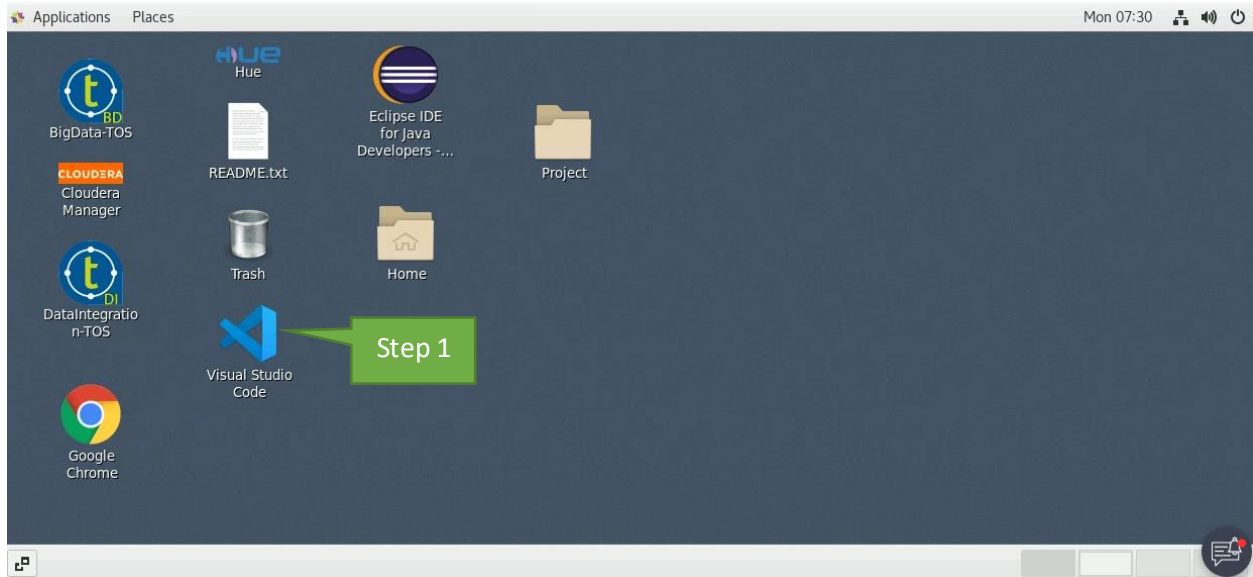
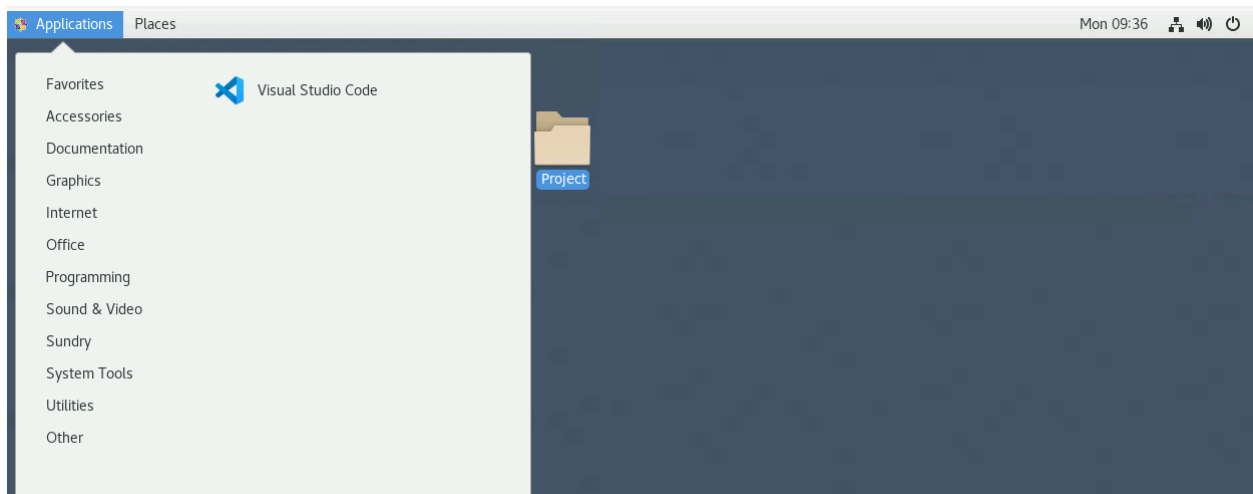

Guide to use VSCODE for PySpark development, sample test execution and making final submission of solution.

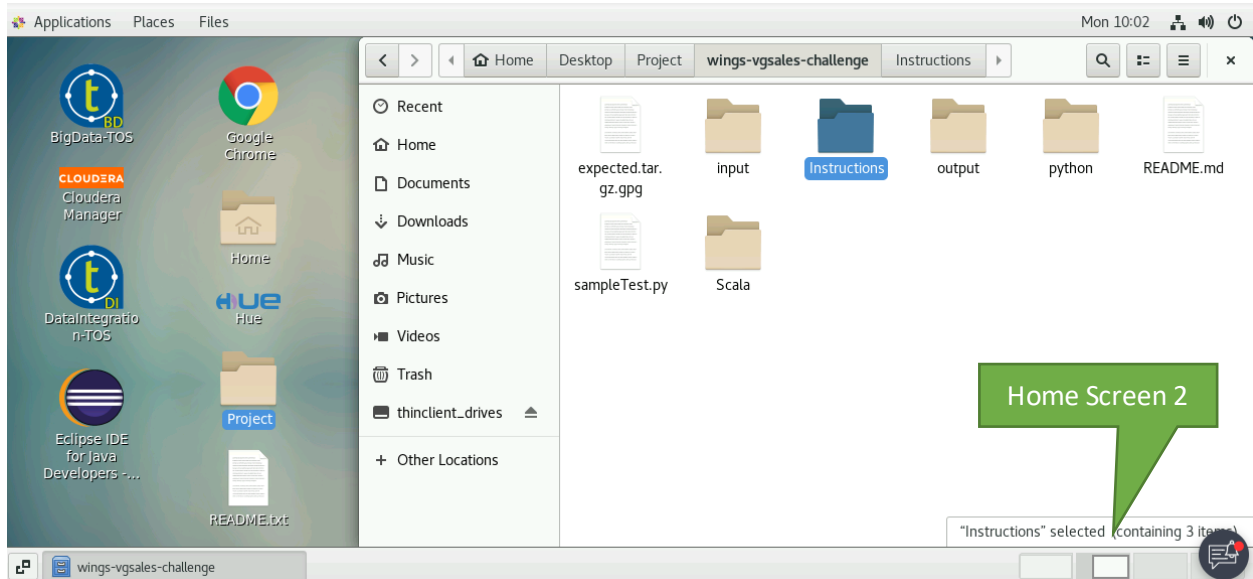
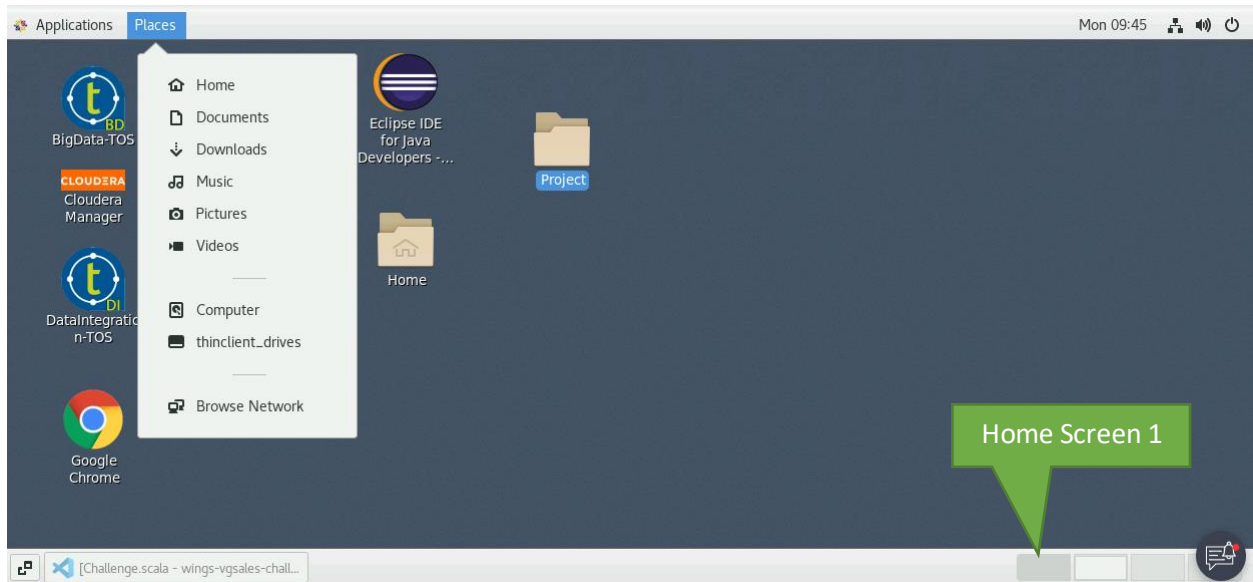
Step 1 – Launch Visual Studio Code from Desktop. Please give couple of minutes to launch to complete.



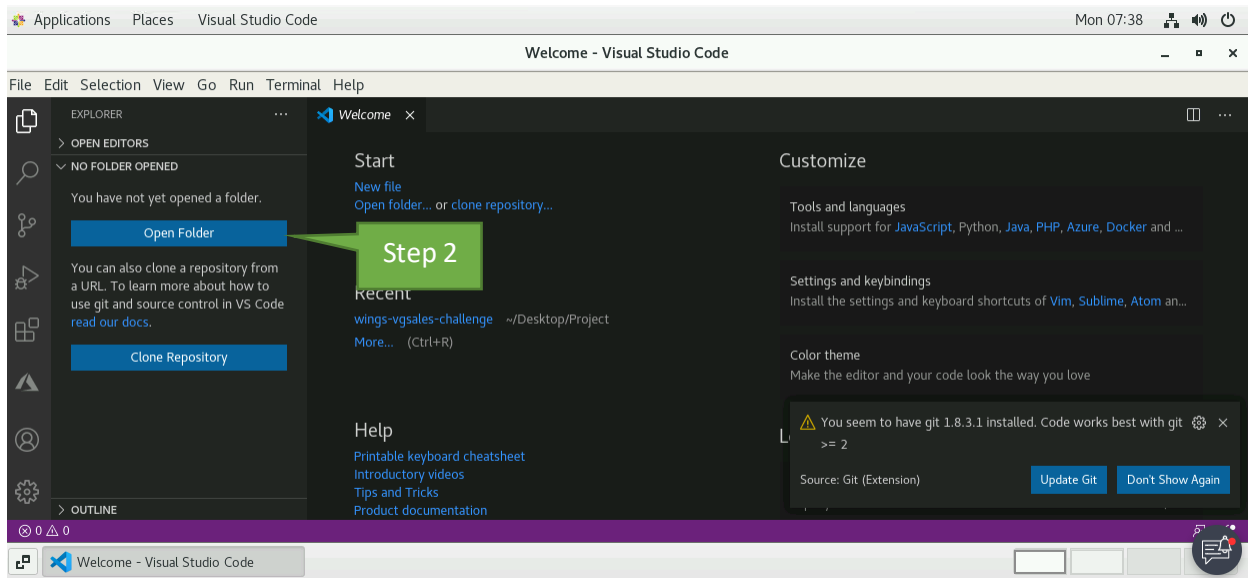
Alternatively, you can launch from “Application -> Programming” menu as shown below.



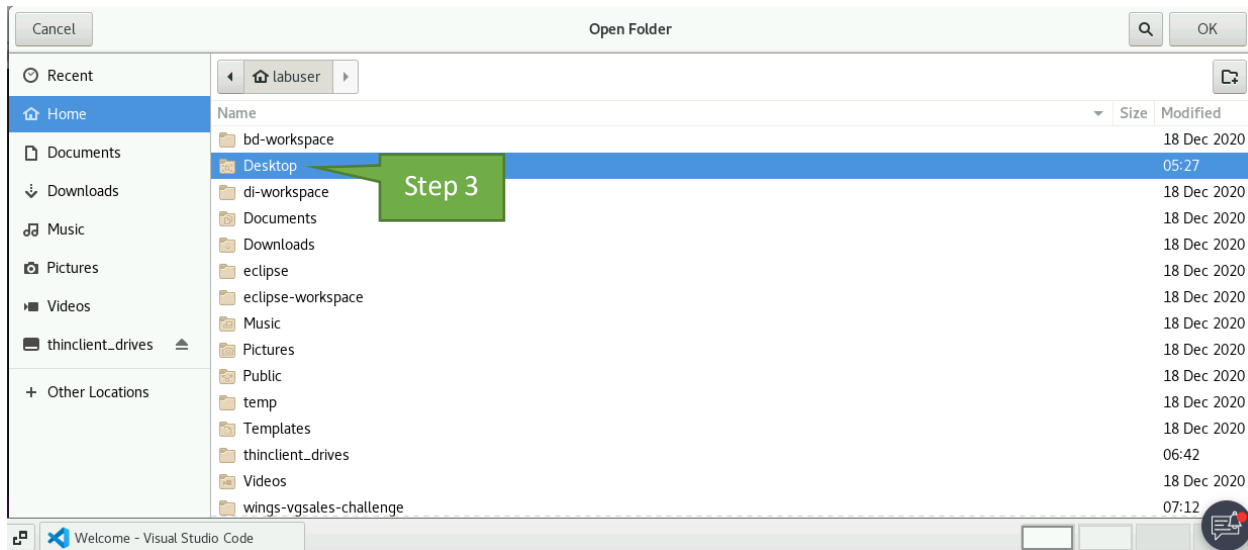
The “Places” menu provides navigational links to other directories. Usually, you have 4 home screens that you can choose as shown in bottom right corner. It may be useful to open VSCODE in one screen and project instructions in another



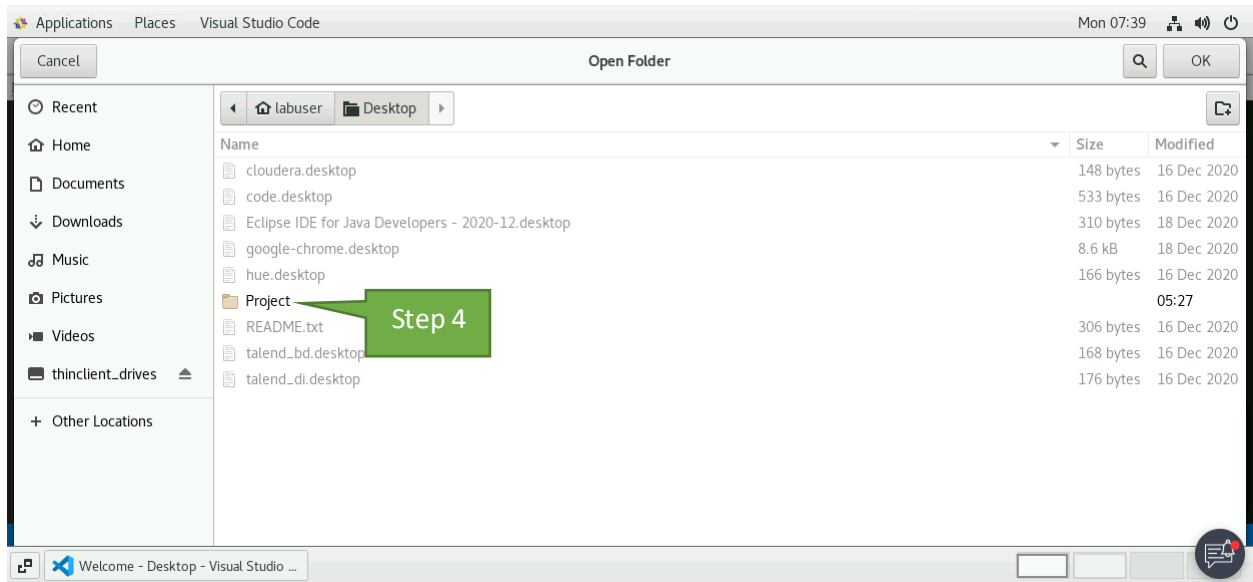
Step 2 – Click Open folder from VSCode home screen



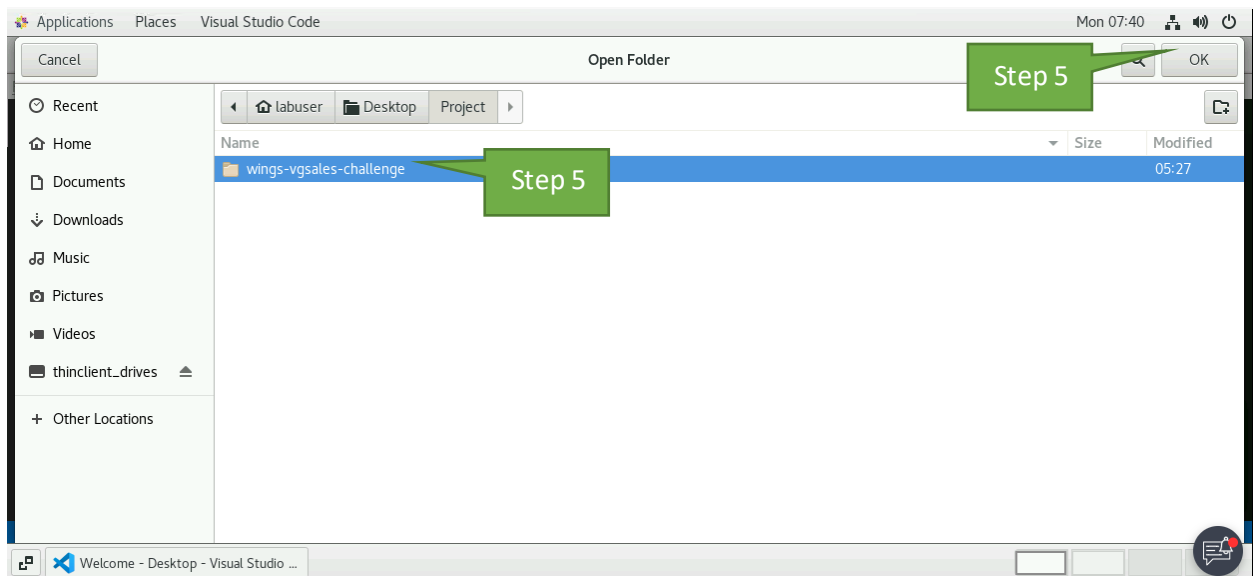
Step 3 – Traverse and double click Desktop folder



Step 4 – Traverse and double click Project folder

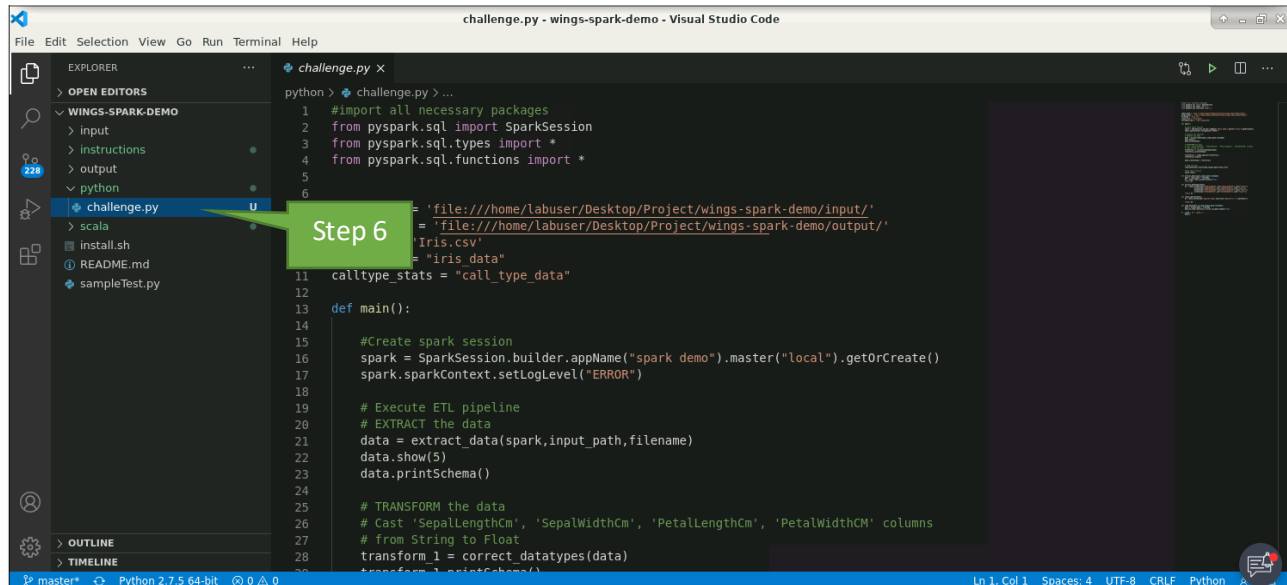


Step 5 – Click on the wings-vgsales-challenge folder and select “OK”. **Note – This is just for instructional purpose. The folder name may vary during the exam.**

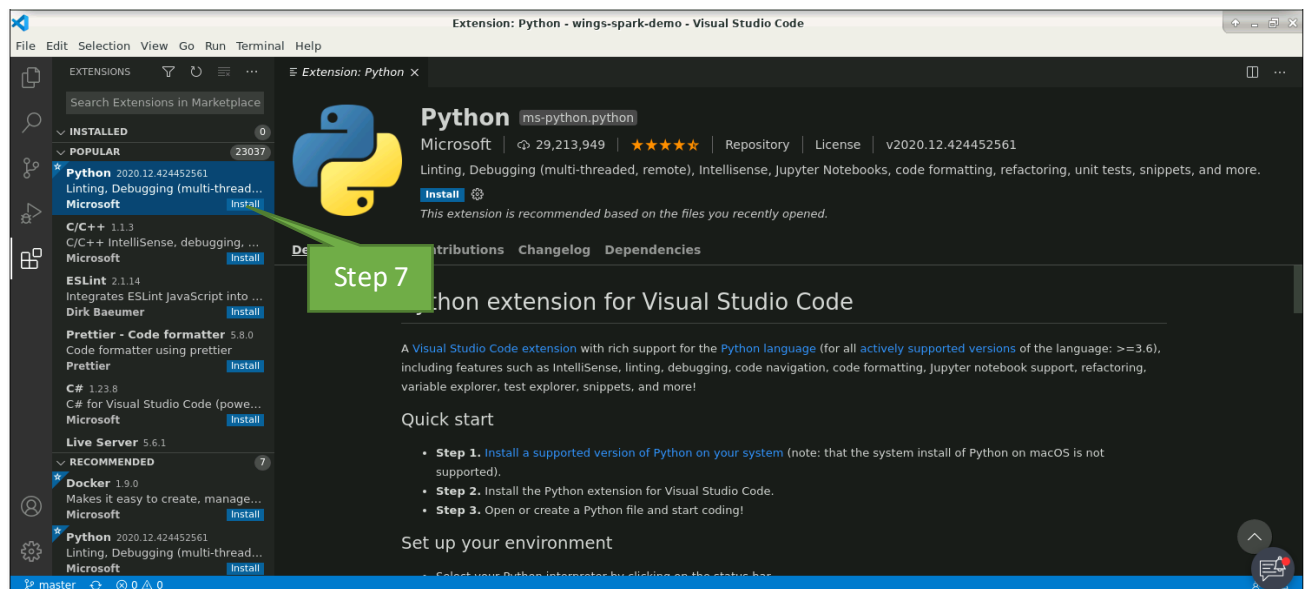


Step 6 – Open the “challenge.py” file within “python” folder

Note: This file contains code blocks which is left intentionally blank. But most of the program flow is already built for you. You need to fill in the code in the blank section where you are specifically asked to as per the problem statement.



Step 7 – Python extension may already be installed for you. If not, please install the extension as shown below.



Step 8 – As shown below (right) you are provided with a file with incomplete code block. You are required to read the problem statement from “Challenge.html” and find the function where you need to add your code in “challenge.py”. Complete the code as per the instructions. Example for Problem 1 is shown below.

Problem Statement from Challenge.html

Code implementation in challenge.py

Problem Statements:

PROBLEM 1:

Extract the input csv data from local storage into a dataframe.

Instructions:

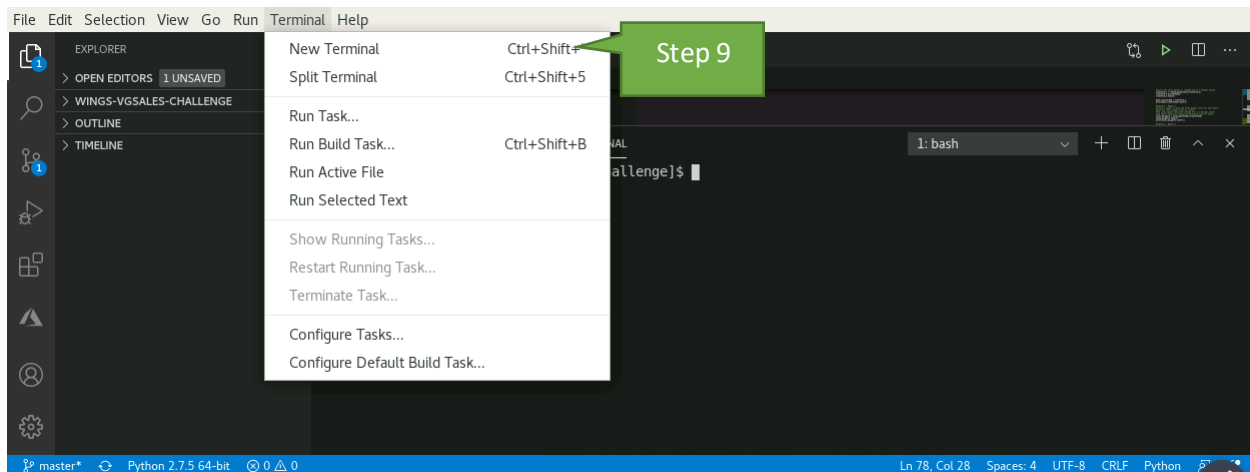
- Complete the function “extract_data” to solve this problem.
- Following details have already been defined for you in the program
 - Input File name: “vgsales.csv”
 - Input File path: “Project/wings-vgsales-challenge/input/”
 - Schema: It is already defined and stored as “schema”

Step 8

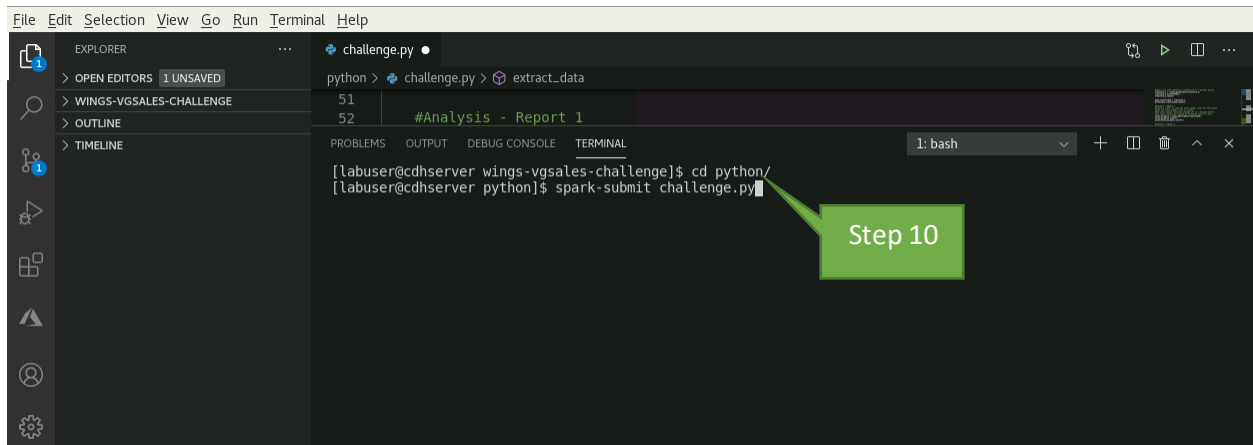
```
python > challenge.py > ...
67
68
69 # LOAD the data
70 load_data(data_transformed,output_path,final_file)
71 load_data(sales_by_genre,output_path,report1)
72 load_data(sport_max_sales,output_path,report2)
73
74 #Stop spark session
75 spark.stop()
76
77 def extract_data(spark,input_path,filename):
78     #WRITE YOUR CODE BELOW THIS LINE
79
80     #WRITE YOUR CODE ABOVE THIS LINE
81     return df
82
```

Step 8

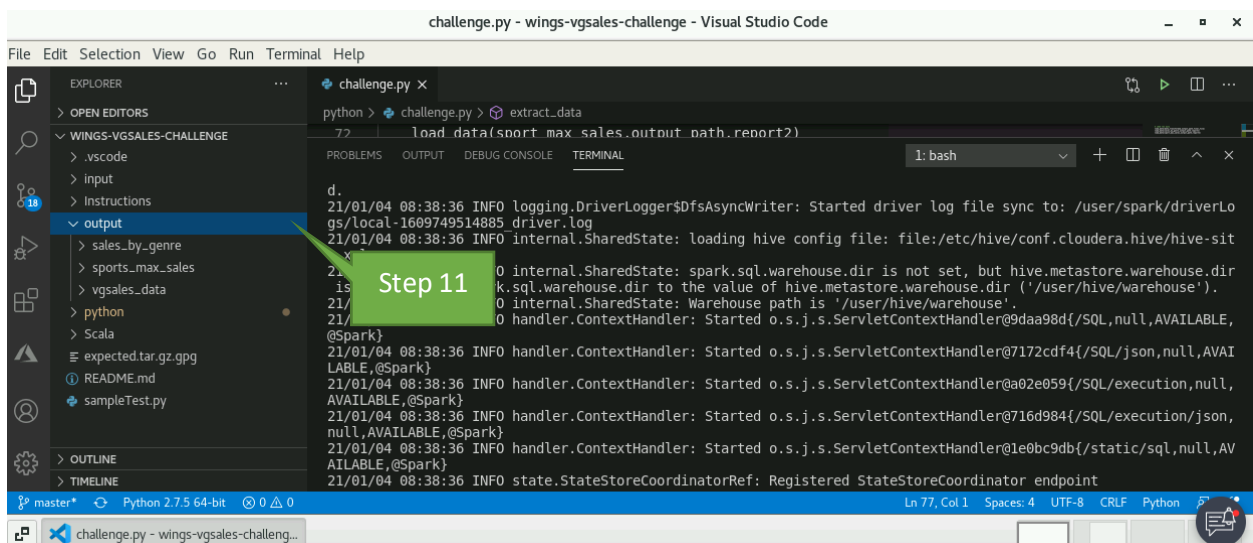
Step 9 – Once you complete your code and when ready to execute it, open a new terminal by clicking “Terminal-> New Terminal” as shown below.



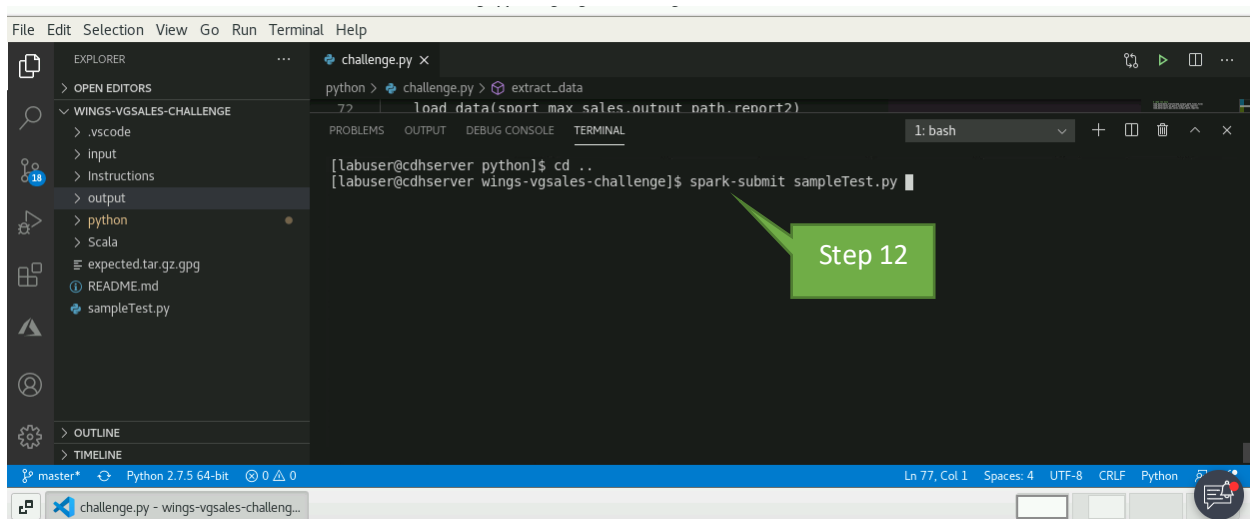
Step 10 – Make sure you are inside the “python” folder. Else navigate into “python” folder by executing command “cd python” in terminal. Once within the folder, submit the “challenge.py” file to cluster by executing the command “spark-submit challenge.py” in terminal.



Step 11 – On successful execution of the job, the required output files should be created under the “output” folder as shown below.



Step 12 – You have an option to execute sample test and confirm if you are on right track. **Note:** as name indicates these are just sample tests. Actual scoring will validate various aspects of your code and data. Make sure you are inside the challenge folder. Else navigate into that folder by executing command “cd ..” in terminal (assuming you come here after Step 10, 11). Once within the folder, submit the sampleTest.py file to cluster by executing the command “spark-submit sampleTest.py” in terminal.



The screenshot shows the VS Code interface with the Explorer panel on the left displaying the file structure of the 'WINGS-VGSALES-CHALLENGE' project. The file 'sampleTest.py' is highlighted. The main editor shows the 'challenge.py' file with the following code:

```
python > challenge.py > extract_data
77 | load_data(sport_max_sales.output_path.report2)
```

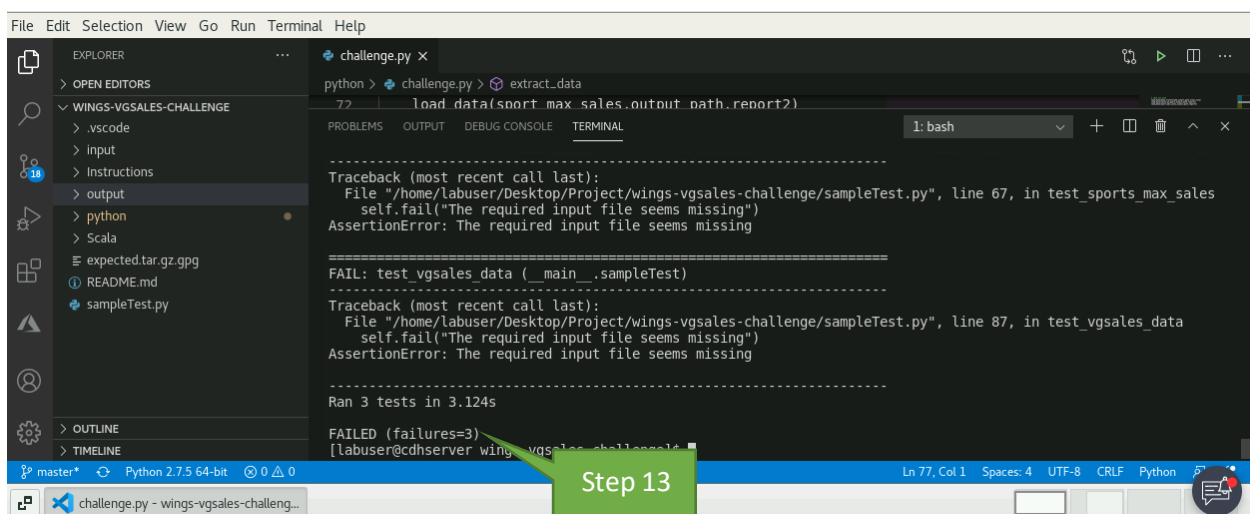
The TERMINAL panel at the bottom shows the following commands and output:

```
[labuser@cdhserver python]$ cd ..
[labuser@cdhserver wings-vgsales-challenge]$ spark-submit sampleTest.py
```

A green callout box labeled 'Step 12' points to the terminal output.

Step 13 – The results of execution is displayed in terminal. If any error found, make sure you revisit the code.

Example of tests failed



The screenshot shows the VS Code interface with the Explorer panel on the left displaying the file structure of the 'WINGS-VGSALES-CHALLENGE' project. The file 'sampleTest.py' is highlighted. The main editor shows the 'challenge.py' file with the following code:

```
python > challenge.py > extract_data
77 | load_data(sport_max_sales.output_path.report2)
```

The TERMINAL panel at the bottom shows the following output:

```
Traceback (most recent call last):
  File "/home/labuser/Desktop/Project/wings-vgsales-challenge/sampleTest.py", line 67, in test_sports_max_sales
    self.fail("The required input file seems missing")
AssertionError: The required input file seems missing

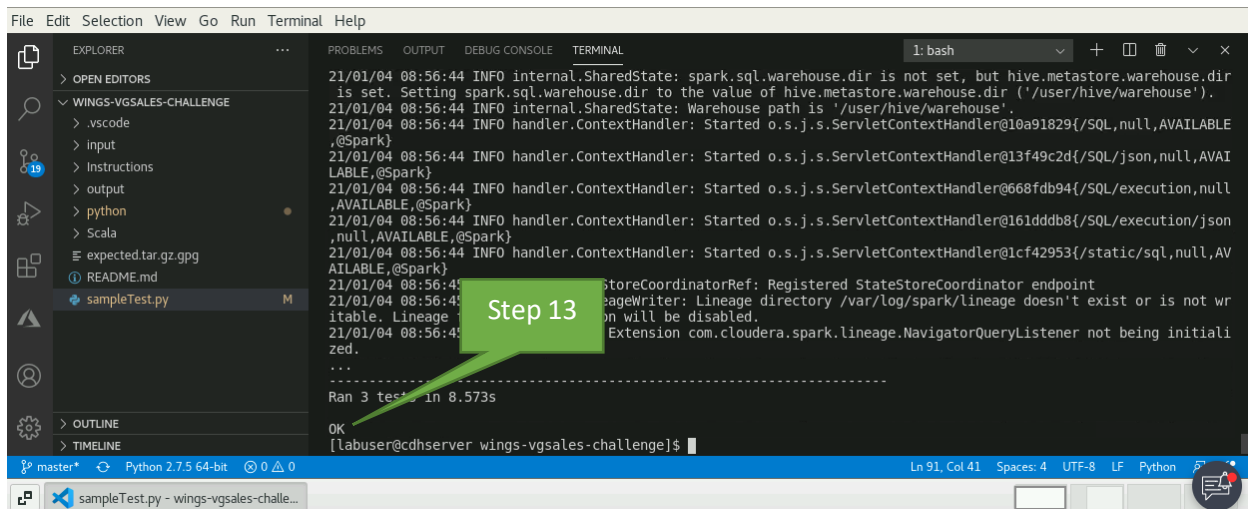
=====
FAIL: test_vgsales_data (_main_.sampleTest)
=====
Traceback (most recent call last):
  File "/home/labuser/Desktop/Project/wings-vgsales-challenge/sampleTest.py", line 87, in test_vgsales_data
    self.fail("The required input file seems missing")
AssertionError: The required input file seems missing

=====
Ran 3 tests in 3.124s

FAILED (failures=3)
[labuser@cdhserver wing_vgsales_challenge]$
```

A green callout box labeled 'Step 13' points to the terminal output.

Example of tests passed



The screenshot shows a VS Code interface with a terminal window open. The terminal displays the output of running tests for a file named 'sampleTest.py'. The output shows several INFO messages from the Spark SQL engine, indicating that the tests passed successfully. A green callout box labeled 'Step 13' points to the terminal output.

```
File Edit Selection View Go Run Terminal Help
EXPLORER
> OPEN EDITORS
WINGS-VGSALLES-CHALLENGE
> .vscode
> input
> Instructions
> output
> python
> Scala
expected.tar.gz.gpg
README.md
sampleTest.py M
OUTLINE
TIMELINE

21/01/04 08:56:44 INFO internal.SharedState: spark.sql.warehouse.dir is not set, but hive.metastore.warehouse.dir
is set. Setting spark.sql.warehouse.dir to the value of hive.metastore.warehouse.dir ('/user/hive/warehouse').
21/01/04 08:56:44 INFO internal.SharedState: Warehouse path is '/user/hive/warehouse'.
21/01/04 08:56:44 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@10a91829{/SQL,null,AVAILABLE
,@Spark}
21/01/04 08:56:44 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@13f49c2d{/SQL/json,null,AVAI
LABLE,@Spark}
21/01/04 08:56:44 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@668fdb94{/SQL/execution,null
,AVAILABLE,@Spark}
21/01/04 08:56:44 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@161dddb8{/SQL/execution/json
,null,AVAILABLE,@Spark}
21/01/04 08:56:44 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@1cf42953{/static/sql,null,AV
AILABLE,@Spark}
21/01/04 08:56:44 INFO StateStoreCoordinatorRef: Registered StateStoreCoordinator endpoint
21/01/04 08:56:44 INFO LineageWriter: Lineage directory /var/log/spark/lineage doesn't exist or is not wr
itable. Lineage tracking will be disabled.
21/01/04 08:56:44 INFO Extension com.cloudera.spark.lineage.NavigatorQueryListener not being initiali
zed.
...
Ran 3 tests in 8.573s
OK
[labuser@cdhserver wings-vgsales-challenge]$
```

Step 14 – Once you are all set, you can submit your code by clicking “Submit” button on top right corner and then clicking “CONFIRM” as shown below. Once you submit, you may not be able to take the test again. Please **note** that at the end of the timer, the solution will be auto submitted for scoring if not manually submitted.

