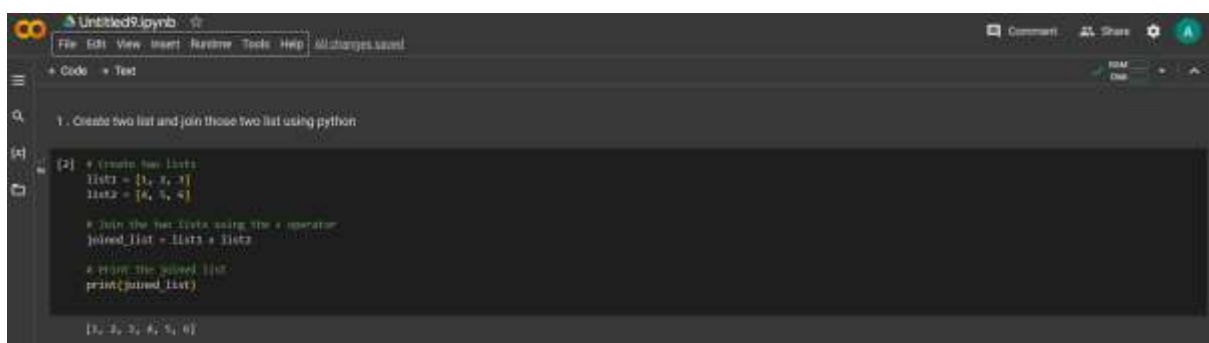


## DA ASSIGNMENT – 4

NAME : SAKTHIVEL A

REGISTER NO : 723920243044

Create two list and join those two list :



A screenshot of a Jupyter Notebook titled 'Untitled9.ipynb'. The interface shows a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. Below the menu bar, there are tabs for '+ Code' and '+ Text'. The main area contains a code cell with the following Python code:

```
1. Create two list and join those two list using python

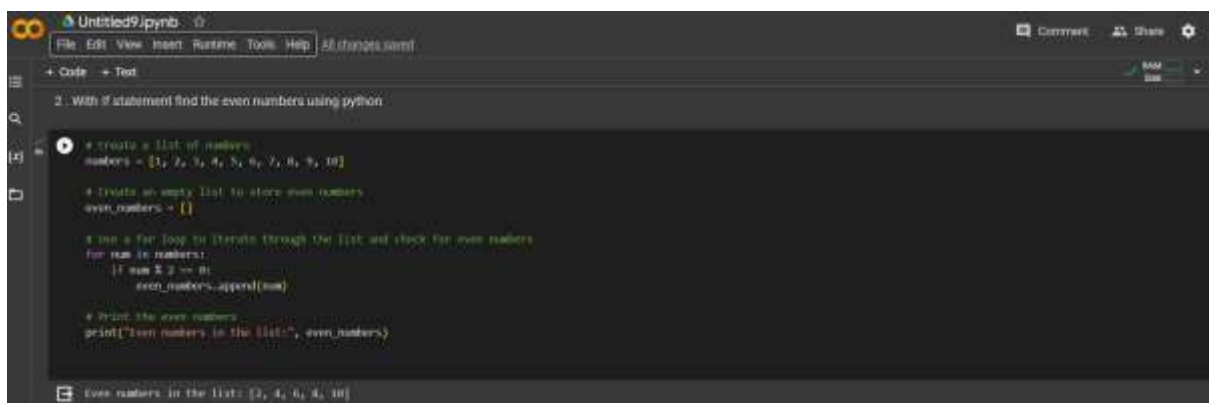
[2]: # create two lists
      lists = [1, 2, 3]
      lists2 = [4, 5, 6]

      # join the two lists using the + operator
      joined_list = lists + lists2

      # print the joined list
      print(joined_list)
```

The output of the code cell is displayed at the bottom: `[1, 2, 3, 4, 5, 6]`.

With If statement find the even numbers :



A screenshot of a Jupyter Notebook titled 'Untitled9.ipynb'. The interface shows a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. Below the menu bar, there are tabs for '+ Code' and '+ Text'. The main area contains a code cell with the following Python code:

```
2. With if statement find the even numbers using python

# create a list of numbers
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

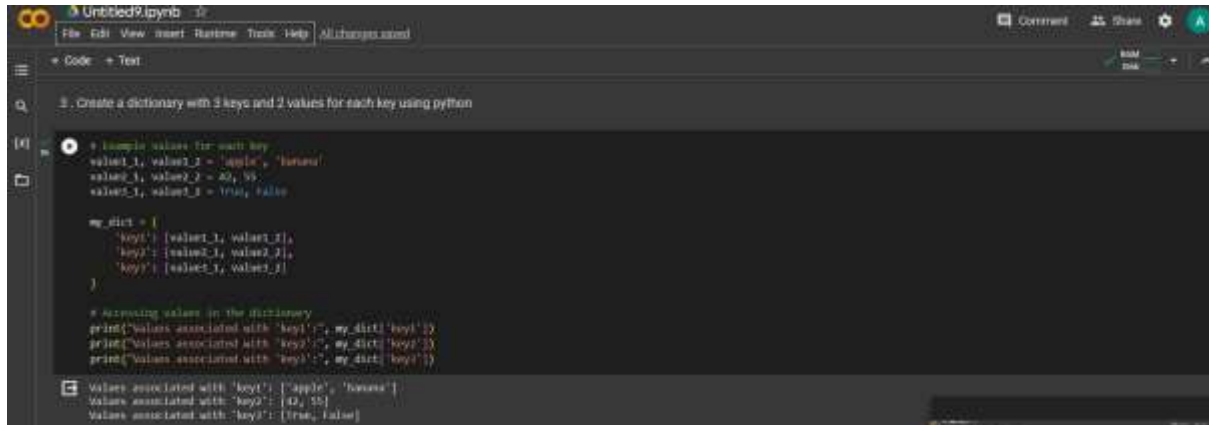
# create an empty list to store even numbers
even_numbers = []

# use a for loop to iterate through the list and check for even numbers
for num in numbers:
    if num % 2 == 0:
        even_numbers.append(num)

# print the even numbers
print("Even numbers in the list:", even_numbers)
```

The output of the code cell is displayed at the bottom: `Even numbers in the list: [2, 4, 6, 8, 10]`.

Create a dictionary with 3 keys and 2 values for each key :



The screenshot shows a Jupyter Notebook interface with a file named 'Untitled9.ipynb'. The code cell contains the following Python code:

```
# Example values for each key
value1_1, value1_2 = 'apple', 'banana'
value2_1, value2_2 = 42, 55
value3_1, value3_2 = True, False

my_dict = {
    'key1': [value1_1, value1_2],
    'key2': [value2_1, value2_2],
    'key3': [value3_1, value3_2]
}

# Accessing values in the Dictionary
print("Values associated with 'key1':", my_dict['key1'])
print("Values associated with 'key2':", my_dict['key2'])
print("Values associated with 'key3':", my_dict['key3'])
```

The output of the code is displayed below the code cell:

```
Values associated with 'key1': ['apple', 'banana']
Values associated with 'key2': [42, 55]
Values associated with 'key3': [True, False]
```

Create a function with If statement which is used to find the odd numbers :



The screenshot shows a Jupyter Notebook interface with a file named 'Untitled9.ipynb'. The code cell contains the following Python code:

```
def find_odd_numbers(numbers):
    # Create an empty list to store odd numbers
    odd_numbers = []

    # Use a for loop to iterate through the list and check for odd numbers
    for num in numbers:
        if num % 2 != 0:
            odd_numbers.append(num)

    # Return the list of odd numbers
    return odd_numbers

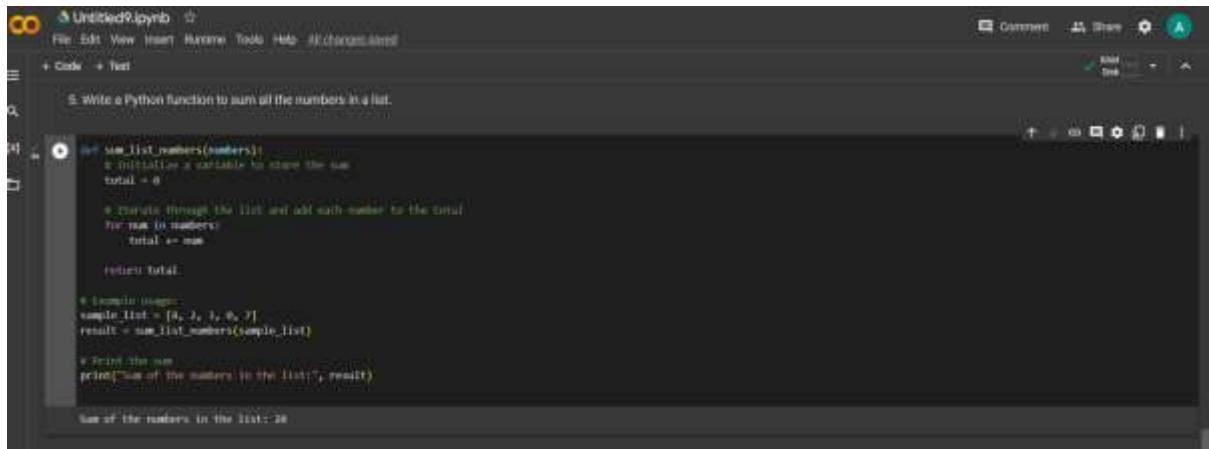
# Sample input
numbers_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
result = find_odd_numbers(numbers_list)

# Print the odd numbers
print("Odd numbers in the list:", result)
```

The output of the code is displayed below the code cell:

```
Odd numbers in the list: [1, 3, 5, 7, 9]
```

Write a Python function to sum all the numbers in a list :



The screenshot shows a Jupyter Notebook interface with a dark theme. The notebook is titled 'Untitled9.ipynb'. The code cell contains a function definition and a test case. The function 'sum\_list\_numbers' takes a list of numbers and returns their sum. The test case creates a sample list [8, 2, 1, 0, 7], calls the function, and prints the result. The output of the code cell is 'Sum of the numbers in the list: 28'.

```
def sum_list_numbers(numbers):  
    # Initialize a variable to store the sum  
    total = 0  
  
    # Iterate through the list and add each number to the total  
    for num in numbers:  
        total += num  
  
    return total  
  
# Sample usage:  
sample_list = [8, 2, 1, 0, 7]  
result = sum_list_numbers(sample_list)  
  
# Print the sum  
print("Sum of the numbers in the list:", result)
```

Sum of the numbers in the list: 28