



Masters in Applied Statistics and Data Science(MASDS)

Department of Statistics

Jahangirnagar University

Savar, Dhaka-1342, Bangladesh

Course : Introduction To Data Science with Python

Course Code: MASDS04

**Assignment 03**

Submitted to

**Assistant Professor Farhana Afrin Duty**

Submitted by

**MD. RAFIUL HOQUE FAKIR**

MASDS 9th Batch, Section: A

Roll: 20229019, 01687882905

1. Write a Python program to count the number of characters (character frequency) in a string.  
Sample String : google.com

Solution:

```
string = "google.com"

char_counts = {}

for char in string:

    if char in char_counts:

        char_counts[char] += 1

    else:

        char_counts[char] = 1

for char, count in char_counts.items():

    print(f"{char}: {count}")
```

2. Write a Python program to check if a given key already exists in a dictionary.

Solution:

```
my_dict = {'apple': 1, 'banana': 2, 'orange': 3}

key_to_check = input("Enter the key to check: ")

if key_to_check in my_dict:

    print(f"The key '{key_to_check}' exists in the dictionary.")

else:

    print(f"The key '{key_to_check}' does not exist in the dictionary.")
```

3. Write a Python program that accepts a sequence of comma-separated numbers from the user and generates a list and a tuple of those numbers. Sample data : 1, 5, 7, 23

Expected Output :

List : ['1', ' 5', ' 7', ' 23']

Tuple : ('1', ' 5', ' 7', ' 23')

Solution:

```
input_string = input("Enter a sequence of comma-separated numbers: ")
```

```
input_list = input_string.split(',')
```

```
input_list = [s.strip() for s in input_list]
```

```
input_list = [int(s) for s in input_list]
```

```
input_tuple = tuple(input_list)
```

```
print("List:", input_list)
```

```
print("Tuple:", input_tuple)
```

4. Write a Python program to calculate the number of days between two dates. Sample dates : (2014, 7, 2), (2014, 7, 11)

Expected output : 9 days

Solution:

```
from datetime import date
```

```
date1 = (2014, 7, 2)
```

```
date2 = (2014, 7, 11)
```

```
date1 = date(*date1)
```

```
date2 = date(*date2)
```

```
delta = date2 - date1
```

```
days_between = delta.days
```

```
print("The number of days between", date1, "and", date2, "is", days_between, "days.")
```

5. Write a Python program to solve  $(x + y) * (x + y)$ . Test Data :  $x = 4, y = 3$

Expected Output :  $(4 + 3) ^ 2 = 49$

Solution:

```
x = 4
```

```
y = 3
```

```
result = (x + y) ** 2
```

```
print("{} + {} ^ 2 = {}".format(x, y, result))
```

6. Write a Python program to compute the future value of a specified principal amount, rate of interest, and number of years. Test Data :  $\text{amt} = 10000, \text{int} = 3.5, \text{years} = 7$

Expected Output : 12722.79

Solution:

```
amt = 10000
```

```
interest = 3.5
```

```
years = 7
```

```
future_value = amt * (1 + (interest/100)) ** years
```

```
print("Future value: {:.2f}".format(future_value))
```

7. Write a Python program that removes duplicate elements from a given array of numbers so that each element appears only once and returns the new length of the array. Sample Input: [0,0,1,1,2,2,3,3,4,4,4]

Sample Output:

The list of unique numbers: [0,1,2,3,4]

length of the unique list : 5

Solution:

```
def remove_duplicates(arr):  
    unique_arr = list(set(arr))  
    print("The list of unique numbers:", unique_arr)  
    return len(unique_arr)  
  
input_arr = [0,0,1,1,2,2,3,3,4,4,4]  
length = remove_duplicates(input_arr)  
print("Length of the unique list:", length)
```

8. Write a Python program to sort (ascending and descending) a dictionary by key value.

Solution:

```
my_dict = {'c': 5, 'a': 3, 'b': 2, 'd': 1}
```

```
# Sort by ascending order
```

```
sorted_dict_asc = {}
```

```
for key in sorted(my_dict.keys()):
```

```
    sorted_dict_asc[key] = my_dict[key]
```

```
print("Sorted dictionary by key (ascending):")
```

```
print(sorted_dict_asc)
```

```
# Sort by descending order
```

```
sorted_dict_desc = {}
```

```
for key in sorted(my_dict.keys(), reverse=True):
```

```
    sorted_dict_desc[key] = my_dict[key]
```

```
print("\nSorted dictionary by key (descending):")
```

```
print(sorted_dict_desc)
```

9. Write a Python program to add key to a dictionary. Go to the editor

Sample Dictionary : {0: 10, 1: 20}

Expected Result : {0: 10, 1: 20, 2: 30}

Solution:

```
dict1 = {0: 10, 1: 20}
```

```
dict1[2] = 30
```

```
print(dict1)
```

10. Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels. Sample Python dictionary data and list labels:

```
exam_data = {'name': ['Arif', 'Asir', 'Arik', 'Anis', 'Anil', 'Ashish', 'Anahi', 'Alia', 'Alvin', 'Asim'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
```

Solution:

```
import numpy as np

import pandas as pd

exam_data = {'name': ['Arif', 'Asir', 'Arik', 'Anis', 'Anil', 'Ashish', 'Anahi', 'Alia', 'Alvin', 'Asim'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

df = pd.DataFrame(exam_data, index=labels)

print(df)
```