QUAID-E-AWAM UNIVERSITY OF ENGINEERING, SCIENCE & TECHNOLOGY NAWABSHAH
DEPARTMENT OF ARTIFICIAL INTELLIGENCE

PROGRAMMING FUNDAMENTALS

Lab Experiment #06

OBJECTIVE:
Advanced data types: lists in Python

TOOLS REQUIRED:
Personal computer with windows and Python installed

DESCRIPTION:
This lab deals with one of the advanced data types of Python i.e., lists. Table -1 shows a comparison among these data structures. Each one of these data structures has their own usage depending upon the requirements of the program.

Table -1

List	**Tuple**	**Set**	**Dictionary**
non-homogeneous data structure which stores the elements in single row and multiple rows and columns	non-homogeneous data structure which stores single row and multiple rows and columns	non-homogeneous data structure but stores in single row	non-homogeneous data structure which stores key value pairs
represented by [ ]	represented by ( )	represented by { }	represented by { }
allows duplicate elements	allows duplicate elements	Does not allow duplicate elements	Does not allow duplicate keys.
can use nested	can use nested	can use nested	can use nested
Example: [1, 2, 3, 4, 5]	Example: (1, 2, 3, 4, 5)	Example: {1, 2, 3, 4, 5}	Example: {1, 2, 3, 4, 5}
can be created using **list()** function	can be created using **tuple()** function.	can be created using **set()** function	can be created using **dict()** function.
List is mutable i.e we can make any changes in list.	Tuple is immutable i.e we can not make any changes in tuple	Set is mutable i.e we can make any changes in set. But elements are not duplicated.	Dictionary is mutable. But Keys are not duplicated.
List is ordered	Tuple is ordered	Set is unordered	Dictionary is ordered
Creating an empty list l=[]	Creating an empty Tuple t=()	Creating a set a=set()	Creating an empty dictionary d={}

Following lab tasks are designed to practice and learn the concepts of the above-mentioned advanced data structures of Python.

LAB TASK:

1. Open Python IDLE terminal and then create a new file. Name it "lab6_1.py". Write a program that generates a list of odd numbers from 1 to 1000, prints numbers and then prints the sum of numbers.

2. Create "lab6_2.py". Write a program that generates a list of 20 random numbers, prints the list and then finds the (1) largest number, and (2) smallest number using linear search method.

3. Create "lab6_3.py". Write a Python program that prompts the user to enter 5 numbers, stores them in a list, and then prints the sum of all even numbers in the list.

4. Create "lab6_4.py". Create a tuple containing the names of 5 countries. Write a program that iterates through the tuple and prints only the countries with a length greater than 7 characters.

5. Create "lab6_5.py". Generate a list of 50 integers. Write a program that iterates through the list and prints the square of each number if it's even, and the cube if it's odd.

6. Create "lab6_6.py". Write a program that takes two sets as input and prints the union of the two sets.

7. Create "lab6_7.py". Write a Python program to convert a list of characters into a string.

8. Create "lab6_8.py". Write a program that randomly selects an item from list_1 and then maps it to a randomly selected item of list_2

9. Create "lab6_9.py". Write a Python program that takes a list of student names and their corresponding grades as key-value pairs in a dictionary. The program should then prompt the user to enter a student name, and it should output the grade associated with that student. If the student is not found in the dictionary, the program should print a message saying "Student not found.

QUESTIONS:

Q # 1: Three lists i.e., list_1, list_2 and list_3 are initialized with 10 items each. Write Python code that will print items from the lists using *zip* method.

Ans.

Q # 2: What is the difference between append and extend methods of a list?

Ans.

Q # 3: Predict the output

```
grocery_list = ['flour','cheese','carrots']
for idx,val in enumerate(grocery_list):
    print("%s: %s" % (idx, val))
```

Ans.

Q # 4: Consider the code below:

```
items=[3,45,66,5,90,101]
items2=[55,77]
items.remove(5)
print(items)
items.extend(items2)
print(items.pop(6))
```

What will be the output of the above code?

Ans.

Name: _____

Roll #: _____

Date: _____

Subject Teacher

Remarks: