Experiment 1

OBJECTIVE: To understand the basics of Python by the following programs:

(i) Write a program to add two numbers to illustrate the use of print statement.

(ii) Write a program to illustrate the use of conditional statements by checking if the input number is odd or even.

(iii) Write a program to illustrate the use of functions.

(iv) Write a program to access the values in a dictionary.

(v) Write a program to perform various string operations.

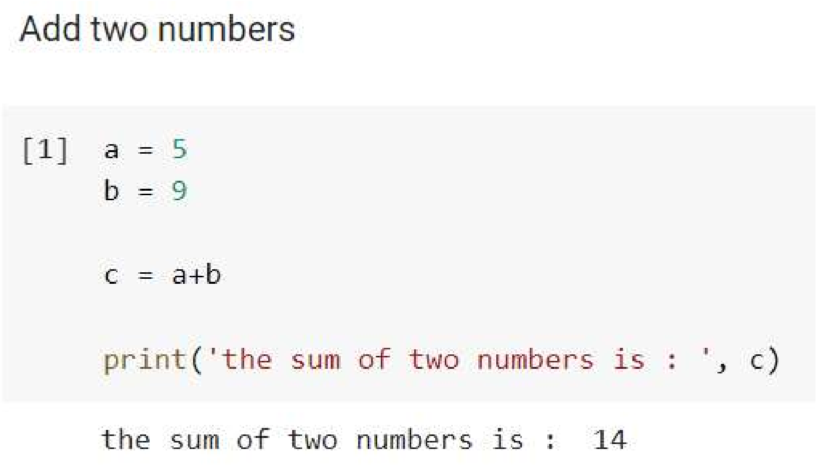
RELATED THEORY

(i) Python is a powerful high-level, object-oriented programming language. It has simple easy-to-use syntax, making it the perfect language for someone trying to learn computer programming for the first time.

(ii) In the first program, two numbers are added to illustrate the use of print statement. In the second program, use of conditional statements is demonstrated. In order to write useful programs, we almost always need the ability to check conditions and change the behavior of the program accordingly. Conditional statements give us this ability.

(iii) Next, python functions are explained where a function is a block of organized, reusable code that is used to perform a single, related action. In the fourth program values in a dictionary are accessed. Each key is separated from its value by a colon (:), the items are separated by commas, and the whole thing is enclosed in curly braces.

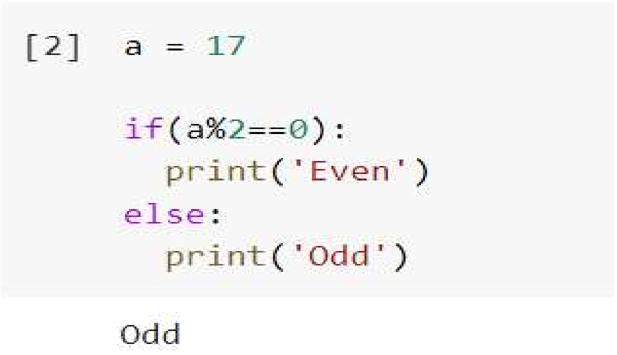
(iv) Finally, various string operations are demonstrated using string functions. Python treats single quotes the same as double quotes.



Add two numbers

```
[1]  a = 5
     b = 9

     c = a+b

     print('the sum of two numbers is : ', c)

     the sum of two numbers is :  14
```

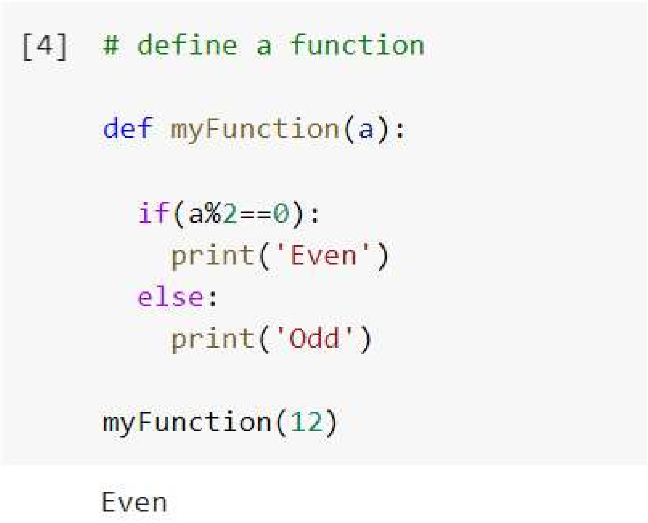
Conditional Statement



```
[2]  a = 17

     if(a%2==0):
       print('Even')
     else:
       print('Odd')
```

```
Odd
```

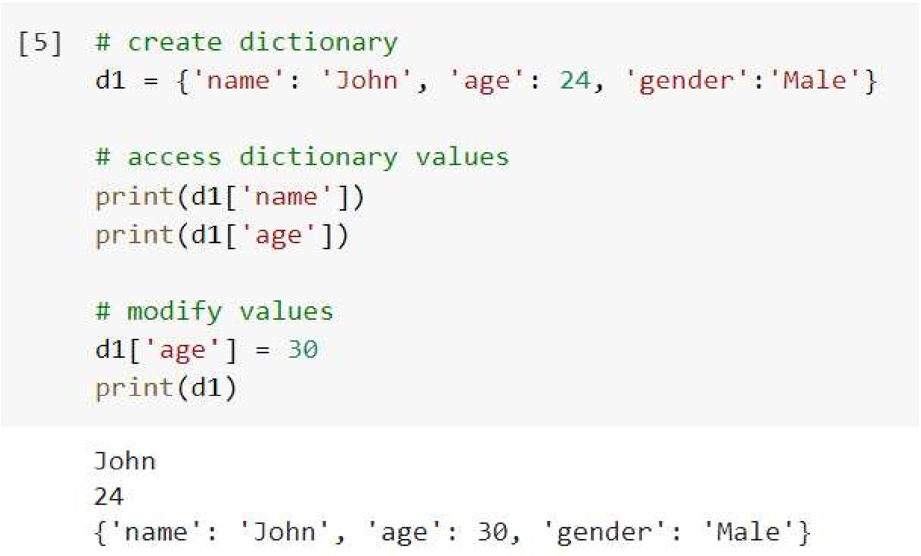
Define and call function in python



```
[4]  # define a function

     def myFunction(a):

       if(a%2==0):
         print('Even')
       else:
         print('Odd')

     myFunction(12)
```

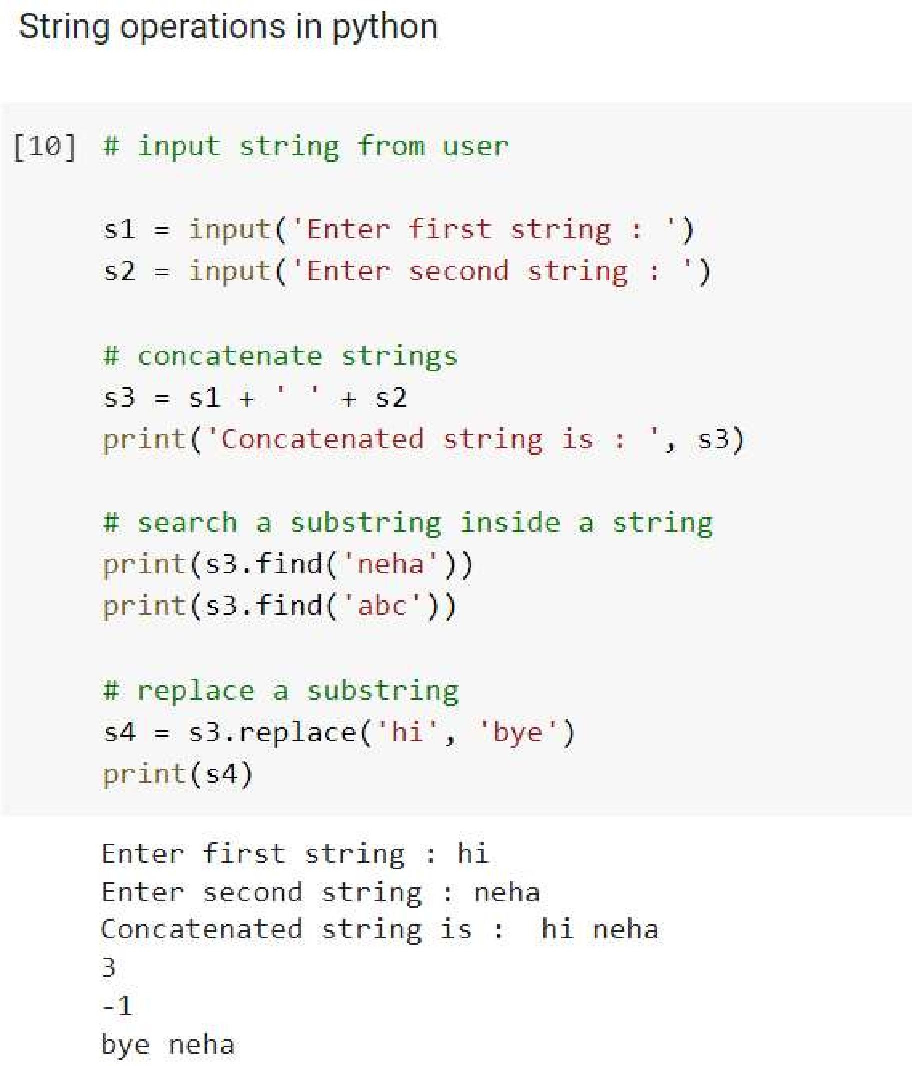
```
Even
```

Python Dictionary



```
[5]  # create dictionary
     d1 = {'name': 'John', 'age': 24, 'gender':'Male'}

     # access dictionary values
     print(d1['name'])
     print(d1['age'])

     # modify values
     d1['age'] = 30
     print(d1)
```

```
John
24
{'name': 'John', 'age': 30, 'gender': 'Male'}
```



String operations in python

```
[10] # input string from user

     s1 = input('Enter first string : ')
     s2 = input('Enter second string : ')

     # concatenate strings
     s3 = s1 + ' ' + s2
     print('Concatenated string is : ', s3)

     # search a substring inside a string
     print(s3.find('neha'))
     print(s3.find('abc'))

     # replace a substring
     s4 = s3.replace('hi', 'bye')
     print(s4)
```

```
Enter first string : hi
Enter second string : neha
Concatenated string is :  hi neha
3
-1
bye neha
```

CONCLUSION

From the above programs, we conclude that python provides rich features for programming. It supports functional and structured programming methods as well as OOP. Python has few keywords, simple structure, and a clearly defined syntax. Python code is more clearly defined and visible to the eyes. Python's bulk of the library is very portable.