LAB # 10

TUPLE AND DICTIONARY

OBJECTIVE

Getting familiar with other data storing techniques - Tuple and Dictionary.

THEORY

Tuple:
A tuple is a sequence of immutable Python objects. Tuples are like lists, but their elements are fixed, that once a tuple is created, you cannot add new elements, delete elements, replace elements, or reorder the elements in the tuple.

Syntax for creating tuple:

```
tup1 =()   #Empty tuple
tup2 = ('physics', 'chemistry')  #Tuple of string)
tup3 = (1, 2, 3, 4, 5 )   #Tuple of integer
```

```
#Create a tuple from a list
tup4 = tuple([2 * x for x in range(1, 5)])     # (2, 4, 6, 8)
#Create Single item in tuple
tup1 = (50,)
```

Example: Accessing Values in Tuples

```
tup1 = ('physics', 'chemistry', 1997, 2000)
tup2 = (1, 2, 3, 4, 5, 6, 7 )
print ("tup1[0]: ", tup1[2])
print ("tup2[1:5]: ", tup2[1:5])
```

Output:

```
>>> %Run task1.py
tup1[2]:  1997
tup2[1:5]:  (2, 3, 4, 5)
```

Tuple Functions:

cmp(tuple1, tuple2)	Compares elements of both tuples.
len(tuple)	Gives the total length of the tuple.
max(tuple)	Returns item from the tuple with max value.
min(tuple)	Returns item from the tuple with min value.
tuple(seq)	Converts a list into tuple.

Dictionary:

A dictionary is a container object that stores a collection of key/value pairs. It enables fast retrieval, deletion, and updating of the value by using the key. A dictionary is also known as a map, which maps each key to a value.

Syntax for creating dictionary:

```
dict1={} # Create an empty dictionary
dict2={1: 'apple', 2: 'ball'} # dictionary with integer keys
```

Example: Accessing Values in Dictionary

```
my_dict = {'name':'xyz', 'age': 26}
print("value:",my_dict['name'])
print("value:",my_dict.get('age'))
```

Output:

```
>>> %Run task2.py
value: xyz
value: 26
```

Example: Adding, Modifying, Retrieving and Deleting Values

To add,modify and retrieve an item to a dictionary, use the syntax: ***dictionaryName[key] = value***

To delete an item from a dictionary, use the syntax: ***del dictionaryName[key], dictionaryName.pop(key)***

```
students = {"111-31":"John", "111-32":"Peter"}
students["111-33"] = "Susan"   # Add a new item
print("Add item:",students)
del students["111-31"]         # Delete item
print("Delete item:",students)
```

Output:

```
>>> %Run task3.py
Add item:{'111-31':'John', '111-32':'Peter', '111-33': 'Susan'}
Delete item: {'111-32': 'Peter', '111-33': 'Susan'}
```

Dictionary Methods:

keys(): tuple	Returns a sequence of keys in form of tuple.
values(): tuple	Returns a sequence of values.
items(): tuple	Returns a sequence of tuples. Each tuple is (key, value) for an item.
clear(): None	Deletes all entries.
get(key): value	Returns the value for the key.
popitem(): tuple	Returns a randomly selected key/value pair as a tuple and removes the selected item.

EXERCISE

A. Point out the errors, if any, and paste the output also in the following Python programs.

1. Code

```
t = (1, 2, 3)
t.append(4)
t.remove(0)
del tup[0]
```

Output

2. Code

```
1user_0=['username':'efermi','first':'enrico','last':'fermi',]
for key, value in 1user_0.items():
    print("\nKey: " ,key)
    print("Value: " ,value)
```

Output:

What will be the output of the following programs:

1. Code

```
tuple1 = ("green", "red", "blue")
tuple2 = tuple([7, 1, 2, 23, 4, 5])
tuple3 = tuple1 + tuple2
print(tuple3)
tuple3 = 2 * tuple1
print(tuple3)
print(tuple2[2 : 4])
print(tuple1[-1])
```

Output

2. Code

```
def main():
    d = {"red":4, "blue":1, "green":14, "yellow":2}
    print(d["red"])
    print(list(d.keys()))
    print(list(d.values()))
    print("blue" in d)
    print("purple" in d)
    d["blue"] += 10
    print(d["blue"])
main() # Call the main function
```

Output

C. Write Python programs for the following:

1. Write a program that create a buffet-style restaurant offers only five basic foods. Think of five simple foods, and store them in a tuple. (Hint:Use a for loop to print each food the restaurant offers. Also the restaurant changes its menu, replacing two of the items with different foods and display the menu again.

2. Write a program for "Guess the capitals" using a dictionary to store the pairs of states and capitals so that the questions are randomly displayed. The program should keep a count of the number of correct and incorrect responses.

3. Write a pogram that make a dictionary called favorite_places. Think of three names to use as keys in the dictionary, and store three favorite places for each person through list. Loop through the dictionary, and print each person's name and their favorite places.

Output look alike:

abc likes the following places:

- Bear Mountain
- Death Valley
- Tierra Del Fuego