UNIVERSITY OF ENGINEERING AND TECHNOLOGY PESHAWAR, JALOZAI CAMPUS

Lab 3: List, Tuple, and Dictionary

Lab Title: EE-271 “OOP & Data Structures Lab”

Note: Using the internet is encouraging for finding relevant code and modifying it for the problem at hand.

Time: 10 min/ Task

Lab Report Work

1. Make a list that contains the student name, the last four digits of the registration number as an integer, the CGPA as a float, and a list that contains GPA of all semesters. **(Nested List)**

2. Use list inducing in the following nested list and access each and every element.

```
x = ["a", ["bb", ["ccc", "ddd"], "ee", "ff"], "g", ["hh", "ii"], "j"]
```

Lab practice

List:

1. Make a list of 5 any integers.

2. Make a list that contains the student's name, the last four digits of the registration number as an integer, CGPA as a float, and a list that contains the GPA of all semesters. **(Nested List)**

3. In the above list print the student's CGPA. **(Index)**

4. Print the list elements using the while loop and use the length function. **(len)**

5. Access the list of all semester GPAs using negative indexing.

6. Make a list of the first 20 integers using the range function.

7. Make a list of odd integers from 20 to 40 using the range function.

8. Consider the student list.

   Student = [Ibrahim, Husain, Irfan, Ayub, Usman]

   Use **in** and **not in** operators to ensure that the student is in the list or not.

9. Print all the student lists in task 8 using the for loop.

10. Observe what is print, what is the first thing print, what the last thing print. Consider task 8 for the student list.

    Student [1:3] and student [2:4]

11. Make an appropriate list and use and demonstrate the use of the following mutator Methods.

    a. `.append(obj)`

    b. `.extend(iterable)`

c. .insert(index, obj)

d. .remove(obj)

e. .pop([index=-1])

12. Use list inducing in the following nested list and access each and every element.

```
x = ["a", ["bb", ["ccc", "ddd"], "ee", "ff"], "g", ["hh", "ii"], "j"]
```

Tuple:

1. Make a tuple containing first five letters of English. **(definition)**
2. Use the index to access the third and last element.
3. Make an appropriate list and tuple and use different operations that will show that list is mutable and tuple is immutable.
4. Make a tuple num = (1, 2, 3, 4).

    a. Observe the output num[2] = 9

Dictionary:

1. Consider the following structure and observe.

```
Fd= { ‘one’: ‘aik’, , ‘three’: ‘teen’, ‘ten’ : ‘das’, ‘fifteen’ : ‘pandra’}
```

- Bracket type
- Print Fd
- Fd[‘one’]
- Fd[‘ten’]
- del Fd[‘three’] and then print Fd.
- Len(Fd).
- Use the key method and print the list of keys.
- Use the value function to print the list of values.
- Observe the output of Fd.items().
- Observe the output Fd.has_key(‘one) and Fd.has_key(‘two’) and think where such type of output will help and required.
- To copy the dictionary, one may use one of the following. Verify by printing copy and alias.
    - copy = Fd.copy()
    - alias = Fd
    - Now let alias[‘three’] = ‘3’, now print alias and observe the result.
    - Also let copy[‘three’] = ‘3’, now print copy and observe the result.

2. Make an appropriate dictionary and use the following built-in function and observe the result.

   a. `.clear()`

   b. `.get()`

   c. `.item()`

   d. `.keys()`

   e. `.vallues()`

List comprehensions:

a. Consider the following code and print the square.

```
squares = []

for x in range(10):

    squares.append(x * x)
```

b. Replace the above code with the following. Print the square. This sample one-line code is called list comprehension.

```
squares = [x * x for x in range(10)]
```

c. Chane `x*x` by any function of x and observe the resulting list.

d. List comprehensions can filter values based on some arbitrary condition that decides whether or not the resulting value becomes a part of the output list.

```
even_squares = [x * x for x in range(10) if x % 2 == 0]
```

   1. The generalized syntax:

```
values = [expression for item in collection if condition]
```

e. This new list comprehension can be transformed into an equivalent for-loop:

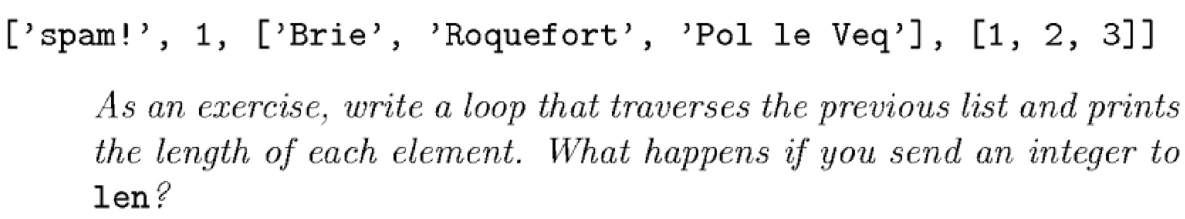
```
even_squares = []

for x in range(10):

    if x % 2 == 0:

        even_squares.append(x * x)
```

Dictionary comprehension:

a. `{ x: x * x for x in range(5) }`

Encouraging

1. This will be appreciated, and this is an indicator for strong coding skills.



```
['spam!', 1, ['Brie', 'Roquefort', 'Pol le Veq'], [1, 2, 3]]
```

As an exercise, write a loop that traverses the previous list and prints the length of each element. What happens if you send an integer to `len`*?*

2. Download the Result CSV and Pima Indians Diabetes Database from the GitHub link below.

   https://github.com/irshadarif/Pythonic-OOP/tree/main/Data_Set_CSV

   - Read the CSV as a list.
   - Read this CSV as a list of dictionaries.
   - Also calculate the mean and standard deviation. Calculation of all statistical perimeter will be a plus.

Recommended reading

1. Read from pages 81 to 112 from “How to think like a computer scientist” “learning with python” by Allen Downey.
2. Also watch the youtube lecture from this playlist (From 34 to 44). https://www.youtube.com/watch?v=UjeNA_JtXME&list=PLlRFEj9H3Oj7Bp8-DfGpfAfDBiblRfl5p