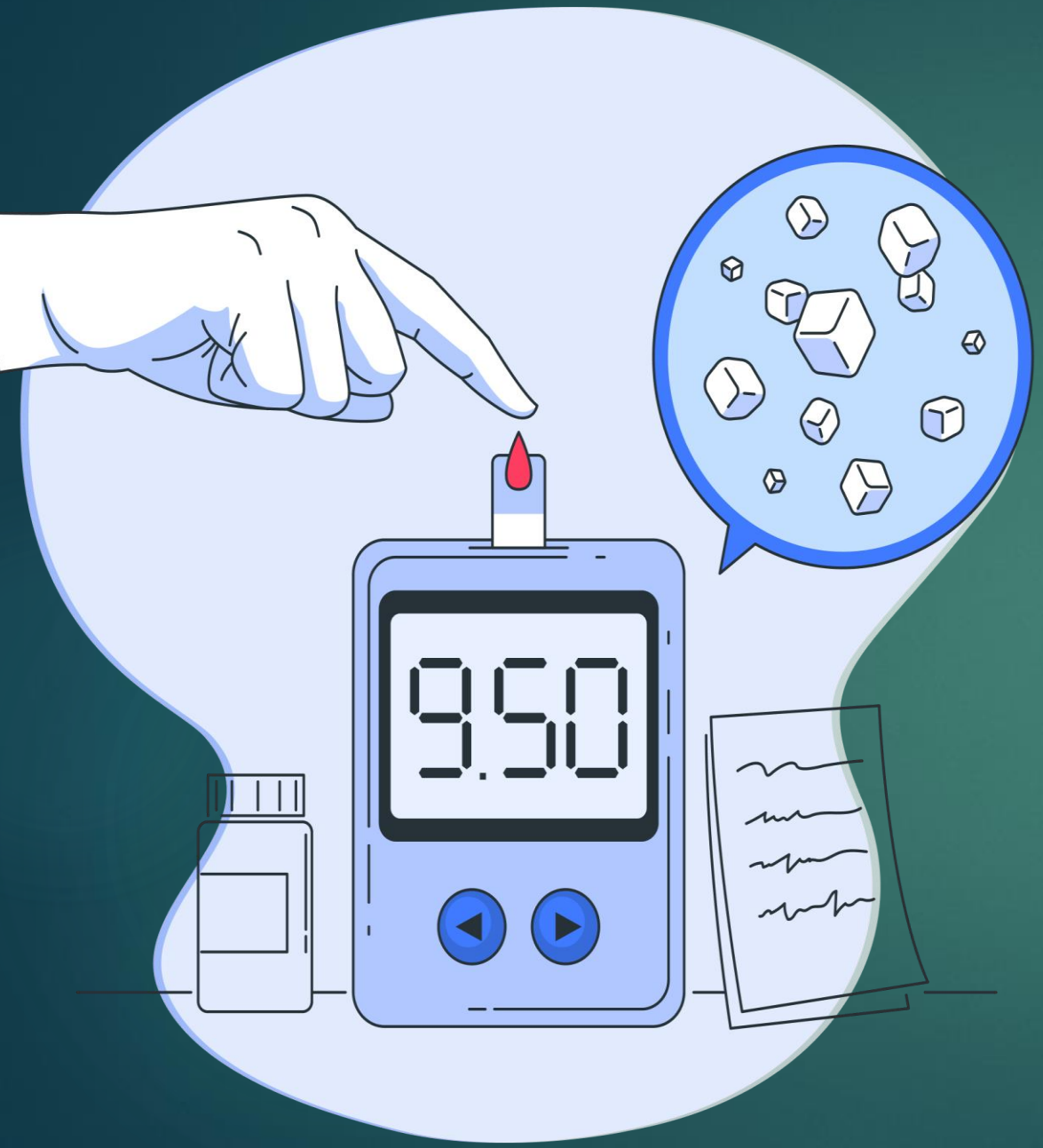
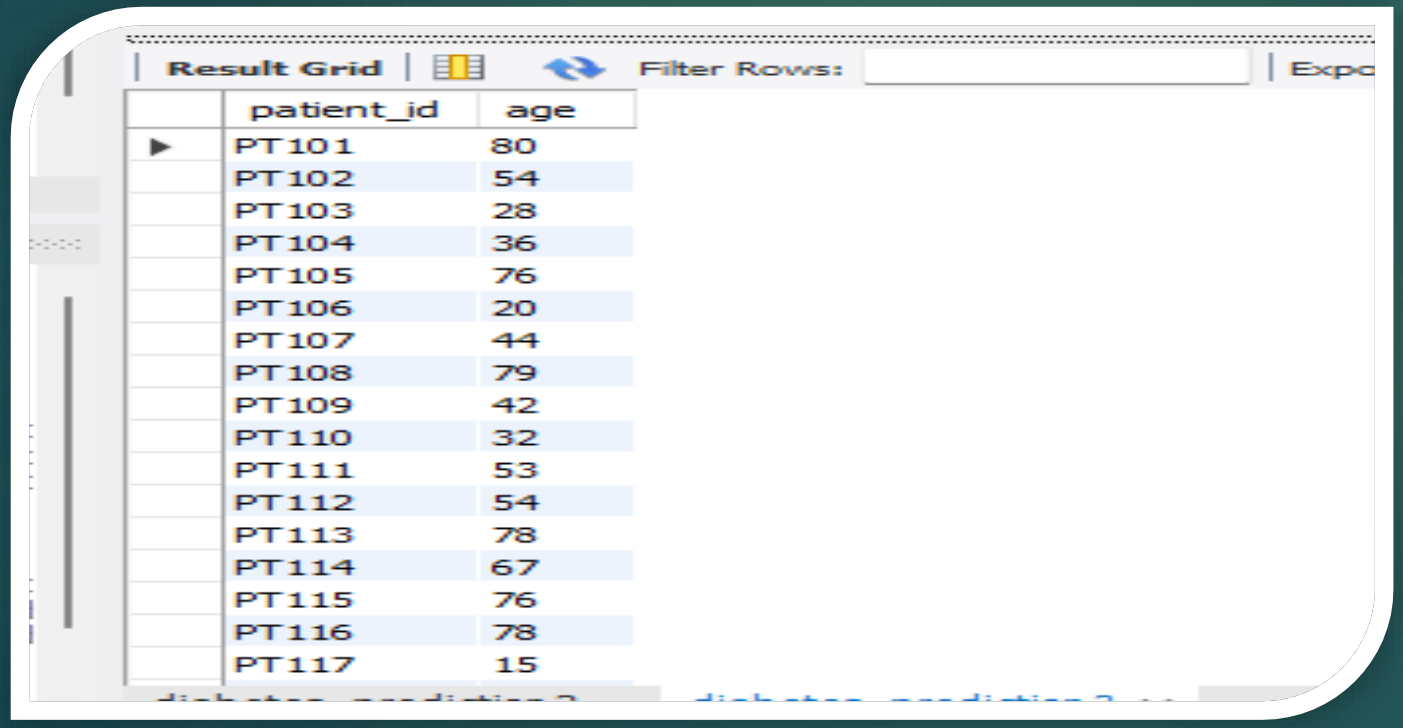


DIABETES PREDICTION ANALYSIS

1. Retrieve the Patient_id and ages of all patients.

- SELECT * FROM diabetes_prediction;
- select patient_id ,age from diabetes_prediction ;

O/p:

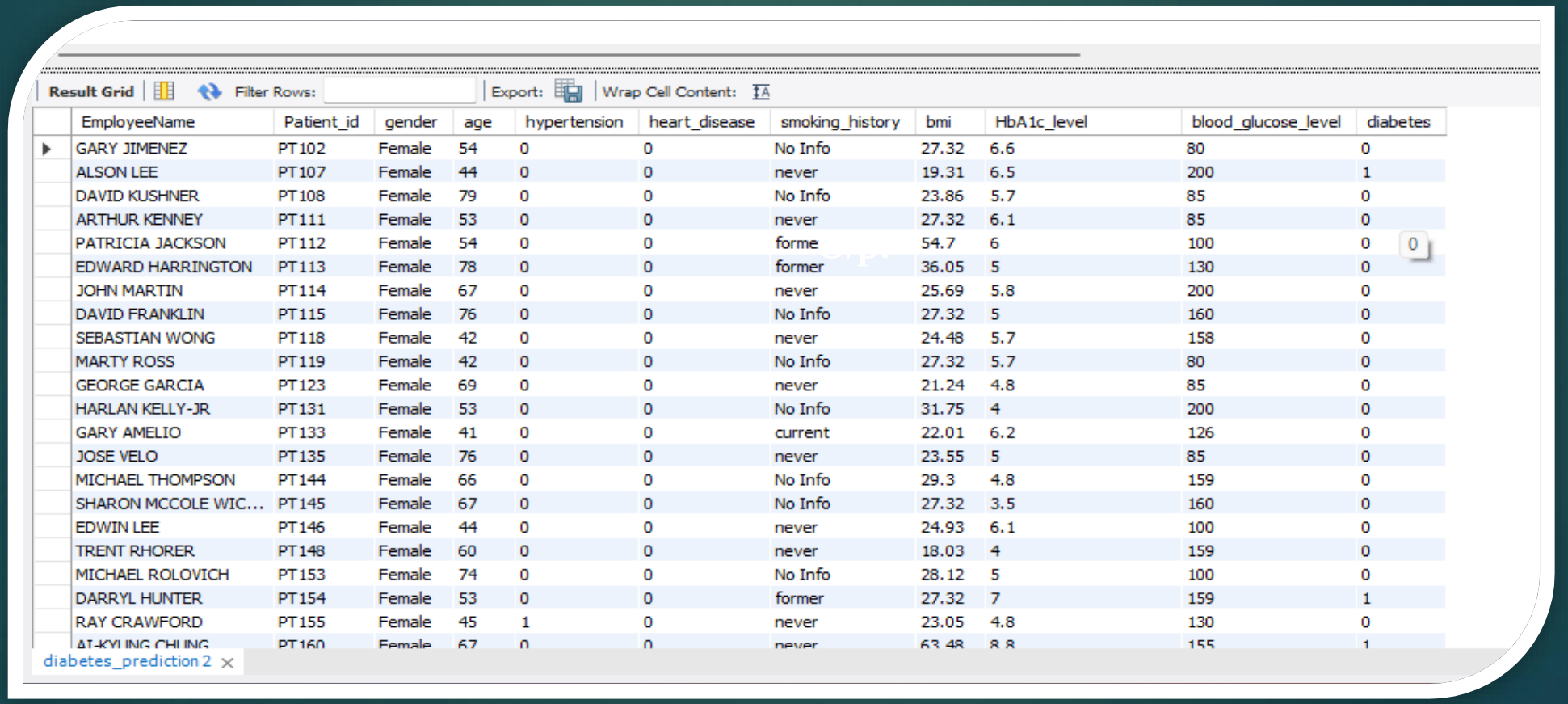


patient_id	age
PT101	80
PT102	54
PT103	28
PT104	36
PT105	76
PT106	20
PT107	44
PT108	79
PT109	42
PT110	32
PT111	53
PT112	54
PT113	78
PT114	67
PT115	76
PT116	78
PT117	15

2. Select all female patients who are older than40.

- select * from diabetes_prediction where gender = "female" and age > 40;

O/p:



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

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	HbA1c_level	blood_glucose_level	diabetes
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	forme	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
HARLAN KELLY-JR	PT131	Female	53	0	0	No Info	31.75	4	200	0
GARY AMELIO	PT133	Female	41	0	0	current	22.01	6.2	126	0
JOSE VELO	PT135	Female	76	0	0	never	23.55	5	85	0
MICHAEL THOMPSON	PT144	Female	66	0	0	No Info	29.3	4.8	159	0
SHARON MCCOLE WIC...	PT145	Female	67	0	0	No Info	27.32	3.5	160	0
EDWIN LEE	PT146	Female	44	0	0	never	24.93	6.1	100	0
TRENT RHORER	PT148	Female	60	0	0	never	18.03	4	159	0
MICHAEL ROLOVICH	PT153	Female	74	0	0	No Info	28.12	5	100	0
DARRYL HUNTER	PT154	Female	53	0	0	former	27.32	7	159	1
RAY CRAWFORD	PT155	Female	45	1	0	never	23.05	4.8	130	0
AI-KYLING CHUNG	PT160	Female	67	0	0	never	63.48	8.8	155	1

diabetes_prediction 2

3. Calculate the average BMI of patients.

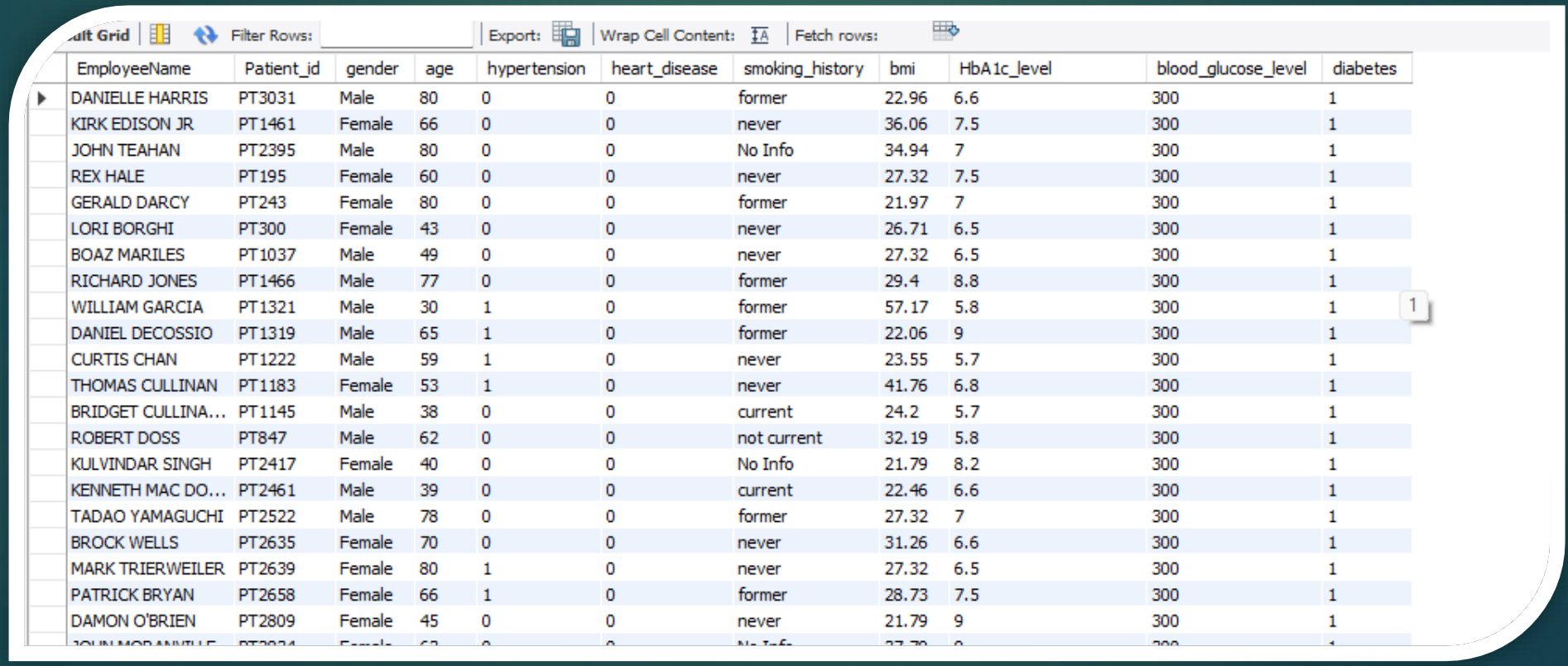
- select avg(bmi) as average_bmi from diabetes_prediction;

O/p:

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

- select * from diabetes_prediction order by blood_glucose_level desc;

O/p:



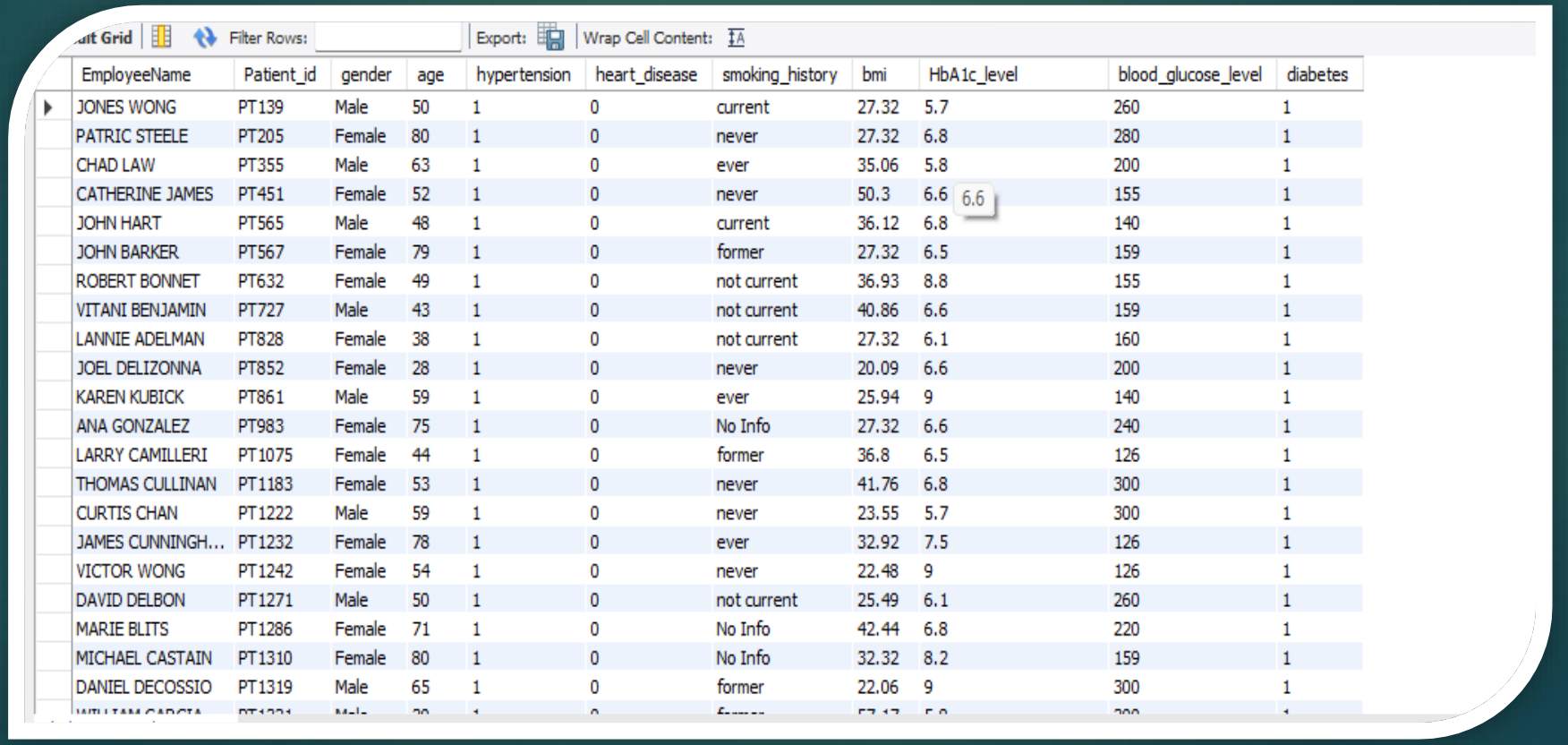
Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	HbA1c_level	blood_glucose_level	diabetes
DANIELLE HARRIS	PT3031	Male	80	0	0	former	22.96	6.6	300	1
KIRK EDISON JR	PT1461	Female	66	0	0	never	36.06	7.5	300	1
JOHN TEAHAN	PT2395	Male	80	0	0	No Info	34.94	7	300	1
REX HALE	PT195	Female	60	0	0	never	27.32	7.5	300	1
GERALD DARCY	PT243	Female	80	0	0	former	21.97	7	300	1
LORI BORGHI	PT300	Female	43	0	0	never	26.71	6.5	300	1
BOAZ MARILES	PT1037	Male	49	0	0	never	27.32	6.5	300	1
RICHARD JONES	PT1466	Male	77	0	0	former	29.4	8.8	300	1
WILLIAM GARCIA	PT1321	Male	30	1	0	former	57.17	5.8	300	1
DANIEL DECOSSIO	PT1319	Male	65	1	0	former	22.06	9	300	1
CURTIS CHAN	PT1222	Male	59	1	0	never	23.55	5.7	300	1
THOMAS CULLINAN	PT1183	Female	53	1	0	never	41.76	6.8	300	1
BRIDGET CULLINA...	PT1145	Male	38	0	0	current	24.2	5.7	300	1
ROBERT DOSS	PT847	Male	62	0	0	not current	32.19	5.8	300	1
KULVINDAR SINGH	PT2417	Female	40	0	0	No Info	21.79	8.2	300	1
KENNETH MAC DO...	PT2461	Male	39	0	0	current	22.46	6.6	300	1
TADAO YAMAGUCHI	PT2522	Male	78	0	0	former	27.32	7	300	1
BROCK WELLS	PT2635	Female	70	0	0	never	31.26	6.6	300	1
MARK TRIERWEILER	PT2639	Female	80	1	0	never	27.32	6.5	300	1
PATRICK BRYAN	PT2658	Female	66	1	0	former	28.73	7.5	300	1
DAMON O'BRIEN	PT2809	Female	45	0	0	never	21.79	9	300	1

5. Find patients who have hypertension and diabetes.

- select * from diabetes_prediction where hypertension = 1 and diabetes = 1;

O/p:



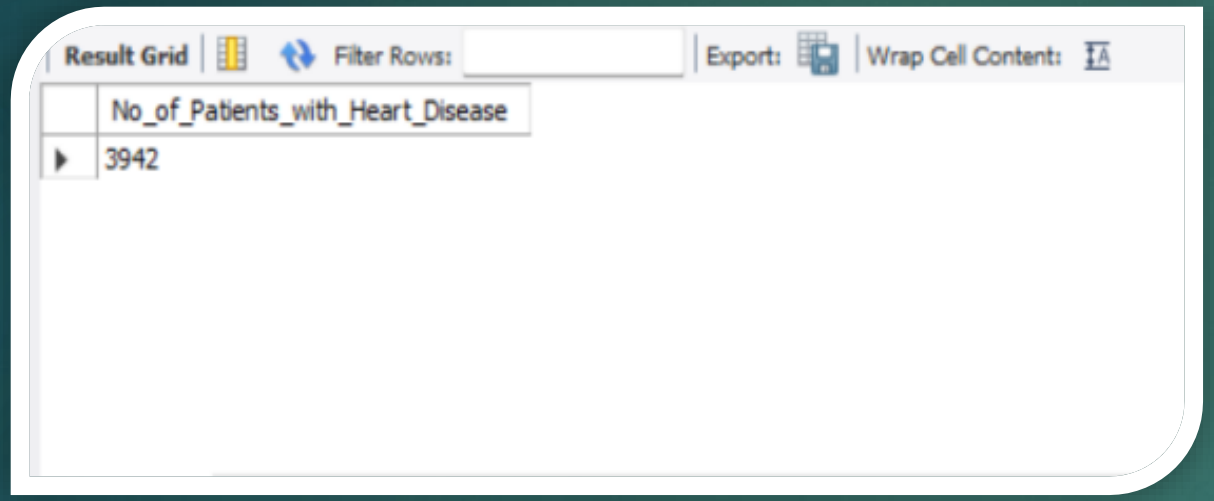
Filter Rows: Export: Wrap Cell Content:

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
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
LARRY CAMILLERI	PT1075	Female	44	1	0	former	36.8	6.5	126	1
THOMAS CULLINAN	PT1183	Female	53	1	0	never	41.76	6.8	300	1
CURTIS CHAN	PT1222	Male	59	1	0	never	23.55	5.7	300	1
JAMES CUNNINGH...	PT1232	Female	78	1	0	ever	32.92	7.5	126	1
VICTOR WONG	PT1242	Female	54	1	0	never	22.48	9	126	1
DAVID DELBON	PT1271	Male	50	1	0	not current	25.49	6.1	260	1
MARIE BLITS	PT1286	Female	71	1	0	No Info	42.44	6.8	220	1
MICHAEL CASTAIN	PT1310	Female	80	1	0	No Info	32.32	8.2	159	1
DANIEL DECOSSIO	PT1319	Male	65	1	0	former	22.06	9	300	1

6. Determine the number of patients with heart disease.

- Select count(Patient_id) as No_of_Patients_with_Heart_Disease from diabetes_prediction where heart_disease = '1';

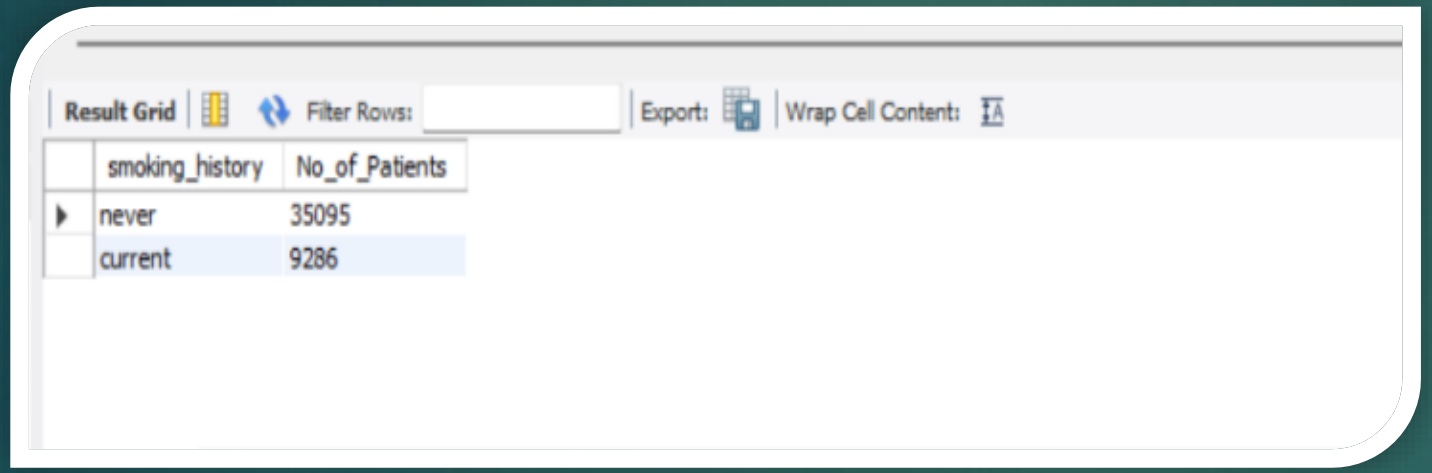
O/p:



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

- select smoking_history, count(*) as No_of_Patients from diabetes_prediction where smoking_history in ('current','never') group by 1;

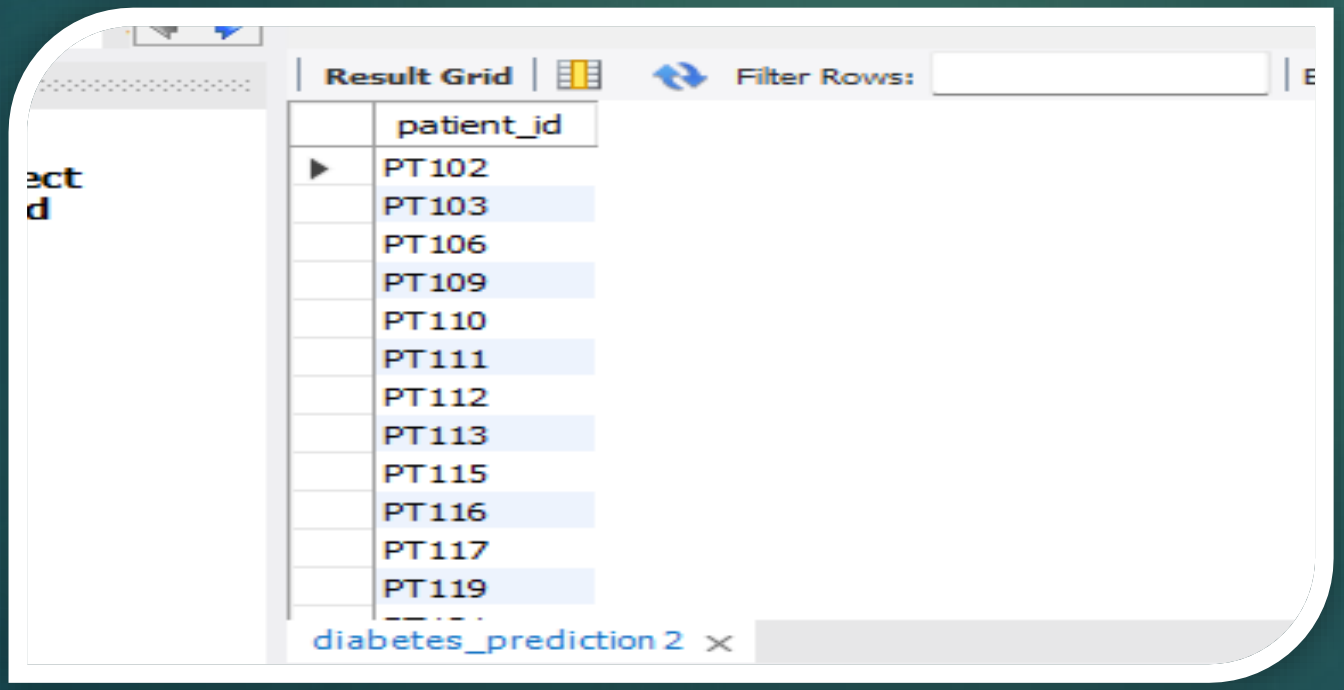
O/p:



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

- select patient_id from diabetes_prediction where bmi >(select avg(bmi) from diabetes_prediction);

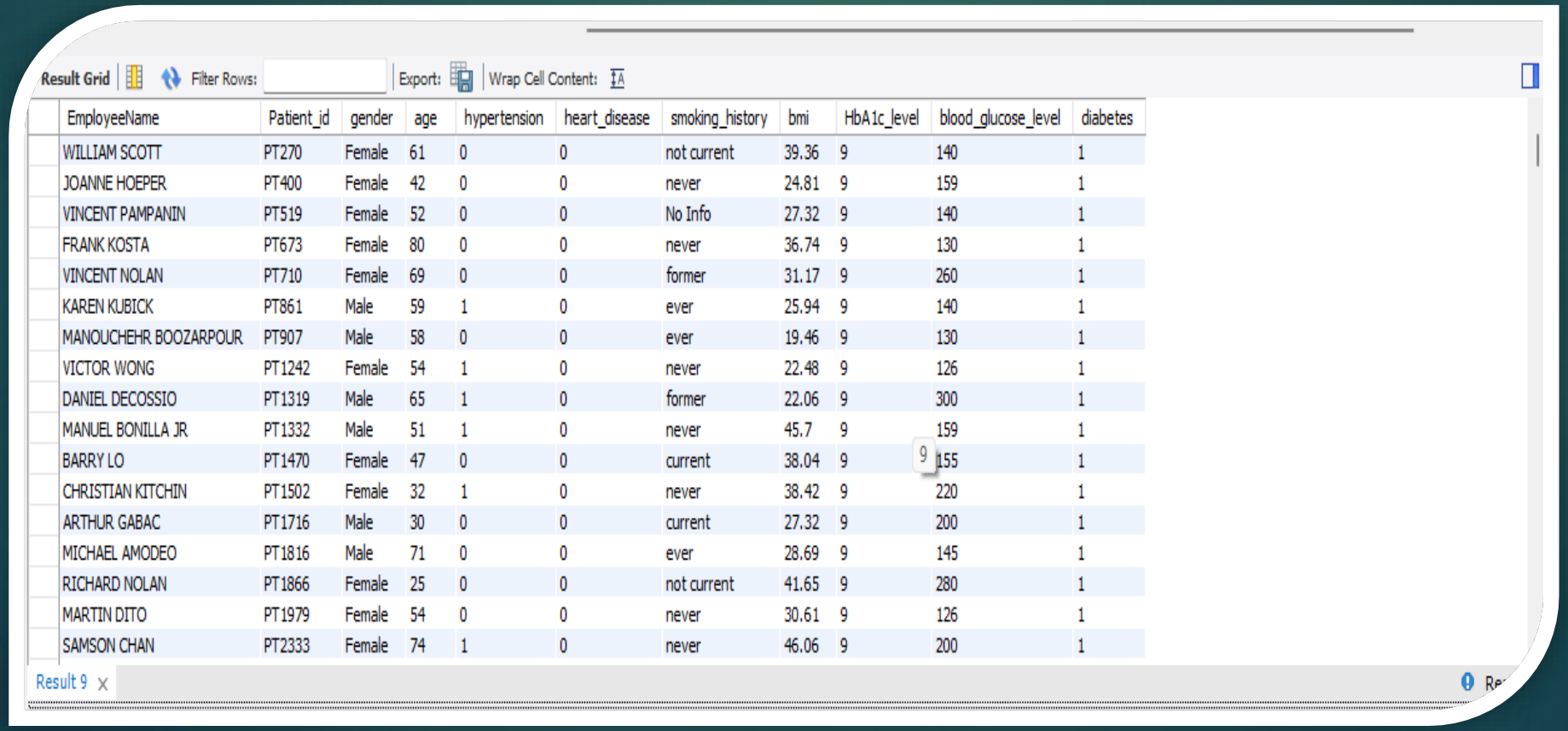
O/p:



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

- select * from diabetes_prediction where HbA1c_level = (select max(HbA1c_level) from diabetes_prediction) union select * from diabetes_prediction where HbA1c_level = (select min(HbA1c_level)from diabetes_prediction);

O/p:



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

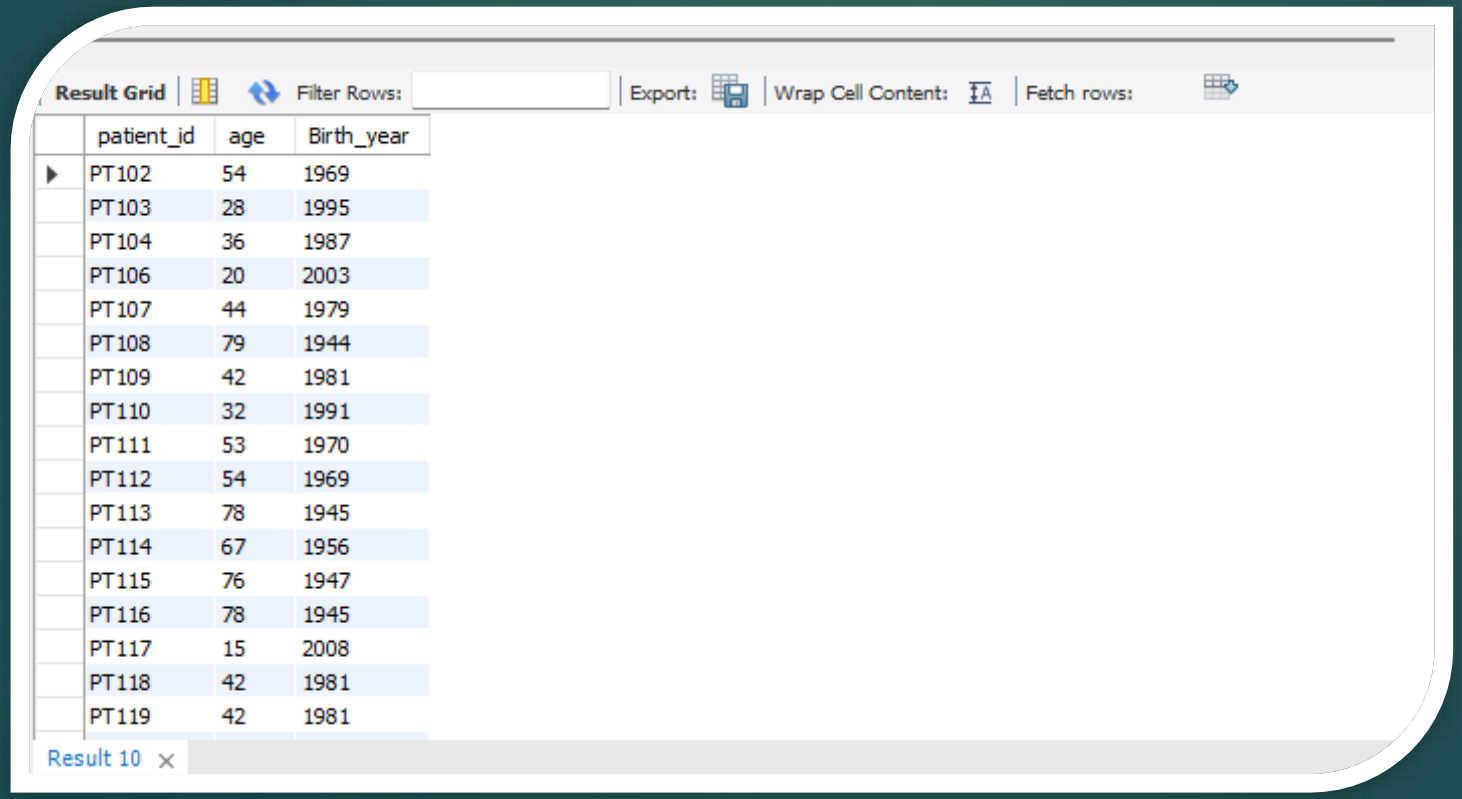
EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	HbA1c_level	blood_glucose_level	diabetes
WILLIAM SCOTT	PT270	Female	61	0	0	not current	39.36	9	140	1
JOANNE HOEPER	PT400	Female	42	0	0	never	24.81	9	159	1
VINCENT PAMPANIN	PT519	Female	52	0	0	No Info	27.32	9	140	1
FRANK KOSTA	PT673	Female	80	0	0	never	36.74	9	130	1
VINCENT NOLAN	PT710	Female	69	0	0	former	31.17	9	260	1
KAREN KUBICK	PT861	Male	59	1	0	ever	25.94	9	140	1
MANOUCHEHR BOOZARPOUR	PT907	Male	58	0	0	ever	19.46	9	130	1
VICTOR WONG	PT1242	Female	54	1	0	never	22.48	9	126	1
DANIEL DECOSSIO	PT1319	Male	65	1	0	former	22.06	9	300	1
MANUEL BONILLA JR	PT1332	Male	51	1	0	never	45.7	9	159	1
BARRY LO	PT1470	Female	47	0	0	current	38.04	9	155	1
CHRISTIAN KITCHIN	PT1502	Female	32	1	0	never	38.42	9	220	1
ARTHUR GABAC	PT1716	Male	30	0	0	current	27.32	9	200	1
MICHAEL AMODEO	PT1816	Male	71	0	0	ever	28.69	9	145	1
RICHARD NOLAN	PT1866	Female	25	0	0	not current	41.65	9	280	1
MARTIN DITO	PT1979	Female	54	0	0	never	30.61	9	126	1
SAMSON CHAN	PT2333	Female	74	1	0	never	46.06	9	200	1

Result 9

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

- select patient_id,age,year(current_date()) - age as 'Birth_year'from diabetes_prediction;

O/p:

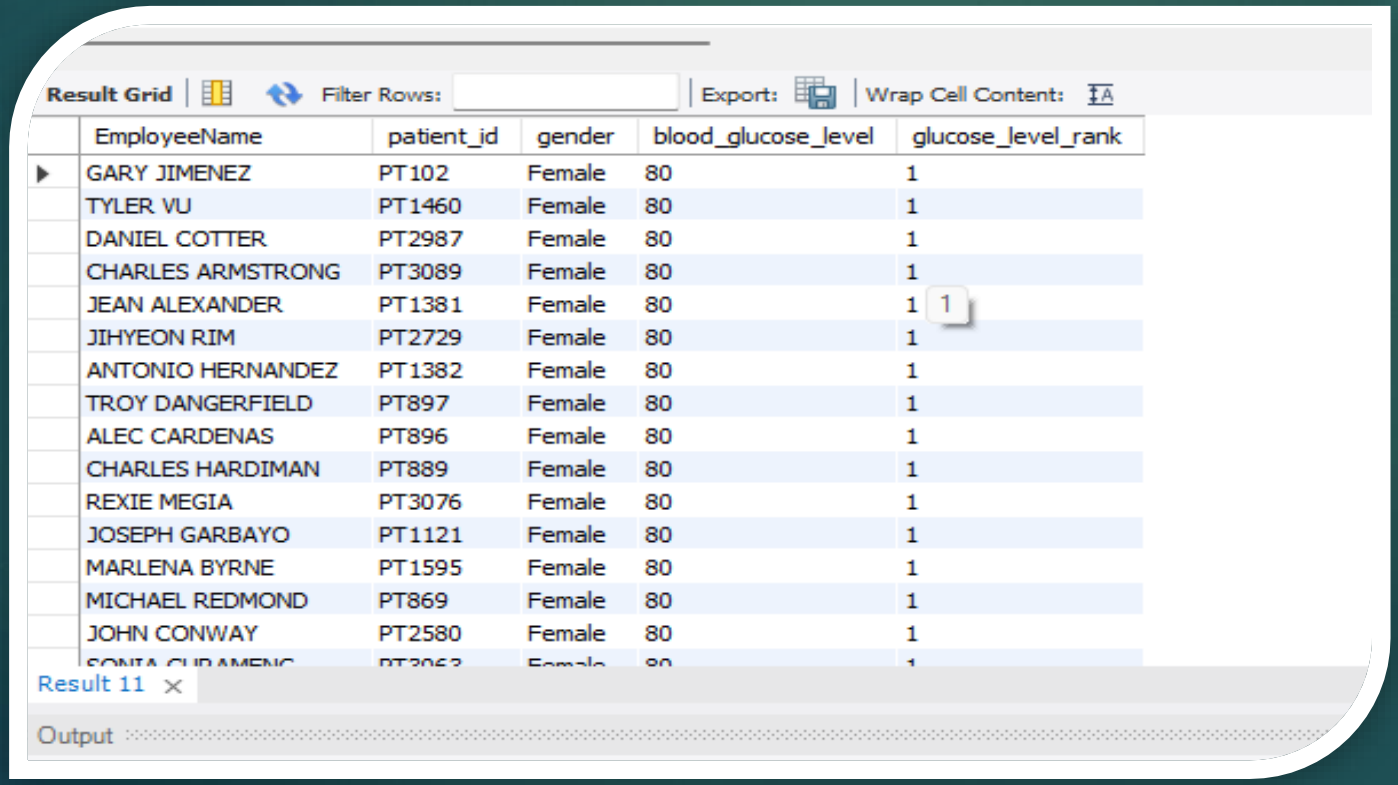


patient_id	age	Birth_year
PT102	54	1969
PT103	28	1995
PT104	36	1987
PT106	20	2003
PT107	44	1979
PT108	79	1944
PT109	42	1981
PT110	32	1991
PT111	53	1970
PT112	54	1969
PT113	78	1945
PT114	67	1956
PT115	76	1947
PT116	78	1945
PT117	15	2008
PT118	42	1981
PT119	42	1981

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

- select EmployeeName , patient_id, gender, blood_glucose_level,dense_rank () over (partition by gender order by blood_glucose_level) as glucose_level_rank from diabetes_prediction;

O/p:

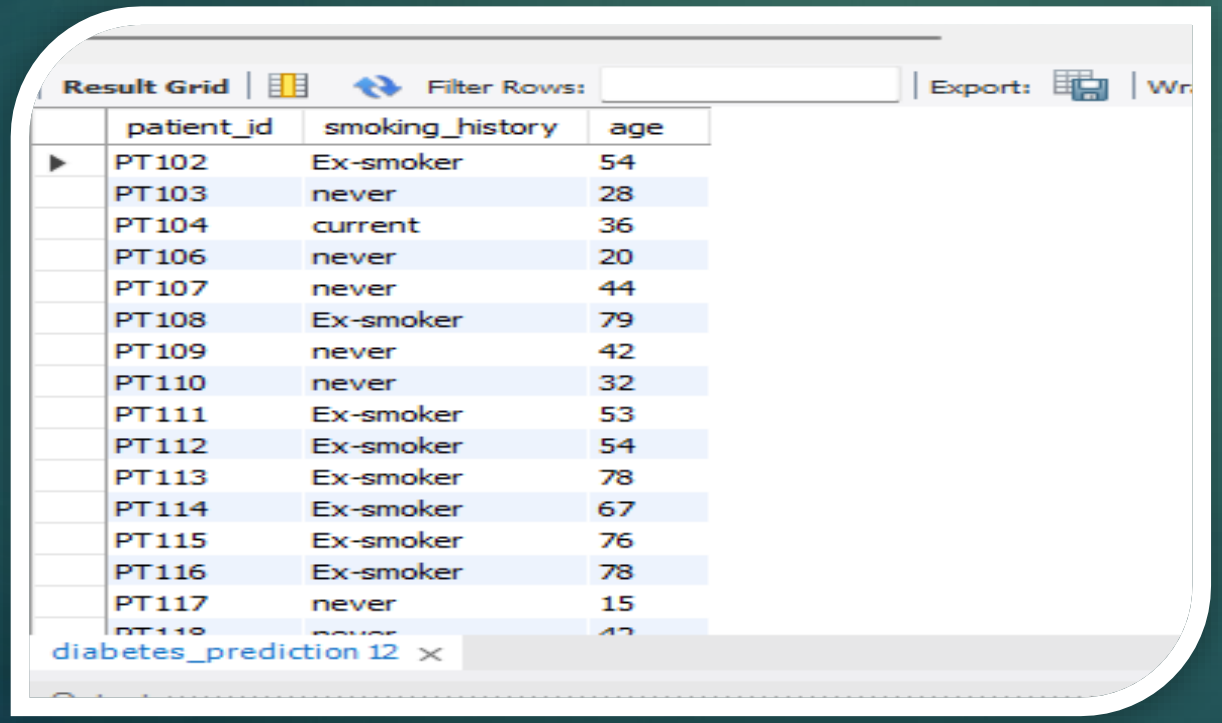


EmployeeName	patient_id	gender	blood_glucose_level	glucose_level_rank
GARY JIMENEZ	PT102	Female	80	1
TYLER VU	PT1460	Female	80	1
DANIEL COTTER	PT2987	Female	80	1
CHARLES ARMSTRONG	PT3089	Female	80	1
JEAN ALEXANDER	PT1381	Female	80	1
JIHYEON RIM	PT2729	Female	80	1
ANTONIO HERNANDEZ	PT1382	Female	80	1
TROY DANGERFIELD	PT897	Female	80	1
ALEC CARDENAS	PT896	Female	80	1
CHARLES HARDIMAN	PT889	Female	80	1
REXIE MEGIA	PT3076	Female	80	1
JOSEPH GARBAYO	PT1121	Female	80	1
MARLENA BYRNE	PT1595	Female	80	1
MICHAEL REDMOND	PT869	Female	80	1
JOHN CONWAY	PT2580	Female	80	1

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

- update internship.diabetes_prediction set smoking_history = 'Ex-smoker' where age > 50;
- select patient_id,smoking_history,age from internship.diabetes_prediction;

O/p:

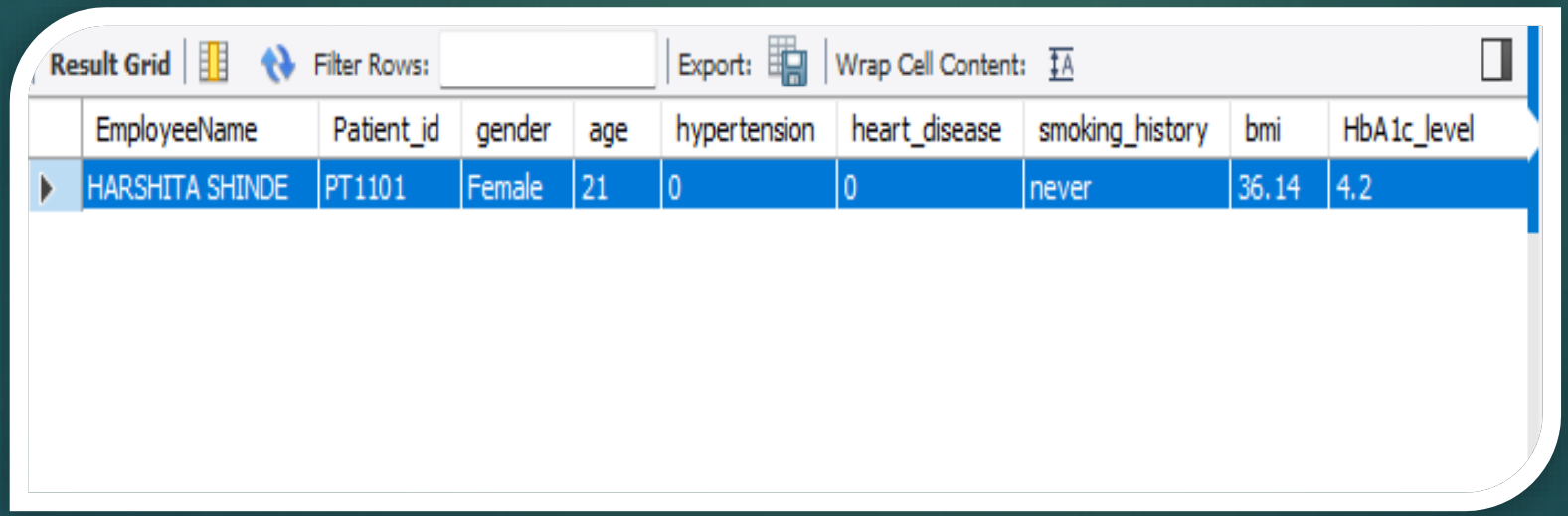


patient_id	smoking_history	age
PT102	Ex-smoker	54
PT103	never	28
PT104	current	36
PT106	never	20
PT107	never	44
PT108	Ex-smoker	79
PT109	never	42
PT110	never	32
PT111	Ex-smoker	53
PT112	Ex-smoker	54
PT113	Ex-smoker	78
PT114	Ex-smoker	67
PT115	Ex-smoker	76
PT116	Ex-smoker	78
PT117	never	15

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

- Insert into diabetes_prediction (employeename,patient_id, gender, age, hypertension, heart_disease,smoking_history,bmi, HbA1c_level,blood_glucose_level, diabetes) values ('HARSHITA SHINDE', 'PT1101', 'Female', '21', '0', 0, 'never', '36.14',4.2, 100, 0);

O/p:

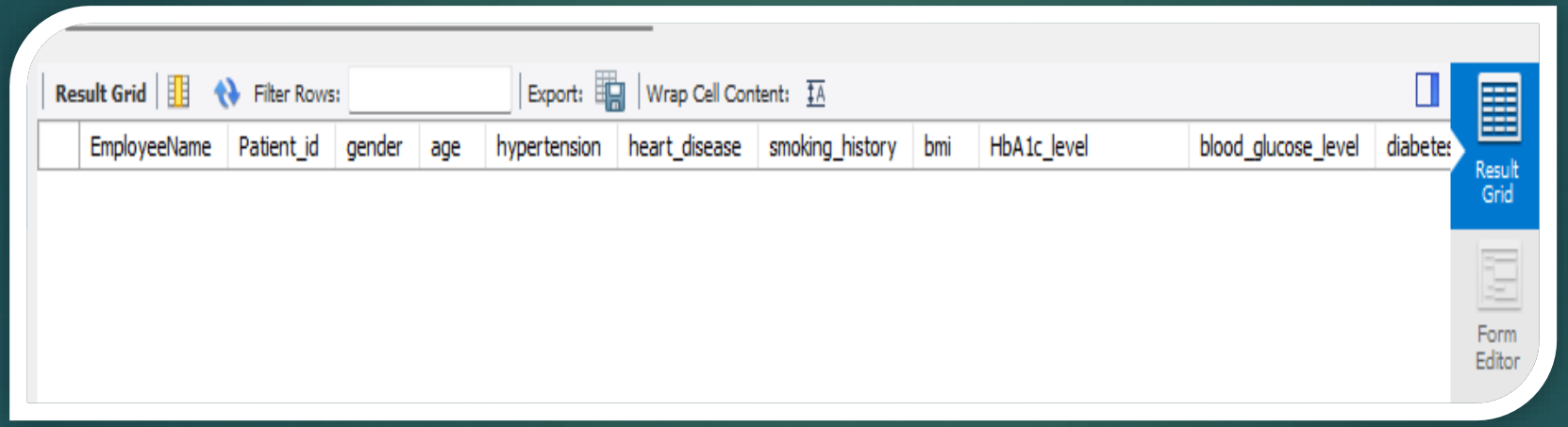


EmployeeName	Patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	HbA1c_level
HARSHITA SHINDE	PT1101	Female	21	0	0	never	36.14	4.2

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

- delete from diabetes_prediction where heart_disease = 1 ;
- select * from diabetes_prediction where heart_disease = 1 ;

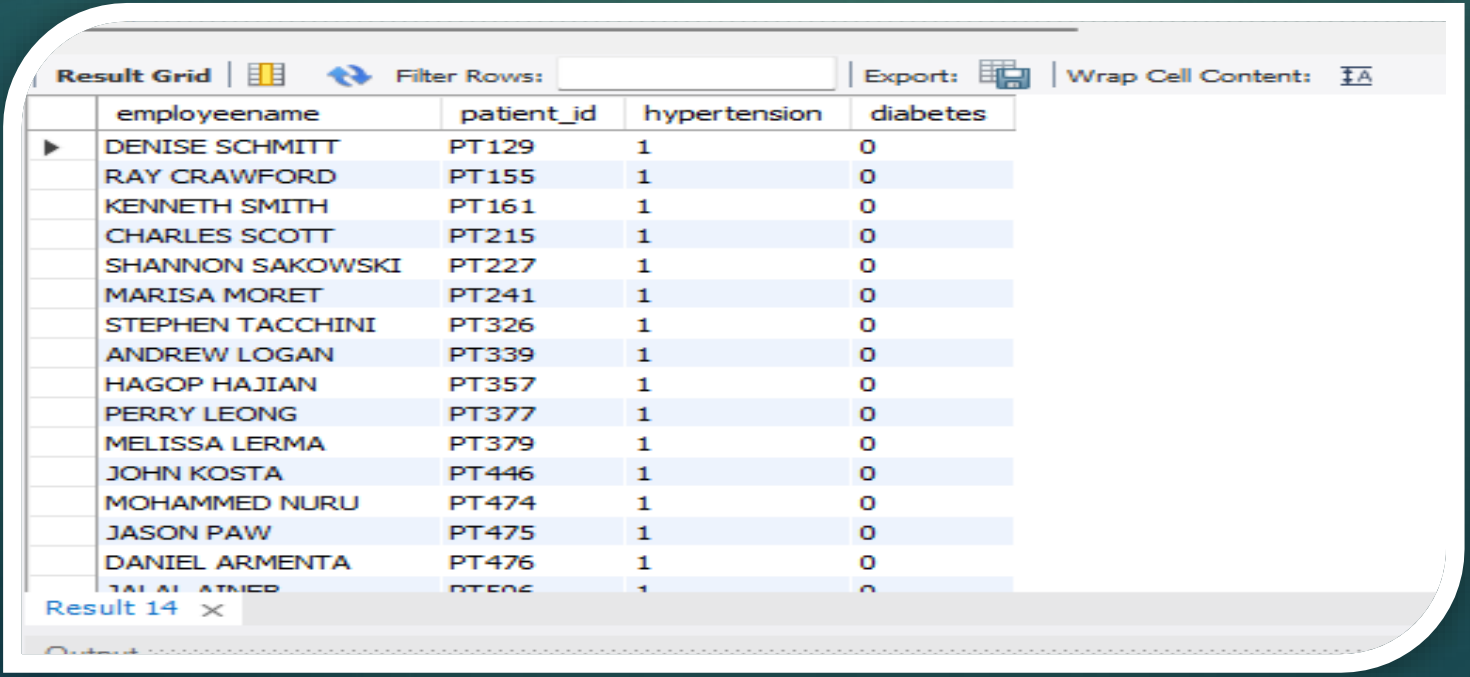
O/p:



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

- select employeename, patient_id,hypertension, diabetes from diabetes_prediction where hypertension = 1
except
select employeename, patient_id,hypertension, diabetes from diabetes_prediction where diabetes = 1;

O/p:



employeename	patient_id	hypertension	diabetes
DENISE SCHMITT	PT129	1	0
RAY CRAWFORD	PT155	1	0
KENNETH SMITH	PT161	1	0
CHARLES SCOTT	PT215	1	0
SHANNON SAKOWSKI	PT227	1	0
MARISA MORET	PT241	1	0
STEPHEN TACCHINI	PT326	1	0
ANDREW LOGAN	PT339	1	0
HAGOP HAJIAN	PT357	1	0
PERRY LEONG	PT377	1	0
MELISSA LERMA	PT379	1	0
JOHN KOSTA	PT446	1	0
MOHAMMED NURU	PT474	1	0
JASON PAW	PT475	1	0
DANIEL ARMENTA	PT476	1	0

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

- ALTER TABLE diabetes_prediction ADD CONSTRAINT unique_patient_id UNIQUE(patient_id);

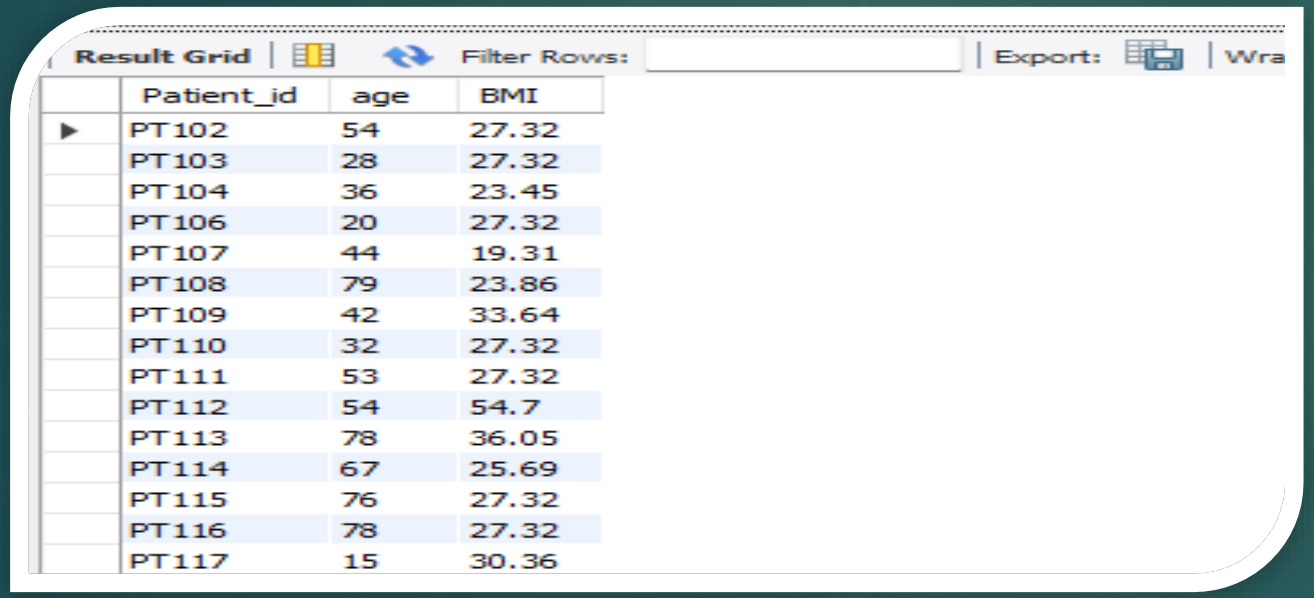
O/p:

22:32:13 ALTER TABLE diabetes_prediction ADD CONSTRAINT unique_patient_id UNIQUE(patient_id) 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 0.000 sec

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

- create view Patients as select Patient_id, age, BMI from diabetes_prediction ;
- select * from Patients;

O/p:



Patient_id	age	BMI
PT102	54	27.32
PT103	28	27.32
PT104	36	23.45
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
PT114	67	25.69
PT115	76	27.32
PT116	78	27.32
PT117	15	30.36

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

- Reducing data redundancy and improving data integrity are crucial aspects of efficient database schema.

Avoid Using Fields for Multiple Purposes: A column should have a single, clear purpose. Avoid using the same column for different types of data depending on the context.

Normalization: Analyze tables for normalization opportunities. It helps eliminate data redundancy by organizing data into separate tables and linking them through relationships.

Primary and Foreign Keys: Define primary keys for each table to ensure a unique identifier for each record . Establish foreign key relationships between tables to maintain referential integrity.

Data Types and Constraints: Choose appropriate data types for columns to optimize storage and retrieval. For eg. Apply constraints such as NOT NULL, UNIQUE, and CHECK to enforce data integrity at the database level.

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

- Optimizing the performance of sql queries on this dataset can involves:

Indexing: Identity columns frequently used in WHERE clauses or JOIN conditions and create indexes on those columns. It can significantly speed up data retrieval.

Regularly Update Statistics: Keep database statistics up-to-date to help the query optimizer generate efficient execution plans.

Partitioning Tables: If dataset is large, consider partitioning tables based on specific criteria (e.g., date ranges). This can improve query performance, especially for certain types of aggregations.

Query Optimization: Write efficient queries by minimizing the use of SELECT *, which retrieves all columns. Only select the columns you need. Use proper JOIN types (INNER JOIN, LEFT JOIN, etc.) based on the actual relationships between tables.

Avoiding SELECT DISTINCT: Use SELECT DISTINCT sparingly, as it can be resource-intensive. If possible, find alternative ways to achieve the desired result.