Big Data Hadoop and Spark Developer

Lesson-End Project Solution

simpl¦learn

Get Certified. Get Ahead.

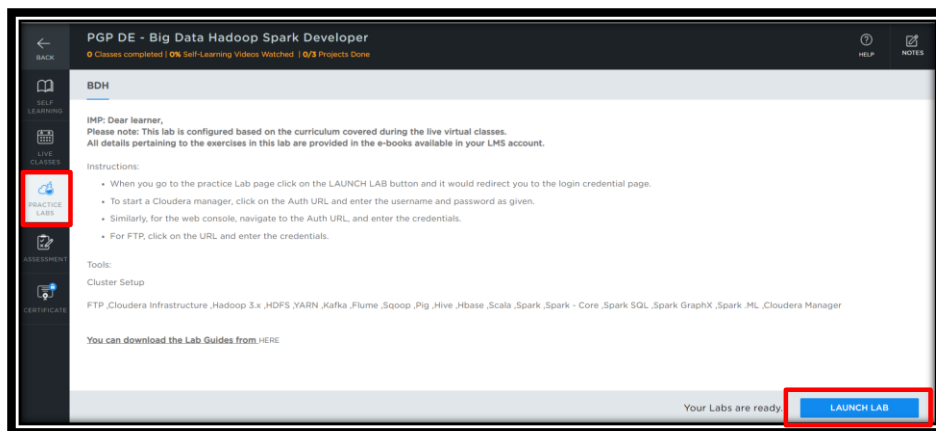**Banking Data Standardization in Python**

Steps to Perform:

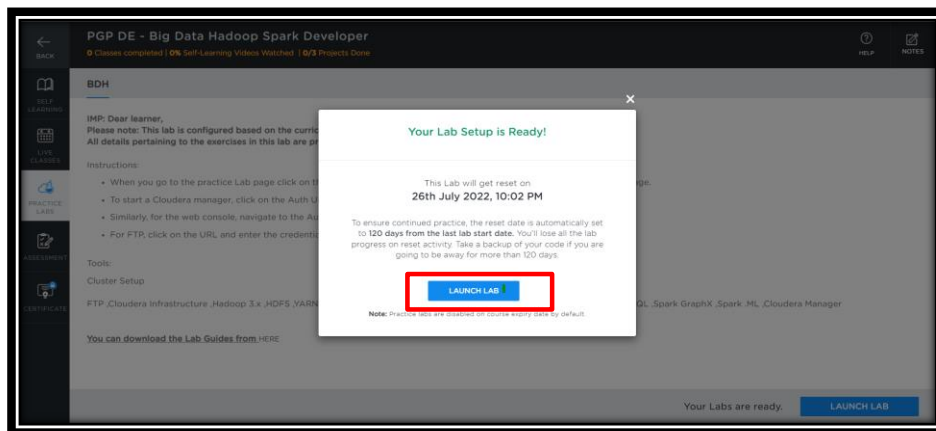**Step 1:** Log in to your LMS account

Step 2: Open the course **"Big Data Hadoop and Spark Developer"**

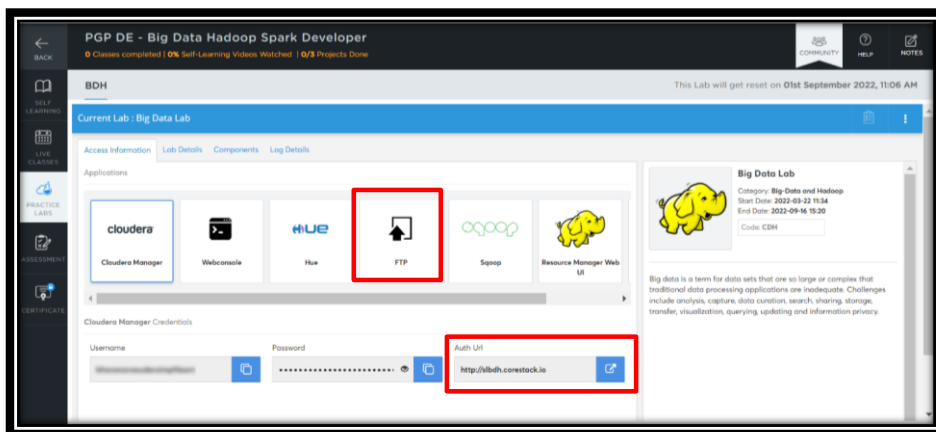**Step 3:** Download the Lesson 13 datasets from the **Course Resources** section

Step 4: Click on the **"PRACTICE LABS"** tab on the left side and select **"LAUNCH LAB"**

Step 5: Click on the **"LAUNCH LAB"** button

Step 6: Click on **"FTP"** to upload the datasets

STEP 7: Log in to **"FTP"** and click on **"Upload files"** to upload the CSV file downloaded

STEP 8: Click on **"Webconsole"** and then on **"Auth Url"**

Step 9: Copy the "**Username"** and "**Password"** provided to log in to the

"**Webconsole"**

Step 10: Paste the "**Username"** and "**Password"** on the console and click on Enter

Note: The password will not be visible when pasted on the console.

Step 11: Log in to the Python shell

Command:

python3

```
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$ python3
Python 3.7.3 (default, Mar 27 2019, 22:11:17)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Step 12: Import the CSV package into the Python shell

Command:

Import csv

```
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$ python3
Python 3.7.3 (default, Mar 27 2019, 22:11:17)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import csv
>>>
```

Step 13: Read the CSV file and save the data in a list

Command:

  data_list=[]

  with open ('Lesson_13_Dataset.csv', 'r') as file:

   csv_reader = csv.reader(file, delimiter=',')

   for row in csv_reader:

     data_list.append(row)

```
>>> data_list=[]
>>> with open ('Lesson_13_Dataset.csv', 'r') as file:
...     csv_reader = csv.reader(file, delimiter=',')
...     for row in csv_reader:
...         data_list.append(row)
...
>>>
```

Step 14: Show five records from the list by creating and calling the function

 Command:

def show(data_list):

   for row in range(0,5):

     print(data_list[row])

show(data_list)

```
>>> def show(data_list):
...       for row in range(0,5):
...           print(data_list[row])
...
>>> show(data_list)
['geo', 'name', 'time', 'population']
['afg', 'Afghanistan', '1800', '3280000']
['afg', 'Afghanistan', '1801', '3280000']
['afg', 'Afghanistan', '1802', '3280000']
['afg', 'Afghanistan', '1803', '3280000']
>>>
```