# System Requirements Specification Index

For

# Pyspark Usecase

## Car sales analysis L1

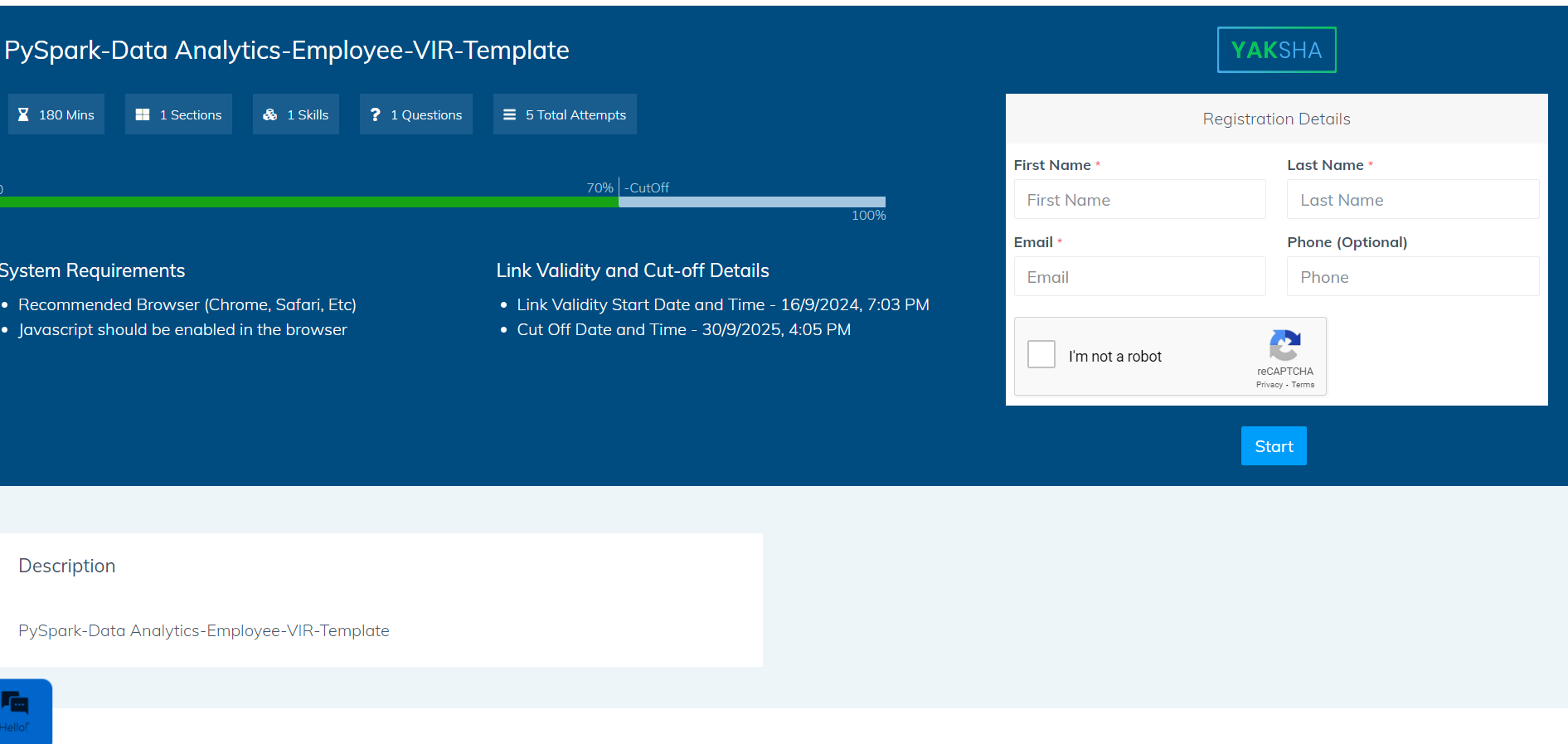
1.0

## IIHT Pvt. Ltd.

[fullstack@iiht.com](mailto:fullstack@iiht.com)

**Step to access the work environment**

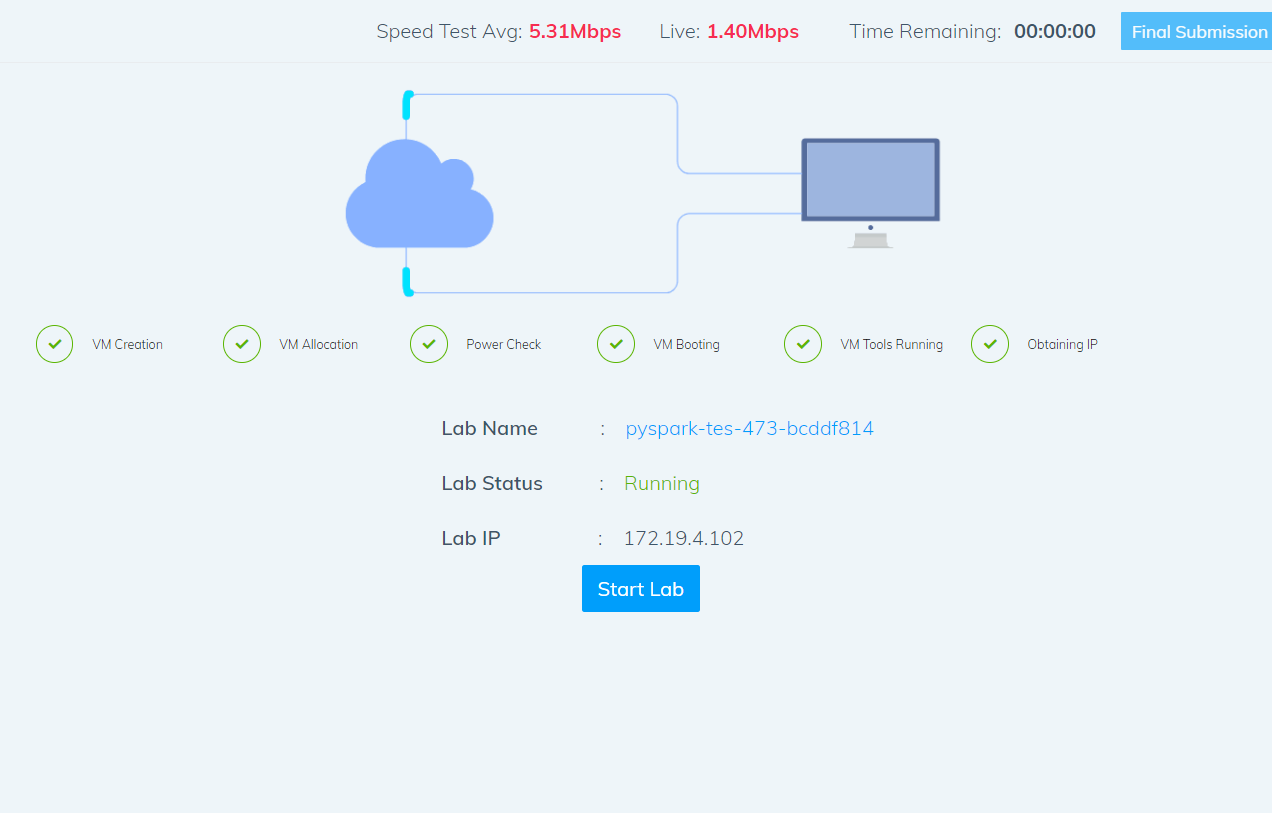
Step 1 use the URL to login provide the username and password



Step 2 Click on the launch assessment Environment

A screenshot of a computer

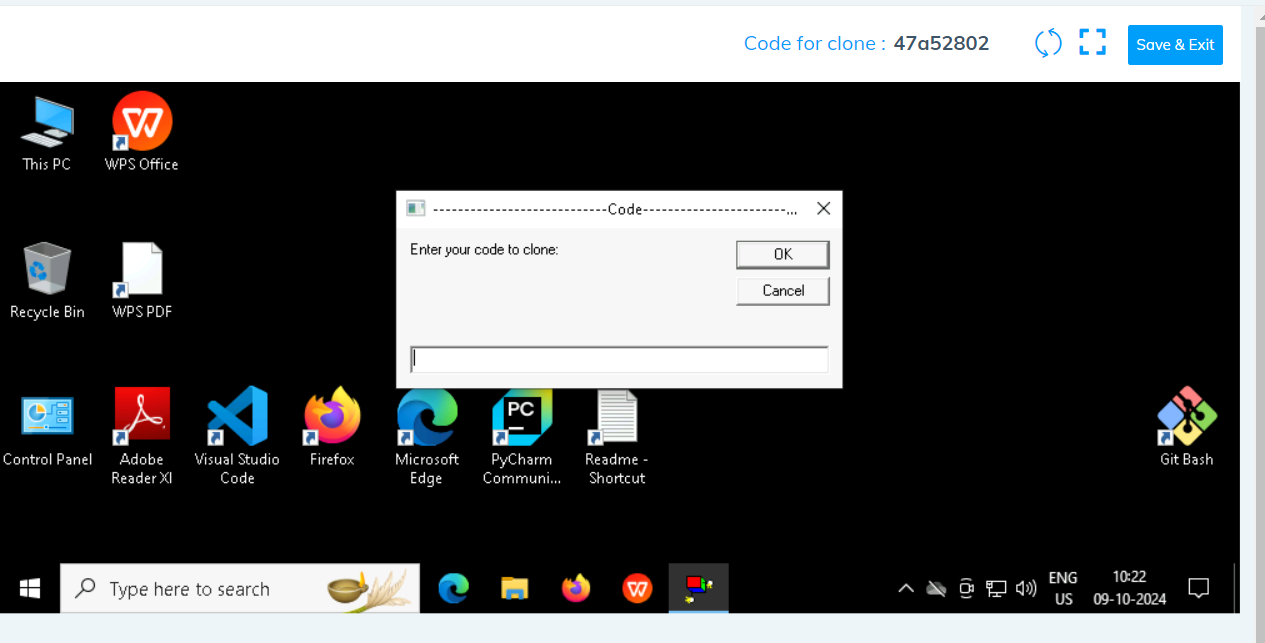
Description automatically generated



Step 3 Click on the start lab button

Step 4 you will get a window you need to type the code from that top corner

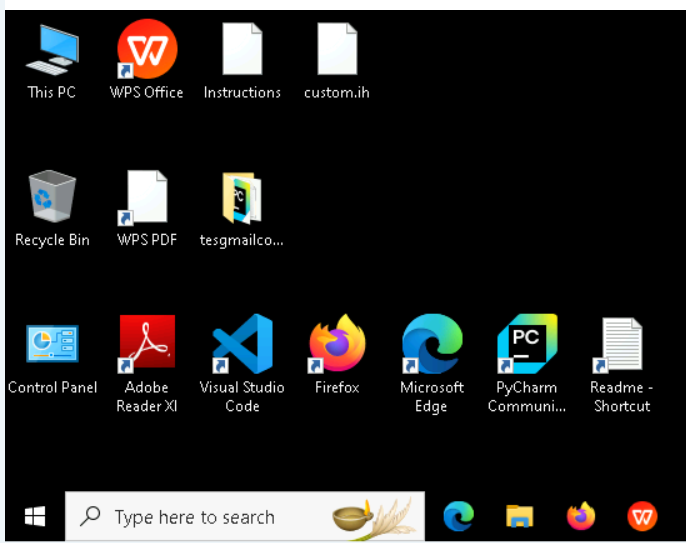
* You need to type the code in the window . It will take few minutes to start the window



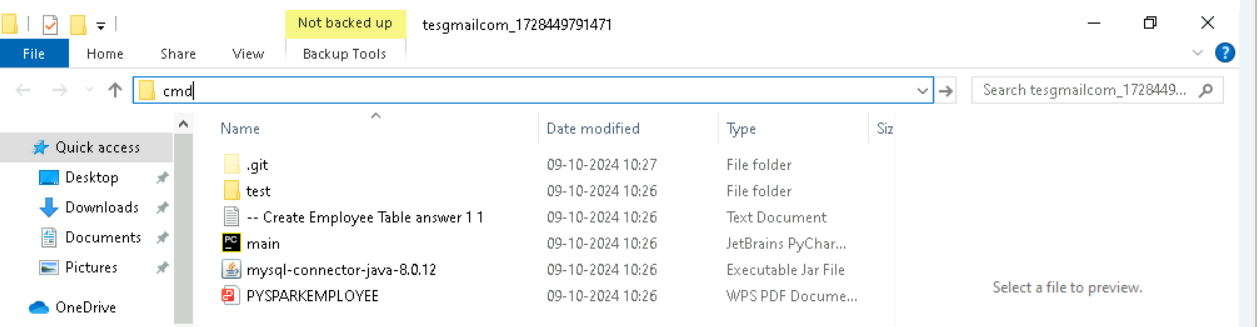
Click on ok

Step 5 after few seconds we can see that the your folder is cloned in the desktop .

Folder cloned



Step 6 go inside the folder type cmd in the top of the file explorer



* Type code. And hit enter you can see that workspace is opened in the visual code

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* You can see that workspace is ready to code

Note Please only work with visual code not with any other IDE

* In the folder cloned you will have all the project files needed .

Problem Statement : **car sales analysis with pyspark**

Description : Use relevant methods operations to perform specified activities which are given in the instructions.

#### PYSPARK TASK -L1

#### The Car Sales Data Analysis program processes a dataset containing car sales information. It performs operations such as calculating total sales, identifying top-selling models, and checking for the presence of data (non-empty dataset). The system ensures accurate analysis and prevents running calculations on empty datasets. The use case also includes boundary and exceptional test cases to ensure system robustness.

#### Primary Actor:

#### System: The Car Sales Analysis system that processes sales data to generate insights such as total sales, average prices, and best-performing models.

#### Stakeholders:

#### Business Analysts: Use the analysis results to make data-driven decisions on car models.

#### Data Engineers: Ensure the pipeline and data processing are working correctly.

#### Test Engineers: Validate the accuracy and reliability of the system using unit and functional tests.

#### Preconditions:

#### The car sales dataset is available in CSV format (e.g., car.csv).

#### The dataset is loaded and cleaned using the load\_data() and clean\_data() functions.

**Solve these Questions**

a. **What is the total sales amount from the dataset?**

b. **Which car model is the top-selling model in the dataset?**

c. **What is the average price for Toyota cars in the dataset?**

d. **How many cars were sold within the price range of $20,000 to $50,000?**

e. **What is the total quantity of cars sold in the dataset?**

f. **What is the average price of cars in the dataset?**

g. **What was the total sales amount for the year 2023?**

h. **Which car model is the least-selling model in the dataset?**

**Execution Steps to Follow:**

1. All actions like build, compile, running application,running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) -> Terminal -> New Terminal
3. This editor Auto Saves the code
4. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the

internal git/repository. Else the code will not be available in the next login.

1. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
2. To setup environment:

You can run the application without importing any packages

1. To launch application:

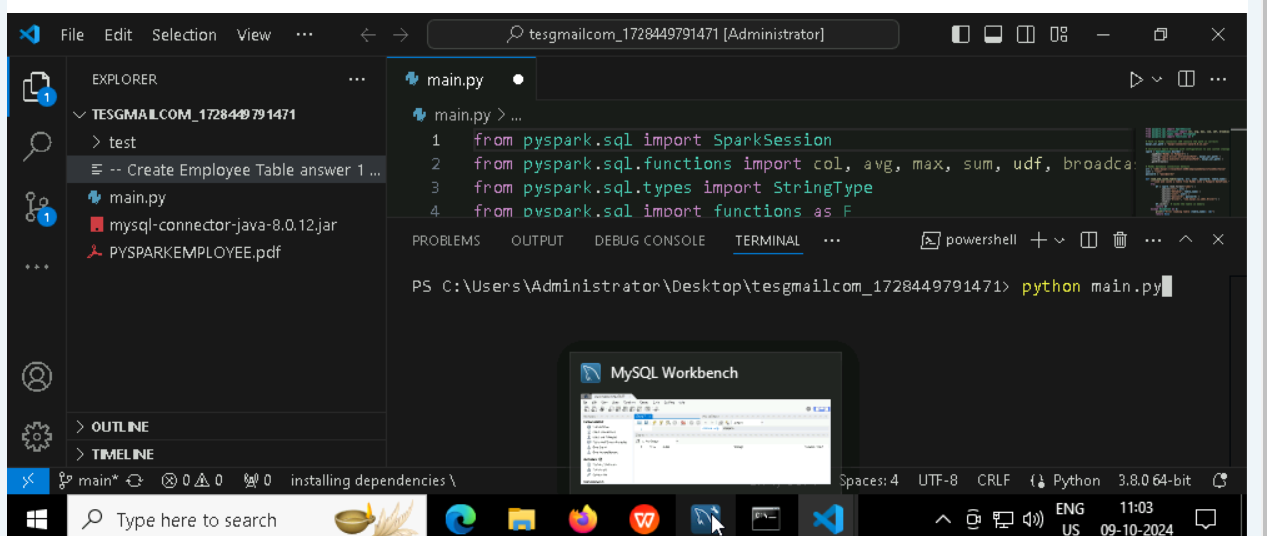
Python main.py

1. To run Test cases:

python -m unittest

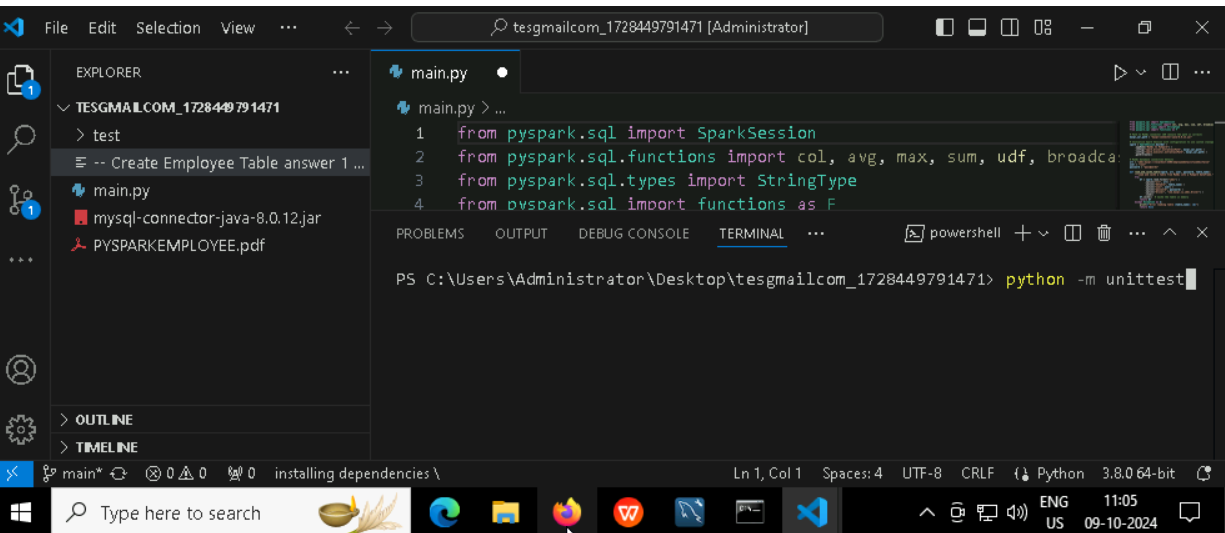
Before Final Submission also, you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository for code

**Screen shot to run the program**



**To run the application**

* **Python main.py**



**To run the testcase**

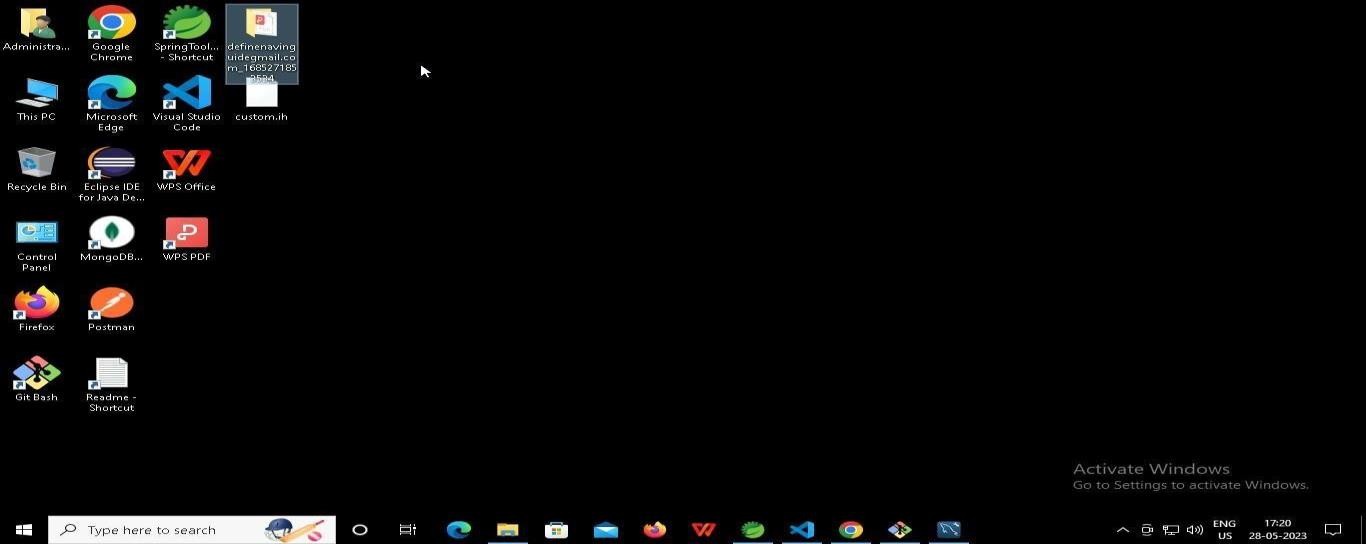
* **Python -m unittest**

**Screenshot to push the application to github**

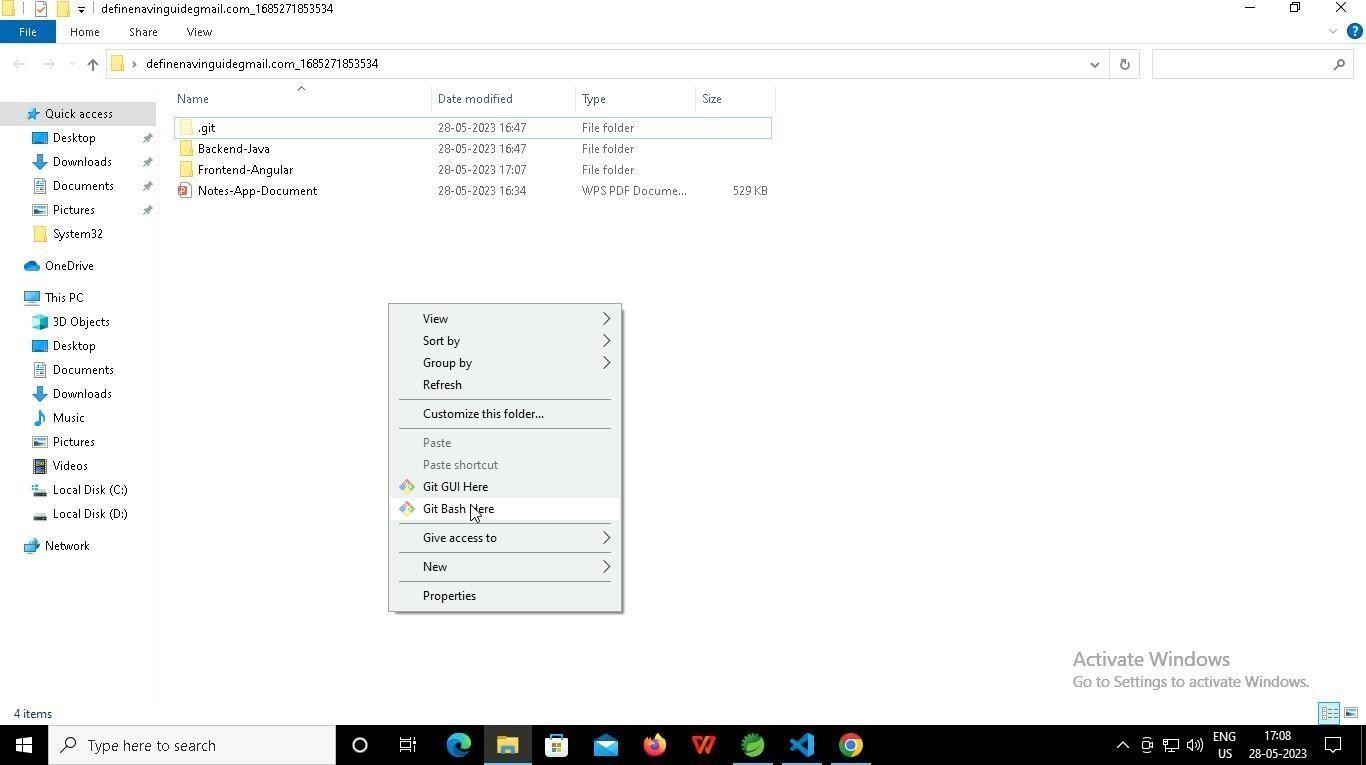
### -----x-----

#### You can run test cases as many numbers of times and at any stage of Development, to check how many test cases are passed/failed and accordingly refactor your code.

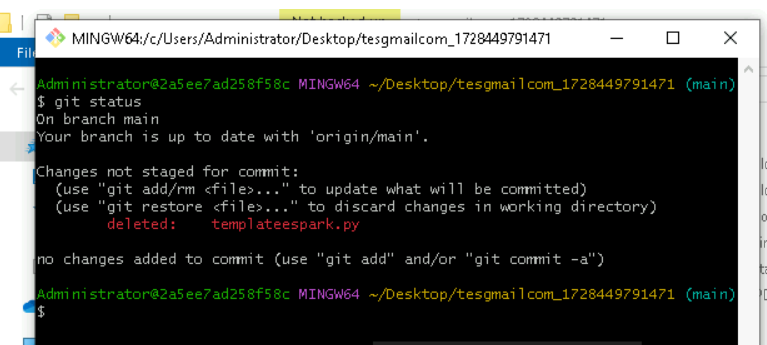
* 1. **Make sure before final submission you commit all changes to git**. For that open the project folder available on desktop



* + 1. **Right click in folder and open Git Bash**



* + 1. **In Git bash terminal, run following commands**
    2. **git status**



* + 1. **git add .**

A black screen with yellow and purple text

Description automatically generated

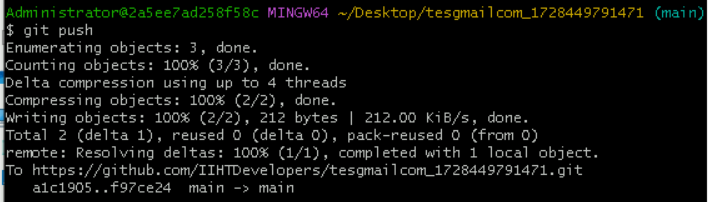
* + 1. git commit -m “First commit”

(You can provide any message every time you commit)

A screenshot of a computer

Description automatically generated

* + 1. **git push**



### After you have pushed your code Finally click on the final submission button

### 

### You should see a screen like this you will have to wait for the results . after getting this page you can leave the system

### A blue screen with white text Description automatically generated

### -----x-----