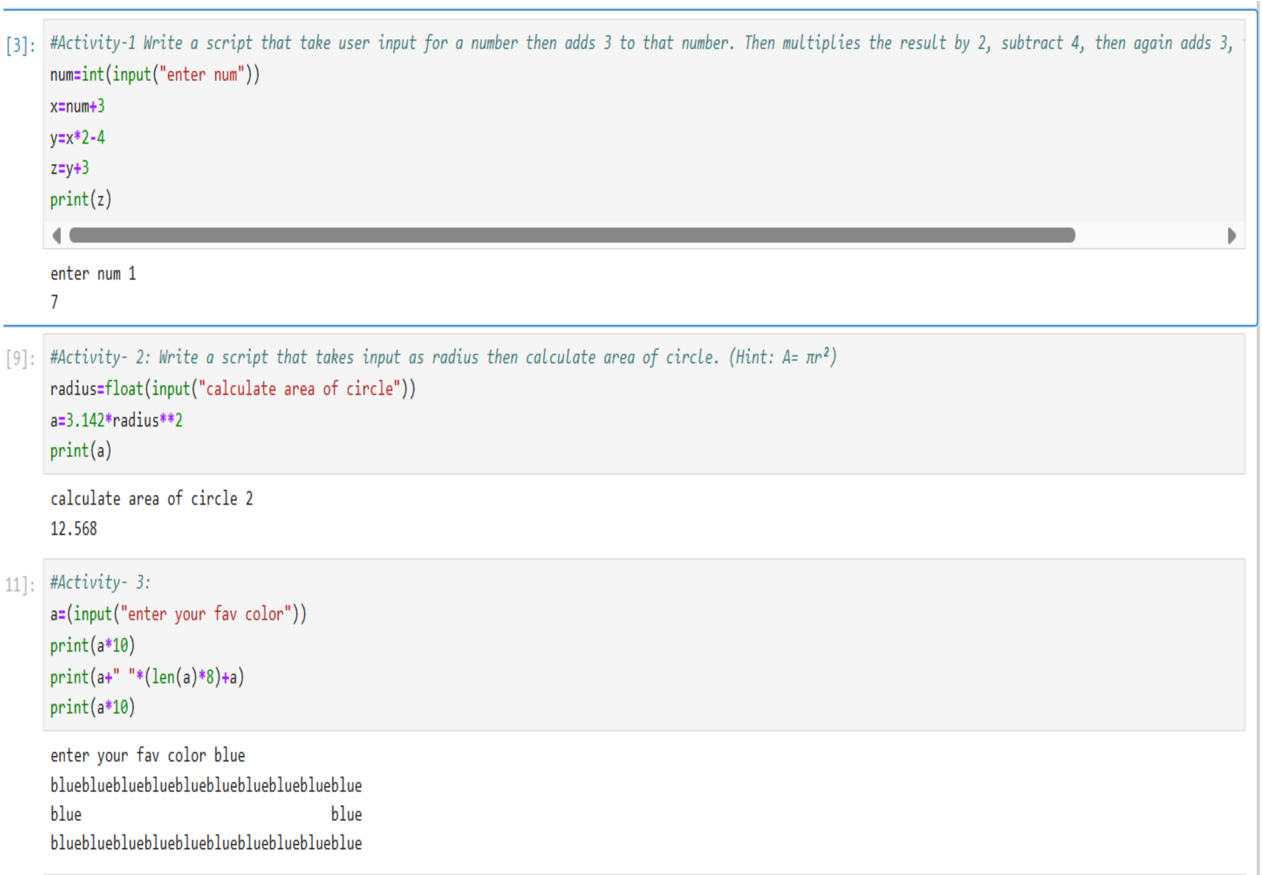
LAB# 1

Intro to strings,loops and conditional statements:



```
[3]: #Activity-1 Write a script that take user input for a number then adds 3 to that number. Then multiplies the result by 2, subtract 4, then again adds 3,
     num=int(input("enter num"))
     x=num+3
     y=x*2-4
     z=y+3
     print(z)
```

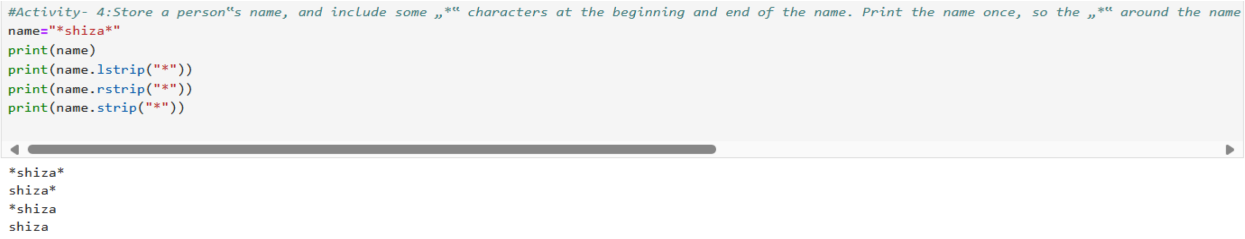
```
enter num 1
7
```

```
[9]: #Activity- 2: Write a script that takes input as radius then calculate area of circle. (Hint: A= πr²)
     radius=float(input("calculate area of circle"))
     a=3.142*radius**2
     print(a)
```

```
calculate area of circle 2
12.568
```

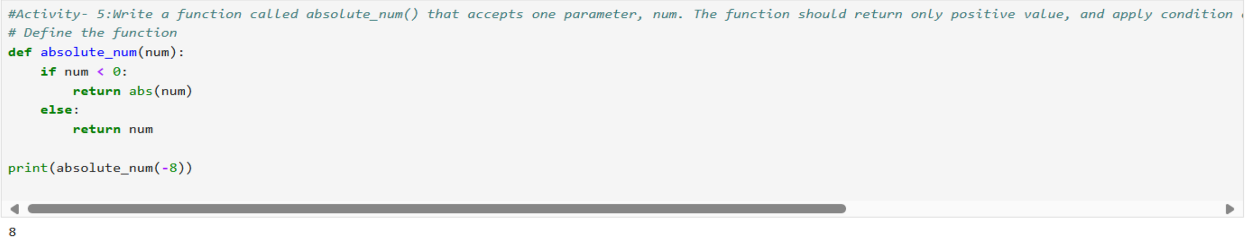
```
11]: #Activity- 3:
     a=(input("enter your fav color"))
     print(a*10)
     print(a+" "*(len(a)*8)+a)
     print(a*10)
```

```
enter your fav color blue
blueblueblueblueblueblueblueblueblueblue
blue                                blue
blueblueblueblueblueblueblueblueblueblue
```



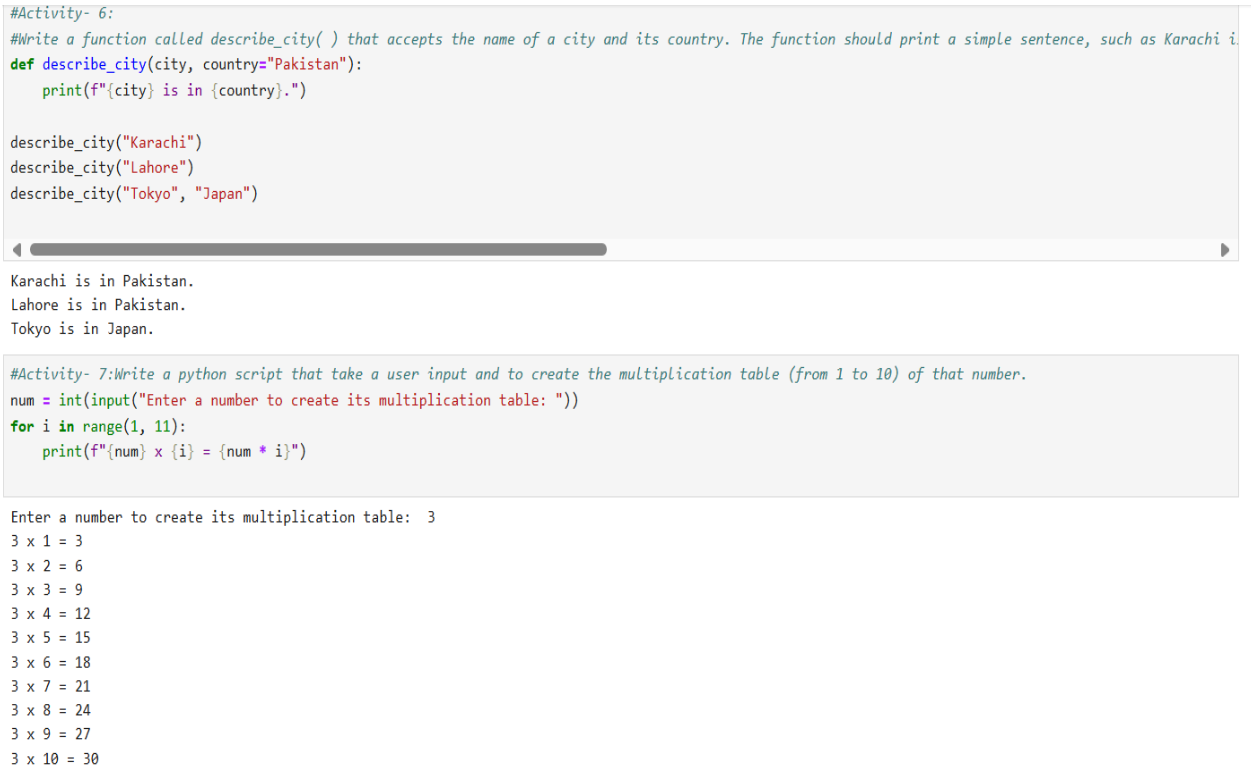
```
#Activity- 4:Store a person“s name, and include some „*“ characters at the beginning and end of the name. Print the name once, so the „*“ around the name
name="*shiza*"
print(name)
print(name.lstrip("*"))
print(name.rstrip("*"))
print(name.strip("*"))
```

```
*shiza*
shiza*
*shiza
shiza
```



```
#Activity- 5:Write a function called absolute_num() that accepts one parameter, num. The function should return only positive value, and apply condition
# Define the function
def absolute_num(num):
    if num < 0:
        return abs(num)
    else:
        return num

print(absolute_num(-8))
```

```
8
```

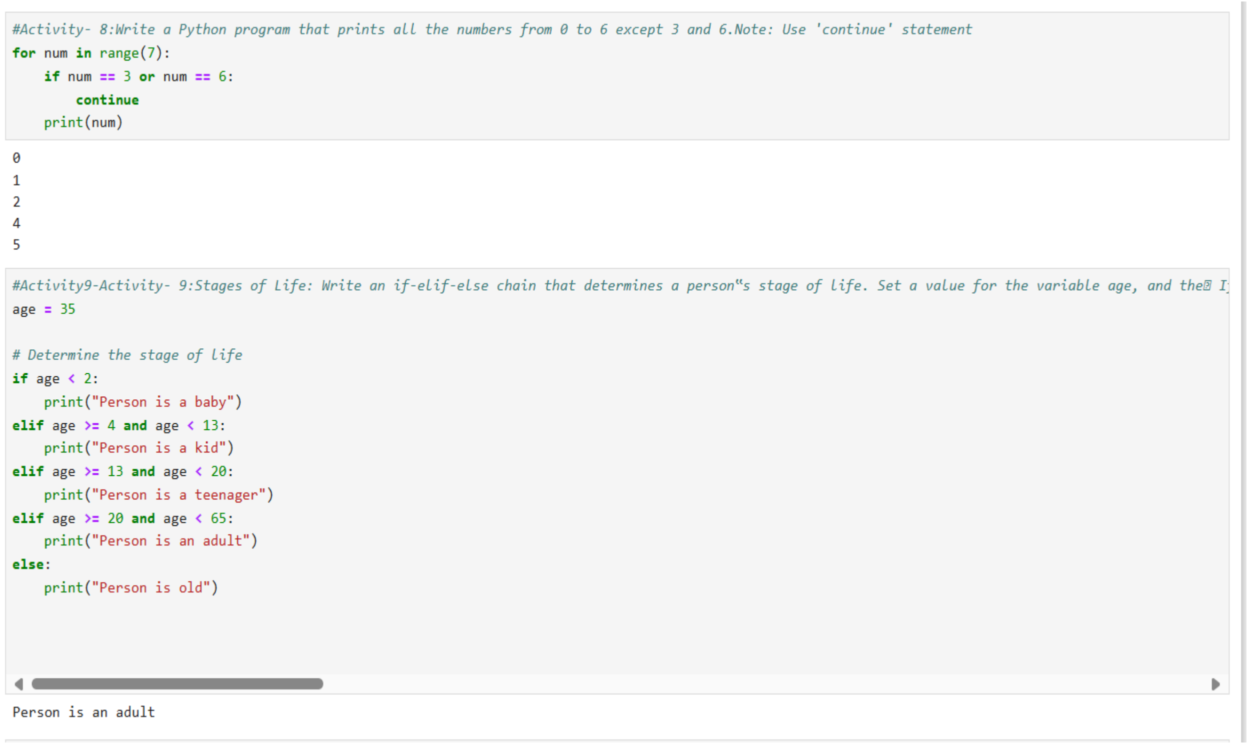


```
#Activity- 6:
#Write a function called describe_city( ) that accepts the name of a city and its country. The function should print a simple sentence, such as Karachi i
def describe_city(city, country="Pakistan"):
    print(f"{city} is in {country}.")

describe_city("Karachi")
describe_city("Lahore")
describe_city("Tokyo", "Japan")
```

```
Karachi is in Pakistan.
Lahore is in Pakistan.
Tokyo is in Japan.
```

```
#Activity- 7:Write a python script that take a user input and to create the multiplication table (from 1 to 10) of that number.
num = int(input("Enter a number to create its multiplication table: "))
for i in range(1, 11):
    print(f"{num} x {i} = {num * i}")
```

```
Enter a number to create its multiplication table:  3
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30
```



```
#Activity- 8:Write a Python program that prints all the numbers from 0 to 6 except 3 and 6.Note: Use 'continue' statement
for num in range(7):
    if num == 3 or num == 6:
        continue
    print(num)
```

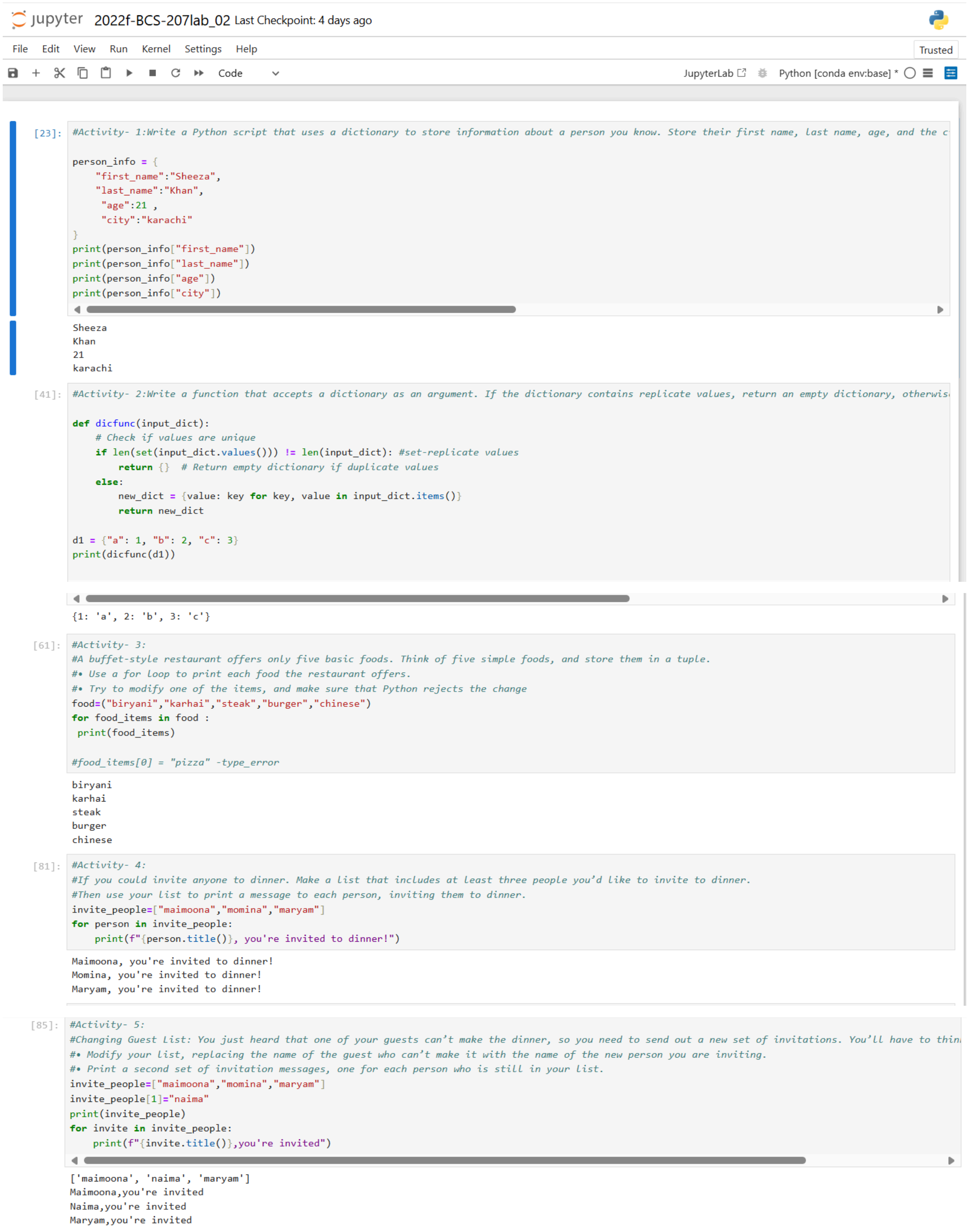
```
0
1
2
4
5
```

```
#Activity9-Activity- 9:Stages of Life: Write an if-elif-else chain that determines a person‟s stage of life. Set a value for the variable age, and the I
age = 35

# Determine the stage of life
if age < 2:
    print("Person is a baby")
elif age >= 4 and age < 13:
    print("Person is a kid")
elif age >= 13 and age < 20:
    print("Person is a teenager")
elif age >= 20 and age < 65:
    print("Person is an adult")
else:
    print("Person is old")
```

```
Person is an adult
```

LAB# 02

Functions,list,Tuples&Dictionary:


jupyter 2022f-BCS-207lab_02 Last Checkpoint: 4 days ago
File
Edit
View
Run
Kernel
Settings
Help
Trusted
Code
JupyterLab
Python [conda env:base] *

[23]:

```python
#Activity- 1:Write a Python script that uses a dictionary to store information about a person you know. Store their first name, last name, age, and the c

person_info = {
    "first_name":"Sheeza",
    "last_name":"Khan",
     "age":21 ,
     "city":"karachi"
}
print(person_info["first_name"])
print(person_info["last_name"])
print(person_info["age"])
print(person_info["city"])
```

```
Sheeza
Khan
21
karachi
```

[41]:

```python
#Activity- 2:Write a function that accepts a dictionary as an argument. If the dictionary contains replicate values, return an empty dictionary, otherwis

def dicfunc(input_dict):
    # Check if values are unique
    if len(set(input_dict.values())) != len(input_dict): #set-replicate values
        return {}  # Return empty dictionary if duplicate values
    else:
        new_dict = {value: key for key, value in input_dict.items()}
        return new_dict

d1 = {"a": 1, "b": 2, "c": 3}
print(dicfunc(d1))
```

```
{1: 'a', 2: 'b', 3: 'c'}
```

[61]:

```python
#Activity- 3:
#A buffet-style restaurant offers only five basic foods. Think of five simple foods, and store them in a tuple.
#• Use a for loop to print each food the restaurant offers.
#• Try to modify one of the items, and make sure that Python rejects the change
food=("biryani","karhai","steak","burger","chinese")
for food_items in food :
 print(food_items)

#food_items[0] = "pizza" -type_error
```

```
biryani
karhai
steak
burger
chinese
```

[81]:

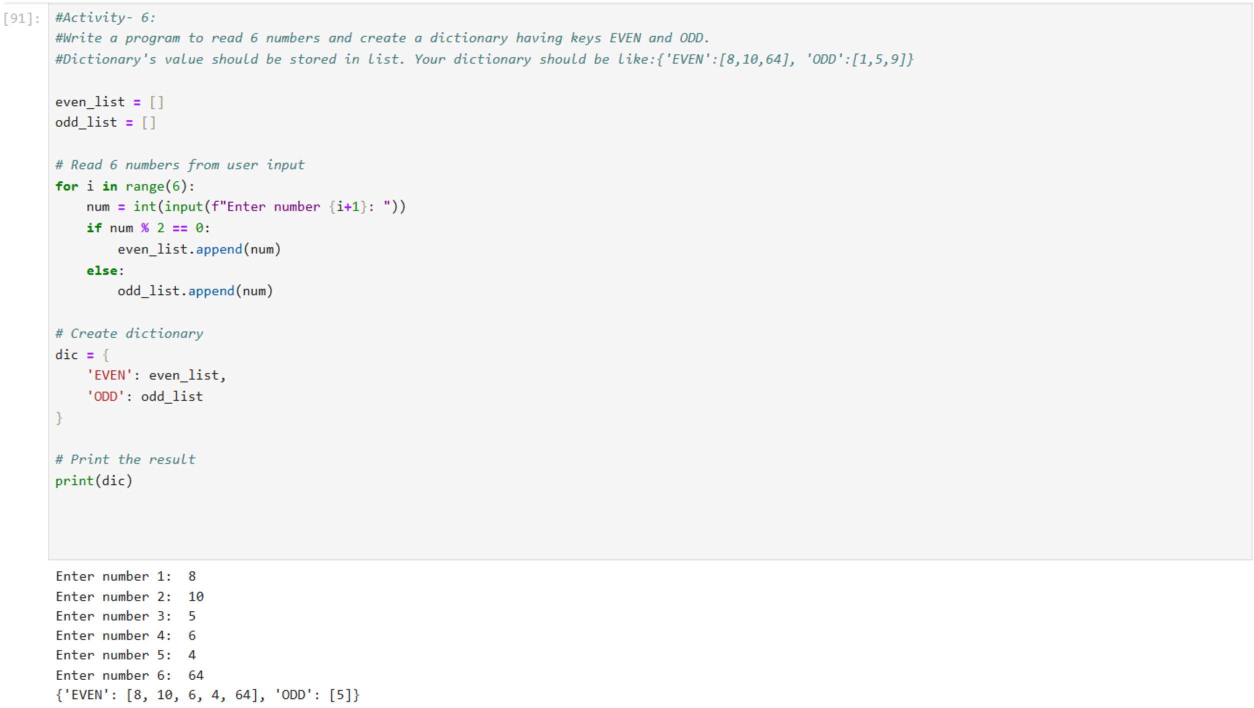
```python
#Activity- 4:
#If you could invite anyone to dinner. Make a list that includes at least three people you'd like to invite to dinner.
#Then use your list to print a message to each person, inviting them to dinner.
invite_people=["maimoona","momina","maryam"]
for person in invite_people:
    print(f"{person.title()}, you're invited to dinner!")
```

```
Maimoona, you're invited to dinner!
Momina, you're invited to dinner!
Maryam, you're invited to dinner!
```

[85]:

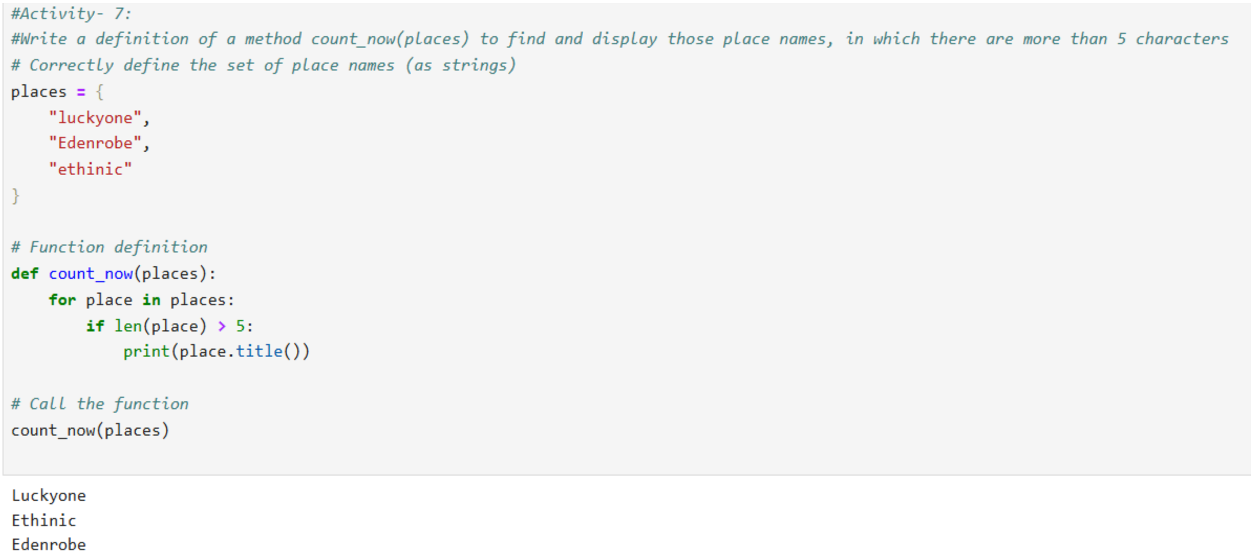
```python
#Activity- 5:
#Changing Guest List: You just heard that one of your guests can't make the dinner, so you need to send out a new set of invitations. You'll have to thin
#• Modify your list, replacing the name of the guest who can't make it with the name of the new person you are inviting.
#• Print a second set of invitation messages, one for each person who is still in your list.
invite_people=["maimoona","momina","maryam"]
invite_people[1]="naima"
print(invite_people)
for invite in invite_people:
    print(f"{invite.title()},you're invited")
```

```
['maimoona', 'naima', 'maryam']
Maimoona,you're invited
Naima,you're invited
Maryam,you're invited
```



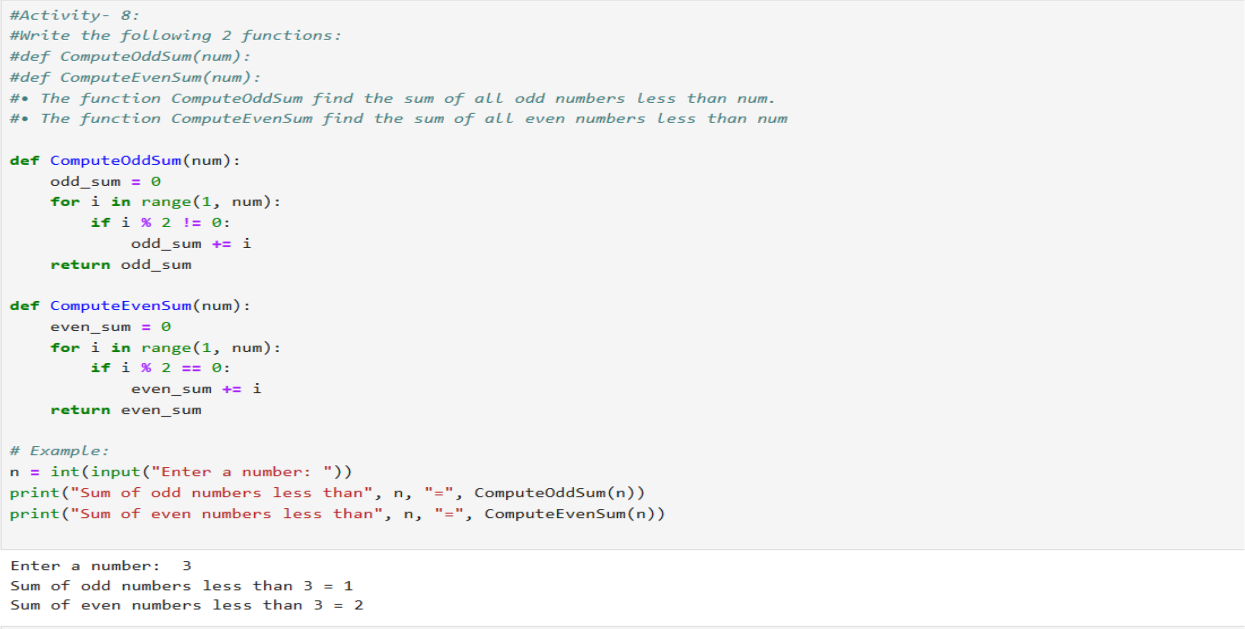
```
[91]: #Activity- 6:
      #Write a program to read 6 numbers and create a dictionary having keys EVEN and ODD.
      #Dictionary's value should be stored in list. Your dictionary should be like:{'EVEN':[8,10,64], 'ODD':[1,5,9]}

      even_list = []
      odd_list = []

      # Read 6 numbers from user input
      for i in range(6):
          num = int(input(f"Enter number {i+1}: "))
          if num % 2 == 0:
              even_list.append(num)
          else:
              odd_list.append(num)

      # Create dictionary
      dic = {
          'EVEN': even_list,
          'ODD': odd_list
      }

      # Print the result
      print(dic)
```

```
Enter number 1:  8
Enter number 2:  10
Enter number 3:  5
Enter number 4:  6
Enter number 5:  4
Enter number 6:  64
{'EVEN': [8, 10, 6, 4, 64], 'ODD': [5]}
```



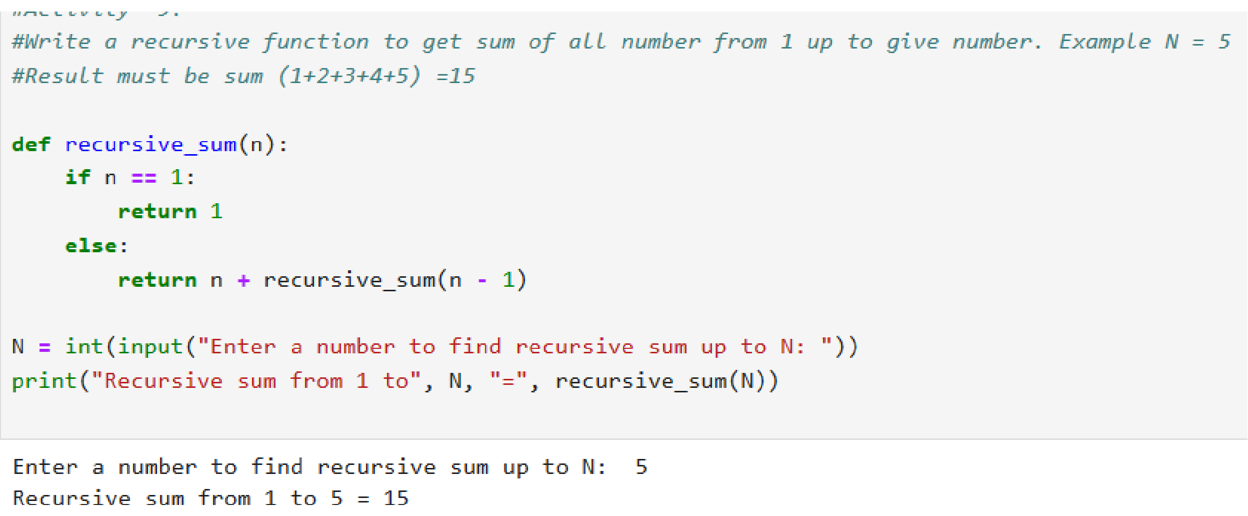
```
#Activity- 7:
#Write a definition of a method count_now(places) to find and display those place names, in which there are more than 5 characters
# Correctly define the set of place names (as strings)
places = {
    "luckyone",
    "Edenrobe",
    "ethinic"
}

# Function definition
def count_now(places):
    for place in places:
        if len(place) > 5:
            print(place.title())

# Call the function
count_now(places)
```

```
Luckyone
Ethinic
Edenrobe
```



```
#Activity- 8:
#Write the following 2 functions:
#def ComputeOddSum(num):
#def ComputeEvenSum(num):
#• The function ComputeOddSum find the sum of all odd numbers less than num.
#• The function ComputeEvenSum find the sum of all even numbers less than num

def ComputeOddSum(num):
    odd_sum = 0
    for i in range(1, num):
        if i % 2 != 0:
            odd_sum += i
    return odd_sum

def ComputeEvenSum(num):
    even_sum = 0
    for i in range(1, num):
        if i % 2 == 0:
            even_sum += i
    return even_sum

# Example:
n = int(input("Enter a number: "))
print("Sum of odd numbers less than", n, "=", ComputeOddSum(n))
print("Sum of even numbers less than", n, "=", ComputeEvenSum(n))
```

```
Enter a number:  3
Sum of odd numbers less than 3 = 1
Sum of even numbers less than 3 = 2
```



```
#Activity- 9:
#Write a recursive function to get sum of all number from 1 up to give number. Example N = 5
#Result must be sum (1+2+3+4+5) =15

def recursive_sum(n):
    if n == 1:
        return 1
    else:
        return n + recursive_sum(n - 1)

N = int(input("Enter a number to find recursive sum up to N: "))
print("Recursive sum from 1 to", N, "=", recursive_sum(N))
```

```
Enter a number to find recursive sum up to N:  5
Recursive sum from 1 to 5 = 15
```