Diabetes Prediction Analysis

BY RAHUL ACHARYA

Retrieve the Patient_id and ages of all patients

```
4 •  SELECT Patient_id, age
5    FROM diabetes_prediction;
6
7
8
9
```

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

Patient_id	age
PT101	80
PT102	54
PT103	28
PT104	36
PT105	76
PT106	20
PT107	44
PT108	79
PT109	42
PT110	32
PT111	53

diabetes_prediction 17

Output

Action Output

#	Time	Action	Message
1	01:12:36	SELECT Patient_id, age FROM diabetes_prediction	100000 row(s) returned

Select all female patients who are older than 40

```
15 •  SELECT *
16    FROM diabetes_prediction
17    WHERE gender = "female" AND age>40;
18
19
20
```

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

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	hbA1c_level	blood_glucose_level	diabetes
NATHANIEL FORD	PT101	Female	80	0	1	never	25.19	6.6	140	0
GARY JIMENEZ	PT102	Female	54	0	0	No Info	27.32	6.6	80	0
ALSON LEE	PT107	Female	44	0	0	never	19.31	6.5	200	1
DAVID KUSHNER	PT108	Female	79	0	0	No Info	23.86	5.7	85	0
ARTHUR KENNEY	PT111	Female	53	0	0	never	27.32	6.1	85	0
PATRICIA JACKSON	PT112	Female	54	0	0	former	54.7	6	100	0
EDWARD HARRINGTON	PT113	Female	78	0	0	former	36.05	5	130	0
JOHN MARTIN	PT114	Female	67	0	0	never	25.69	5.8	200	0
DAVID FRANKLIN	PT115	Female	76	0	0	No Info	27.32	5	160	0
SEBASTIAN WONG	PT118	Female	42	0	0	never	24.48	5.7	158	0
MARTY ROSS	PT119	Female	42	0	0	No Info	27.32	5.7	80	0
GEORGE GARCIA	PT123	Female	69	0	0	never	21.24	4.8	85	0
VICTOR WYRSCH	PT124	Female	72	0	1	former	27.94	6.5	130	0

diabetes_prediction 51

Output

Action Output

#	Time	Action	Message
1	01:55:21	SELECT * FROM diabetes_prediction WHERE gender = "female" AND age>40	31155 row(s) returned

Calculate the average BMI of patients

```
SELECT avg(bmi) as Average_bmi
FROM diabetes_prediction;
```

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

Average_bmi
27.32076709999422

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	age	hypertension	heart_disease	smoking_history	bmi	hbA1c_level	blood_glucose_level	diabetes
ELIZABETH MOSER	PT30101	Female	39	0	0	current	30.73	8.8	300	1
BRIAN HINZE	PT30286	Male	38	0	0	No Info	27.32	7	300	1
JANE JOHNSON	PT30361	Male	68	1	0	No Info	25.82	6.5	300	1
TONYA BREAUX	PT30561	Female	80	0	0	never	27.32	8.2	300	1
ISSEL ALVAREZ	PT30968	Male	58	1	0	No Info	27.32	6.8	300	1
VICKI LEE	PT28652	Female	80	0	0	never	20.67	6.6	280	1
ENRIQUE MORA	PT28707	Female	80	0	0	never	29.39	7.5	280	1
WINSTON LOUIE	PT28746	Female	42	0	0	never	72.89	6.8	280	1
ALBERTO GUAD...	PT28863	Male	58	0	0	former	42	7	280	1
BETTY WONG	PT29077	Male	65	0	1	never	27.32	8.8	280	1
KEVIN MATTIAS	PT29189	Male	39	0	0	never	27.79	6.2	280	1
STEPHANIE COE	PT29215	Female	30	0	0	never	45.21	6.1	280	1
ZHENHUA LI	PT29271	Male	80	0	0	former	27.32	6	280	1

Find patients who have hypertension and diabetes

```
38 •  SELECT *
39    FROM diabetes_prediction
40    WHERE hypertension=1 AND diabetes=1;
41
42
```

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

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	hbA1c_level	blood_glucose_level	diabetes
JONES WONG	PT139	Male	50	1	0	current	27.32	5.7	260	1
PATRIC STEELE	PT205	Female	80	1	0	never	27.32	6.8	280	1
ARTHUR STELLINI	PT343	Male	57	1	1	not current	27.77	6.6	160	1
CHAD LAW	PT355	Male	63	1	0	ever	35.06	5.8	200	1
CATHERINE JAMES	PT451	Female	52	1	0	never	50.3	6.6	155	1
JOHN HART	PT565	Male	48	1	0	current	36.12	6.8	140	1
JOHN BARKER	PT567	Female	79	1	0	former	27.32	6.5	159	1
ROBERT BONNET	PT632	Female	49	1	0	not current	36.93	8.8	155	1
VITANI BENJAMIN	PT727	Male	43	1	0	not current	40.86	6.6	159	1
LANNIE ADELMAN	PT828	Female	38	1	0	not current	27.32	6.1	160	1
JOEL DELIZONNA	PT852	Female	28	1	0	never	20.09	6.6	200	1
KAREN KUBICK	PT861	Male	59	1	0	ever	25.94	9	140	1
ANA GONZALEZ	PT983	Female	75	1	0	No Info	27.32	6.6	240	1

diabetes_prediction 55

Output

Action Output

#	Time	Action	Message
1	01:57:08	SELECT * FROM diabetes_prediction WHERE hypertension=1 AND diabetes=1	2088 row(s) returned

Determine the number of patients with heart disease

```
SELECT COUNT(*) as Heart_disease_patients
FROM diabetes_prediction
WHERE heart_disease=1;
```

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

Heart_disease_patients
3942

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

```
SELECT smoking_history, COUNT(*) as Count
FROM diabetes_prediction
WHERE smoking_history = "current" OR smoking_history="never"
GROUP BY smoking_history;
```

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

smoking_history	Count
never	35095
current	9286

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

```
# Avg bmi is 27.3207670999422
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

diabetes_prediction 30

Output

Action Output

#	Time	Action	Message
1	01:26:32	SELECT Patient_id, bmi FROM diabetes_prediction WHERE bmi > (SELECT AVG(bmi) FROM diabetes_predicti...	33768 row(s) returned

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

```
SELECT EmployeeName, Patient_id, hbA1c_level as Max_hbA1c_level
FROM diabetes_prediction
ORDER BY hbA1c_level DESC
LIMIT 1;
```

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

EmployeeName	Patient_id	Max_hbA1c_level
MICHAEL THOMPSON	PT141	9

```
SELECT EmployeeName, Patient_id, hbA1c_level as Min_hbA1c_level
FROM diabetes_prediction
ORDER BY hbA1c_level
LIMIT 1;
```

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

EmployeeName	Patient_id	Min_hbA1c_level
ELLEN MOFFATT	PT120	3.5

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

```sql
SELECT EmployeeName, Patient_id,
    YEAR(NOW()) - age AS Birth_year,
    YEAR(NOW()) - YEAR(NOW()) + age AS Current_age
FROM diabetes_prediction;
```

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

EmployeeName	Patient_id	Birth_year	Current_age
NATHANIEL FORD	PT101	1943	80
GARY JIMENEZ	PT102	1969	54
ALBERT PARDINI	PT103	1995	28
CHRISTOPHER CHONG	PT104	1987	36
PATRICK GARDNER	PT105	1947	76
DAVID SULLIVAN	PT106	2003	20
ALSON LEE	PT107	1979	44
DAVID KUSHNER	PT108	1944	79
MICHAEL MORRIS	PT109	1981	42
JOANNE HAYES-WHITE	PT110	1991	32
ARTHUR KENNEY	PT111	1970	53
PATRICIA JACKSON	PT112	1969	54
EDWARD HARRINGTON	PT113	1945	78

Result 37

Output

Action Output

#	Time	Action	Message
1	01:33:31	SELECT EmployeeName, Patient_id, YEAR(NOW()) - age AS Birth_year, YEAR(NOW()) - YEAR(NOW()) + ...	100000 row(s) returned

Rank patients by blood glucose level within each gender group

```sql
SELECT Patient_id, gender, blood_glucose_level,
    RANK() OVER (PARTITION BY gender ORDER BY blood_glucose_level) AS blood_glucose_ranks_as_per_gender
FROM diabetes_prediction;
```

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

Patient_id	gender	blood_glucose_level	blood_glucose_ranks_as_per_gender
PT47949	Female	80	1
PT47157	Female	80	1
PT47163	Female	80	1
PT47171	Female	80	1
PT49143	Female	80	1
PT46271	Female	85	4199
PT46110	Female	85	4199
PT48066	Female	85	4199
PT46675	Female	85	4199
PT49723	Female	85	4199
PT46114	Female	85	4199
PT48119	Female	85	4199

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;

SELECT * FROM diabetes_prediction;
```

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

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	hbA1c_level	blood_glucose_level	diabetes
NATHANIEL FORD	PT101	Female	80	0	1	Ex-smoker	25.19	6.6	140	0
GARY JIMENEZ	PT102	Female	54	0	0	Ex-smoker	27.32	6.6	80	0
ALBERT PARDINI	PT103	Male	28	0	0	never	27.32	5.7	158	0
CHRISTOPHER CHONG	PT104	Female	36	0	0	current	23.45	5	155	0
PATRICK GARDNER	PT105	Male	76	1	1	Ex-smoker	20.14	4.8	155	0
DAVID SULLIVAN	PT106	Female	20	0	0	never	27.32	6.6	85	0
ALSON LEE	PT107	Female	44	0	0	never	19.31	6.5	200	1
DAVID KUSHNER	PT108	Female	79	0	0	Ex-smoker	23.86	5.7	85	0
MICHAEL MORRIS	PT109	Male	42	0	0	never	33.64	4.8	145	0
JOANNE HAYES-WHITE	PT110	Female	32	0	0	never	27.32	5	100	0
ARTHUR KENNEY	PT111	Female	53	0	0	Ex-smoker	27.32	6.1	85	0
PATRICIA JACKSON	PT112	Female	54	0	0	Ex-smoker	54.7	6	100	0
EDWARD HARRINGTON	PT113	Female	78	0	0	Ex-smoker	36.05	5	130	0

diabetes_prediction 39

Output

Action Output

#	Time	Action	Message
4	01:37:08	UPDATE diabetes_prediction SET smoking_history="Ex-smoker" WHERE age>50	38463 row(s) affected Rows matched: 38463 Changed: 38463 Warnings: 0
5	01:37:20	SELECT * FROM diabetes_prediction	100000 row(s) returned

Insert a new patient into the database with sample data

```
INSERT INTO diabetes_prediction
VALUES("Harry Potter", "PT100101","Male", 30, 0, 0, "never", 29.30, 5, 100, 0);

SELECT * FROM diabetes_prediction;

#Q14)Delete all patients with heart disease from the database.
#Will do it at the end
```

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

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	hbA1c_level	blood_glucose_level	diabetes
Tanya J Bustamante	PT100090	Female	26	0	0	No Info	27.32	5	158	0
Armando A Aguilar	PT100091	Male	39	0	0	No Info	27.32	6.1	100	0
Sandra A May	PT100092	Male	22	0	0	current	29.65	6	80	0
James B Roberson	PT100093	Female	26	0	0	never	34.34	6.5	160	0
Ruth S Bacuyani	PT100094	Female	40	0	0	never	40.69	3.5	155	0
Jessica K Aldaz	PT100095	Female	36	0	0	No Info	24.6	4.8	145	0
William Chun	PT100096	Female	80	0	0	Ex-smoker	27.32	6.2	90	0
Antoinette L Wells	PT100097	Female	2	0	0	No Info	17.37	6.5	100	0
Richard D Swart	PT100098	Male	66	0	0	Ex-smoker	27.83	5.7	155	0
Vivian Chu	PT100099	Female	24	0	0	never	35.42	4	100	0
Savitree Satram	PT100100	Female	57	0	0	Ex-smoker	22.43	6.6	90	0
Harry Potter	PT100101	Male	30	0	0	never	29.3	5	100	0

diabetes_prediction 40

Output

Action Output

#	Time	Action	Message
6	01:38:13	INSERT INTO diabetes_prediction VALUES("Harry Potter", "PT100101","Male", 30, 0, 0, "never", 29.30, 5, 1...	1 row(s) affected
7	01:38:17	SELECT * FROM diabetes_prediction	100001 row(s) returned

Delete all patients with heart disease from the database

```
117 •  DELETE FROM diabetes_prediction
118    WHERE heart_disease = 1;
119 •  SELECT * FROM diabetes_prediction;
120
121
```

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

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	hbA1c_level	blood_glucose_level	diabetes
SEBASTIAN WONG	PT118	Female	42	0	0	never	24.48	5.7	158	0
MARTY ROSS	PT119	Female	42	0	0	No Info	27.32	5.7	80	0
ELLEN MOFFATT	PT120	Male	37	0	0	ever	25.72	3.5	159	0
VENUS AZAR	PT121	Male	40	0	0	current	36.38	6	90	0
JUDY MELINEK	PT122	Male	5	0	0	No Info	18.8	6.2	85	0
GEORGE GARCIA	PT123	Female	69	0	0	never	21.24	4.8	85	0
JOSEPH DRISCOLL	PT125	Female	4	0	0	No Info	13.99	4	140	0
GREGORY SUHR	PT126	Male	30	0	0	never	33.76	6.1	126	0
RAYMOND GUZMAN	PT128	Male	40	0	0	former	27.85	5.8	80	0
DENISE SCHMITT	PT129	Male	45	1	0	never	26.47	4	158	0
MONICA FIELDS	PT130	Male	43	0	0	never	26.08	6.1	155	0
HARLAN KELLY-JR	PT131	Female	53	0	0	No Info	31.75	4	200	0
DAVID SHINN	PT132	Male	50	0	0	No Info	25.15	4	145	0

diabetes_prediction 59

Output

Action Output

#	Time	Action	Message
4	01:58:41	SELECT EmployeeName, Patient_id, hbA1c_level as Max_hbA1c_level FROM diabetes_prediction ORDER B...	1 row(s) returned
5	02:01:46	DELETE FROM diabetes_prediction WHERE heart_disease = 1	3942 row(s) affected
6	02:02:13	SELECT * FROM diabetes_prediction	96058 row(s) returned

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

```
SELECT Patient_id, hypertension, diabetes
FROM diabetes_prediction WHERE hypertension=1
EXCEPT
SELECT Patient_id, hypertension, diabetes
FROM diabetes_prediction WHERE diabetes=1;
```

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

Patient_id	hypertension	diabetes
PT105	1	0
PT129	1	0
PT143	1	0
PT155	1	0
PT161	1	0
PT215	1	0
PT220	1	0
PT227	1	0
PT241	1	0
PT326	1	0
PT339	1	0
PT357	1	0
PT377	1	0
PT379	1	0

Result 12

Output

Action Output

#	Time	Action	Message
1	13:08:30	SELECT Patient_id, hypertension, diabetes FROM diabetes_prediction WHERE hypertension=1 EXCEPT SELE...	5397 row(s) returned

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

```sql
ALTER TABLE diabetes_prediction
ADD CONSTRAINT Patient_id UNIQUE (Patient_id);

INSERT INTO diabetes_prediction
VALUES ("Harry Potter", "PT100101", "Male", 30, 0, 0, "never", 29.3, 5, 100, 0);
```

Output

Action Output

#	Time	Action	Message
1	01:41:40	INSERT INTO diabetes_prediction VALUES ("Harry Potter", "PT100101", "Male",30,0, 0, "never", 29.3, 5, 100,...	Error Code: 1062. Duplicate entry 'PT100101' for key 'diabetes_prediction.Patient_id'

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

```sql
CREATE VIEW Patient_data AS (
    SELECT Patient_id, age, bmi
    FROM diabetes_prediction
);
SELECT * FROM Patient_data;
```

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

Patient_id	age	bmi
PT101	80	25.19
PT102	54	27.32
PT103	28	27.32
PT104	36	23.45
PT105	76	20.14
PT106	20	27.32
PT107	44	19.31
PT108	79	23.86
PT109	42	33.64
PT110	32	27.32
PT111	53	27.32
PT112	54	54.7
PT113	78	36.05

Patient_data 42

Output

Action Output

#	Time	Action	Message
1	01:43:52	CREATE VIEW Patient_data AS (SELECT Patient_id, age, bmi FROM diabetes_prediction)	0 row(s) affected
2	01:44:08	SELECT * FROM Patient_data	100001 row(s) returned

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.