Name : **Bhavana Soge**

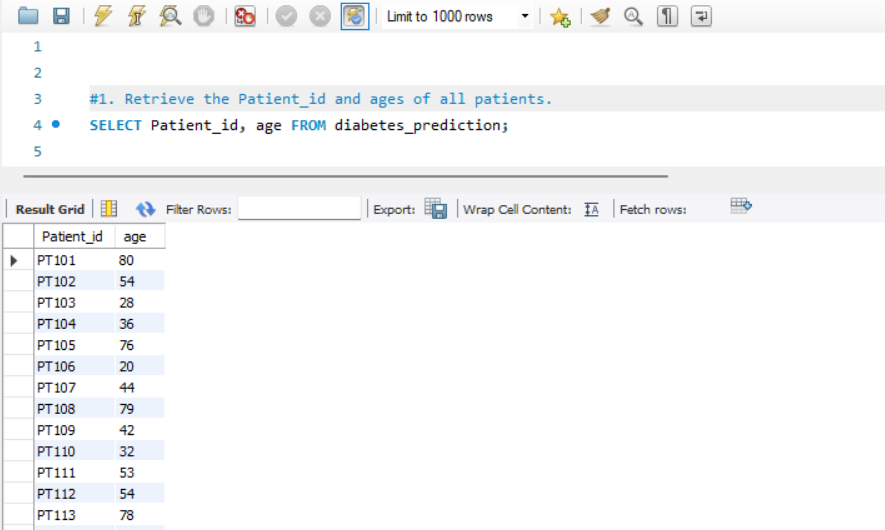
Email : soge.bhavana@gmail.com

LinkedIn : https://www.linkedin.com/in/bhavana-soge-15a76a226/

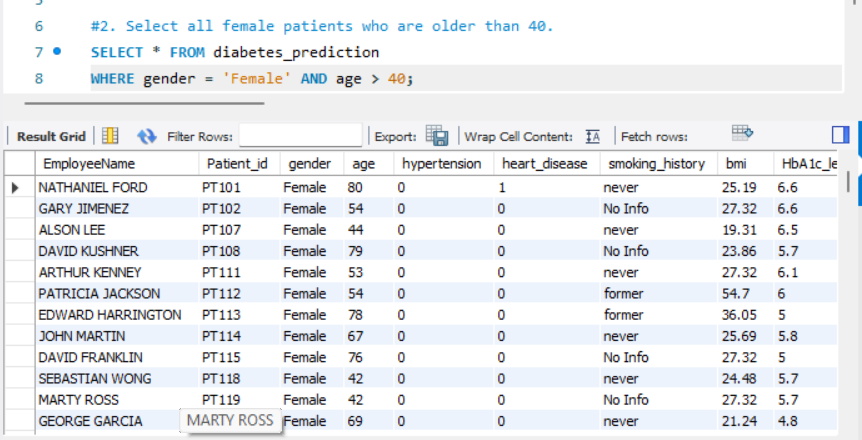
PSYLIQ

**DIABETES PREDICTION ASSESSMENT**

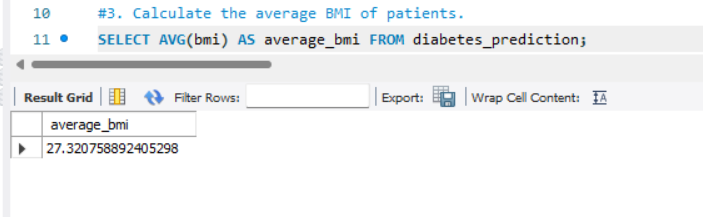
1. **Retrieve the Patient\_id and ages of all patients.**

****

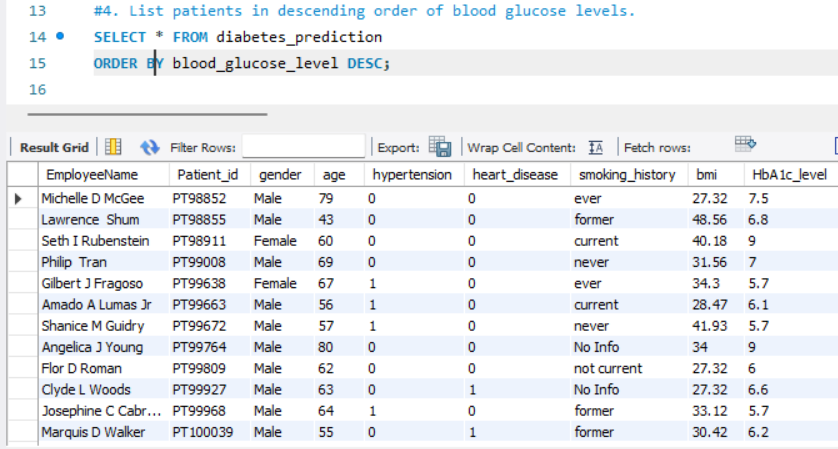
1. **Select all female patients who are older than 40.**

****

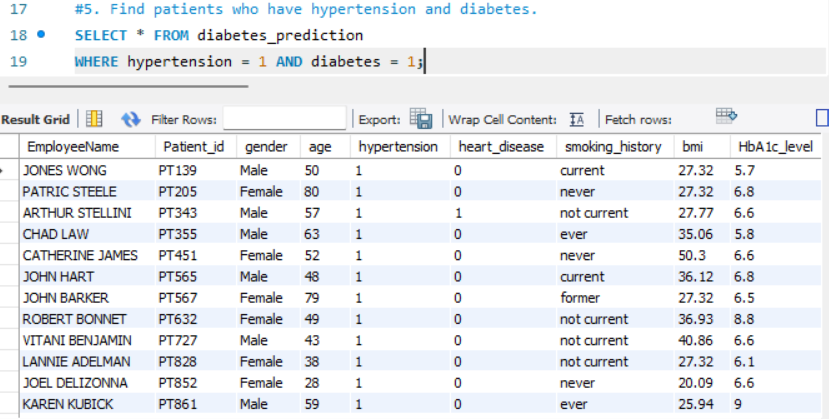
1. **Calculate the average BMI of patients.**

****

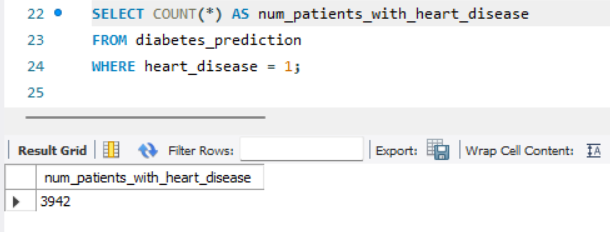
1. **List patients in descending order of blood glucose levels.**

****

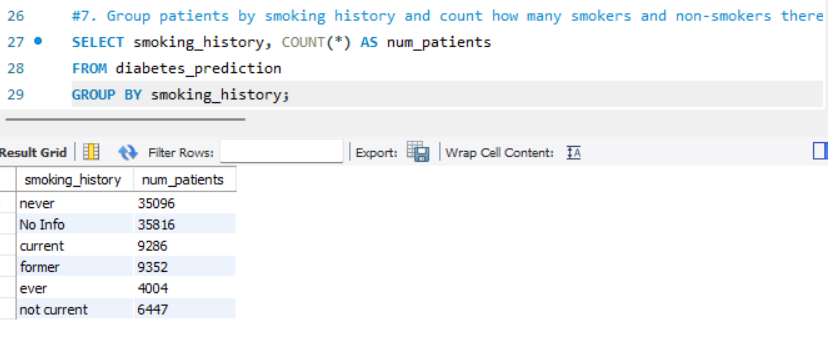
1. **Find patients who have hypertension and diabetes.**

****

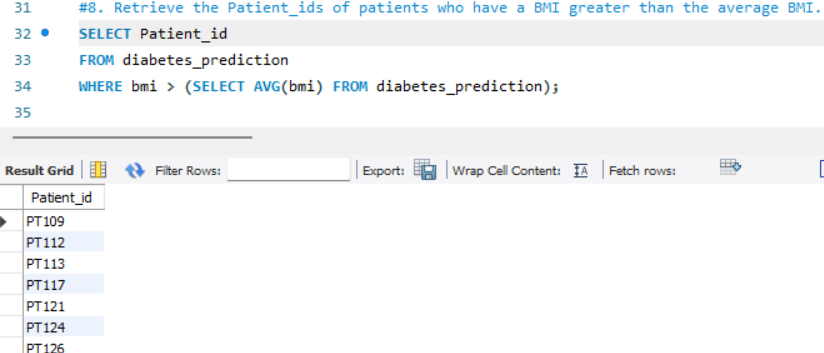
1. **Determine the number of patients with heart disease.**

****

1. **Group patients by smoking history and count how many smokers and nonsmokers there are.**

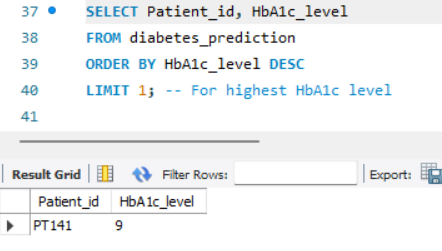
****

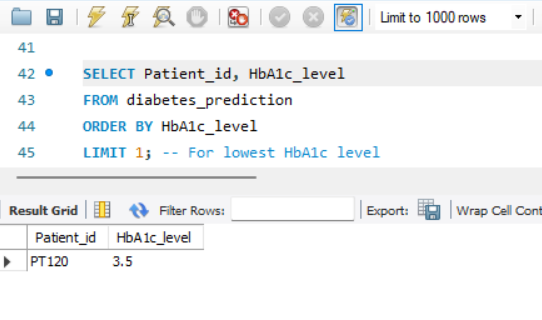
1. **Retrieve the Patient\_ids of patients who have a BMI greater than the average BMI.**

****

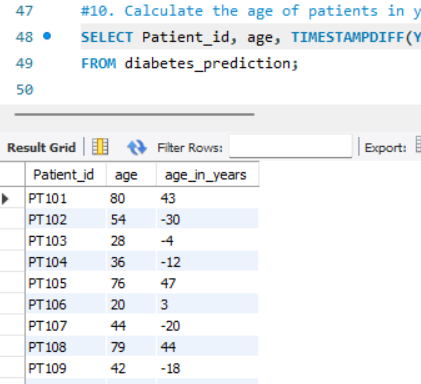
**9. Find the patient with the highest HbA1c level and the patient with the lowest**

**HbA1clevel.**

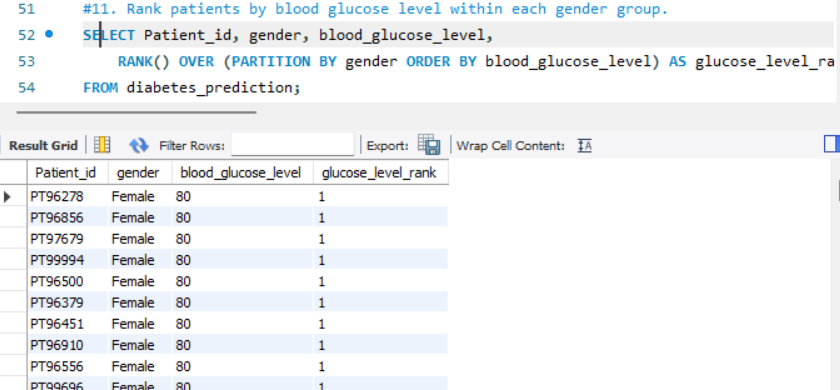
****

****

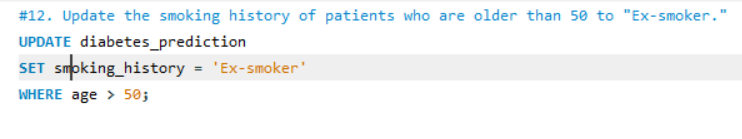
1. **Calculate the age of patients in years (assuming the current date as of now).**

****

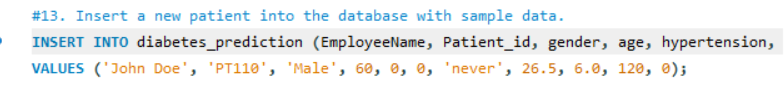
1. **Rank patients by blood glucose level within each gender group.**

****

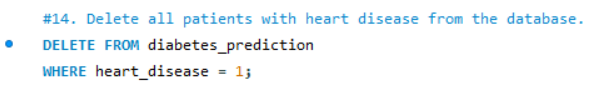
1. **Update the smoking history of patients who are older than 50 to "Ex-smoker."**

****

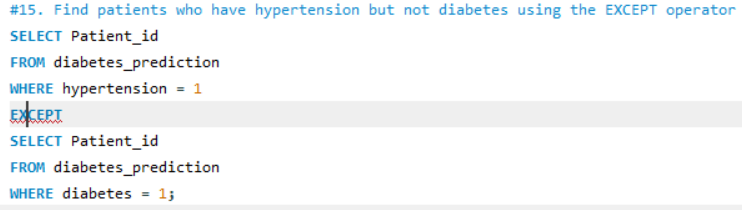
1. **Insert a new patient into the database with sample data.**

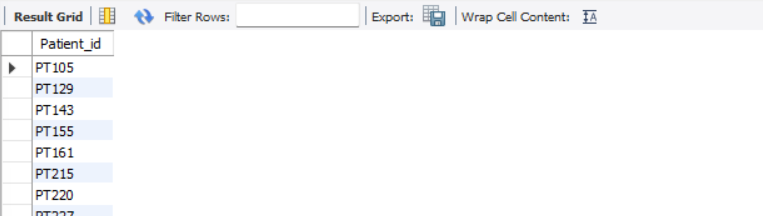
****

1. **Delete all patients with heart disease from the database.**

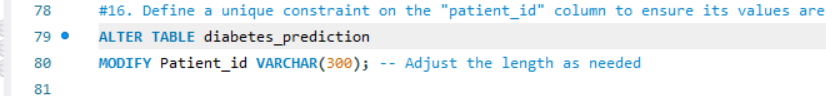
****

1. **Find patients who have hypertension but not diabetes using the EXCEPT operator.**

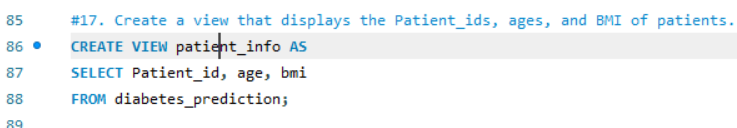
****

****

1. **Define a unique constraint on the "patient\_id" column to ensure its values are unique.**

****

**17. Create a view that displays the Patient\_ids, ages, and BMI of patients.**

****

**18. Suggest improvements in the database schema to reduce data redundancy and**

**improve data integrity.**

* Normalization: Ensure the database is normalized to reduce redundancy. For example, you might have a separate table for patient demographics (e.g., name, gender), and the main table would reference the demographics using a foreign key.
* Foreign Keys: Use foreign keys to establish relationships between tables. For instance, you could have a separate table for smoking history with a foreign key in the main table.
* Data Types: Use appropriate data types for each column to minimize storage and enhance data integrity.
* Indexes: Add indexes on columns frequently used in search conditions to improve query performance.

**19. Explain how you can optimize the performance of SQL queries on this dataset**

**Indexes: Properly index columns used in WHERE clauses and JOIN conditions.**

* Avoid SELECT : Only select the columns you need instead of using SELECT .
* Use Joins Efficiently: Optimize JOIN operations, and use INNER JOIN, LEFT JOIN, etc., as needed.
* Partitioning: If the dataset is large, consider partitioning the table based on certain criteria.
* Update Statistics: Keep statistics up-to-date to help the query optimizer make better decisions.
* Limit Results: Use LIMIT or TOP to restrict the number of rows returned, especially when displaying data in applications.
* Review Execution Plans: Analyze the query execution plans to identify areas for improvement.
* Caching: Utilize caching mechanisms where appropriate to avoid redundant queries.