Diabetes Prediction

Internship Project [PSYLIQ]

Introduction

This dataset is provided by PSYLIQ , In this project, I will use the Diabetes Prediction data set to explore various aspects Diabetes and how they affect patients. The data set contains information about 100000 patients which are diabetes patients and their details such as Patient_id,gender, age, hypertension, heart_disease,smoking_history, bmi, HbA1c_level blood_glucose_level, diabetes .

1. Retrieve the Patient_id and ages of all patients.

```
select patient_id,age from diabetes_prediction;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

patient_id	age
PT101	31
PT102	31
PT103	31
PT104	31
PT105	35
PT106	35
PT107	35
PT108	35
PT109	35
PT110	34
PT111	34
PT112	34
PT113	34
PT114	34

diabetes_prediction 2

2. Select all female patients who are older than 40.

```
3
4 • select * from diabetes_prediction where gender="female" and age>40;
5
```

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

EmployeeName	Patient_id	gender	D.O.B	hypertension	heart_disease	smoking_history	Age	bmi	HbA1c_level	blood_glucose_level	diabetes

Because I don’t have age greater than 40. so it retrieve nothing after execution

3. Calculate the average BMI of patients.

```
select avg(bmi) as AVG_BMI from diabetes_prediction ;
```

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

AVG_BMI
27.321033180382557

4. List patients in descending order of blood glucose levels.

```
select * from diabetes_prediction order by blood_glucose_level desc;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

EmployeeName	Patient_id	gender	D.O.B	hypertension	heart_disease	smoking_history	Age	bmi	HbA1c_level	blood_glucose_level	diabetes
Adrian G Mendez	PT98419	Male	29-09-1995	0	0	not current	28	27.32	6.5	300	1
Lenora G Banks	PT98454	Female	29-09-1995	1	0	never	28	38.59	6.6	300	1
Dante Rogayan	PT98461	Male	29-09-1995	0	0	No Info	28	27.72	6.6	300	1
Tinisha C Bishop	PT98500	Male	29-09-1995	0	0	No Info	28	27.32	8.8	300	1
Tualatai Auimatagi	PT98538	Female	30-09-1995	0	0	never	28	26.52	8.2	300	1
Michelle D McGee	PT98852	Male	30-09-1995	0	0	ever	28	27.32	7.5	300	1
Lawrence Shum	PT98855	Male	30-09-1995	0	0	former	28	48.56	6.8	300	1
Seth I Rubenstein	PT98911	Female	30-09-1995	0	0	current	28	40.18	9	300	1
Philip Tran	PT99008	Male	01-10-1995	0	0	never	28	31.56	7	300	1
Gilbert J Fragoso	PT99638	Female	23-09-1995	1	0	ever	28	34.3	5.7	300	1

diabetes_prediction 2

5. Find patients who have hypertension and diabetes.

```
9
10 • select * from diabetes_prediction where hypertension="1" and diabetes="1";
11
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

EmployeeName	Patient_id	gender	D.O.B	hypertension	heart_disease	smoking_history	Age	bmi	HbA1c_level	blood_glucose_level	diabetes
Gilbert J Fragoso	PT99638	Female	23-09-1995	1	0	ever	28	34.3	5.7	300	1
Amado A Lumas Jr	PT99663	Male	23-09-1995	1	0	current	28	28.47	6.1	300	1
Shanice M Guidry	PT99672	Male	23-09-1995	1	0	never	28	41.93	5.7	300	1
Angelica J Young	PT99764	Male	24-09-1995	0	0	No Info	28	34	9	300	1
Flor D Roman	PT99809	Male	24-09-1995	0	0	not current	28	27.32	6	300	1
Clyde L Woods	PT99927	Male	24-09-1995	0	1	No Info	28	27.32	6.6	300	1
Josephine C Cabrera	PT99968	Male	24-09-1995	1	0	former	28	33.12	5.7	300	1
Marquis D Walker	PT100039	Male	24-09-1995	0	1	former	28	30.42	6.2	300	1
Clair Wildman	PT92189	Female	21-09-1995	0	0	No Info	28	27.32	6.2	300	1
John C Lynch	PT92506	Male	22-09-1995	0	0	No Info	28	29.26	6	300	1
Peter Po Kwong Yu	PT92513	Male	22-09-1995	0	0	former	28	27.32	6.1	300	1
Edson Marquez	PT92581	Male	22-09-1995	0	0	never	28	31.65	7.5	300	1
Mary Ann Moran	PT92871	Female	23-09-1995	0	0	never	28	53.4	5.8	300	1
Anthony Bruce	PT93259	Male	24-09-1995	0	0	ever	28	27.32	7.5	300	1

diabetes_prediction 5

6. Determine the number of patients with heart disease.

```
select count(patient_id) from diabetes_prediction where heart_disease=1;
```

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

count(patient_id)
3937

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

```
select smoking_history,count(patient_id) from diabetes_prediction
group by smoking_history;
```

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

smoking_history	count(patient_id)
never	35045
No Info	35753
current	9265
former	9324
ever	3997
not current	6434

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

```
select patient_id,bmi from diabetes_prediction where bmi> (select avg(bmi)
 from diabetes_prediction);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

patient_id	bmi
PT109	33.64
PT112	54.7
PT113	36.05
PT117	30.36
PT121	36.38
PT124	27.94
PT126	33.76
PT128	27.85
PT131	31.75
PT140	56.43
PT143	32.02
PT144	29.3
PT149	28.27
PT153	28.12

diabetes_prediction 9

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

```
(select patient_id,hba1c_level from diabetes_prediction order by HbA1c_level asc limit 1)
UNION
(select Patient_id,HbA1c_level from diabetes_prediction order by HbA1c_level desc limit 1);
```

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

patient_id	hba1c_level
PT120	3.5
PT141	9

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

```
select patient_id, abs(age-year(now())) as Year_of_Birth from diabetes_prediction;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

patient_id	Year_of_Birth
PT101	1993
PT102	1993
PT103	1993
PT104	1993
PT105	1989
PT106	1989
PT107	1989
PT108	1989
PT109	1989
PT110	1990
PT111	1990
PT112	1990
PT113	1990
PT114	1990

Result 11

11. Rank patients by blood glucose level within each gender group.

```
select patient_id,gender,blood_glucose_level, RANK() OVER (PARTITION BY gender order by
blood_glucose_level)
as Glucose_level_rank from diabetes_prediction ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

patient_id	gender	blood_glucose_level	Glucose_level_rank
PT98584	Female	80	1
PT96552	Female	80	1
PT98149	Female	80	1
PT97519	Female	80	1
PT96556	Female	80	1
PT97682	Female	80	1
PT97685	Female	80	1
PT99316	Female	80	1
PT99179	Female	80	1
PT96389	Female	80	1
PT97756	Female	80	1
PT98295	Female	80	1
PT97868	Female	80	1

Result 12

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

```
SET SQL_SAFE_UPDATES = 0;
update diabetes_prediction set smoking_history="Ex-smoker" where age>50;
```

13. Insert a new patient into the database with sample data.

```
insert into diabetes_prediction
values("Vyas Kumar","PT101010","Male","30-03-1999",0,0,"never",25.5,6.1,120,0,24);
```

```
select * from diabetes_prediction;
```

esult Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

EmployeeName	Patient_id	gender	D.O.B	hypertension	heart_disease	smoking_history	Age	bmi	HbA1c_level	blood_glucose_level	diabetes
Ruth S Bacuyani	PT100094	Female	24-09-1995	0	0	never	28	40.69	3.5	155	0
Jessica K Aldaz	PT100095	Female	24-09-1995	0	0	No Info	28	24.6	4.8	145	0
William Chun	PT100096	Female	24-09-1995	0	0	No Info	28	27.32	6.2	90	0
Antoinette L Wells	PT100097	Female	24-09-1995	0	0	No Info	28	17.37	6.5	100	0
Richard D Swart	PT100098	Male	24-09-1995	0	0	former	28	27.83	5.7	155	0
Vivian Chu	PT100099	Female	24-09-1995	0	0	never	28	35.42	4	100	0
Savitree Satram	PT100100	Female	25-09-1995	0	0	current	28	22.43	6.6	90	0
Vyas Kumar	PT101010	Male	30-03-1999	0	0	never	26	6.1	120	0	24

14. Delete all patients with heart disease from the database.

```
• delete from diabetes_prediction where heart_disease=1;
```

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

```
38 •  select * from diabetes_prediction where hypertension=1
39 ⊠  EXCEPT
40    select * from diabetes_prediction where diabetes=1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

EmployeeName	Patient_id	gender	D.O.B	hypertension	heart_disease	smoking_history	Age	bmi	HbA1c_level	blood_glucose_level	diabetes
DENISE SCHMITT	PT129	Male	29-06-1989	1	0	never	34	26.47	4	158	0
RAY CRAWFORD	PT155	Female	02-01-1997	1	0	never	27	23.05	4.8	130	0
KENNETH SMITH	PT161	Male	09-03-1997	1	0	current	26	27.86	6.6	145	0
CHARLES SCOTT	PT215	Female	08-06-1997	1	0	never	26	34.2	5.7	140	0
SHANNON SAKOWSKI	PT227	Male	02-07-1997	1	0	No Info	26	28.73	6.6	160	0
MARISA MORET	PT241	Female	13-07-1997	1	0	never	26	44.06	6.5	160	0
STEPHEN TACCHINI	PT326	Female	28-08-1997	1	0	never	26	36.73	6.6	126	0
ANDREW LOGAN	PT339	Male	05-09-1997	1	0	No Info	26	25.31	6	130	0
HAGOP HAJIAN	PT357	Female	13-09-1997	1	0	never	26	21.46	4	80	0
PERRY LEONG	PT377	Female	25-09-1997	1	0	No Info	26	24.29	3.5	90	0

Result 13

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

```
alter table diabetes_prediction modify Patient_id varchar(255) unique;
```

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

```
create view patient_info as select patient_id,age,bmi from diabetes_prediction;

select * from patient_info;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

patient_id	age	bmi
PT102	31	27.32
PT103	31	27.32
PT104	31	23.45
PT106	35	27.32
PT107	35	19.31
PT108	35	23.86
PT109	35	33.64
PT110	34	27.32
PT111	34	27.32

patient_info 14

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

To reduce data redundancy and improve data integrity in a database schema, you can consider the following best practices:

- 1. Normalization: Break down large tables, apply normalization techniques (1NF, 2NF, 3NF).
- 2. Primary Keys: Ensure each table has a unique primary key.
- 3. Foreign Keys: Establish relationships between tables for referential integrity.
- 4. Avoid Redundant Columns: Refrain from duplicating data across multiple tables.
- 5. Data Types: Use appropriate data types for columns.
- 6. Default Values and Constraints: Set defaults, use constraints to enforce rules.
- 7. Indexes: Apply indexes for frequently queried columns (considering trade-offs).
- 8. Avoid Nulls: Minimize NULL values; use defaults or separate tables.
- 9. Data Validation: Enforce validation at both application and database levels.
- 10. Use of Views: Presents data from multiple tables without duplicating information, simplifying queries, and maintaining a consistent view.

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

- **Indexing:**
  - Create indexes on key columns for faster data retrieval.
- **Query Optimization:**
  - Simplify queries by minimizing joins and subqueries.
- **Result Set Management:**
  - Use the LIMIT clause to restrict returned rows.
  - Fetch only necessary columns, avoiding SELECT *.
- **Stored Procedures:**
  - Encapsulate frequent queries in stored procedures.
- **Statistics Update:**
  - Regularly update table statistics for accurate query optimization.
- **Distinct Usage:**
  - Minimize the use of SELECT DISTINCT; explore alternatives like GROUP BY or refined logic.

THANK YOU

Vyas Kumar
7717706832
vyaskv123@gmail.com
www.linkedin.com/in/vyas-kumar-969388220