

Python – Experiment 2

AIM : Advanced Data types and functions in Python

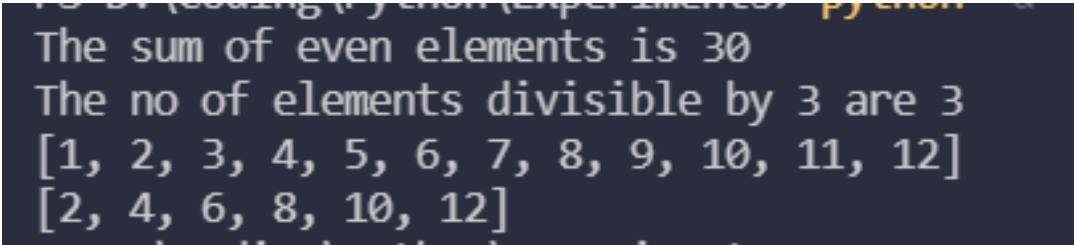
LO - 2 : Advanced Data types and functions in Python

A. Write a python program to perform following operations on an array: Sum of all even elements, count the no. of elements divisible by 3, insert 2 elements at the end of the list, delete all the odd elements from the list.

Code:

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
sum = sum(i for i in a if i % 2 == 0)
print("The sum of even elements is", sum)
print("The no of elements divisible by 3 are", len([x for x in a if x % 3 == 0]))
a.append(11)
a.append(12)
print(a)
a = [x for x in a if x % 2 == 0]
print(a)
```

Output:



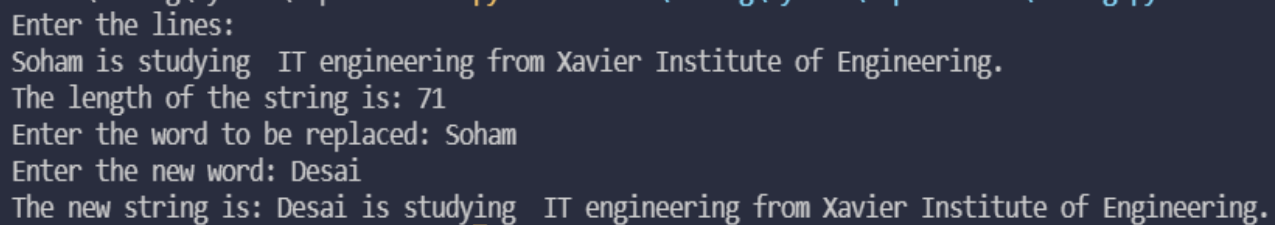
```
The sum of even elements is 30
The no of elements divisible by 3 are 3
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
[2, 4, 6, 8, 10, 12]
```

B. Write a python program to input a multiline string or a paragraph & count the no. of words & characters in string. Also check for a substring & replace each of its occurrences by some other string.

Code:

```
print("Enter the lines: ")
lines = ""
while not lines.endswith("."):
    lines = str(input())

print("The length of the string is:", len(lines))
rep_word = input("Enter the word to be replaced: ")
new_word = input("Enter the new word: ")
lines = lines.replace(rep_word, new_word)
print("The new string is:", lines)
```

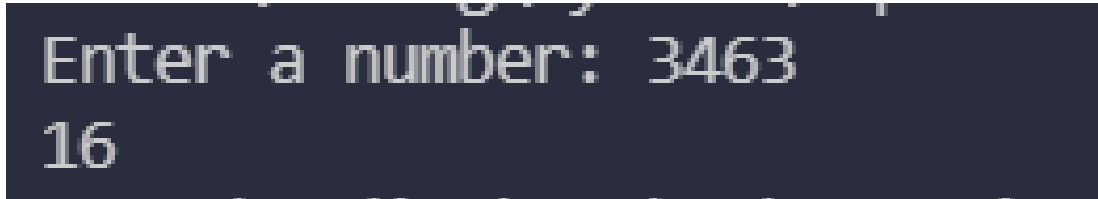
Output:

```
Enter the lines:
Soham is studying IT engineering from Xavier Institute of Engineering.
The length of the string is: 71
Enter the word to be replaced: Soham
Enter the new word: Desai
The new string is: Desai is studying IT engineering from Xavier Institute of Engineering.
```

C. Write a recursive function in python to find the Sum of the digits of the numbers

Code:

```
def sum_digits(n):
    if n < 10:
        return n
    else:
        return n % 10 + sum_digits(n // 10)
n = int(input("Enter a number: "))
print(sum_digits(n))
```

Output:A terminal window with a dark background. The text 'Enter a number: 3463' is displayed on the first line, and the number '16' is displayed on the second line.

D. Write a menu-driven python program to :

- 1) Add students marks information in terms of tuples. Calculate the total and average marks.
- 2) Display students with specified key.
- 3) Enter students' admission date in the form (dd/mm/yyyy) to introduce nested tuple. Display students having admission in the same year.

Code:

```
while True:
```

```
    print("1.Calculate average and total marks")
```

```
    print("2.Display students with specific key")
```

```
    print("3.Enter admission date")
```

```
    print("0.Exit")
```

```
    choice = int(input("Enter your choice: "))
```

```
    a = (78,45,58,69,90)
```

```
    if choice == 1:
```

```
        print("Students marks: ", a)
```

```
        print("Total marks: ", sum(a))
```

```
        print("Average marks: ", sum(a)/len(a))
```

```
    elif choice == 2:
```

```
        key = int(input("Enter the key: "))
```

```
        print("The student at given key is:", a[key])
```

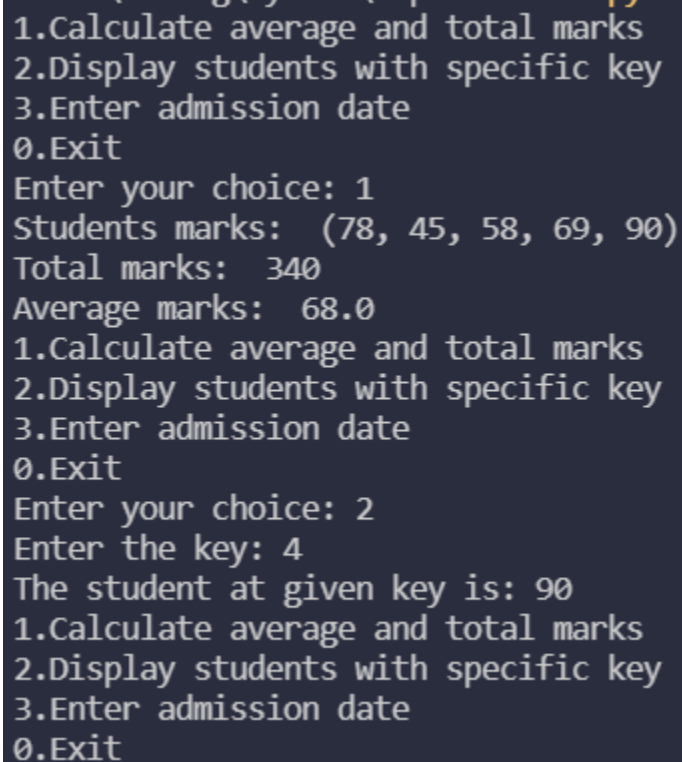
```
    elif choice == 3:
```

```
        student_date = ((12,2,2022),(11,2,2022),(10,2,2022),(9,2,2022),(8,2,2022))
```

```
        studentsID = 100
```

```
        toCheck = False
```

```
toGetYear = int(input("Enter year (2020 or 2021) to get student's having admission in the
year : "))
if 2020 or 2021 in toGetYear:
    for addDate in a:
        if toGetYear in addDate:
            print(f"Student {studentsID} ", end=" ")
        toCheck = True
        studentsID = studentsID + 1
        if toCheck :
            print("Having Admission In The Year",toGetYear)
    else:
        print("enter valid year!!!!(2020-2021)")
elif choice == 0:
    exit()
```

Output:

```
1. Calculate average and total marks
2. Display students with specific key
3. Enter admission date
0. Exit
Enter your choice: 1
Students marks: (78, 45, 58, 69, 90)
Total marks: 340
Average marks: 68.0
1. Calculate average and total marks
2. Display students with specific key
3. Enter admission date
0. Exit
Enter your choice: 2
Enter the key: 4
The student at given key is: 90
1. Calculate average and total marks
2. Display students with specific key
3. Enter admission date
0. Exit
```

E. Write a menu-driven program to demonstrate the use of set in python:

- i) Accept two strings from the user.
- ii) Display common letters in two input strings (set intersection).
- iii) Display letters which are in the first string but not in the second string (set difference).
- iv) Display set of all letters from both the strings (set union)
- v) Display set of letters which are in two strings but not common (Symmetric Difference).

Code:

```
a = input("Enter first string:")
b = input("Enter second string:")
f = set(a)
f1 = set(b)
for i in range(len(f)):
    f.add(a[i])
for i in range(len(f1)):
    f1.add(b[i])
while True:
    print("1.Display common elements")
    print("2.Display elements in a but not in b")
    print("3.Display union of a and b")
    print("4.Display elements present in both a and b but not common")
    print("0.Exit")
    choice = int(input("Enter your choice: "))
    if choice == 1:
        print("Common elements: ", f.intersection(f1))
    elif choice == 2:
        print("Elements in a but not in b: ", f.difference(f1))
    elif choice == 3:
        print("Union of a and b: ", f.union(f1))
    elif choice == 4:
        print("Elements present in both a and b but not common: ", f.symmetric_difference(f1))
    elif choice == 0:
        exit()
    else:
        print("Invalid choice")
```

Output:

```
Enter first string:Soham
Enter second string:Desai
1.Display common elements
2.Display elements in a but not in b
3.Display union of a and b
4.Display elements present in both a and b but not common
0.Exit
Enter your choice: 1
Common elements: {'a'}
1.Display common elements
2.Display elements in a but not in b
3.Display union of a and b
4.Display elements present in both a and b but not common
0.Exit
Enter your choice: 2
Elements in a but not in b: {'h', 's', 'm', 'o'}
1.Display common elements
2.Display elements in a but not in b
3.Display union of a and b
4.Display elements present in both a and b but not common
0.Exit
Enter your choice: 3
Union of a and b: {'e', 's', 'o', 's', 'a', 'i', 'D', 'm', 'h'}
1.Display common elements
2.Display elements in a but not in b
3.Display union of a and b
4.Display elements present in both a and b but not common
0.Exit
Enter your choice: 4
Elements present in both a and b but not common: {'e', 's', 'o', 'h', 's', 'D', 'm', 'i'}
1.Display common elements
2.Display elements in a but not in b
3.Display union of a and b
4.Display elements present in both a and b but not common
0.Exit
Enter your choice: 0
```

F. Write a menu-driven program to demonstrate the use of dictionary in python:

- i) Create key/value pair dictionary.
- ii) Update/concatenate and delete item from existing dictionary.

Find a key and print its value.

Code:

```
Keys = int(input("Enter the number of keys you want to add in dictionary : "))
myDict = {}
for i in range(Keys):
    keys = input("Enter the Keys In The Dictionary : ")
    value = input("Enter the Values In The Dictionary : ")
    toAdd = f"{keys} : {value}"
    myDict[keys] = value
while True:
    print("1.Update dictionary")
    print("2.Delete any key from dictionary")
    print("3.Find any value of keys from dictionary")
    print("0.Exit")
    choice = int(input("Enter your choice :"))
    if choice==1:
        uKey = input("Enter key to update :")
        if uKey in myDict:
            uValue = input("Enter the updated value : ")
            myDict[uKey]=uValue
            print(myDict)
        else:
            print("Enter a valid input")
    elif choice==2:
        dKey = input("Enter key to delete :")
        if dKey in myDict:
            del myDict[dKey]
            print(myDict)
        else:
            print("Enter a valid input")
    elif choice==3:
```

```
fKey = input("Enter key to find :")
if fKey in myDict:
    print(myDict[fKey])
else:
    print("Enter a valid input")
elif choice==0:
    exit()
```

Output:

```
Enter the number of keys you want to add in dictionary : 2
Enter the Keys In The Dictionary : 1
Enter the Values In The Dictionary : S
Enter the Keys In The Dictionary : 2
Enter the Values In The Dictionary : F
1.Update dictionary
2.Delete any key from dictionary
3.Find any value of keys from dictionary
0.Exit
Enter your choice :1
Enter key to update :2
Enter the updated value : VC
{'1': 'S', '2': 'VC'}
1.Update dictionary
2.Delete any key from dictionary
3.Find any value of keys from dictionary
0.Exit
Enter your choice :2
Enter key to delete :1
{'2': 'VC'}
1.Update dictionary
2.Delete any key from dictionary
3.Find any value of keys from dictionary
0.Exit
Enter your choice :3
Enter key to find :2
VC
1.Update dictionary
2.Delete any key from dictionary
3.Find any value of keys from dictionary
0.Exit
Enter your choice :0
```

Conclusion: From this experiment we have learned about how to deal with dictionary, tuples and many form of loops.