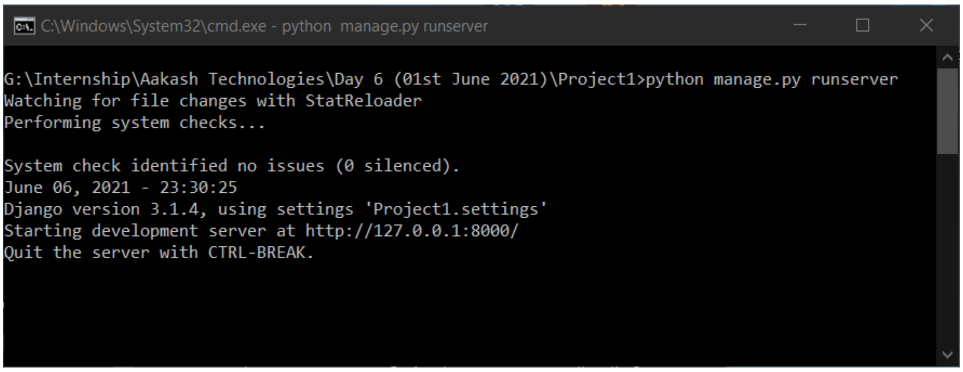
```
Microsoft Windows [Version 10.0.19043.985]
(c) Microsoft Corporation. All rights reserved.

G:\Internship\Aakash Technologies\Day 6 (01st June 2021)>django-admin startproject Project1

G:\Internship\Aakash Technologies\Day 6 (01st June 2021)>cd Project1

G:\Internship\Aakash Technologies\Day 6 (01st June 2021)\Project1>python manage.py startapp app1

G:\Internship\Aakash Technologies\Day 6 (01st June 2021)\Project1>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
```

Folder Directory:



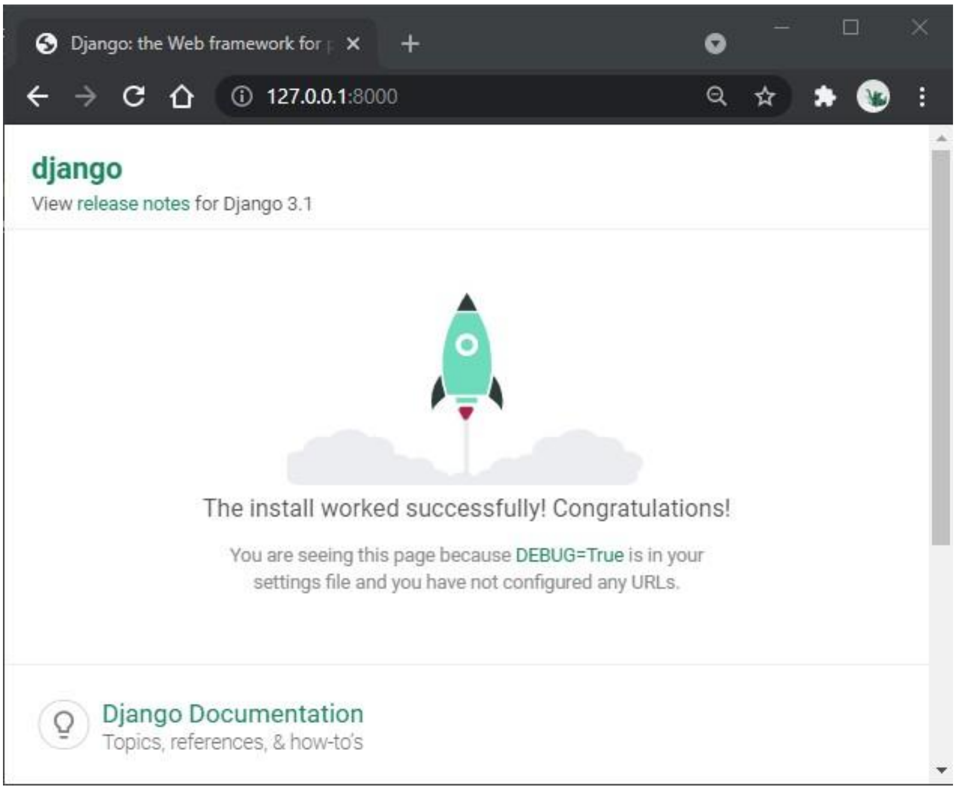
Running the following command:

Python manage.py runserver



```
G:\Internship\Aakash Technologies\Day 6 (01st June 2021)\Project1>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
June 06, 2021 - 23:30:25
Django version 3.1.4, using settings 'Project1.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Browser Window:



Django: the Web framework for p
127.0.0.1:8000
django
View release notes for Django 3.1

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.

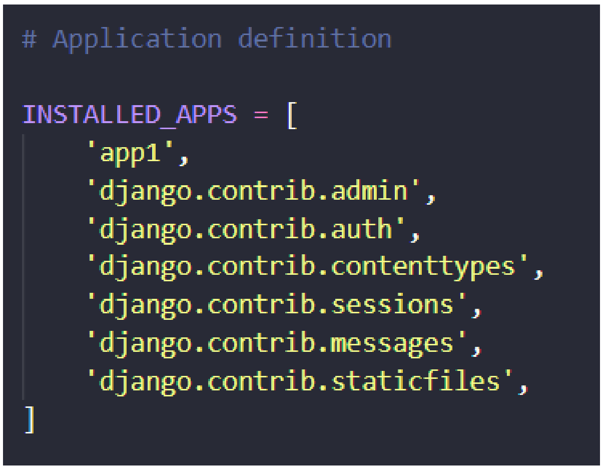
Django Documentation
Topics, references, & how-to's

TASK FOR DAY-7

AIM: Display Hello World Text in Browser in Django

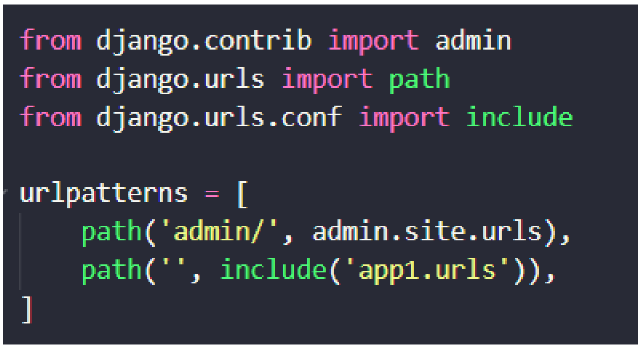
After Creating Project and Creating an App, Make the following changes:

INSTALLED_APPS in settings.py of Project1:



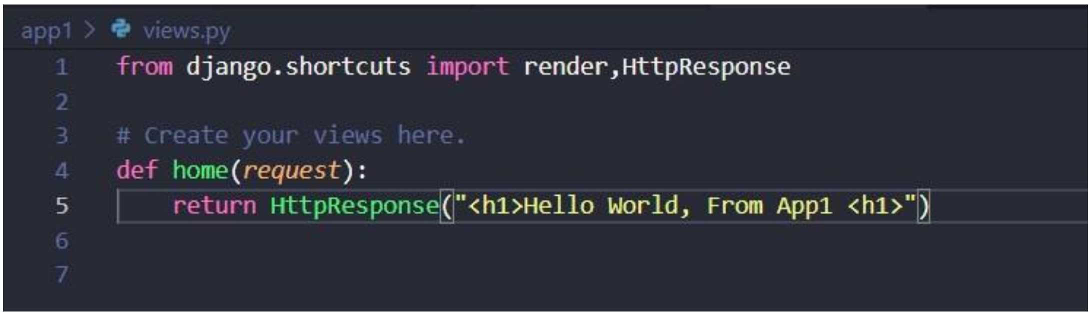
```
# Application definition

INSTALLED_APPS = [
    'app1',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
```

urls.py of Project1:



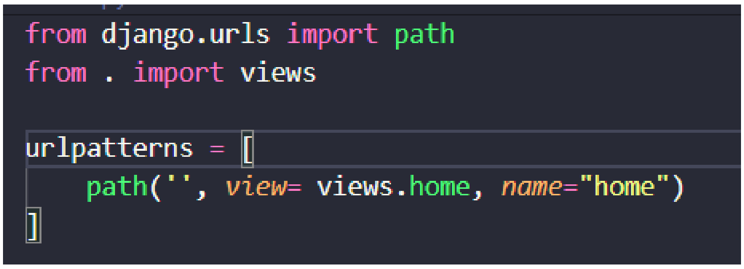
```
from django.contrib import admin
from django.urls import path
from django.urls.conf import include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('app1.urls')),
]
```

views.py in app1:



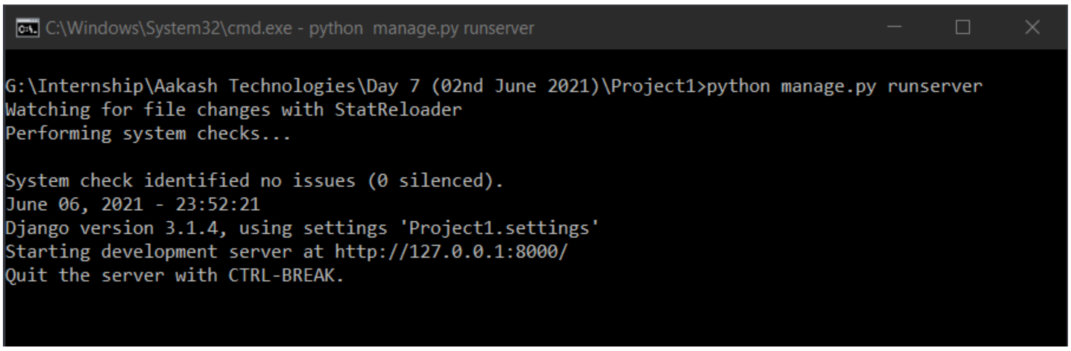
```
from django.shortcuts import render,HttpResponse

# Create your views here.
def home(request):
    return HttpResponse("<h1>Hello World, From App1 <h1>")


```

urls.py of app1:



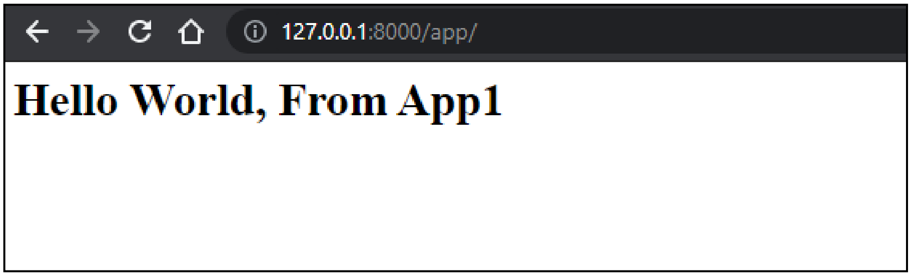
```
from django.urls import path
from . import views

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

Server:



```
G:\Internship\Aakash Technologies\Day 7 (02nd June 2021)\Project1>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
June 06, 2021 - 23:52:21
Django version 3.1.4, using settings 'Project1.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Output:

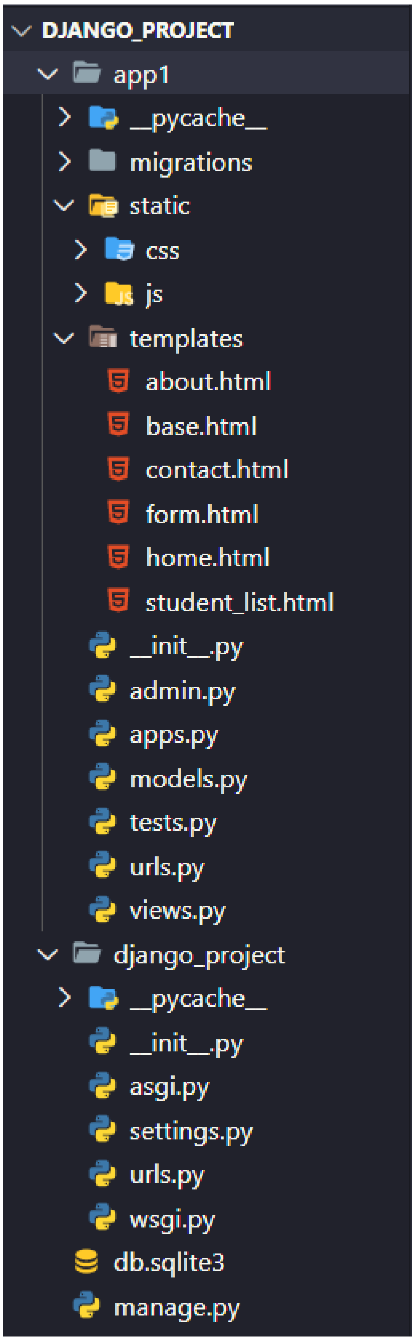


Hello World, From App1

TASK FOR DAY-8

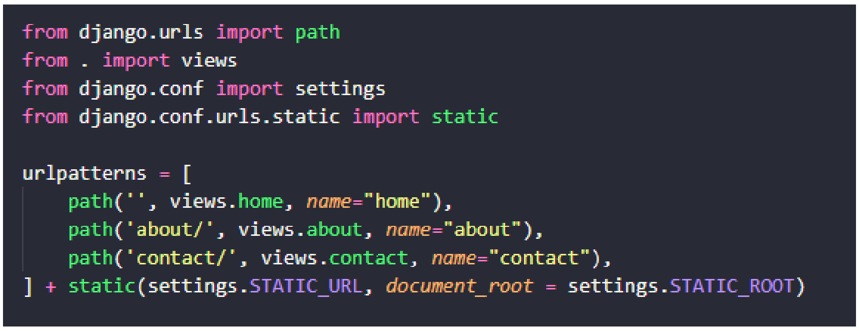
Task: Template Integration and URL Routing

If you have templates with static files then arrange as shown below:



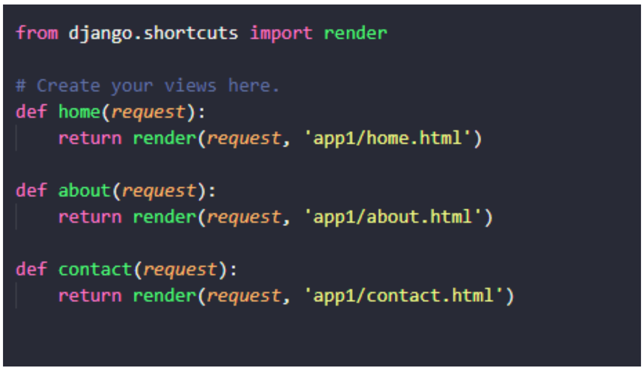
```
DJANGO_PROJECT
  app1
    __pycache__
    migrations
    static
      css
      js
    templates
      about.html
      base.html
      contact.html
      form.html
      home.html
      student_list.html
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  django_project
    __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  db.sqlite3
  manage.py
```

For URL Routing: (in urls.py of app1)



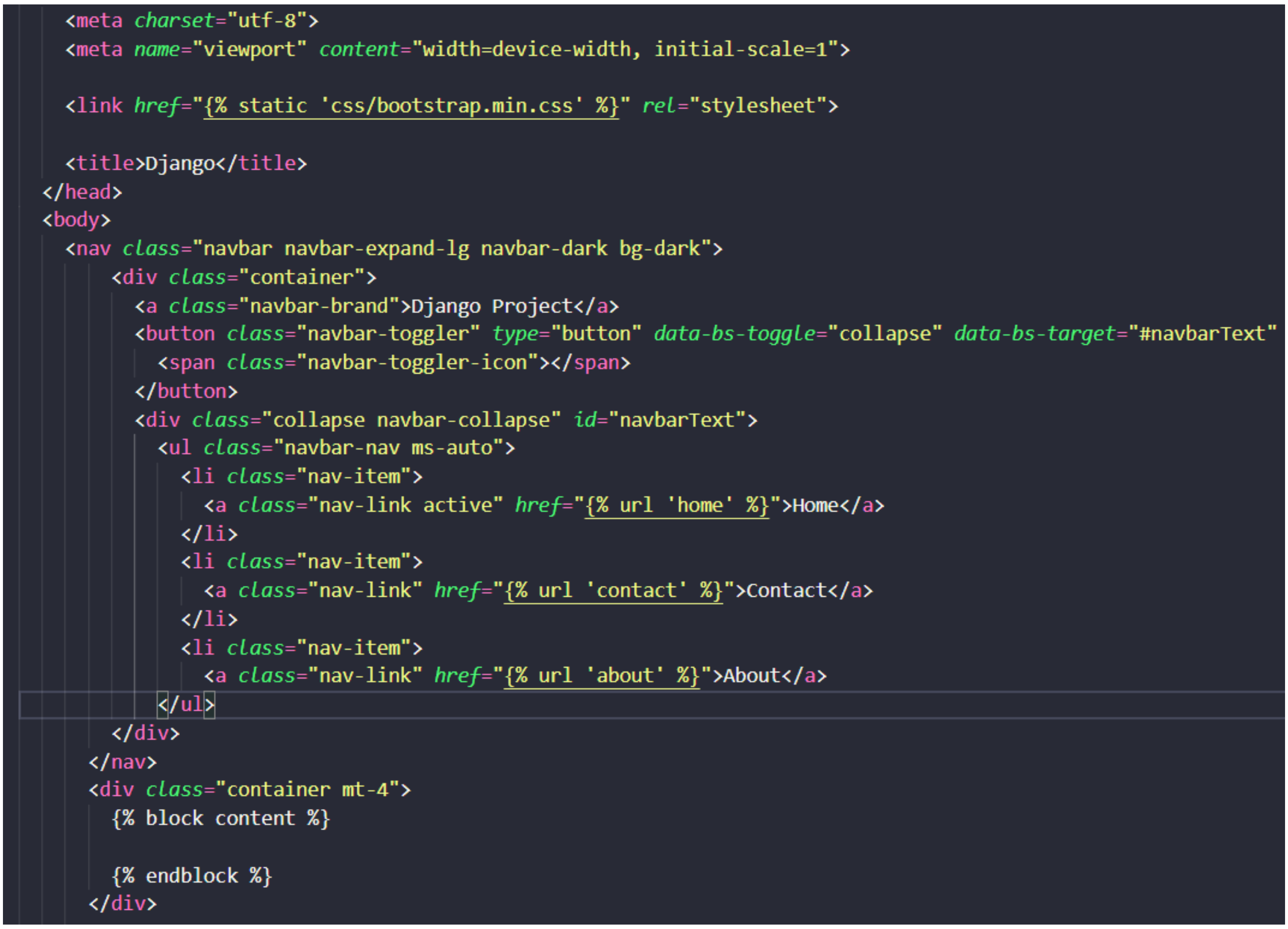
```
from django.urls import path
from . import views
from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('', views.home, name="home"),
    path('about/', views.about, name="about"),
    path('contact/', views.contact, name="contact"),
] + static(settings.STATIC_URL, document_root = settings.STATIC_ROOT)
```

views.py in app1:



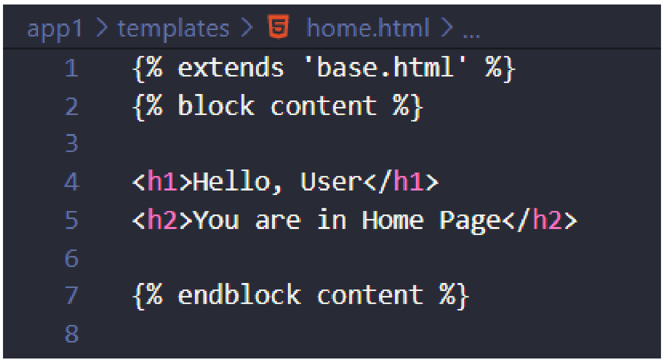
```python
from django.shortcuts import render

# Create your views here.
def home(request):
    return render(request, 'app1/home.html')

def about(request):
    return render(request, 'app1/about.html')

def contact(request):
    return render(request, 'app1/contact.html')
```

Implementing the Static files as follows: (base.html)



```html
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link href="{% static 'css/bootstrap.min.css' %}" rel="stylesheet">

    <title>Django</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <div class="container">
          <a class="navbar-brand">Django Project</a>
          <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarText"
            <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse" id="navbarText">
            <ul class="navbar-nav ms-auto">
              <li class="nav-item">
                <a class="nav-link active" href="{% url 'home' %}">Home</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="{% url 'contact' %}">Contact</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="{% url 'about' %}">About</a>
            </ul>
        </div>
      </nav>
      <div class="container mt-4">
        {% block content %}

        {% endblock %}
      </div>
```

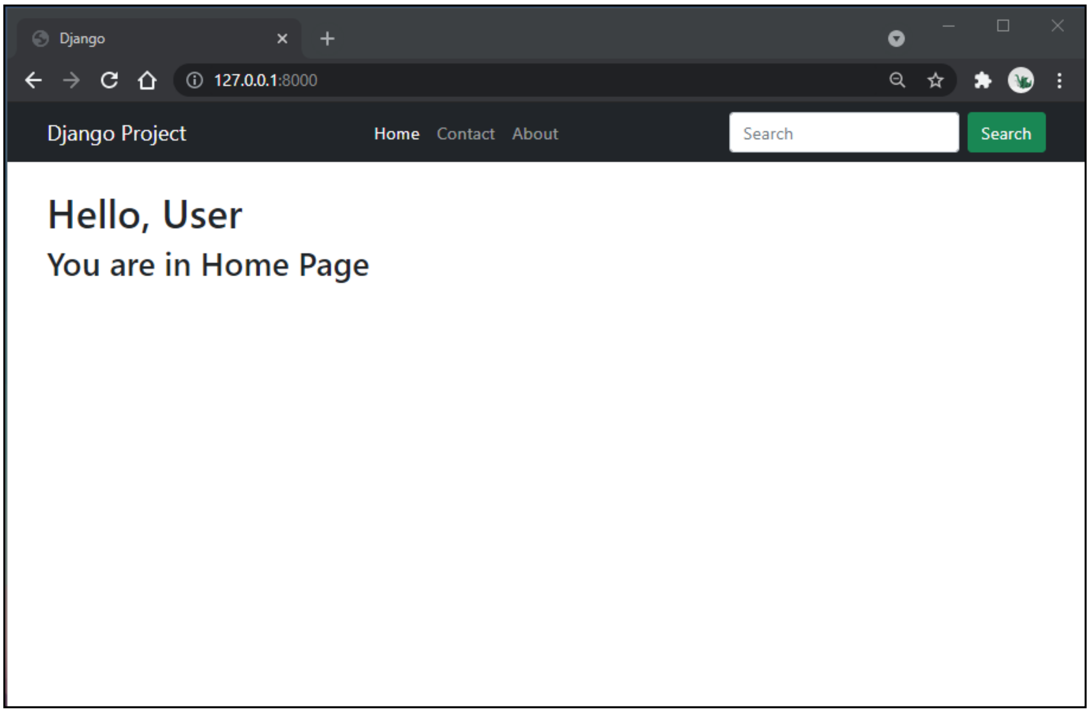
Extending base.html to contact.html and many other html files



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

<h1>Hello, User</h1>
<h2>You are in Home Page</h2>

{% endblock content %}

```

Output:

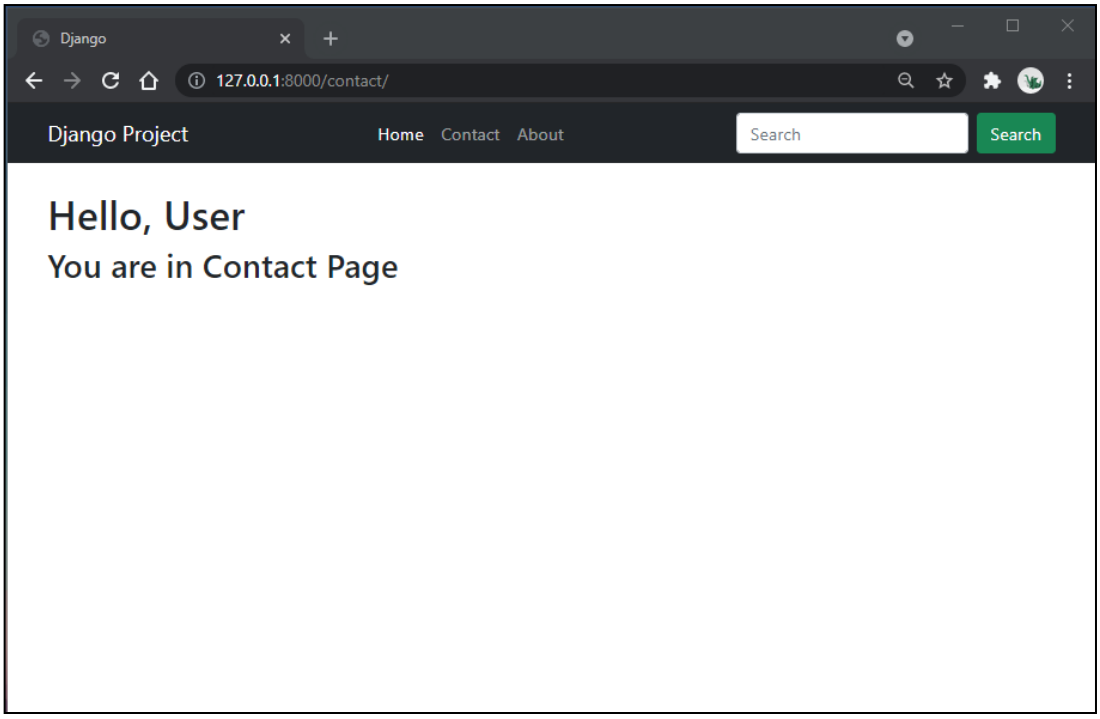
home.html



Hello, User

You are in Home Page

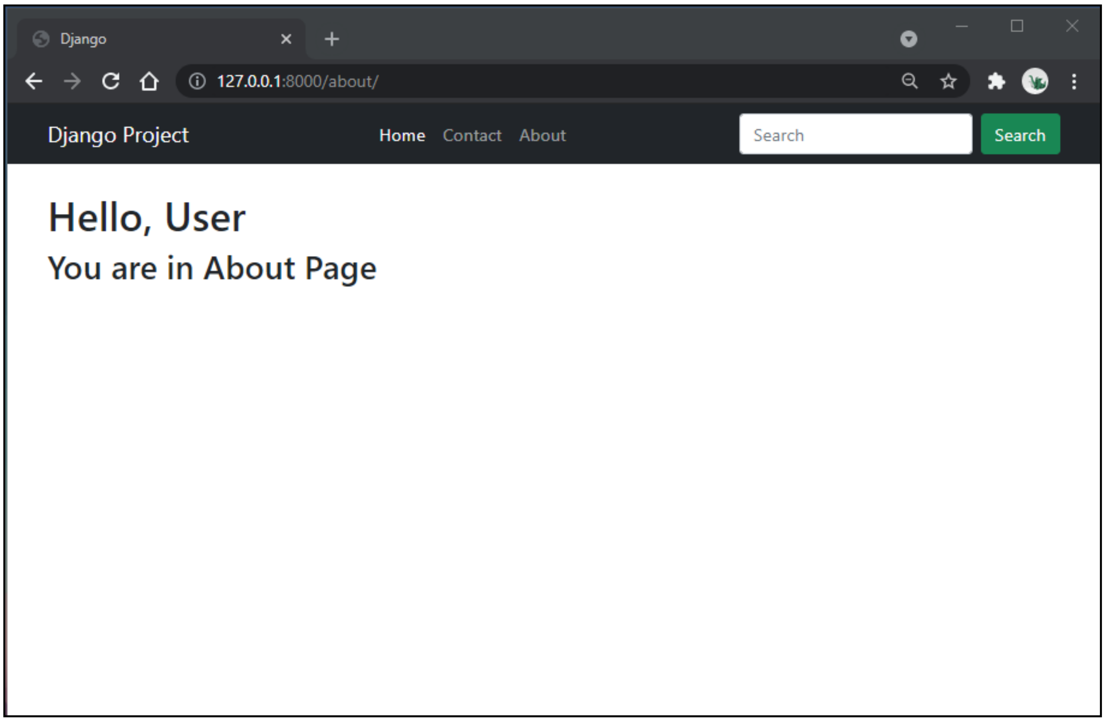
contact.html



Hello, User

You are in Contact Page

about.html



Hello, User

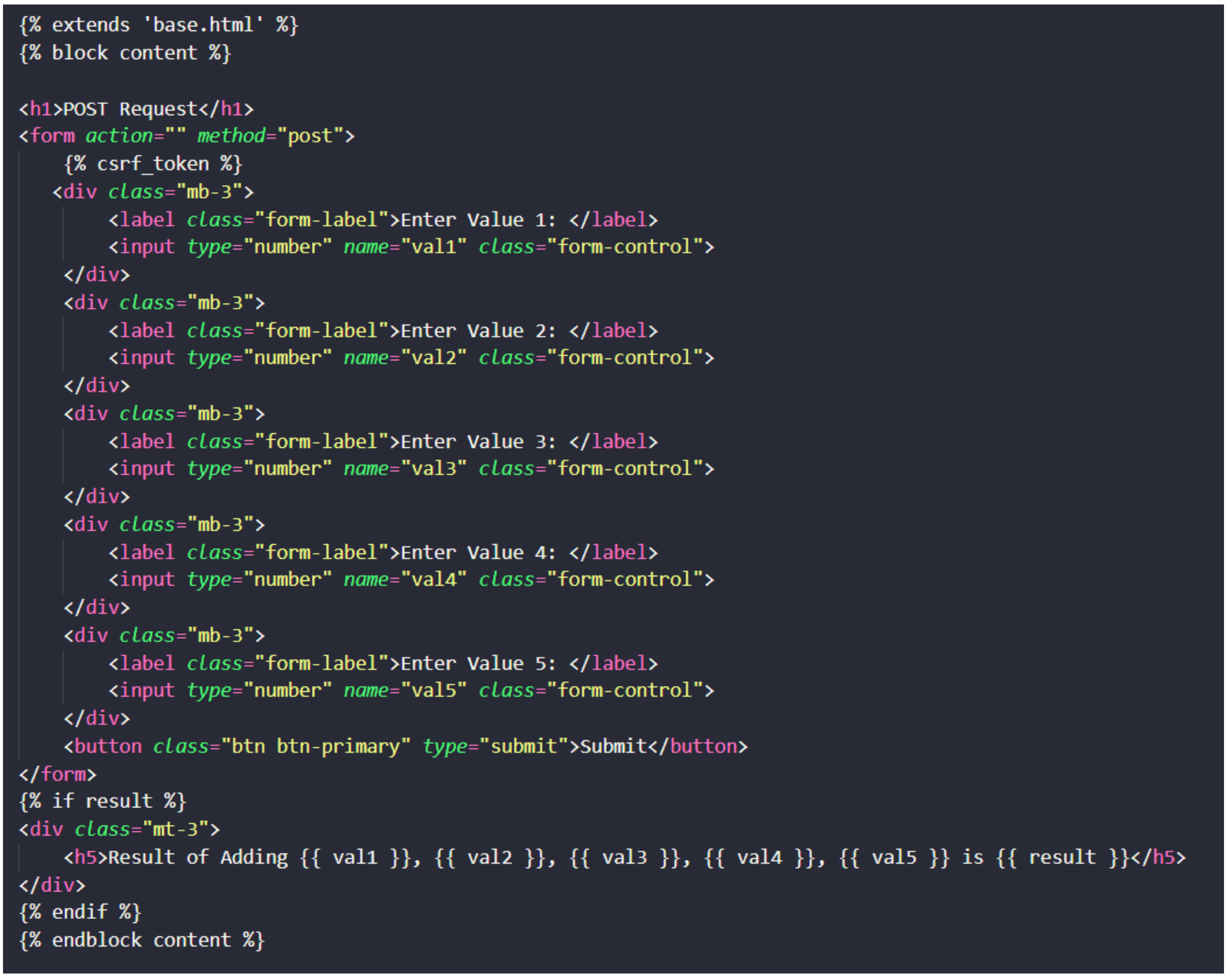
You are in About Page

TASK FOR DAY-9

AIM: Using POST method to get User Values and returning back on page.

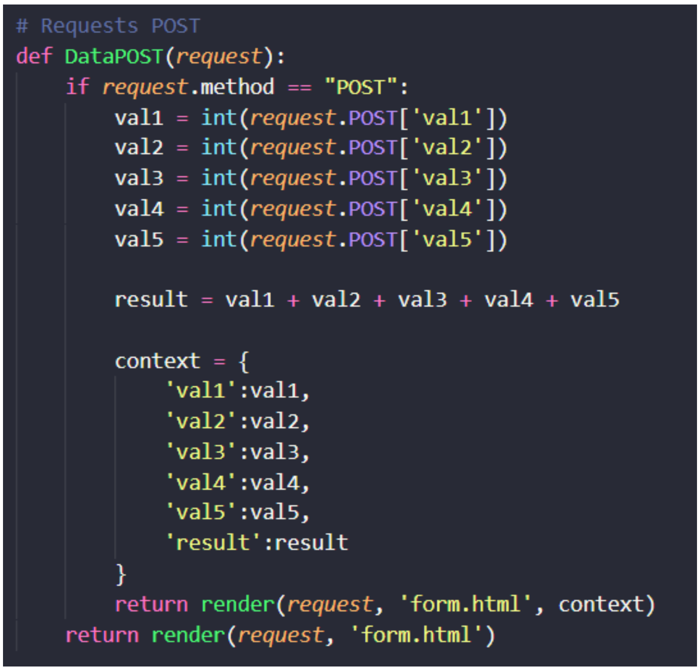
Here I’m going to extend the Project I did on Day 8

form.html



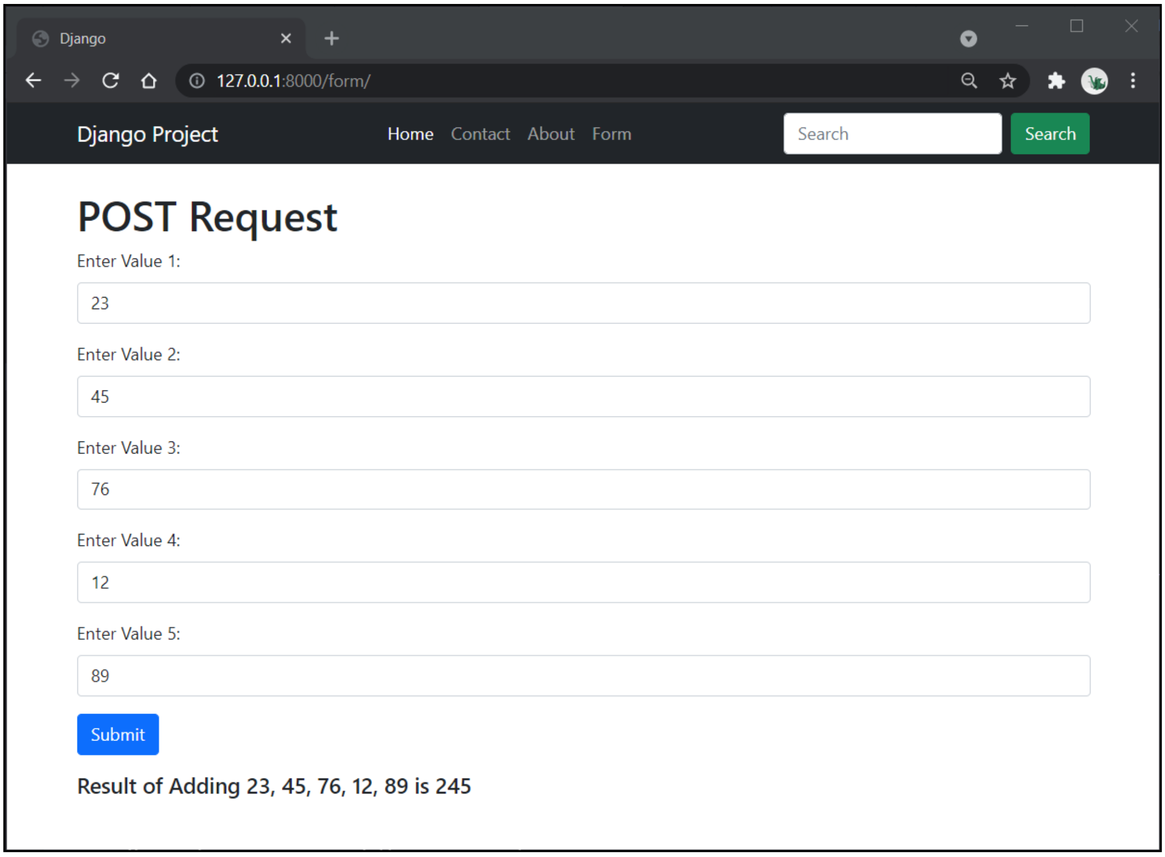
```
{% extends 'base.html' %}
{% block content %}

<h1>POST Request</h1>
<form action="" method="post">
    {% csrf_token %}
   <div class="mb-3">
        <label class="form-label">Enter Value 1: </label>
        <input type="number" name="val1" class="form-control">
    </div>
    <div class="mb-3">
        <label class="form-label">Enter Value 2: </label>
        <input type="number" name="val2" class="form-control">
    </div>
    <div class="mb-3">
        <label class="form-label">Enter Value 3: </label>
        <input type="number" name="val3" class="form-control">
    </div>
    <div class="mb-3">
        <label class="form-label">Enter Value 4: </label>
        <input type="number" name="val4" class="form-control">
    </div>
    <div class="mb-3">
        <label class="form-label">Enter Value 5: </label>
        <input type="number" name="val5" class="form-control">
    </div>
    <button class="btn btn-primary" type="submit">Submit</button>
</form>
{% if result %}
<div class="mt-3">
    <h5>Result of Adding {{ val1 }}, {{ val2 }}, {{ val3 }}, {{ val4 }}, {{ val5 }} is {{ result }}</h5>
</div>
{% endif %}
{% endblock content %}
```

views.py in app1:



```python
# Requests POST
def DataPOST(request):
    if request.method == "POST":
        val1 = int(request.POST['val1'])
        val2 = int(request.POST['val2'])
        val3 = int(request.POST['val3'])
        val4 = int(request.POST['val4'])
        val5 = int(request.POST['val5'])

        result = val1 + val2 + val3 + val4 + val5

        context = {
            'val1':val1,
            'val2':val2,
            'val3':val3,
            'val4':val4,
            'val5':val5,
            'result':result
        }
        return render(request, 'form.html', context)
    return render(request, 'form.html')
```

Output:



POST Request

Enter Value 1:

23

Enter Value 2:

45

Enter Value 3:

76

Enter Value 4:

12

Enter Value 5:

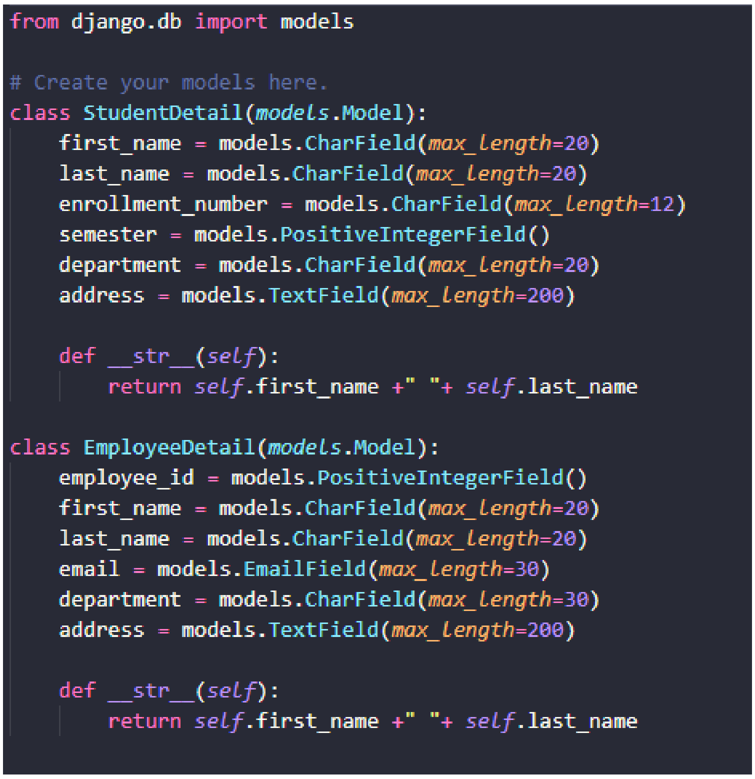
89

Result of Adding 23, 45, 76, 12, 89 is 245

TASK FOR DAY-10

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

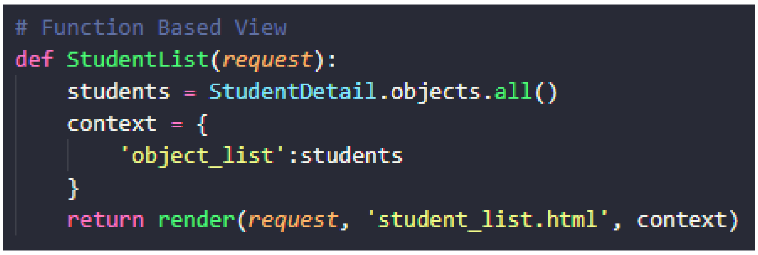
models.py



```
from django.db import models

# Create your models here.
class StudentDetail(models.Model):
    first_name = models.CharField(max_length=20)
    last_name = models.CharField(max_length=20)
    enrollment_number = models.CharField(max_length=12)
    semester = models.PositiveIntegerField()
    department = models.CharField(max_length=20)
    address = models.TextField(max_length=200)

    def __str__(self):
        return self.first_name +" "+ self.last_name

class EmployeeDetail(models.Model):
    employee_id = models.PositiveIntegerField()
    first_name = models.CharField(max_length=20)
    last_name = models.CharField(max_length=20)
    email = models.EmailField(max_length=30)
    department = models.CharField(max_length=30)
    address = models.TextField(max_length=200)

    def __str__(self):
        return self.first_name +" "+ self.last_name
```

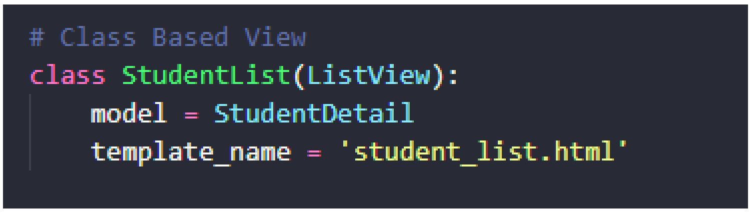
Choose any one of them:

Function Based Views:



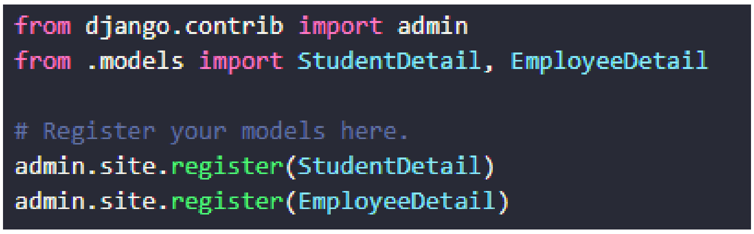
```
# Function Based View
def StudentList(request):
    students = StudentDetail.objects.all()
    context = {
        'object_list':students
    }
    return render(request, 'student_list.html', context)
```

Class Based Views:



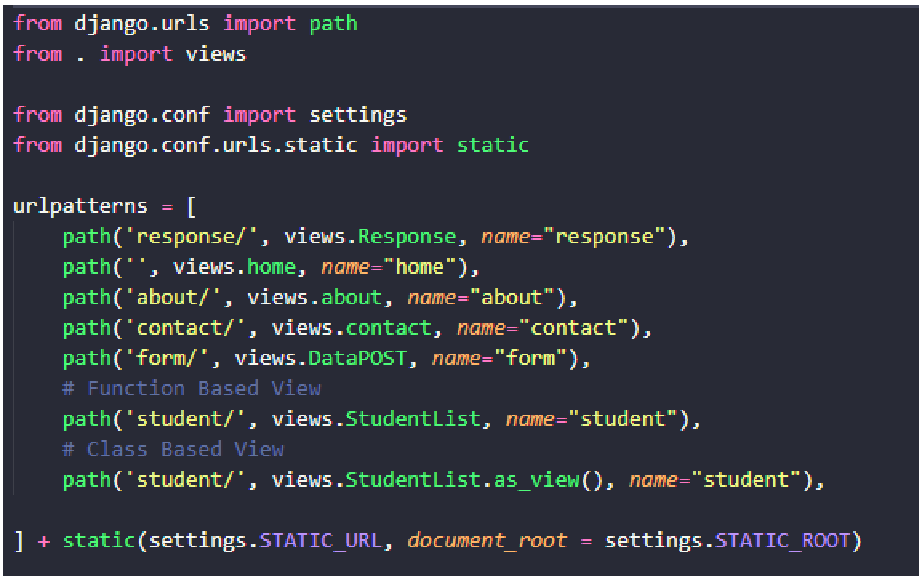
```
# Class Based View
class StudentList(ListView):
    model = StudentDetail
    template_name = 'student_list.html'
```

Registering Models in admin:



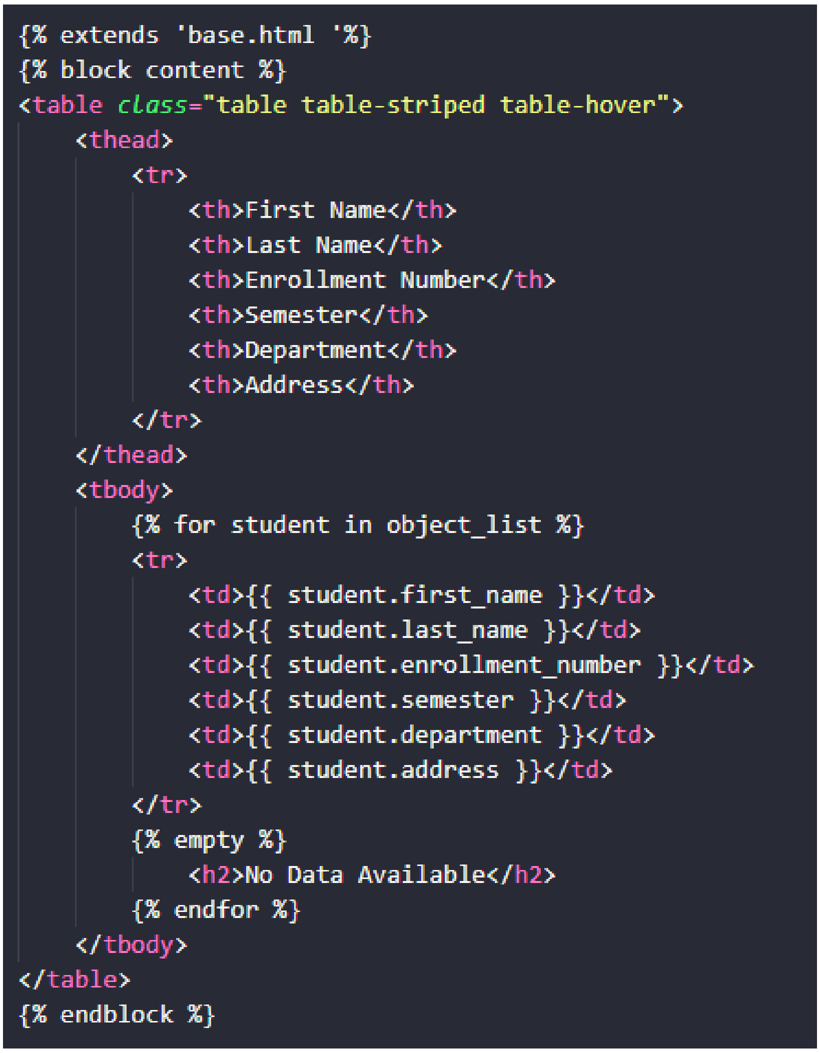
```
from django.contrib import admin
from .models import StudentDetail, EmployeeDetail

# Register your models here.
admin.site.register(StudentDetail)
admin.site.register(EmployeeDetail)
```

Routing URLs:



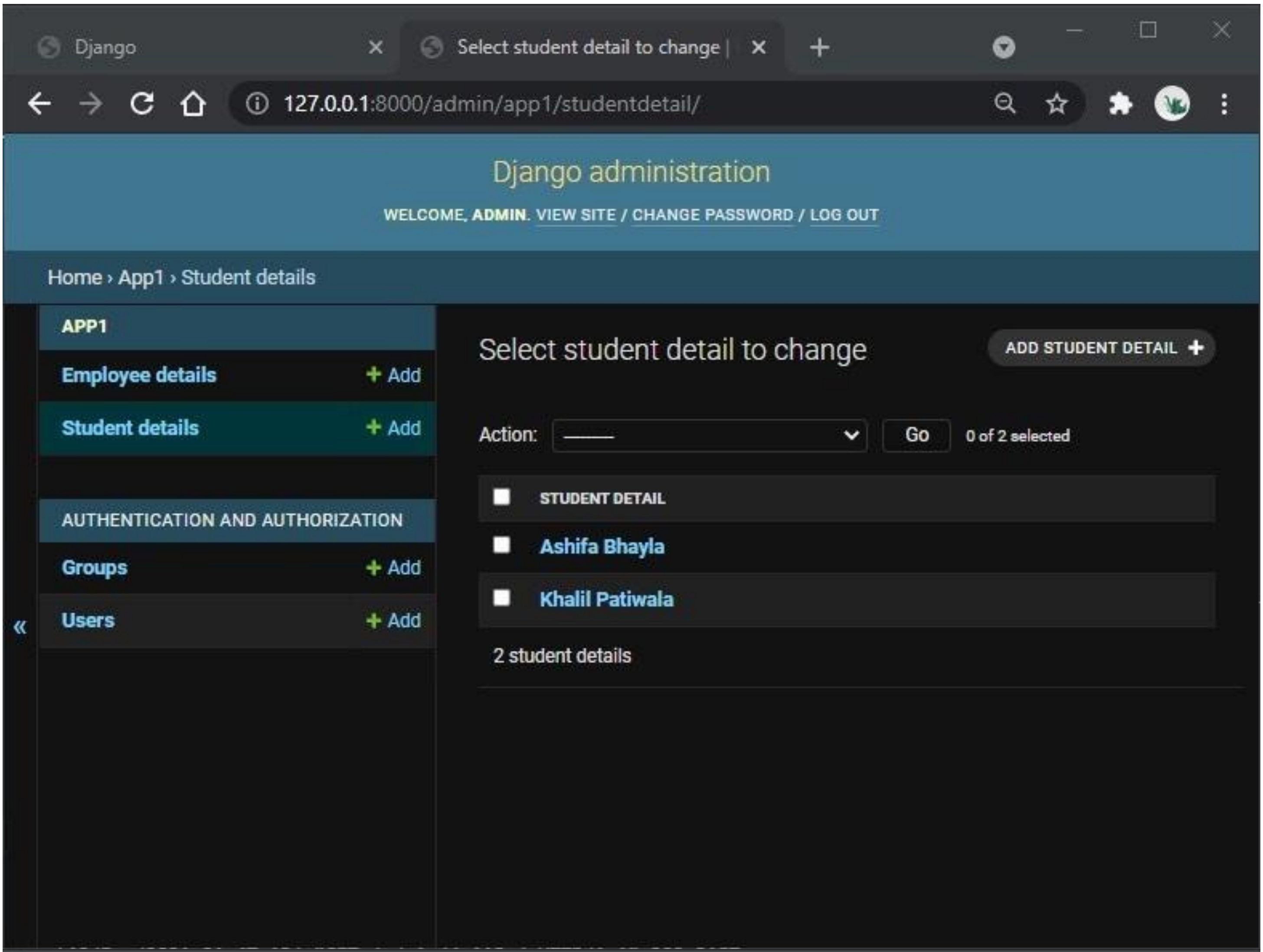
```
from django.urls import path
from . import views

from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('response/', views.Response, name="response"),
    path('', views.home, name="home"),
    path('about/', views.about, name="about"),
    path('contact/', views.contact, name="contact"),
    path('form/', views.DataPOST, name="form"),
    # Function Based View
    path('student/', views.StudentList, name="student"),
    # Class Based View
    path('student/', views.StudentList.as_view(), name="student"),

] + static(settings.STATIC_URL, document_root = settings.STATIC_ROOT)
```

student_list.html



```
{% extends 'base.html '%}
{% block content %}
<table class="table table-striped table-hover">
    <thead>
        <tr>
            <th>First Name</th>
            <th>Last Name</th>
            <th>Enrollment Number</th>
            <th>Semester</th>
            <th>Department</th>
            <th>Address</th>
        </tr>
    </thead>
    <tbody>
        {% for student in object_list %}
        <tr>
            <td>{{ student.first_name }}</td>
            <td>{{ student.last_name }}</td>
            <td>{{ student.enrollment_number }}</td>
            <td>{{ student.semester }}</td>
            <td>{{ student.department }}</td>
            <td>{{ student.address }}</td>
        </tr>
        {% empty %}
            <h2>No Data Available</h2>
        {% endfor %}
    </tbody>
</table>
{% endblock %}
```

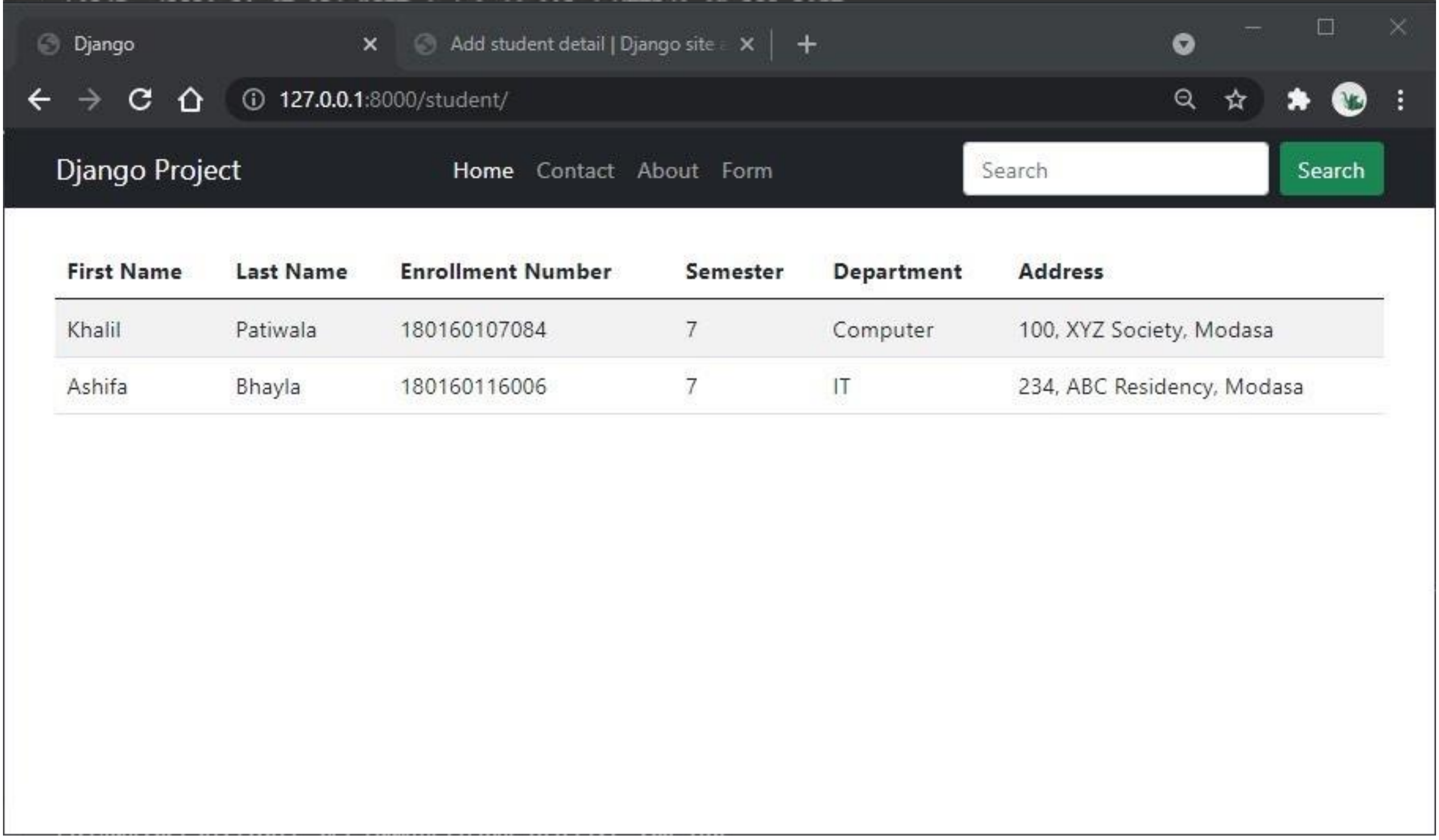
Output:


Django
Select student detail to change |
127.0.0.1:8000/admin/app1/studentdetail/
Django administration
WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › App1 › Student details
APP1
Employee details + Add
Student details + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
«
ADD STUDENT DETAIL +
Action: ---------
Go
0 of 2 selected

Select student detail to change

□	STUDENT DETAIL
□	Ashifa Bhayla
□	Khalil Patiwala

2 student details



First Name	Last Name	Enrollment Number	Semester	Department	Address
Khalil	Patiwala	180160107084	7	Computer	100, XYZ Society, Modasa
Ashifa	Bhayla	180160116006	7	IT	234, ABC Residency, Modasa