

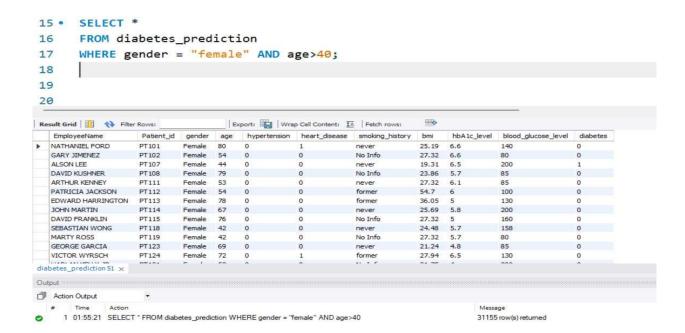
Diabetes Prediction Analysis

BY RAHUL ACHARYA

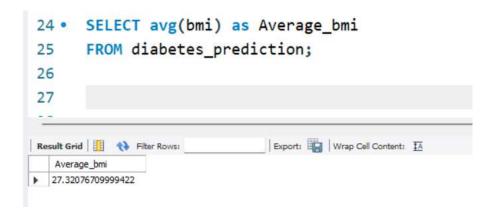
Retrieve the Patient_id and ages of all patients



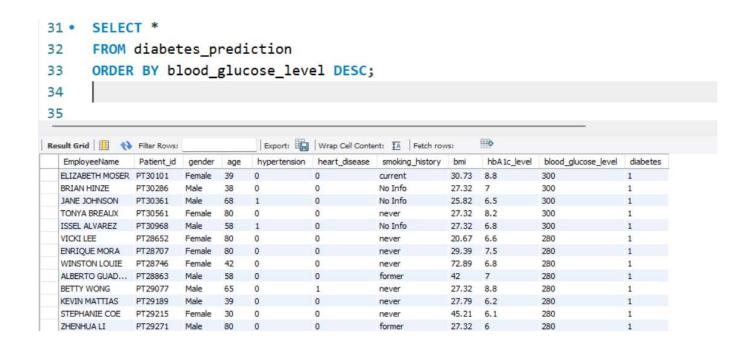
Select all female patients who are older than 40



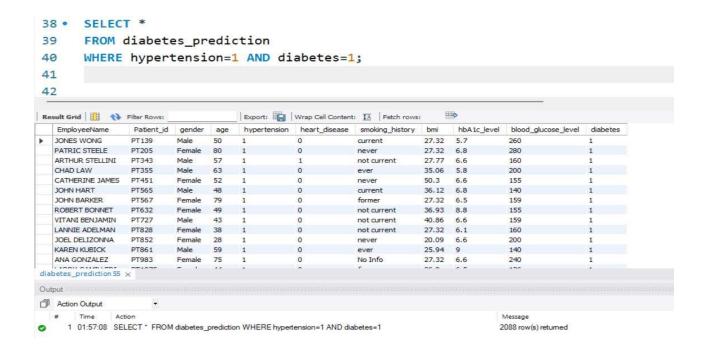
Calculate the average BMI of patients



List patients in descending order of blood glucose levels



Find patients who have hypertension and diabetes



Determine the number of patients with heart disease

```
46 • SELECT COUNT(*) as Heart_disease_patients
47 FROM diabetes_prediction
48 WHERE heart_disease=1;
49
50
51

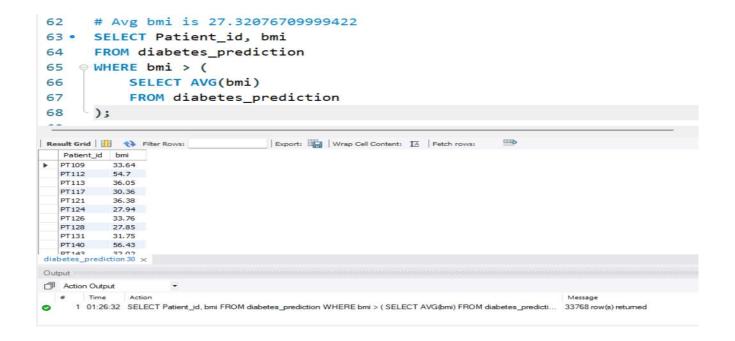
Result Grid  Filter Rows: Export: Wrap Cell Content: A

Heart_disease_patients
3942
```

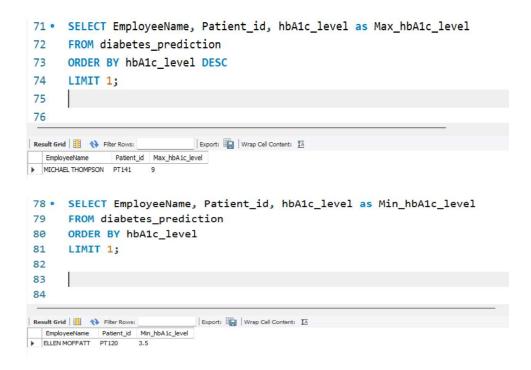
Group patients by smoking history and count how many smokers and non-smokers there are

```
53 • SELECT smoking_history, COUNT(*) as Count
    FROM diabetes_prediction
54
      WHERE smoking_history = "current" OR smoking_history="never"
55
      GROUP BY smoking history;
56
57
58
                             Export: Wrap Cell Content: IA
Result Grid Filter Rows:
  smoking_history Count
never
            35095
  current
            9286
```

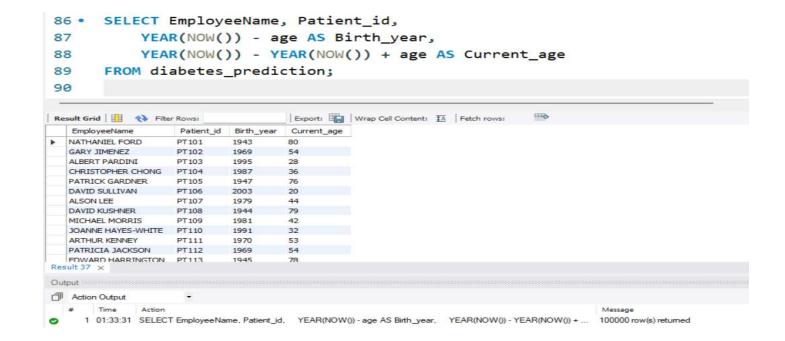
Retrieve the Patient_ids of patients who have a BMI greater than the average BMI



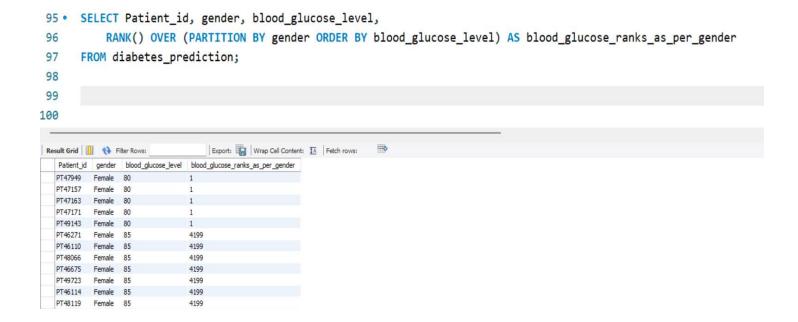
Find the patient with the highest HbA1c level and the patient with the lowest HbA1clevel



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



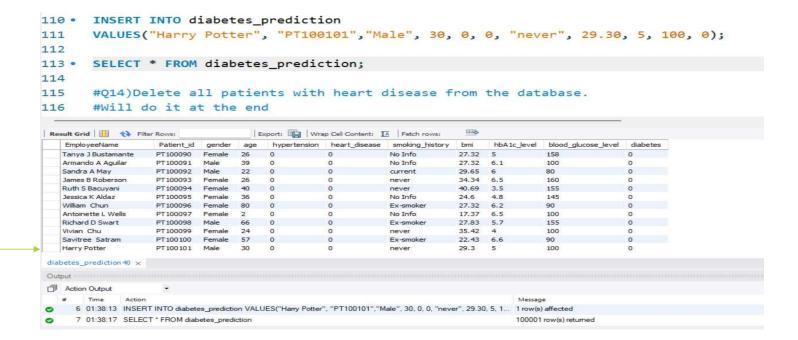
Rank patients by blood glucose level within each gender group



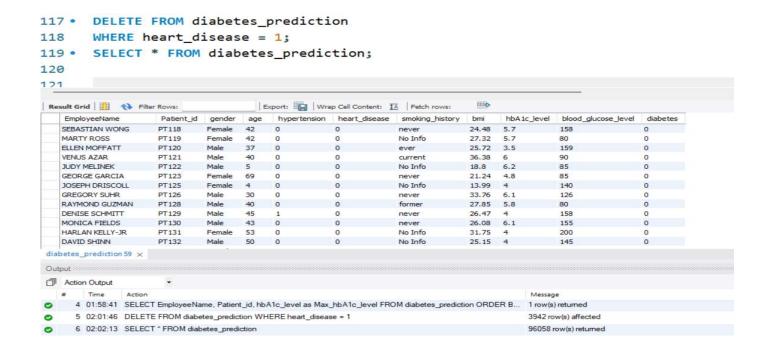
Update the smoking history of patients who are older than 50 to "Ex-smoker"



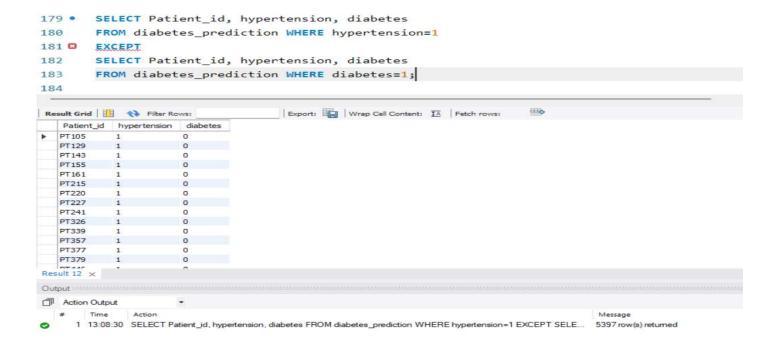
Insert a new patient into the database with sample data



Delete all patients with heart disease from the database



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



Define a unique constraint on the "patient_id" column to ensure its values are unique

```
ALTER TABLE diabetes_prediction

ADD CONSTRAINT Patient_id UNIQUE (Patient_id);

133

134 • INSERT INTO diabetes_prediction

135 VALUES ("Harry Potter", "PT100101", "Male", 30, 0, 0, "never", 29.3, 5, 100, 0);

136

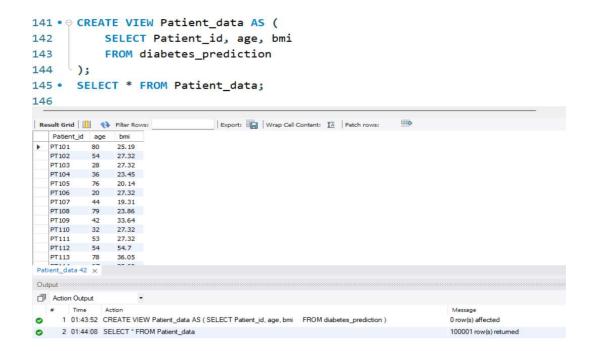
137

Output

# Time | Action Output |
# Time | Action |
Output |
# Time | A
```

Since we added a UNIQUE constraint to 'Patient_id', when we insert a duplicate 'Patient_id' it should throw an error.

Create a view that displays the Patient_ids, ages, and BMI of patients



Suggest improvements in the database schema to reduce data redundancy and improve data integrity

- ► To reduce data redundancy:
- > 1)We can identify columns that may contain redundant information i.e. a piece of data can be derived from other columns and remove it.
- > 2)We should avoid storing calculated or derived values in the database and instead calculate them dynamically.
- > 3)If certain information can be obtained by joining tables or querying existing data, consider avoiding redundant storage of that information.
- For ensuring data integrity:
- ▶ 1)Ensure that each table has a primary key to uniquely identify each record.
- 2)Apply unique constraints to columns that should have unique values within a table. This helps prevent duplicate entries in critical fields.
- > 3)Use check constraints to enforce specific rules on column values. For example, check constraints can ensure that numerical values fall within a specified range or that dates are within a valid range.
- ▶ 4)Choose appropriate data types for columns to ensure accurate representation of data.

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

- ▶ 1) Ensuring that appropriate indexes are created on columns frequently used. Indexing can significantly speed up data retrieval.
- ▶ 2)Instead of using SELECT *, explicitly specify the columns you need. This can improve query performance.
- ▶ 3)Use appropriate join types (e.g., INNER JOIN, LEFT JOIN) and ensure that join conditions are efficient. Also, avoid unnecessary joins.
- 4) Using stored procedures for frequently executed queries. Stored procedures can reduce network overhead and provide better performance for certain types of operations.
- > 5) Using the LIMIT clause to restrict the number of rows returned. Fetch only the data you need to minimize the impact on performance.