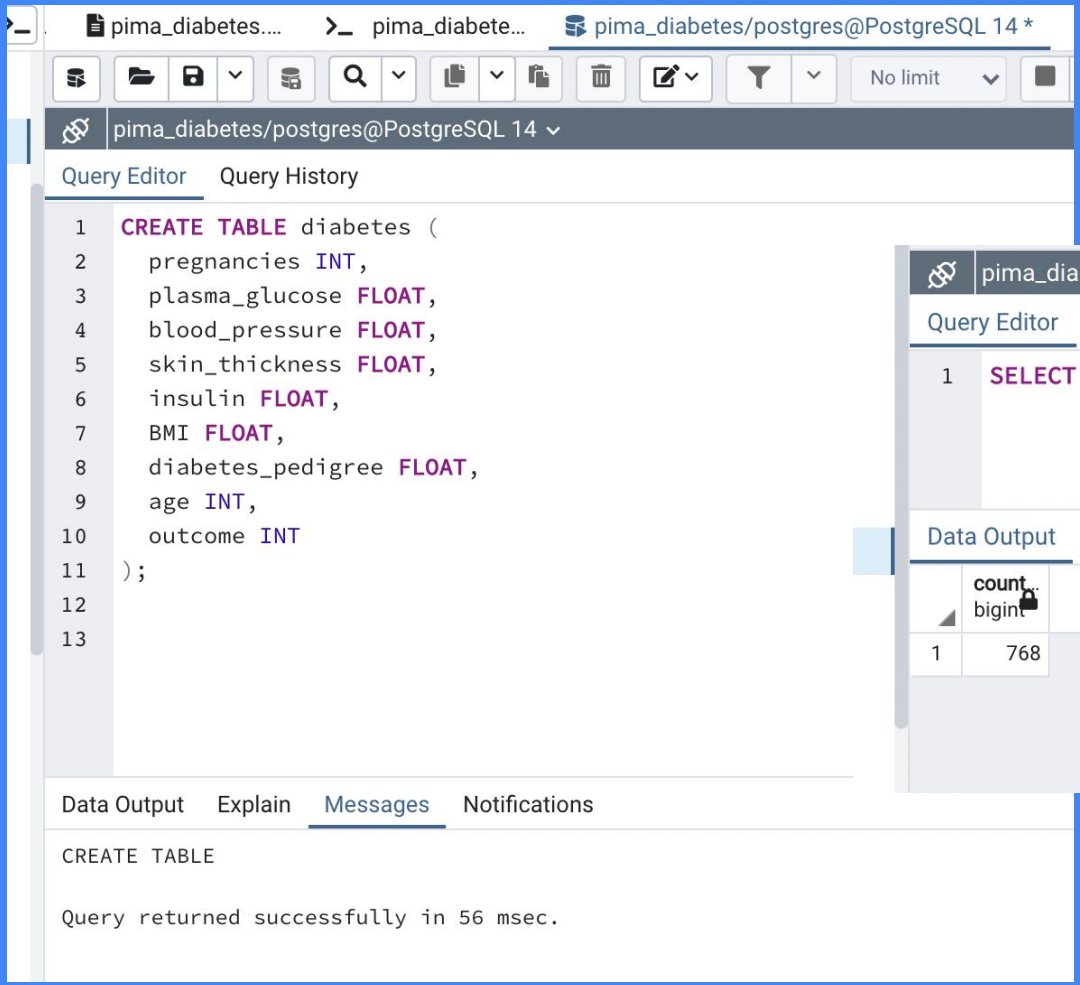
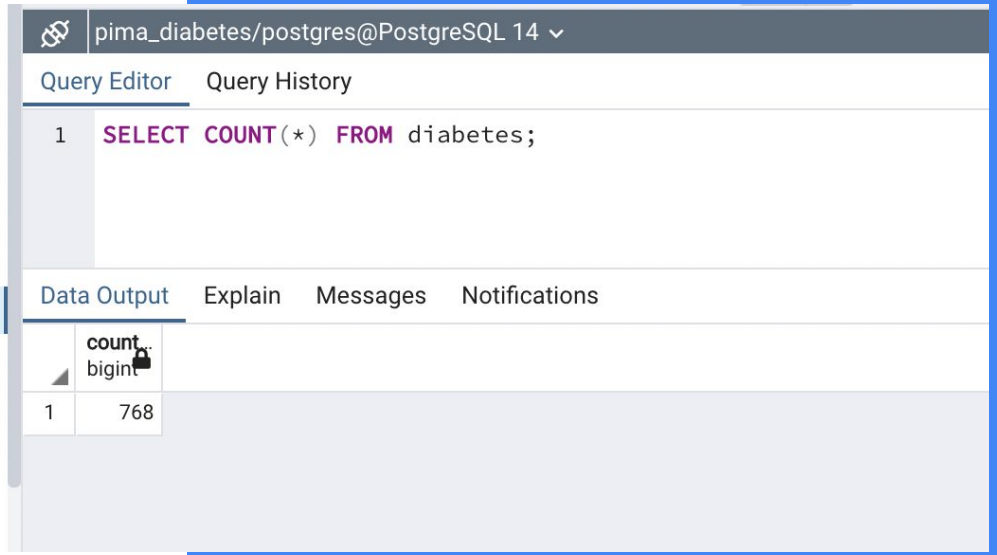
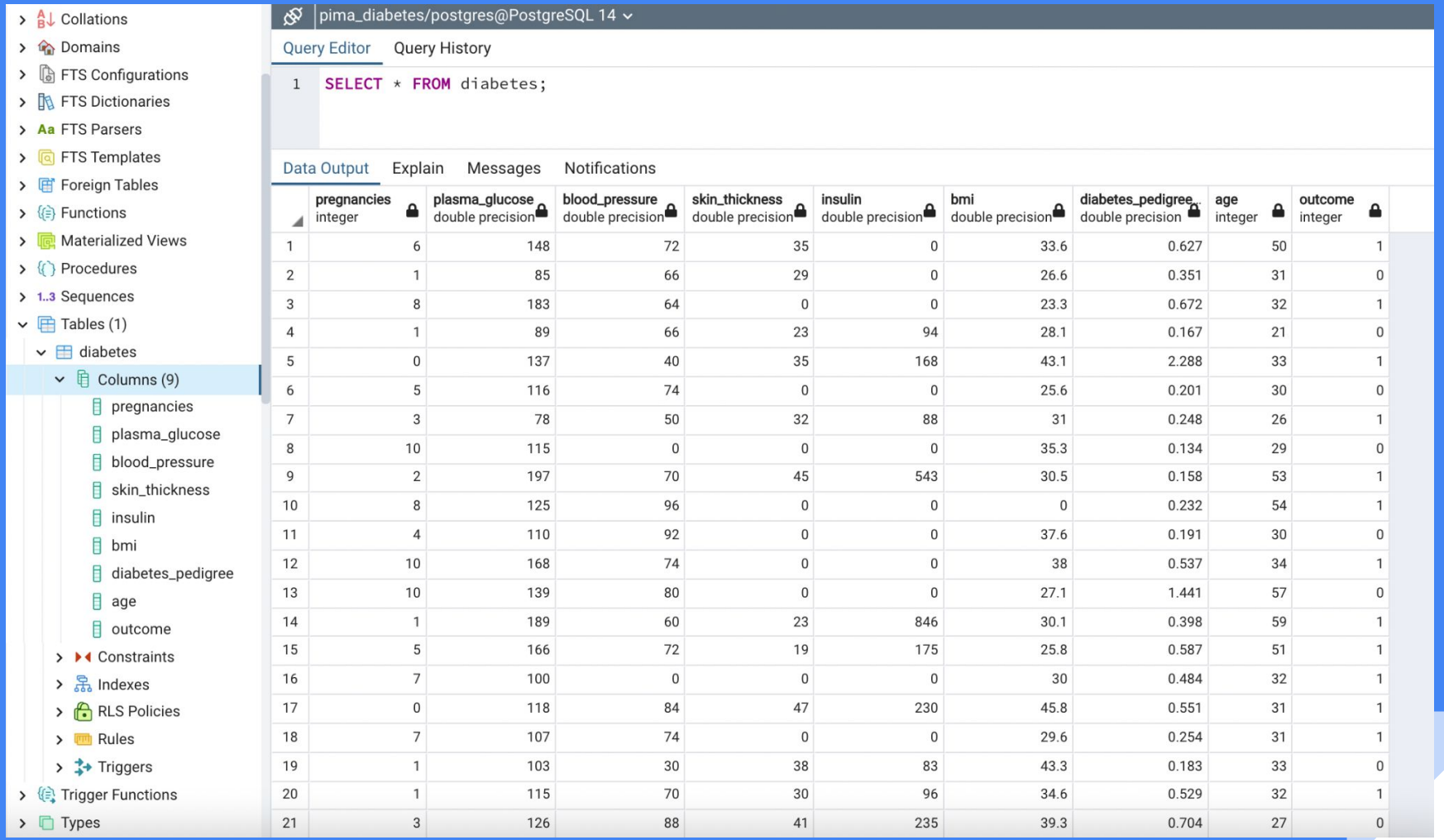
Medical Data Analysis using SQL

Diabetes Dataset

Diana Gekelman


pima_diabetes....
pima_diabete...
pima_diabetes/postgres@PostgreSQL 14 *
No limit
pima_diabetes/postgres@PostgreSQL 14
Query Editor
Query History
1 CREATE TABLE diabetes (
2 pregnancies INT,
3 plasma_glucose FLOAT,
4 blood_pressure FLOAT,
5 skin_thickness FLOAT,
6 insulin FLOAT,
7 BMI FLOAT,
8 diabetes_pedigree FLOAT,
9 age INT,
10 outcome INT
11);
12
13
Data Output
Explain
Messages
Notifications
CREATE TABLE
Query returned successfully in 56 msec.


pima_diabetes/postgres@PostgreSQL 14
Query Editor
Query History
1 SELECT COUNT(*) FROM diabetes;
Data Output
Explain
Messages
Notifications
count
bigint
1
768



	pregnancies integer	plasma_glucose double precision	blood_pressure double precision	skin_thickness double precision	insulin double precision	bmi double precision	diabetes_pedigree double precision	age integer	outcome integer
1	6	148	72	35	0	33.6	0.627	50	1
2	1	85	66	29	0	26.6	0.351	31	0
3	8	183	64	0	0	23.3	0.672	32	1
4	1	89	66	23	94	28.1	0.167	21	0
5	0	137	40	35	168	43.1	2.288	33	1
6	5	116	74	0	0	25.6	0.201	30	0
7	3	78	50	32	88	31	0.248	26	1
8	10	115	0	0	0	35.3	0.134	29	0
9	2	197	70	45	543	30.5	0.158	53	1
10	8	125	96	0	0	0	0.232	54	1
11	4	110	92	0	0	37.6	0.191	30	0
12	10	168	74	0	0	38	0.537	34	1
13	10	139	80	0	0	27.1	1.441	57	0
14	1	189	60	23	846	30.1	0.398	59	1
15	5	166	72	19	175	25.8	0.587	51	1
16	7	100	0	0	0	30	0.484	32	1
17	0	118	84	47	230	45.8	0.551	31	1
18	7	107	74	0	0	29.6	0.254	31	1
19	1	103	30	38	83	43.3	0.183	33	0
20	1	115	70	30	96	34.6	0.529	32	1
21	3	126	88	41	235	39.3	0.704	27	0

Attributes

1. Number pregnancies
2. Plasma glucose concentration
3. Diastolic blood pressure (mm Hg)
4. Triceps skinfold thickness (mm)
5. Insulin level (U/mL)
6. Body mass index (BMI) (kg/m^2)
7. Diabetes pedigree function
8. Age (years)
9. Outcome (1 for diabetes, 0 for no diabetes)

https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT age, plasma_glucose, CAST(ROUND(AVG(pregnancies)) AS INTEGER) AS average_pregnancies
2  FROM diabetes
3  GROUP BY age, plasma_glucose;
4
```

Data Output | Explain | Messages | Notifications

	age integer	plasma_glucose double precision	average_pregnancies integer
1	53	105	2
2	34	83	4
3	30	147	4
4	36	151	6
5	41	184	7
6	28	85	4
7	32	102	3
8	25	121	3
9	21	118	1
10	27	90	2
11	31	95	4
12	44	106	12
13	27	103	3
14	60	179	7
15	31	169	3
16	24	140	0
17	39	153	13
18	41	120	1
19	40	96	3
20	22	151	1
21	29	92	4
22	23	93	3

Summary Statistics

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT percentile_cont(0.5) WITHIN GROUP (ORDER BY plasma_glucose) AS median_glucose
2  FROM diabetes;
3
```

Data Output | Explain | Messages | Notifications

	median_glucose double precision
1	117

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT percentile_cont(0.5) WITHIN GROUP (ORDER BY insulin) AS median_insulin
2  FROM diabetes;
3
```

Data Output | Explain | Messages | Notifications

	median_insulin double precision
1	30.5

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT percentile_cont(0.25) WITHIN GROUP (ORDER BY bmi) AS bmi_q1_value,
2         percentile_cont(0.5) WITHIN GROUP (ORDER BY bmi) AS bmi_median_value,
3         percentile_cont(0.75) WITHIN GROUP (ORDER BY bmi) AS bmi_q3_value
4  FROM diabetes;
5
```

Data Output | Explain | Messages | Notifications

	bmi_q1_value double precision	bmi_median_value double precision	bmi_q3_value double precision
1	27.3	32	36.6

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```sql
SELECT age AS mode_value
FROM (
    SELECT age, COUNT(*) AS occurrence
    FROM diabetes
    GROUP BY age
    ORDER BY COUNT(*) DESC
    LIMIT 1
) AS subquery;
```

Data Output | Explain | Messages | Notifications

	mode_value integer
1	22

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1
2  SELECT ROUND(CAST(STDDEV(blood_pressure) AS NUMERIC), 2) AS blood_pressure_sd
3  FROM diabetes;
```

Data Output | Explain | Messages | Notifications

	blood_pressure_sd numeric
1	19.36

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT ROUND(CAST(STDDEV(diabetes_pedigree) AS NUMERIC), 4) AS diabetes_pedigree_sd
2  FROM diabetes;
3
```

Data Output | Explain | Messages | Notifications

	diabetes_pedigree_sd numeric
1	0.3313

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT CORR(plasma_glucose, age) AS correlation
2  FROM diabetes
3
4
```

Data Output | Explain | Messages | Notifications

	correlation double precision
1	0.2635143198243335

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT CORR(plasma_glucose, bmi) AS correlation
2  FROM diabetes
3
4
```

Data Output | Explain | Messages | Notifications

	correlation double precision
1	0.22107106945898297

pima_diabetes/postgres@PostgreSQL 14

Query Editor | Query History

```
1  SELECT age, plasma_glucose, outcome,
2         COUNT(*) AS total_count,
3         CAST(ROUND(AVG(pregnancies)) AS INT) AS average_pregnancies,
4         MIN(blood_pressure) AS min_blood_pressure,
5         MAX(blood_pressure) AS max_blood_pressure,
6         ROUND(CAST(AVG(bmi) AS NUMERIC), 2) AS average_bmi
7  FROM diabetes
8  GROUP BY age, plasma_glucose, outcome;
```

Data Output | Explain | Messages | Notifications

	age integer	plasma_glucose double precision	outcome integer	total_count bigint	average_pregnancies integer	min_blood_pressure double precision	max_blood_pressure double precision	average_bmi numeric
1	43	167	1	1	8	106	106	37.60
2	22	141	0	1	0	84	84	32.40
3	29	158	1	1	5	84	84	39.40
4	42	136	1	1	11	84	84	28.30
5	41	187	1	1	7	68	68	37.70
6	24	136	0	1	1	74	74	37.40
7	26	99	0	1	3	62	62	21.80
8	24	81	0	1	1	72	72	26.60
9	24	92	0	1	2	62	62	31.60
10	44	117	0	1	0	0	0	33.80
11	29	107	0	1	1	50	50	28.30
12	47	68	0	1	10	106	106	35.50
13	24	135	1	1	0	68	68	42.30
14	24	128	0	2	3	64	70	37.15