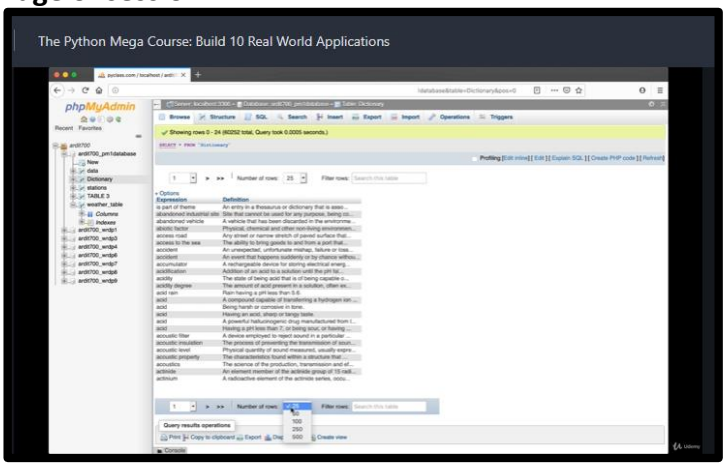
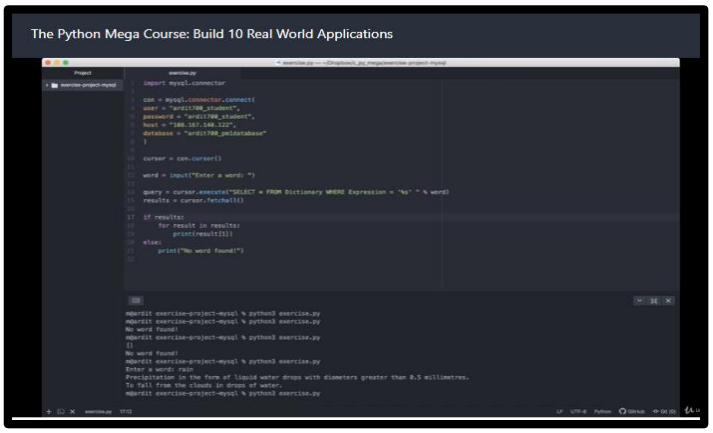
DAILY ASSESSMENT FORMAT

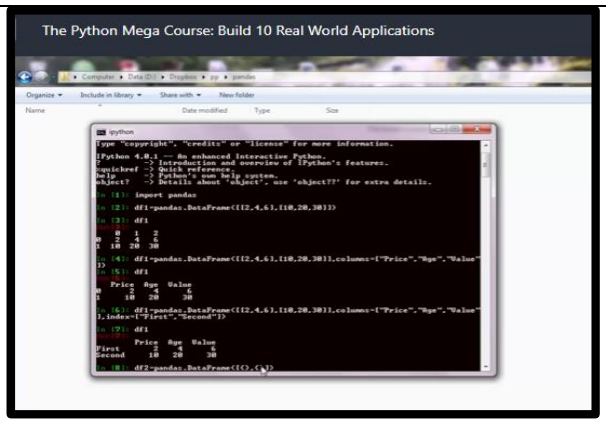
Date:	**21/05/2020**	**Name:**	**Shilpa S**
Course:	**Python**	**USN:**	**4AL14EC078**
Topic:	**Project Exercise with Python and MySQL: Interactive English Dictionary Data Analysis with Pandas**	**Semester & Section:**	**8th – A sec**
Github Repository:	**Shilpa-online**		

FORENOON SESSION DETAILS

Image of session







Report –

Project Exercise with Python and MySQL: Interactive English Dictionary

Introduction to the App & Making of the App –

Code:

```
import mysql.connector

con = mysql.connector.connect(
user = "ardit700_student",
password = "ardit700_student",
host = "108.167.140.122",
database = "ardit700_pm1database"
)

cursor = con.cursor()

word=input("Enter the word: ")

query = cursor.execute("SELECT Definition FROM Dictionary WHERE Expression = '%s'" % word)
results = cursor.fetchall()

if results:
    for result in results:
        print(result[0])
else:
    print("No word found!")
```

Output –

```
Enter the word:  rain
Precipitation in the form of liquid
water drops with diameters greater
than 0.5 millimetres.
To fall from the clouds in drops of
water.
```

In the example we used the following SQL statement in our Python code:

```
query = cursor.execute ("SELECT * FROM Dictionary WHERE Expression = 'rain'")
```

This statement retrieved all the rows of the *Dictionary* table where the value of the column *Expression* was *rain*. The string inside *cursor.execute ()* is SQL code that Python sends to the database. That kind of language is understood by the database.

Data Analysis with Pandas –

Installing Pandas

Make sure you have pandas installed. You can install it with pip:

`pip install pandas` or `pip3 install pandas`

Also, in the next lecture, we will use an enhanced Python interactive shell called IPython. IPython is just like the normal shell you get when you run `python`, but IPython provides better printing for large text. This ability makes IPython suitable for data analysis because the program prints data in a well-structured format. You can install IPython with pip:

`pip install ipython` or `pip3 install ipython`

Loading JSON Files

In the previous lecture you learned that you can load a CSV file with this code:

```
1. import pandas
2. df1 = pandas.read_csv("supermarkets.csv")
```

Try loading the `supermarkets.json` file for this exercise using `read_json` instead of `read_csv`.

The supermarkets.json file can be found inside the supermarkets.zip file attached in the previous lecture.

The code for loading the `supermarkets.json` file in Python with pandas would be this:

```
1. import pandas
2. df2 = pandas.read_json("supermarkets.json")
```

The `df2` data frame should contain this data:

	ID	Address	City	State	Country	Name	Employees
0	1	3666 21st St	San Francisco	CA 94114	USA	Madeira	8
1	2	735 Dolores St	San Francisco	CA 94119	USA	Bready Shop	15
2	3	332 Hill St	San Francisco	California 94114	USA	Super River	25
3	4	3995 23rd St	San Francisco	CA 94114	USA	Ben's Shop	10
4	5	1056 Sanchez St	San Francisco	California	USA	Sanchez	12
5	6	551 Alvarado St	San Francisco	CA 94114	USA	Richvalley	20

Loading Excel Files

To load Excel (.xlsx) files in Python with pandas. Pandas may require the xlrd library as a dependency. If you get an error such as `ModuleNotFoundError: No module named 'xlrd'`, you can fix the error by installing xlrd: `pip install xlrd` or `pip3 install xlrd`

Geopy -

Geopy makes it easy for Python developers to locate the coordinates of addresses, cities, countries, and landmarks across the globe using third-party geocoders and other data sources.

Geopy includes geocoder classes for the Open Street Map Nominatim, Google Geocoding API (V3), and many other geocoding services. The full list is available on the Geocoders doc section. Geocoder classes are located in geopy. Geocoders.