

# EX NO 1 Introduction to DataScience

December 18, 2023

[ ]: URK22AI1115

16/12/23

[ ]: Aim: To perform basic operations in python with lists, sets, tuples, dictionaries, arrays.

[ ]: Description:

## Dictionary

In Python, a dictionary can be created by placing a sequence of elements within curly {} braces, separated by 'comma'. Dictionary holds pairs of values, one being the Key and the other corresponding pair element being its Key:value.

## List

Lists are used to store multiple items in a single variable. Lists are one of 4 built-in data types in Python used to store collections of data

## Set

A Set is an unordered collection data type that is iterable, mutable and has no duplicate elements. Set define in { }

## Tuple

A Tuple is a collection of Python objects separated by commas. In some ways, a tuple is similar to a list in terms of indexing, nested objects and repetition but a tuple is immutable unlike lists which are mutable.

## Numpy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object

## .csv files

To read or access .csv file we need to use csv library.

[ ]: Q1. Create an empty dictionary and fill with some sub\_code and sub\_name as pair by user input. Then take one sub\_code as input from the user and traverse through dictionary to find the corresponding sub\_name and display the same also replace with the user given new input.

```
[2]: dict_ = {}

def fill():
    n = int(input("Enter the num of data to enter:"))
    for x in range(n):
        sub_code = input("Enter subject code: ")
        sub_name = input("Enter subject name: ")
        dict_[sub_code] = sub_name

def update():
    to_find = input("Enter subject code to find: ")
    if to_find in dict_:
        sub_name = dict_[to_find]
        print(f"The corresponding subject name is: {sub_name}")
        new_sub_name = input("Enter the new subject name: ")
        dict_[to_find] = new_sub_name
        print("Dictionary updated successfully.")
    else:
        print("Subject code not found in the dictionary.")

fill()
update()
print("Updated Dictionary:", dict_)

print("DEVENDRA KUMAR")
print("URK22AI1115")
```

Enter the num of data to enter: 3

Enter subject code: 11

Enter subject name: deva

Enter subject code: 22

Enter subject name: web

Enter subject code: 33

Enter subject name: tech

Enter subject code to find: 33

The corresponding subject name is: tech

Enter the new subject name: data

Dictionary updated successfully.

Updated Dictionary: {'11': 'deva', '22': 'web', '33': 'data'}

DEVENDRA KUMAR

URK22AI1115

```
[ ]: Q2. Create an empty list and fill with list of words by user input. Find the list of words that are longer than n from a given list of words and remove from the list then print.
```

```
[3]: def filter_long_words(words_list, n):  
    """Filter the list of words longer than n"""  
    return [word for word in words_list if len(word) > n]  
  
words_list = input("Enter the list of words separated by spaces: ").split(',')  
n = int(input("Enter the number n: "))  
  
filtered_words = filter_long_words(words_list, n)  
print("Words longer than n are:", ' '.join(filtered_words))  
  
print("DEVENDRA KUMAR")  
print("URK22AI1115")
```

```
Enter the list of words separated by spaces: devendra  
Enter the number n: 4  
  
Words longer than n are: devendra  
DEVENDRA KUMAR  
URK22AI1115
```

```
[ ]: Q3. Create an empty set and fill with some values by user input. Check if a given value is present in a set or not. If not present include it. If present remove it.
```

```
[4]: my_set = set()  
  
def fill_set():  
    n = int(input("Enter the number of values: "))  
    for _ in range(n):  
        value = input("Enter a value: ")  
        my_set.add(value)  
  
def check_and_update(value):  
    if value in my_set:  
        my_set.remove(value)  
        print(f"{value} removed from the set.")  
    else:  
        my_set.add(value)  
        print(f"{value} included in the set.")  
  
fill_set()  
  
value_to_check = input("Enter a value to check: ")
```

```

check_and_update(value_to_check)

print("DEVENDRA KUMAR")
print("URK22AI1115")

```

Enter the number of values: 2  
Enter a value: 20  
Enter a value: 40  
Enter a value to check: 20

20 removed from the set.  
DEVENDRA KUMAR  
URK22AI1115

[ ]: Q4. Create an empty tuple and fill with some values by user input. Count the occurrence of a given input number in the tuple and print the unique occurrences.

```

[5]: value_tuple = ()

num_values = int(input("Enter the number of values you want to add to the tuple:
↵ "))
values = []
for i in range(num_values):
    value = int(input("Enter a value: "))
    values.append(value)

value_tuple = tuple(values)

print("Initial tuple:", value_tuple)

n = int(input("Enter the number to count occurrences in the tuple: "))

occurrences = value_tuple.count(n)

print(f"The number {n} occurred {occurrences} time(s) in the tuple.")

print("DEVENDRA KUMAR")
print("URK22AI1115")

```

Enter the number of values you want to add to the tuple: 1  
Enter a value: 2  
Initial tuple: (2,)  
Enter the number to count occurrences in the tuple: 2

The number 2 occurred 1 time(s) in the tuple.  
DEVENDRA KUMAR  
URK22AI1115

[ ]: Q5. Create a menu driven program to perform 2D array addition, subtraction, multiplication and division using numpy.

```
[7]: import numpy as np

def get_matrix():
    rows = int(input("Enter the number of rows: "))
    columns = int(input("Enter the number of columns: "))

    matrix = []
    for i in range(rows):
        row = []
        for j in range(columns):
            element = float(input(f"Enter element at position ({i + 1}, {j + 1}): "))
            row.append(element)
        matrix.append(row)

    return np.array(matrix)

def print_result(operation, result):
    print(f"\nResult of {operation}: \n{result}\n")

while True:
    print("Menu:")
    print("1. Addition")
    print("2. Subtraction")
    print("3. Multiplication")
    print("4. Division")
    print("5. Exit")

    choice = input("Enter your choice (1-5): ")

    if choice == '1':
        matrix1 = get_matrix()
        matrix2 = get_matrix()
        result = np.add(matrix1, matrix2)
        print_result("Addition", result)

    elif choice == '2':
        matrix1 = get_matrix()
        matrix2 = get_matrix()
        result = np.subtract(matrix1, matrix2)
```

```

        print_result("Subtraction", result)

    elif choice == '3':
        matrix1 = get_matrix()
        matrix2 = get_matrix()
        result = np.matmul(matrix1, matrix2)
        print_result("Multiplication", result)

    elif choice == '4':
        matrix1 = get_matrix()
        matrix2 = get_matrix()
        result = np.divide(matrix1, matrix2)
        print_result("Division", result)

    elif choice == '5':
        print("Exiting the program. Goodbye!")
        break

    else:
        print("Invalid choice. Please enter a number between 1 and 5.")

print("DEVENDRA KUMAR")
print("URK22AI1115")

```

Menu:

1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit

Enter your choice (1-5): 5

Exiting the program. Goodbye!

DEVENDRA KUMAR

URK22AI1115

[ ]: Q6. Read an .csv file and display the basic details.

```

[20]: import pandas as pd
      path="submission.csv"
      df=pd.read_csv(path)
      df

```

```

[20]:      total_fare
0          0.0
1          0.0
2          0.0
3          0.0

```

```
4          0.0
...
89856      0.0
89857      0.0
89858      0.0
89859      0.0
89860      0.0
```

```
[89861 rows x 1 columns]
```

```
[ ]: Result:
```

```
Thus, to perform basic operations in python with lists, sets, tuples,
dictionaries, arrays is performed and executed successfully.
```