**A RESEARCH ON DATABASE TESTING**

**DATABASE:**

1. A database is an organized collection of data stored electronically and managed by a Data Base Management System (DBMS).
2. A DBMS manages and controls access to a DB. Examples include MongoDB, MariaDB, SQL.

**DATABASE TESTING:**

1. Database testing is the process of verifying the accuracy, reliability and performance of a Data Base system.
2. It can be performed manually, automatically and in a hybrid way (combined procedure of automation and manual).
3. It is also called as backend testing as it mainly focuses on the server side which isn’t visible to the user.

**IMPORTANCE OF DATABASE TESTING:**

Database testing saves data loss and aborted transaction data

Transaction here refers to the access and retrieval of data and transaction follows a few properties like:

1. ATOMICITY: If any transaction is performed on data, it should happen completely or shouldn’t be implemented.
2. CONSISTENCY: Database must be valid and preserved after the transaction is complete.
3. DURABILITY: Modifications will be kept without any fail when the transaction is committed.
4. ISOLATION

Database testing helps identify and mitigate security vulnerabilities, such as SQL injection attacks, which could compromise sensitive data.

Early detection and resolution of database issues during testing can significantly reduce the cost of fixing problems after deployment.

**TYPES OF DATABASE TESTING:**

There are three types of database testing. They include:

1. STRUCTURAL:

This type of database testing mainly focuses on the internal structure of the Data Base. Problems tested here include:

1. Data Type issues
2. Constraint Testing (primary key, foreign key)
3. Referential integrity checks
4. FUNCTIONAL:

This type of database testing is mainly used for checking whether the Data Base works according to the specified requirements. Problems tested here include:

1. CRUD [ Create-Read-Update-Delete]
2. Data manipulation through application interface
3. NON-FUNCTIONAL:

This type of database testing helps in measuring performance, scalability, reliability and security of the Data Base. Types include:

1. PERFORMANCE TEST- response time for each query
2. LOAD TEST- handling large volume of data/ users
3. SECURITY TEST- finds all possible loopholes/ weaknesses.
4. STRESS TEST- checks whether DB works beyond normal workload
5. COMPATIBILITY TEST- aims to check the developed software application functions on many software and hardware platforms.

The Non-Functional database testing is mainly automated due to its complexity whereas the structural and functional testing can be done in both ways (automated and manual too).

**TOOLS THAT ARE USED FOR DATABASE TESTING:**

1. **MANUAL:**

**1)MySQL Work Bench 2) Oracle SQL Developer 3) pG Admin**

1. **Automated functions and structured DB Testing:**
2. NoSQLUnit- Used for NoSQL databases.
3. pgTAP- SQL Unit Testing Framework
4. SQLTest: Automated testing of queries, procedures and result completion.

Tools for Automated Non-Functional Testing:

1. Apache JMeter: For load and performance testing
2. LoadRunner- Load testing app
3. OWASP-ZAP: Used for security testing

SELF HEALING:

Self-healing is the ability of automated tests to automatically adapt changes in the database schema/ data without manual intervention.

It involves using AI and ML techniques to identify, correct issues arising from Data Base updates.

They are needed in database testing as:

1. Renaming of column names
2. Table structure change
3. Data format change

Self-Healing improves efficiency by accelerating the error detection by automatically detecting and correcting errors.

HOW TO AUTOMATE DATA-BASE TESTING:

STEP1: Define objectives

STEP2: Create good test cases based on our objectives. The test cases include:

1. Data Integrity Test:
2. Primary Key Constraints
3. Foreign key Constraints
4. Unique Constraints
5. Data Type Validations:
6. Range Checks
7. Format Checks

STEP3: Choose the best automation tool like Apache JMeter, Selenium etc.

STEP4: Setup the test environment.

STEP5: Integrate it with CI/CD.

STEP6: Execute the test.

STEP7: VALIDATE RESULTS AND MAINTAIN TEST SCRIPTS

STEP8: Monitor Performance

MY APPROACH TOWARDS SOLVING THIS PROBLEM:

MY APPROACH IS TO USE OLLAMA which is a large language model released by META AI on the month of FEB 2023.

It is an open-source model unlike GPT which is a proprietary model and is highly customizable, strong in code generation.

This will be run locally and then I will have a database having details of few players.

Then, I will be using Python and in Python I am using the unittest module which is used for automating the data validation logic and mysql.connector for connecting python with MySQL.

In this problem LLAMA will be used to heal the broken SQL queries and create a text file to store the error conditions.

PYTHON TESTS

LOG FIX

RETRY QUERY

LLAMA HEALING

SQL ERROR

FIG1: ANALYSIS PATTERN OF SELF HEALING IN OUR DATABASE PROBLEM.

WE ALSO HAVE A FEW DATA VALIDATION RULES TO CHECK FOR WRONG TYPES, DUPLICATES.

**STEPS INVOLVED IN MY PROCEDURE TO TEST SELF HEALING IN A DATABASE:**

Step1: Software installations:  
  
1. IDLE Python- Have the latest version installed (3.13.5). While installing just make sure that you add python.exe to the path which helps in proper installations of modules.

2.MySQL: MySQL is an open-source RDBMS which is fast, secure, reliable and easy to use when compared to the other database managements systems. Work with caution while installing it in your system as the installation is a long procedure.

3. After installing MySQL and Python install the modules which are necessary for completion of our project. Some of the modules which are used here for successful completion of project include:

1. mysql.connector : mysql.connector is a Python library that enables python programs to interact with MySQL databases. In our project one of the important modules is mysql.connector as our project is about self-healing in databases.
2. Unittest: The unittest module is Python’s built in unit testing framework. Along with this, it provides a structured, object-oriented approach to writing and running tests for Python codes.
3. Datetime: We use the datetime module for mentioning the date at when the database testing takes place and the time at which the testing occurs. So, it provides us in having a log on having a note at the errors which occur in the database and tests the validity of the database.
4. Json: Json stands for Java Script Object Notation. It is mainly used for transmitting data between a server and web application and also for storing structured data in the form of key- value pair.
5. re: The re module provides support for regular expressions. Regular expressions are powerful tools that enable tasks like searching, splitting, validating text based patterns.
6. The tool which we use for achieving self-healing of a database along with Python and MySQL is Ollama. Ollama is a tool that is used for running and managing large language models locally on our computer. It simplifies the process of downloading, setting local LLM’s and makes them accessible for research, code generation and automation of testing of DB. SO, this tool plays a very major role in our project.
7. Before we proceed into the testing part, let us firstly create a database which consist of details about the CSK Players who are from Australia. Create another database named original which is not used for testing and used only for viewing and knowing details about it.
8. Along with this we, have used a software called Memcached, which is a high performance, distributed memory object caching system. It acts as a short-term memory for applications. Here, the values are stored in the form of key-value pairs which makes it easy for storing and retrieving data. We use this for checking whether the datatype of a particular column has been changed when compared to the original database and if yes, we use self-healing for returning back to the original database.

Then, here in this project to see the healed query and results of the tests conducted we have a notepad file named healing\_log which logs them. For a better viewing we have created a UI for viewing them. For that, we have used the flask module. Flask is a microframework which is used for web development as it offers a minimal core with optional extentions allowing developers to choose and add functionality.

The flask module is built on top of two powerful libraries:

* + - 1. Jinja2
      2. Werkzeug

**OUTPUT SCREENSHOTS**

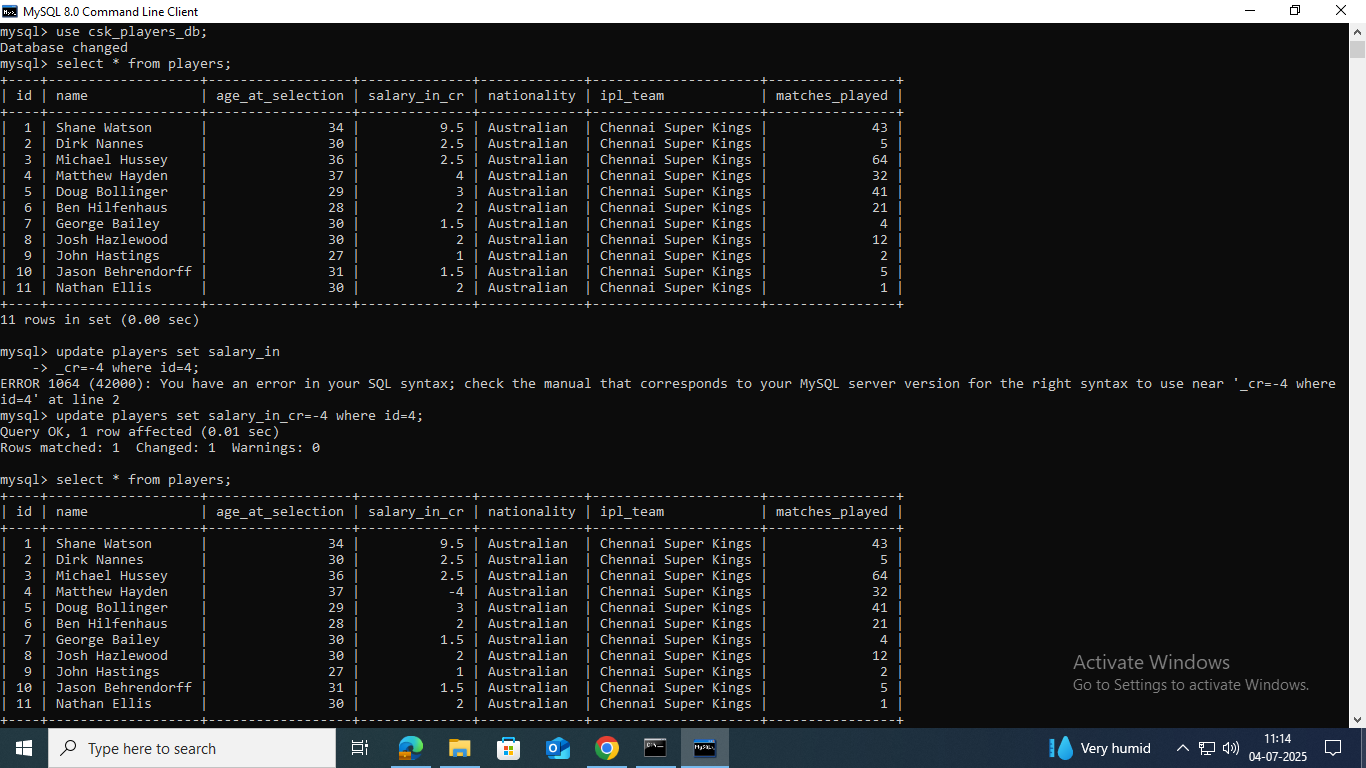
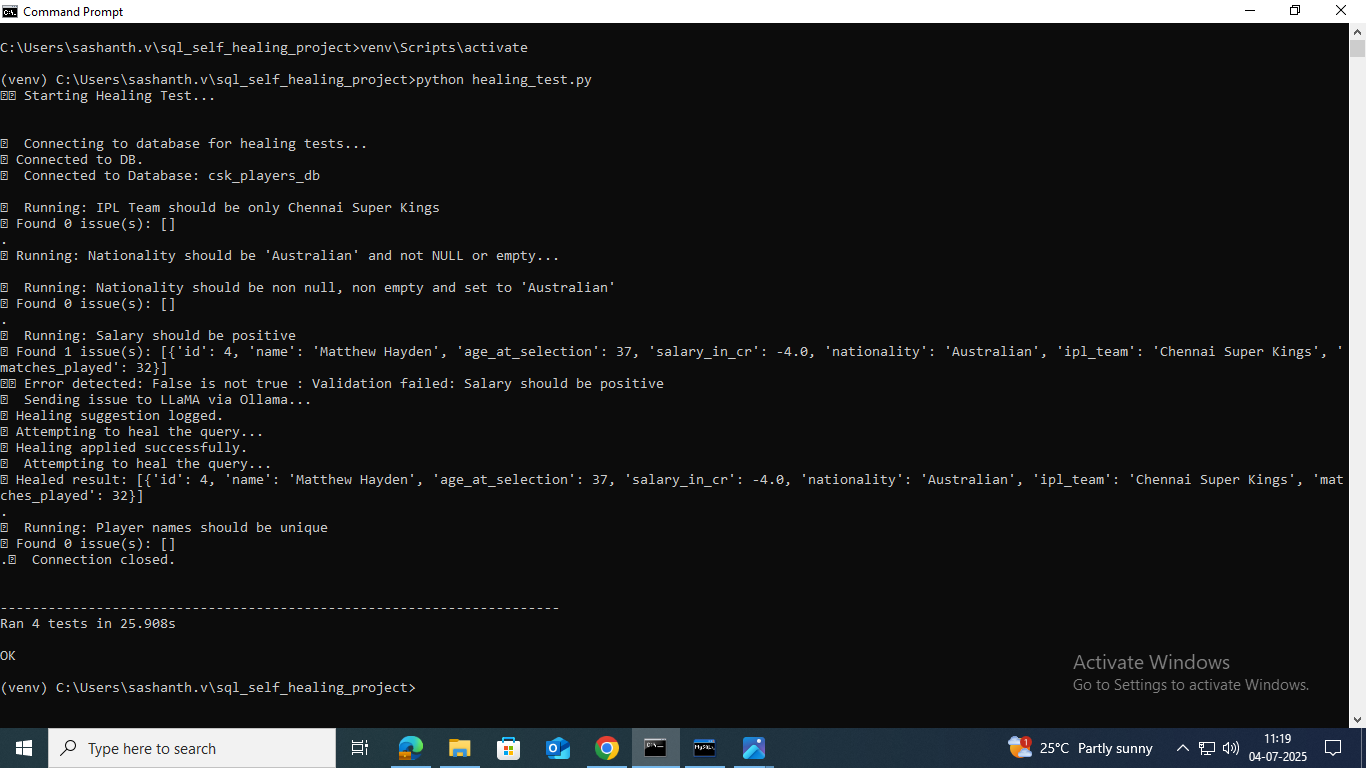
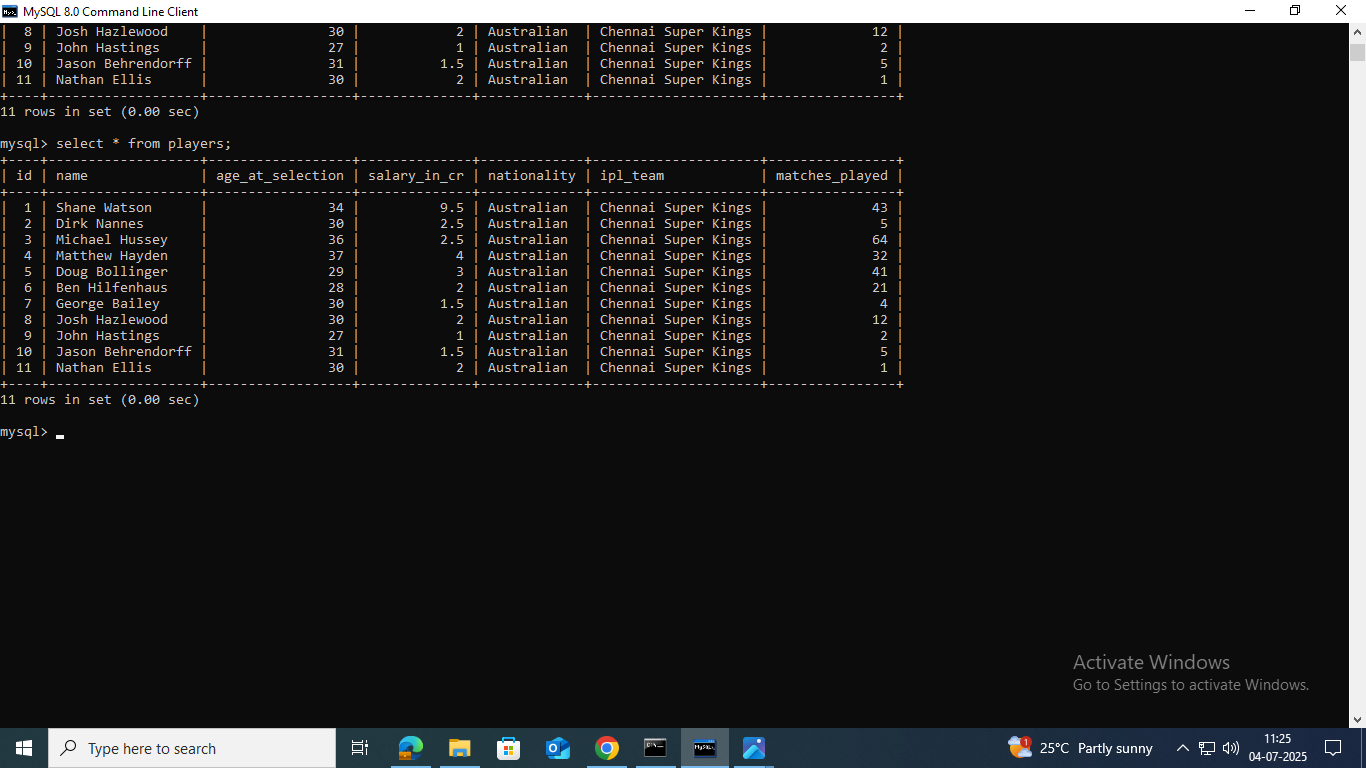
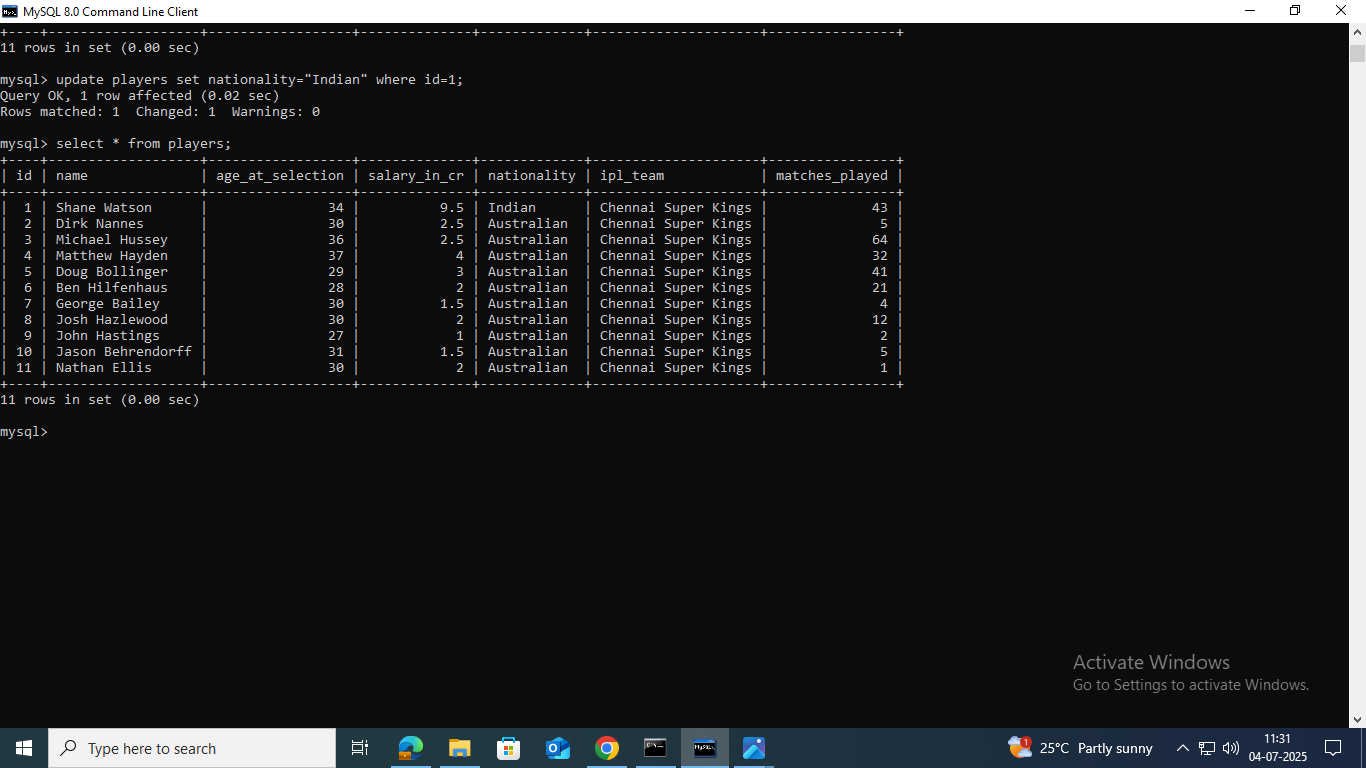
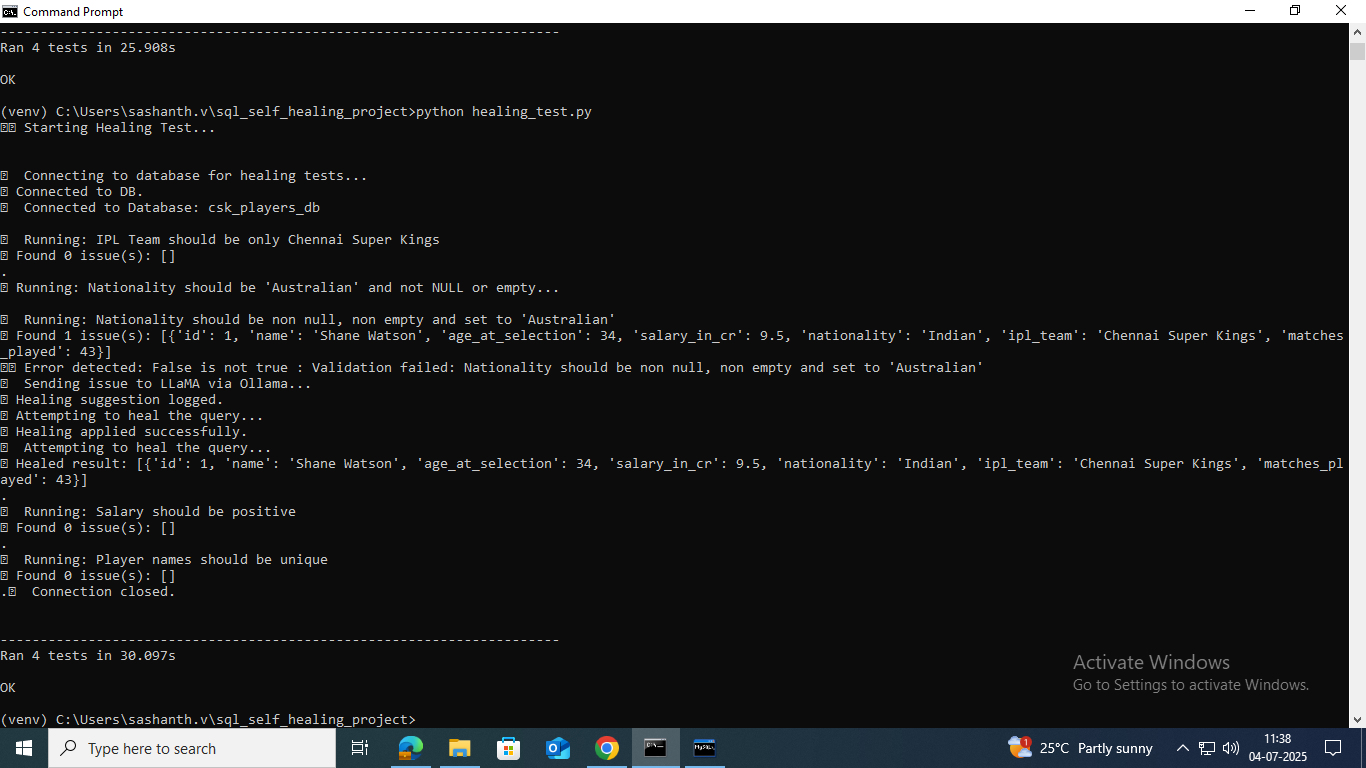
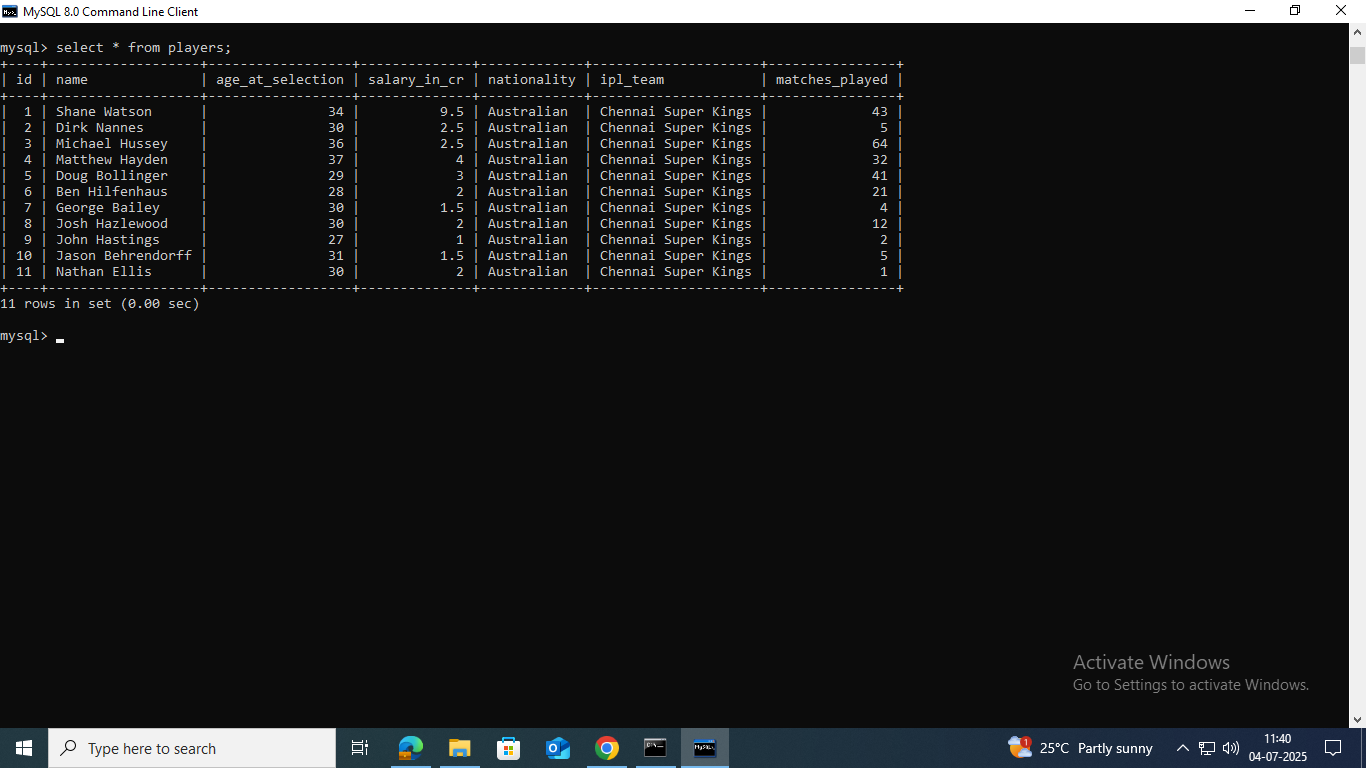


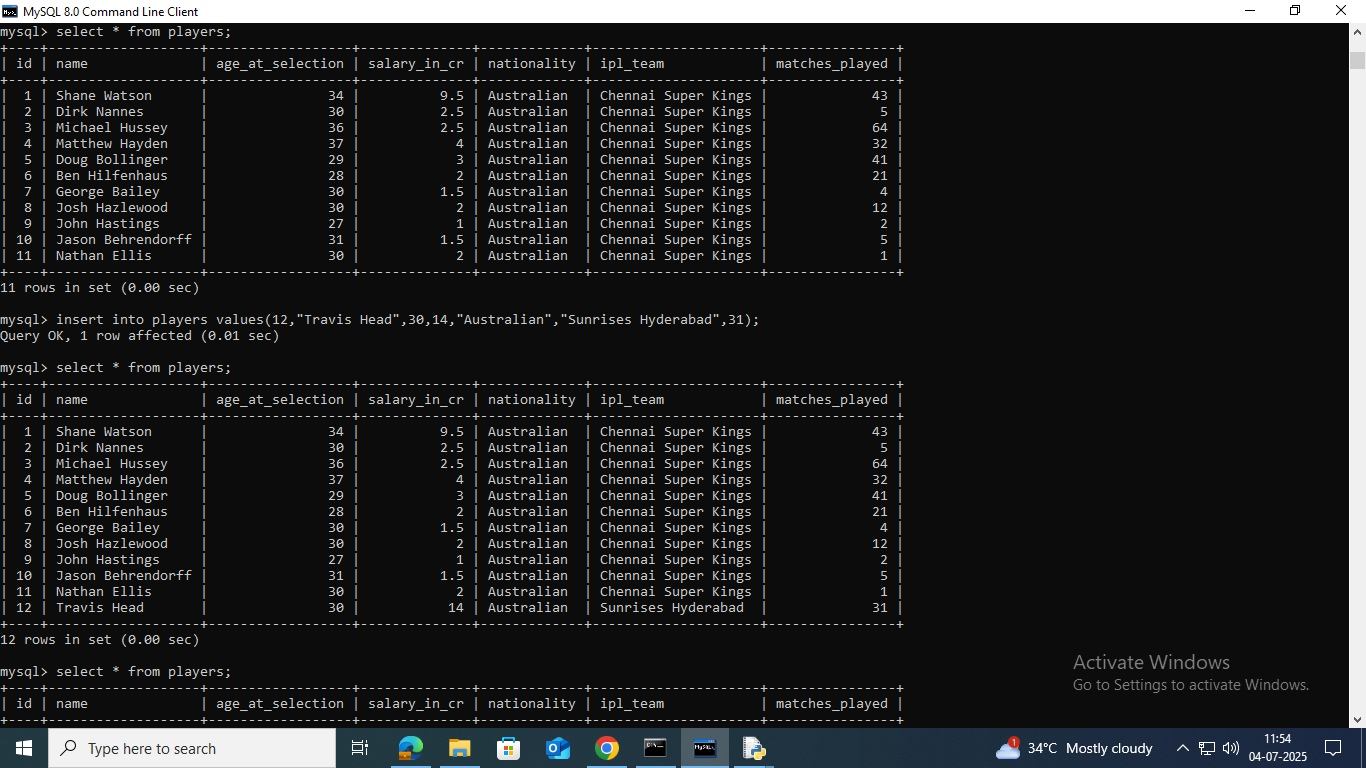
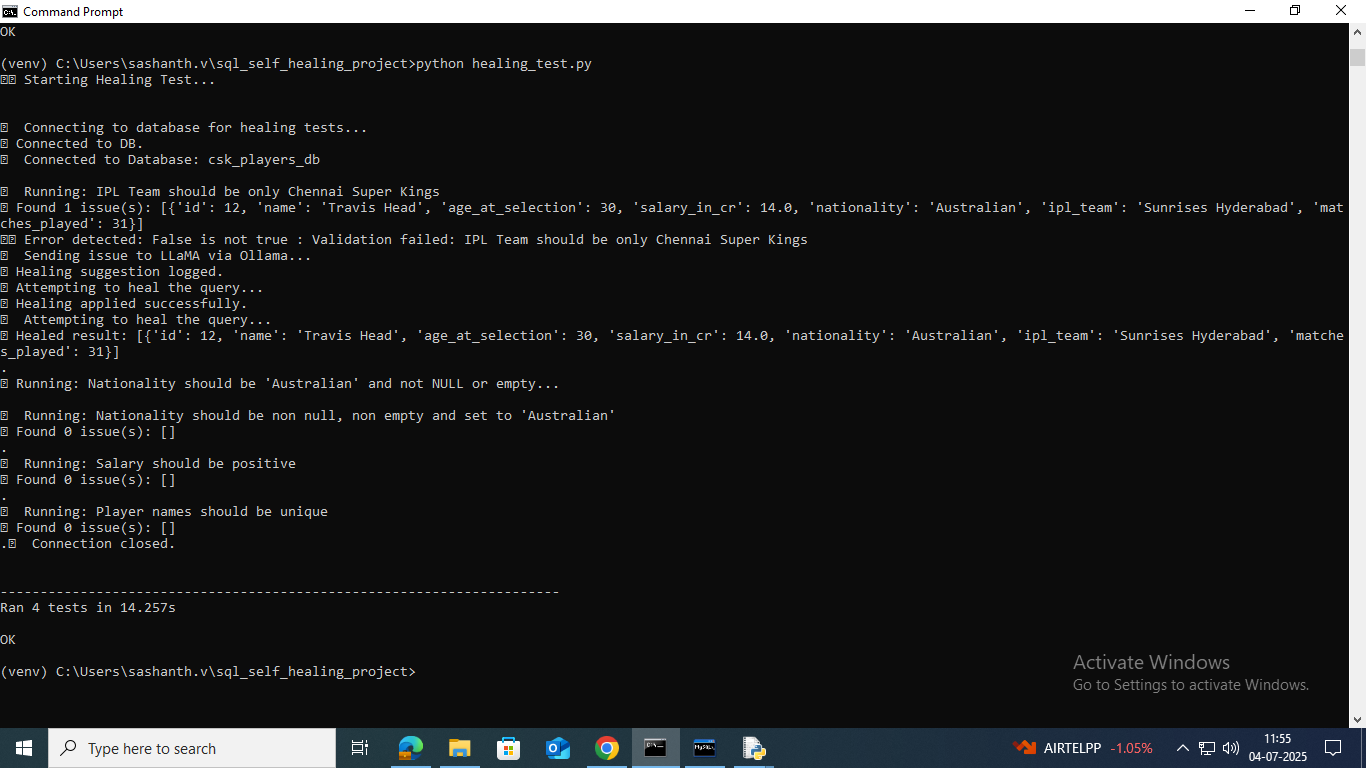
FIG1- Screenshot of the clean database created and the second one is the database with a negative salary mentioned to the person having id=4.

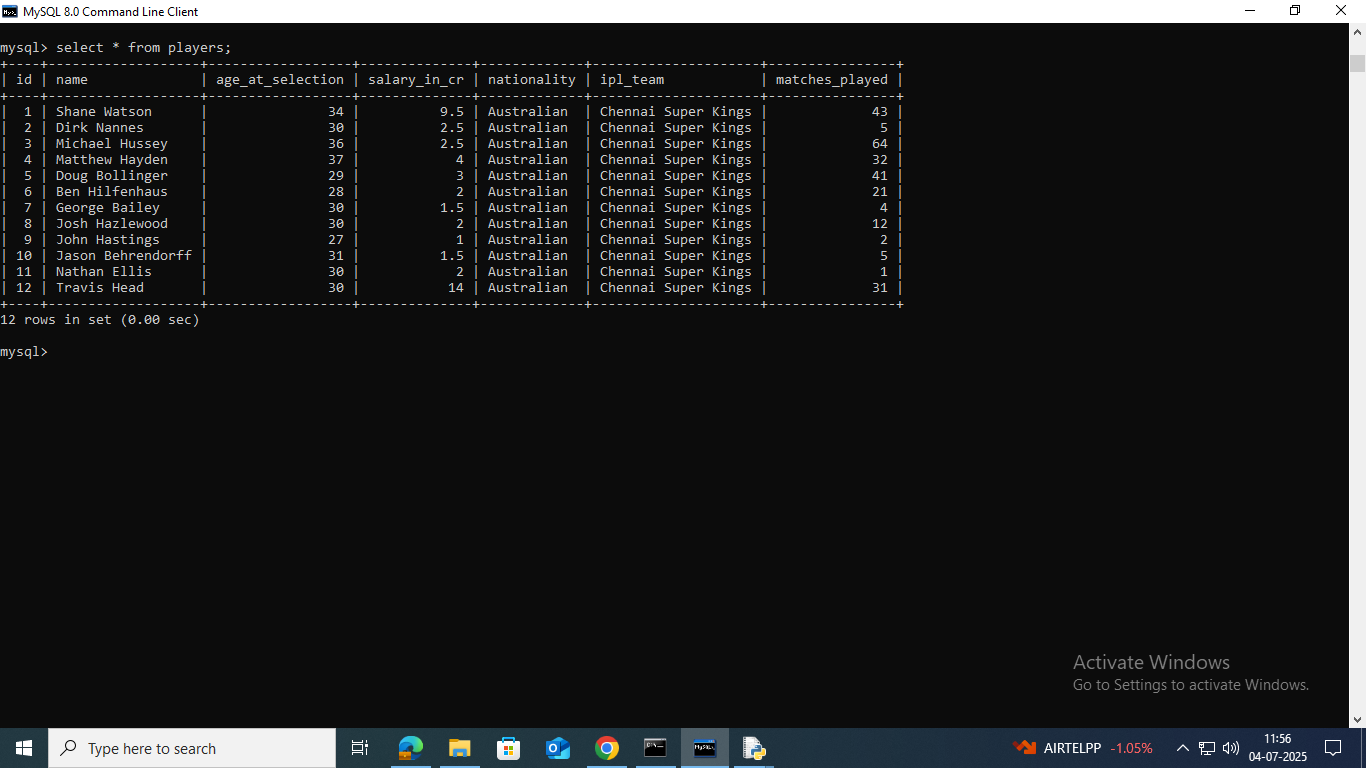
FIG-2: The activation of running the tests in cmd and the healing query which finally removes the negative salary and changes it into the absolute value.

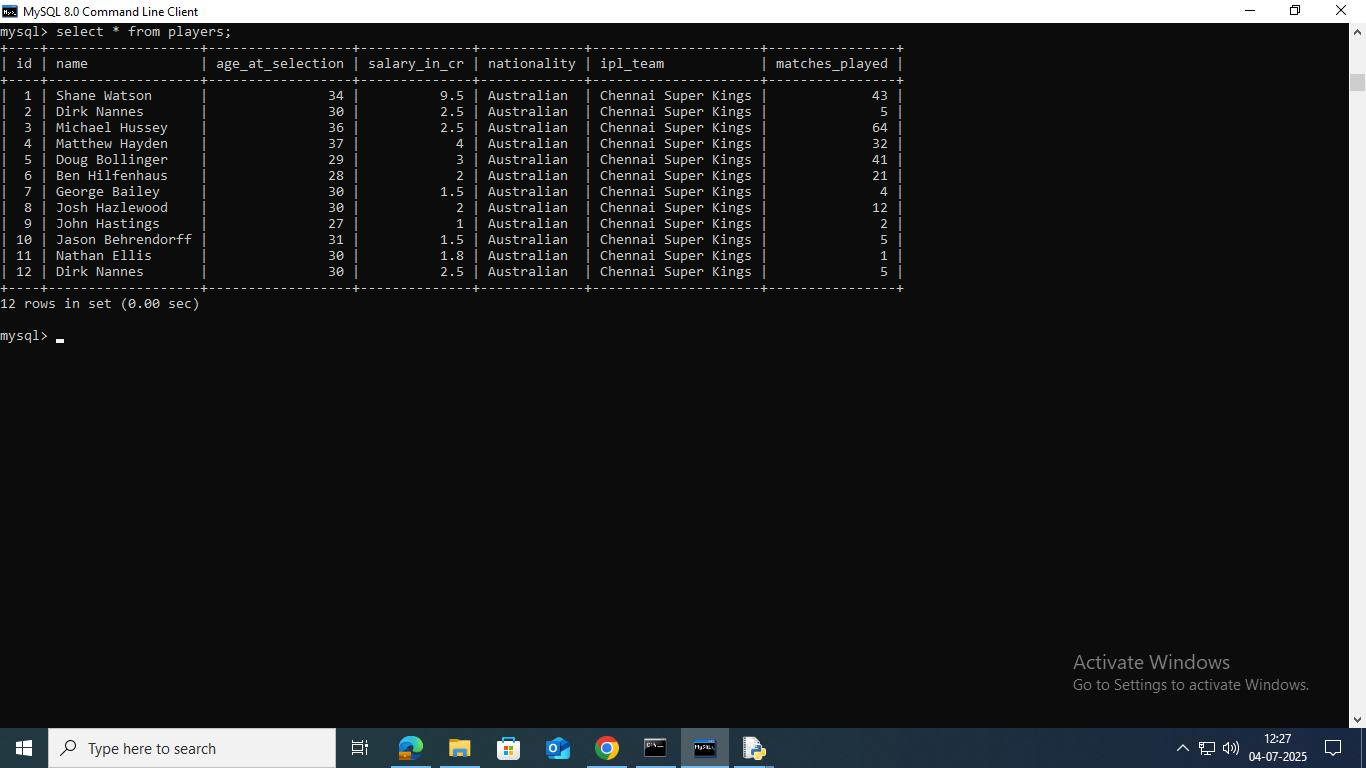
Fig3: The cleaned database after modifying the salary to be positive which validates the test case 1.

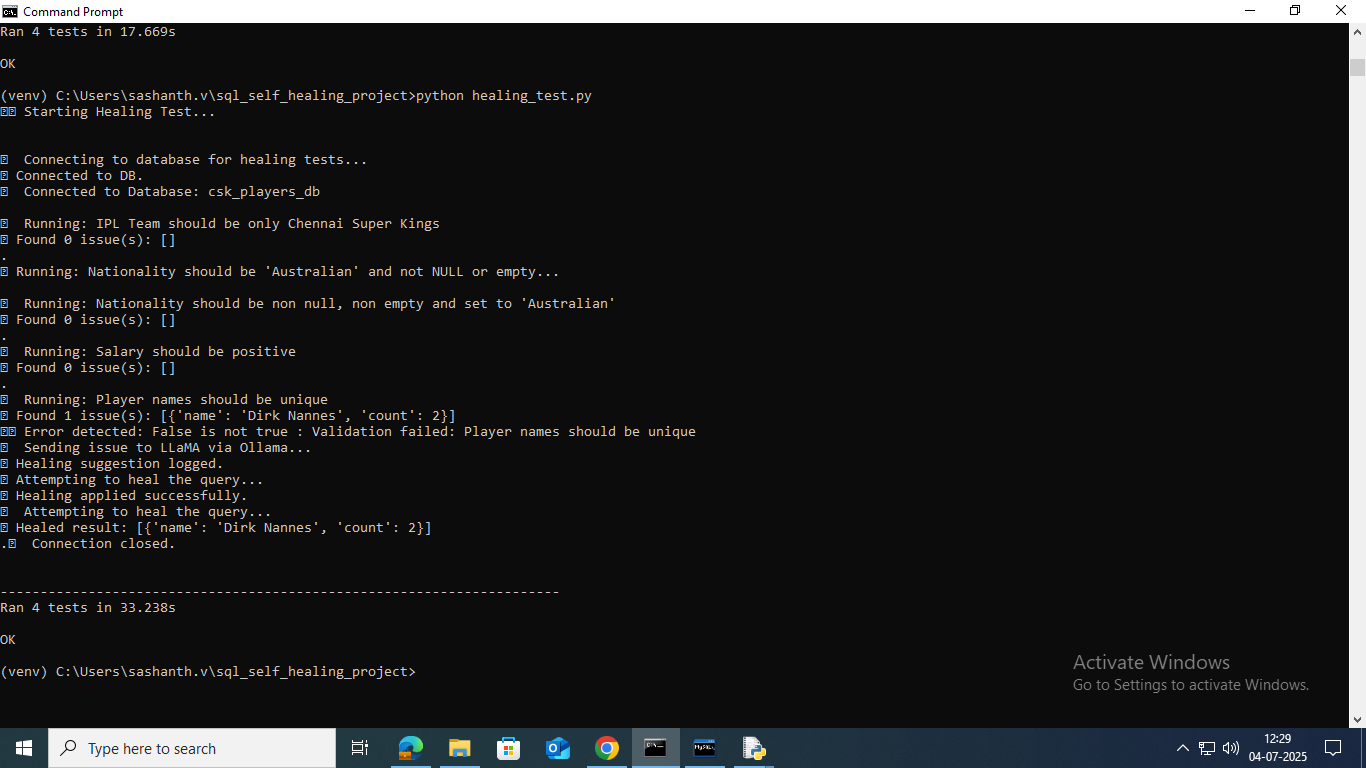
Fig4: The database having the nationality of a player with id=1 as Indian which indicates that this one is an invalid data as the nationality must be Australian.

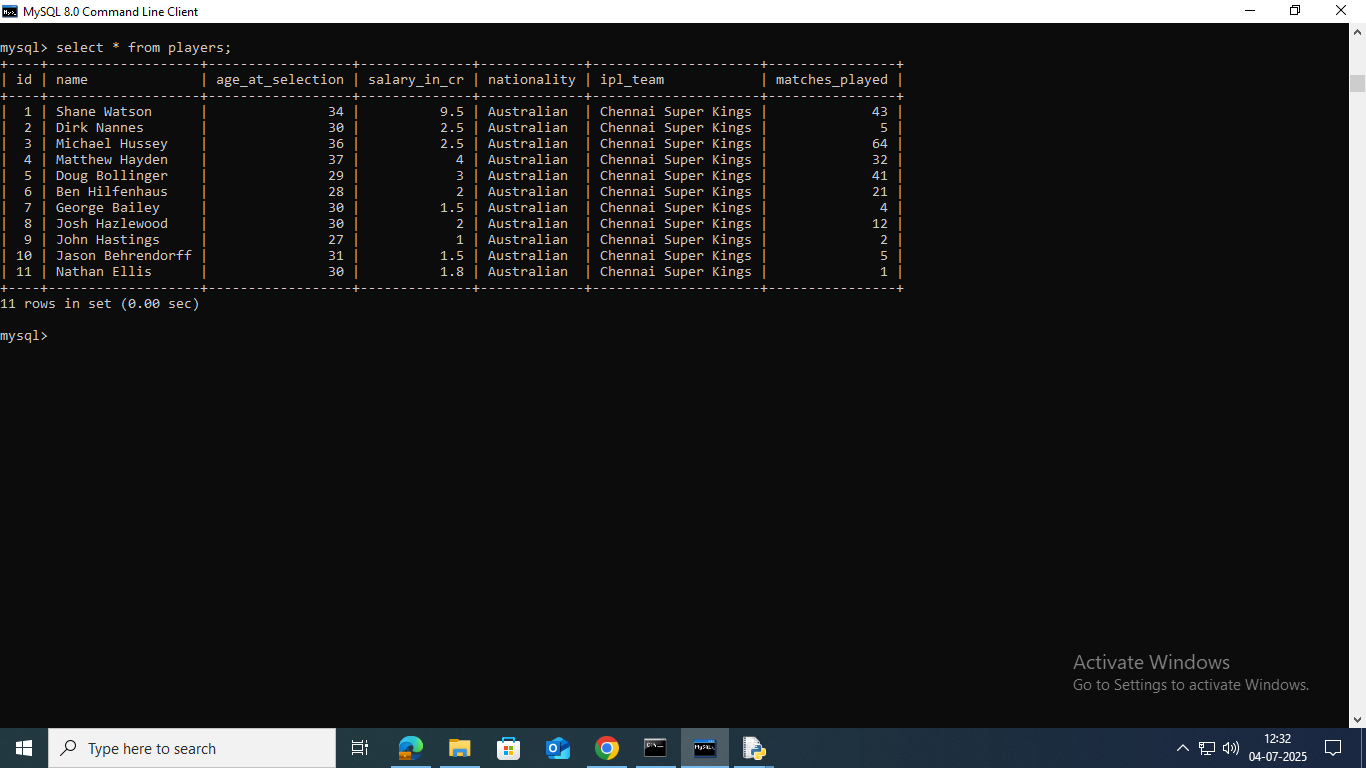
Fig5: The cmd testing for testing the database. The healer identifies an error stating that the nationality is Indian and then the query is being healed using LLaMA.Fig6: The cleaned and validated database with valid entries which satisfies the second test case.

Fig7: The database has a valid entry in the 12th row which is a player from Sunrises Hyderabad. Fig8: The self-healing automation testing is run in the cmd. SO, it validates all the tests and finally checks the invalid entries.

Fig9: The clean database with valid entries.

Fig10: The database having a duplicate entry in the 12th row.

Fig11: The running of tests related to our database. Here, we identify the invalid entry, and it sends to LLaMA via Ollama to rectify it.

Fig12: The cleaned and valid database with valid entries.

**TEST CASES THAT WE WORK ON:**

There are four test cases which we test in our self-healing project:

1. Salary being positive
2. Nationality to be ‘Australian’ unlike others as our sole purpose of the database is to have data of players from Australia.
3. Duplicates to be removed.
4. Ipl Team must be Chennai Super Kings unlike the other teams such as SRH, RCB etc.
5. If the column name changes it must get healed automatically and work accordingly

These 4 test cases are checked and before running the validation tests, we must ensure that all the source codes required for the testing is kept in the same directory and at the same time make sure that all the modules are being installed for its successful completion of testing.

1. If the datatype of a particular column changes ex: int to float manually, the Ollama present must identify the change in the datatype due to caching of the columns at the datatype which is due to memcaching process done and get converted into the original datatype as per the original one.
2. Suppose the name of the table changes, the Ollama present for the self-healing must identify it and then proceed with the tests.

The modules/ software’s required for testing of the database here will be:

1. MySQL connector
2. Python-requests.
3. Memcached
4. Flask
5. Notepad++ (For HTML)

Result:

So, we can finally conclude that self-healing is observed in the database successfully.