Name : Muhammad Husnain Tariq

Reg No: FA20-BCE-024

LAB 2 REPORT

In Lab Tasks:

Task 1:

Execute a simple python program to check the python installation and environment setup.

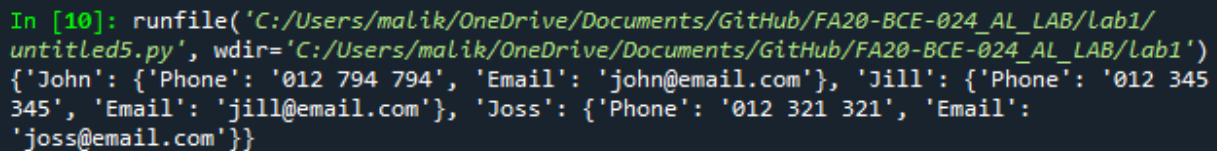\# Data Types - Dictionaries

phonebook = {}

phonebook["John"] = {"Phone": "012 794 794", "Email": "john@email.com"}

phonebook["Jill"] = {"Phone": "012 345 345", "Email": "jill@email.com"}

phonebook["Joss"] = {"Phone": "012 321 321", "Email": "joss@email.com"}

print(phonebook)

After executing the program, you should see an output similar to the following image.

```
In [10]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab1/
untitled5.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab1')
{'John': {'Phone': '012 794 794', 'Email': 'john@email.com'}, 'Jill': {'Phone': '012 345
345', 'Email': 'jill@email.com'}, 'Joss': {'Phone': '012 321 321', 'Email':
'joss@email.com'}}
```

Task 2:

Execute a simple python program to check the python installation and environment setup.

\# Data Types - Dictionaries

phonebook = {}

phonebook["John"] = {"Phone": "012 794 794", "Email": "john@email.com"}

phonebook["Jill"] = {"Phone": "012 345 345", "Email": "jill@email.com"}

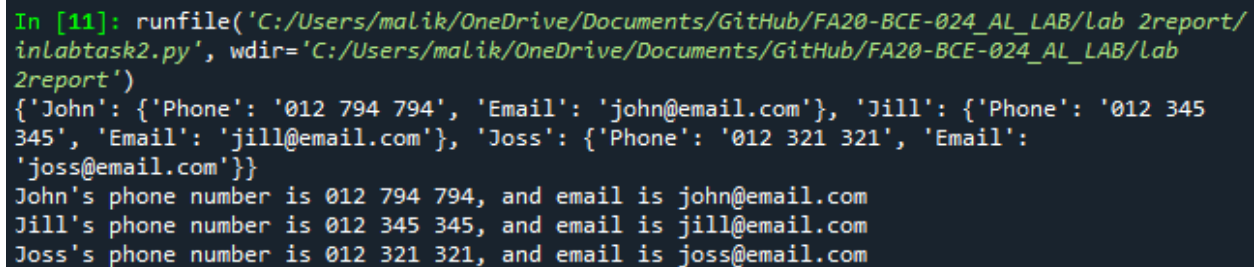phonebook["Joss"] = {"Phone": "012 321 321", "Email": "joss@email.com"}

print(phonebook)

#Using for loop to extract data fron Dictionaries

for name, record in phonebook.items():

   print("{}'s phone number is {}, and email is {}" .format(name,record["Phone"], record["Email"]))

After executing the program, you should see an output similar to the following image.

```
In [11]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask2.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
{'John': {'Phone': '012 794 794', 'Email': 'john@email.com'}, 'Jill': {'Phone': '012 345
345', 'Email': 'jill@email.com'}, 'Joss': {'Phone': '012 321 321', 'Email':
'joss@email.com'}}
John's phone number is 012 794 794, and email is john@email.com
Jill's phone number is 012 345 345, and email is jill@email.com
Joss's phone number is 012 321 321, and email is joss@email.com
```

Task 3:

Execute a simple python program to check the python installation and environment setup.

Data Types - Dictionaries

phonebook = {}

phonebook["John"] = {"Phone": "012 794 794", "Email": "john@email.com"}

phonebook["Jill"] = {"Phone": "012 345 345", "Email": "jill@email.com"}

phonebook["Joss"] = {"Phone": "012 321 321", "Email": "joss@email.com"}

print(phonebook)

```python
#Using for loop to extract data fron Dictionaries


for name, record in phonebook.items():

    print("{}'s phone number is {}, and email is {}" .format(name,record["Phone"],
record["Email"]))


# First `del`


del phonebook["John"]

for name, record in phonebook.items():

    print("{}'s phone number is {}. \ and their email is {}" .format(name, record["Phone"],
record["Email"]))


# Pop returna the record and deletes it


jill_record = phonebook.pop("Jill")

print(jill_record)

for name, record in phonebook.items():

    #you can see that only joss is still left in the system

    print("{}'s phone number is {}. \ and their email is {}" .format(name, record["Phone"],
record["Email"]))


#del phonebook["John"]
```

After executing the program, you should see an output similar to the following image.

```
In [12]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask3.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
{'John': {'Phone': '012 794 794', 'Email': 'john@email.com'}, 'Jill': {'Phone': '012 345
345', 'Email': 'jill@email.com'}, 'Joss': {'Phone': '012 321 321', 'Email':
'joss@email.com'}}
John's phone number is 012 794 794, and email is john@email.com
Jill's phone number is 012 345 345, and email is jill@email.com
Joss's phone number is 012 321 321, and email is joss@email.com
Jill's phone number is 012 345 345. \ and their email is jill@email.com
Joss's phone number is 012 321 321. \ and their email is joss@email.com
{'Phone': '012 345 345', 'Email': 'jill@email.com'}
Joss's phone number is 012 321 321. \ and their email is joss@email.com
```

Task 4:

Execute a simple python program to check the python installation and environment setup.

#Arithmatic Operators

number = 1 +2 * 3 / 4.0

print(number)

remainder = 11 % 3

print(remainder)

power

squared = 7 ** 2

print(squared)

cubed = 2 ** 3

print(cubed)

#%%

List Operators

#
==
=========================

even_numbers = [2, 4, 6, 8]

uneven_numbers = [1, 3, 5, 7]

all_numbers = uneven_numbers + even_numbers

print(all_numbers)

print([1, 2, 3] * 3)

After executing the program, you should see an output similar to the following image.

```
In [13]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask4.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
2.5
2
49
8
[1, 3, 5, 7, 2, 4, 6, 8]
[1, 2, 3, 1, 2, 3, 1, 2, 3]
```

Task 5:

Execute a simple python program to check the python installation and environment setup.

Define the two strings

greeting = "Hello, World!"

repeated_hello = "Hello " * 7

Print the strings

print(greeting)

print(repeated_hello)

After executing the program, you should see an output similar to the following image

```
In [14]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask5.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
Hello, World!
Hello Hello Hello Hello Hello Hello Hello
```

Task 6:

x = 2

print(x == 2)

print(x == 3)

print(x < 3)

name = "John"

4

print(name == "John" and x == 2)

Using `or`

print(name == "John" or name == "Jill")

Using in on lists

print(name in ["John", "Jill", "Jess"])

```
In [15]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask6.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
True
False
True
True
True
True
```

Task 7:

x = 2

y = 10

if x > 2

```python
    print("x > 2")

elif x == 2 and y > 50:

    print("x == 2 and y > 50")

elif x < 10 or y > 50:

    print("x < 10 or y > 50")

else:

    print("Nothing worked.")

name_list1 = ["John", "Jill"]

name_list2 = ["John", "Jill"]

print (not (name_list1 == name_list2))

# Using `is`

name2 = "John"

print(name_list1 == name_list2)

print(name_list1 is name_list2)
```

```
In [16]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask7.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
x < 10 or y > 50
False
True
False
```

Task 8:

```python
numeric_data  = [10, 20, 30, 40, 50]

for number in numeric_data:

    result = number*2 # Perform some operation (e.g., multiplication) print (result) Print the result

print(result)

#Sample string
```

```python
text = "Hello, World!"

#Using a for loop to read and print each character in the string

for char in text:

    print(char)

new_text = ""

for char in text:

    new_text += char.upper() # Convert letters to uppercase else: new_text += char #Keep non-letter characters as they are

    print(new_text)

#Writing Numeric Data

numeric_data = []

for i in range(1,11):

    numeric_data.append(i)

print(numeric_data)
```

```
In [17]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
inlabtask8.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
100
H
e
l
l
o
,

W
o
r
l
d
!
```

```
H
HE
HEL
HELL
HELLO
HELLO,
HELLO,
HELLO, W
HELLO, WO
HELLO, WOR
HELLO, WORL
HELLO, WORLD
HELLO, WORLD!
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Task 9:

count = 1

while count <= 5:

   print(count)

count += 1

2. For Strings

Using a while loop to print each character of a string text= "Hello"

text = "Hello"

index = 0

while index < len(text):

   print(text[index])

   index += 1

student_grades = {"Alice": 92, "Bob": 85, "Charlie": 78}

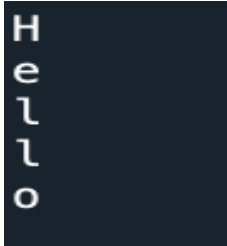keys = list(student_grades.keys()) # Get the keys as a list

index = 0

while index < len (keys):

```
    key = keys[index]

value = student_grades[key]

print (f" (key): {value}")

index += 1
```


Post Lab:

Task:

```python
# Initialize a dictionary to store student names and grades

student_grades = {}


# Define the number of students

num_students = 7


# Input student names and grades

for _ in range(num_students):

    name = input("Enter student name: ")

    grade = float(input("Enter student grade: "))

    student_grades[name] = grade


# Calculate and display the average grade
```

```python
total_grade = sum(student_grades.values())

average_grade = total_grade / num_students

print(f"Average grade: {average_grade:.2f}")


# Categorize each student's grade and display

for student, grade in student_grades.items():

    if grade >= 90:

        category = "Excellent"

    elif grade >= 80:

        category = "Very Good"

    elif grade >= 70:

        category = "Good"

    else:

        category = "Needs Improvement"


    print(f"{student}: {grade} ({category})")


# Search for a specific student's grade
while True:

    search_name = input("Enter student name to search for (or 'quit' to exit): ")


    if search_name.lower() == 'quit':

        break
```

if search_name in student_grades:

    print(f"{search_name}'s grade: {student_grades[search_name]}")

else:

    print("Student not found. Please enter a valid name.")

```
In [21]: runfile('C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab 2report/
postlab.py', wdir='C:/Users/malik/OneDrive/Documents/GitHub/FA20-BCE-024_AL_LAB/lab
2report')
Enter student name: Husnain
Enter student grade: 70
Enter student name: Uzair
Enter student grade: 80
Enter student name: Asad
Enter student grade: 90
Enter student name: Ali
Enter student grade: 70
```

```
Enter student name: Sidra
Enter student grade: 80
Enter student name: Omar
Enter student grade: 60
Enter student name: Bashir
Enter student grade: 80
Average grade: 75.71
Husnain: 70.0 (Good)
Uzair: 80.0 (Very Good)
Asad: 90.0 (Excellent)
Ali: 70.0 (Good)
Sidra: 80.0 (Very Good)
Omar: 60.0 (Needs Improvement)
Bashir: 80.0 (Very Good)
Enter student name to search for (or 'quit' to exit):
```