

# Medical Data History

Database Overview table rows

patients 4530, admissions 5067 ,doctors 27 ,province\_names 13

Schema: patients(patient\_id, first\_name, last\_name, gender, birth\_date, city, province\_id, allergies, height, weight) ↔ admissions(patient\_id, admission\_date, discharge\_date, diagnosis, doctor\_id) ↔ doctors(doctor\_id, first\_name, last\_name, specialty); patients.province\_id ↔ province\_names.province\_i

## Query 1 – Male Patients

### SQL:

```
SELECT first_name, last_name, gender FROM patients WHERE gender = 'M';
```



The screenshot shows a database interface with a 'Result Grid' tab. It displays the results of an SQL query. The grid has columns for 'first\_name', 'last\_name', and 'gender'. There are 12 rows of data, all with 'M' in the gender column. The first row is highlighted with a blue arrow icon on the left. At the bottom of the grid, there is a label 'patients 3' with a close button (X).

	first_name	last_name	gender
▶	Donald	Waterfield	M
	Mickey	Baasha	M
	Jiji	Sharma	M
	Blair	Diaz	M
	Charles	Wolfe	M
	Thomas	ONeill	M
	Sonny	Beckett	M
	Cedric	Coltrane	M
	Hank	Spencer	M
	Rick	Bennett	M
	Woody	Bashir	M

patients 3 X

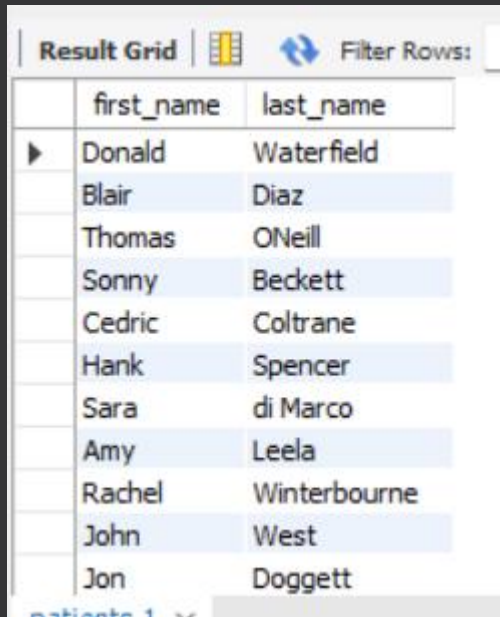
### Insights:

- Shows all patients where gender = 'M'.
- Useful for gender distribution analysis.
- 2468 row(s) returned

## Query 2 – Patients Without Allergies

SQL:

```
SELECT first_name, last_name FROM patients WHERE allergies IS NULL;
```



The screenshot shows a database interface with a 'Result Grid' tab. It displays a table with two columns: 'first\_name' and 'last\_name'. The table contains 12 rows of patient data. The interface includes a 'Filter Rows' button and a small icon of a grid. The table is titled 'patients\_1' at the bottom left.

	first_name	last_name
▶	Donald	Waterfield
	Blair	Diaz
	Thomas	ONeill
	Sonny	Beckett
	Cedric	Coltrane
	Hank	Spencer
	Sara	di Marco
	Amy	Leela
	Rachel	Winterbourne
	John	West
	Jon	Doggett

### Insights:

- Identifies patients with no recorded allergies.
- Useful for updating missing data.
- 2059 row(s) returned.

### Query 3 – First Names Starting with C

SQL:

```
SELECT first_name FROM patients WHERE first_name LIKE 'C%';
```



The screenshot shows a database interface with a 'Result Grid' tab selected. Above the grid, there are controls for 'Filter Rows' (with a search box), 'Export' (with a download icon), and 'Wrap Cell Content' (with a text icon). The grid itself contains 12 rows of data, each with 12 columns: patient\_id, first\_name, last\_name, gender, birth\_date, city, province\_id, allergies, height, weight, and birth\_year. The first names listed are Charles, Cedric, Charles, Cross, Calleigh, Catherine, Caroline, Casanova, Chen, and Charmian.

	patient_id	first_name	last_name	gender	birth_date	city	province_id	allergies	height	weight	birth_year
▶	5	Charles	Wolfe	M	2017-11-19	Orillia	ON	Penicillin	47	10	2017
	10	Cedric	Coltrane	M	1961-11-10	Toronto	ON	NKA	157	61	1961
	39	Charles	Andonuts	M	2016-05-20	Hamilton	ON	NKA	62	15	2016
	45	Cross	Gordon	M	2009-03-20	Ancaster	ON	NKA	125	53	2009
	63	Calleigh	Dean	F	2003-07-06	Hamilton	ON	Penicillin	175	56	2003
	67	Catherine	Minoru	F	1997-02-25	Victoria	BC	Sulphur	184	100	1997
	88	Caroline	Smart	F	1963-04-07	Hamilton	ON	NKA	145	64	1963
	103	Casanova	Dresden	M	1987-12-20	Milton	ON	Penicillin	150	74	1987
	117	Chen	Baasha	M	1992-05-09	Cambridge	ON	NKA	193	140	1992
	152	Charmian	Hardy	F	1992-04-16	Hamilton	ON	NKA	155	93	1992

#### Insights:

- Filtered patient list by first name pattern.
- Supports targeted searches.
- 302 row(s) returned.

**Query 4: Show first name and last name of patients that weigh within the range of 100 to 120 (inclusive)**

SQL:

```
select first_name,last_name from patients where weight between '100' and '120';
```

A screenshot of a database application's 'Result Grid'. The grid has two columns: 'first\_name' and 'last\_name'. It contains 11 rows of data. The first row is highlighted with a blue background. Above the grid, there are icons for 'Filter Rows' and a 'Filter Row' button.

	first_name	last_name
▶	Jiji	Sharma
	Blair	Diaz
	Thomas	ONeill
	Sonny	Beckett
	Tom	Halliwell
	Jon	Doggett
	Angel	Edwards
	John	Farley
	Temple	Russert
	Don	Edwards





Insights:

- In weight 100–120kg: 952

## Query 5 – Update NULL allergies to 'NKA'

SQL :

UPDATE patients SET allergies = 'NKA' WHERE allergies IS NULL ;

Result Grid    Filter Rows: <input type="text"/>   Export:  Wrap Cell Content:    Fetch rows: 											
	patient_id	first_name	last_name	gender	birth_date	city	province_id	allergies	height	weight	birth_year
▶	1	Donald	Waterfield	M	1963-02-12	Barrie	ON	NKA	156	65	1963
	2	Mickey	Baasha	M	1981-05-28	Dundas	ON	Sulfa	185	76	1981
	3	Jiji	Sharma	M	1957-09-05	Hamilton	ON	Penicillin	194	106	1957
	4	Blair	Diaz	M	1967-01-07	Hamilton	ON	NKA	191	104	1967
	5	Charles	Wolfe	M	2017-11-19	Orillia	ON	Penicillin	47	10	2017
	6	Sue	Falcon	F	2017-09-30	Ajax	ON	Penicillin	43	5	2017
	7	Thomas	ONeill	M	1993-01-31	Burlington	ON	NKA	180	117	1993
	8	Sonny	Beckett	M	1952-12-11	Port Hawkesbury	NS	NKA	174	105	1952
	9	Sister	Spitzer	F	1966-10-15	Toronto	ON	Penicillin	173	95	1966
	10	Cedric	Coltrane	M	1961-11-10	Toronto	ON	NKA	157	61	1961

patients 7 x

### Insights:

- Ensures missing allergy info defaults to “No Known Allergies”.
- Improves medical record completeness.

## Query 6 – Full name concatenation

```
SELECT CONCAT(first_name, ' ', last_name) AS full_name FROM patients;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid contains a single column labeled 'full\_name'. The data is as follows:

full_name
Donald Waterfield
Mickey Baasha
Jiji Sharma
Blair Diaz
Charles Wolfe
Sue Falcon
Thomas O'Neill
Sonny Beckett
Sister Spitzer
Cedric Coltrane

At the bottom of the grid, it says 'Result 8' with a close button (X).

### Insights:

- Creates unified full name for reporting.
- Helpful for patient list exports.

## Query 7 – Join with province names

```
SELECT p.first_name, p.last_name, pn.province_name FROM patients p JOIN province_names pn  
ON p.province_id = pn.province_id;
```



The screenshot shows a database interface with a 'Result Grid' tab. It displays the results of a SQL query that joins the 'patients' table with the 'province\_names' table. The grid has three columns: 'first\_name', 'last\_name', and 'province\_name'. There are 11 rows of data. The first 10 rows show patients from Ontario, and the 11th row shows a patient from Nova Scotia. The interface includes a 'Filter Rows' button and a 'Result 9' label at the bottom left.

first_name	last_name	province_name
Donald	Waterfield	Ontario
Mickey	Baasha	Ontario
Jiji	Sharma	Ontario
Blair	Diaz	Ontario
Charles	Wolfe	Ontario
Sue	Falcon	Ontario
Thomas	ONeill	Ontario
Sonny	Beckett	Nova Scotia
Sister	Spitzer	Ontario
Cedric	Coltrane	Ontario

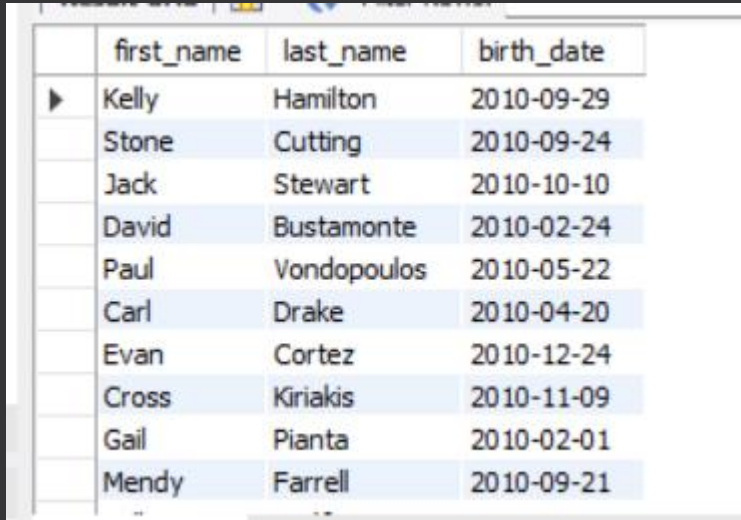
### Insights:

- Adds full province name to patient data.
- Improves readability for reports.



## Query 8 – Birth year is 2010

```
SELECT first_name, last_name, birth_date FROM patients WHERE YEAR(birth_date) = 2010;
```



A screenshot of a database query result window. The window displays a table with three columns: first\_name, last\_name, and birth\_date. The table contains 11 rows of data, representing patients born in 2010. The rows are: Kelly Hamilton (2010-09-29), Stone Cutting (2010-09-24), Jack Stewart (2010-10-10), David Bustamonte (2010-02-24), Paul Vondopoulos (2010-05-22), Carl Drake (2010-04-20), Evan Cortez (2010-12-24), Cross Kiriakis (2010-11-09), Gail Pianta (2010-02-01), and Mendy Farrell (2010-09-21). The first row is highlighted with a mouse cursor.

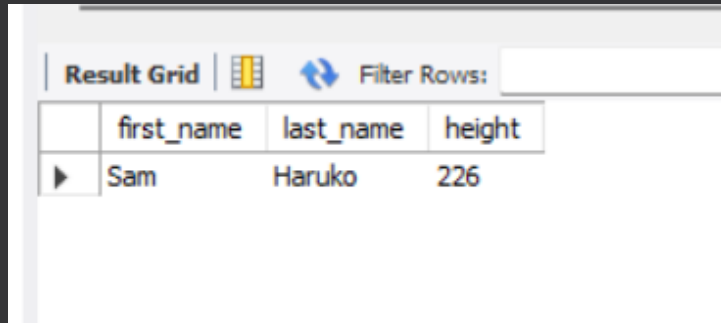
	first_name	last_name	birth_date
▶	Kelly	Hamilton	2010-09-29
	Stone	Cutting	2010-09-24
	Jack	Stewart	2010-10-10
	David	Bustamonte	2010-02-24
	Paul	Vondopoulos	2010-05-22
	Carl	Drake	2010-04-20
	Evan	Cortez	2010-12-24
	Cross	Kiriakis	2010-11-09
	Gail	Pianta	2010-02-01
	Mendy	Farrell	2010-09-21

### Insights:

- Filters patients born in 2010.
- Useful for age-specific health analysis
- 55 row(s) returned.

### Query 9 – Patient(s) with greatest height

SELECT first\_name, last\_name, height FROM patients WHERE height = (SELECT MAX(height) FROM patients);



The screenshot shows a database query result grid. At the top, there is a tab labeled 'Result Grid' and a 'Filter Rows' button. Below this, a table displays the results of the query. The table has three columns: 'first\_name', 'last\_name', and 'height'. A single row is visible, showing the patient 'Sam Haruko' with a height of 226.

	first_name	last_name	height
▶	Sam	Haruko	226

#### Insights:

- Identifies tallest patient(s).
- Could be used for anomaly detection or equipment planning.

## Query 10 – Specific patient IDs

```
SELECT * FROM patients WHERE patient_id IN (1, 45, 534, 879, 1000);
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

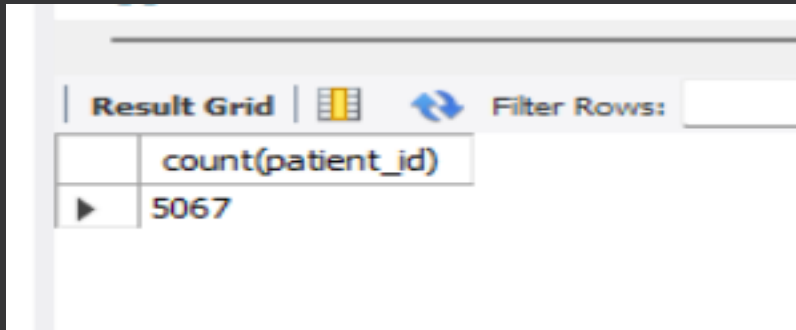
	patient_id	first_name	last_name	gender	birth_date	city	province_id	allergies	height	weight	birth_year
▶	1	Donald	Waterfield	M	1963-02-12	Barrie	ON	NKA	156	65	1963
	45	Cross	Gordon	M	2009-03-20	Ancaster	ON	NKA	125	53	2009
	534	Don	Zatara	M	2008-01-11	Timmins	ON	NKA	136	67	2008
	879	Orla	Shawn	F	1967-09-24	Sarnia	ON	Penicillin	149	65	1967
	1000	Rick	Williams	M	1975-04-13	Hamilton	ON	Penicillin	176	127	1975

### Insights:

- Retrieves detailed info for selected patients.
- Often used for manual checks or special cases.

### Query 11 – Total admissions count

```
SELECT COUNT(*) AS total_admissions FROM admissions;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid contains a single row with the column name 'count(patient\_id)' and the value '5067'. Above the grid, there are icons for a grid, a refresh button, and a 'Filter Rows' input field.

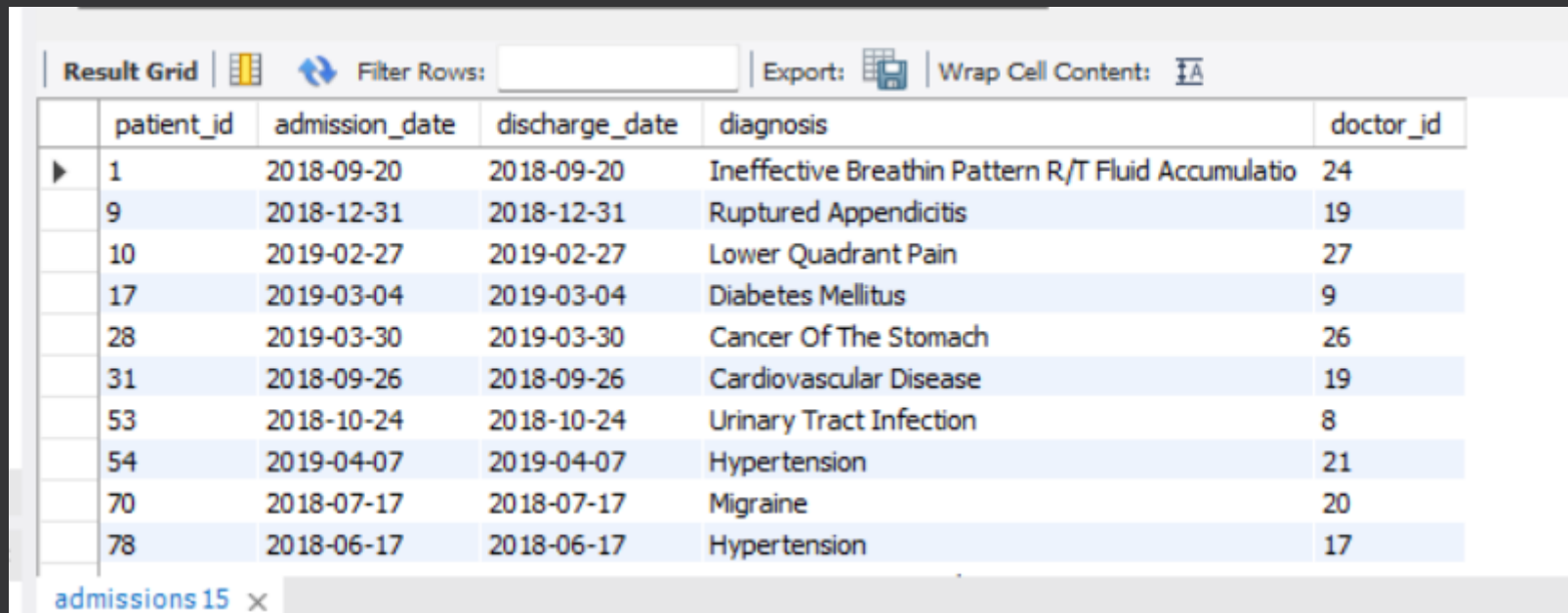
	count(patient_id)
▶	5067

#### Insights:

- Shows overall hospital workload volume.
- Can be benchmarked against capacity.

## Query 12 – Same-day admissions

```
SELECT * FROM admissions WHERE admission_date = discharge_date;
```



The screenshot shows a database query result grid with the following columns: patient\_id, admission\_date, discharge\_date, diagnosis, and doctor\_id. The grid displays 10 rows of data where the admission and discharge dates are the same. The interface includes a 'Result Grid' tab, a 'Filter Rows' input field, and 'Export' and 'Wrap Cell Content' buttons. A tab labeled 'admissions 15' is visible at the bottom left.

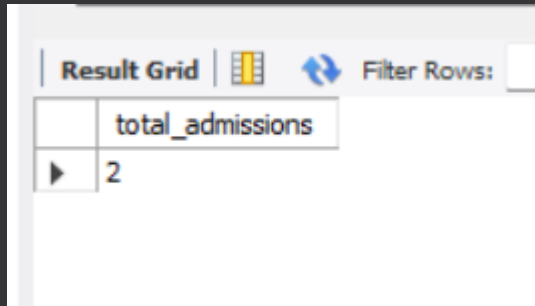
	patient_id	admission_date	discharge_date	diagnosis	doctor_id
▶	1	2018-09-20	2018-09-20	Ineffective Breathin Pattern R/T Fluid Accumulatio	24
	9	2018-12-31	2018-12-31	Ruptured Appendicitis	19
	10	2019-02-27	2019-02-27	Lower Quadrant Pain	27
	17	2019-03-04	2019-03-04	Diabetes Mellitus	9
	28	2019-03-30	2019-03-30	Cancer Of The Stomach	26
	31	2018-09-26	2018-09-26	Cardiovascular Disease	19
	53	2018-10-24	2018-10-24	Urinary Tract Infection	8
	54	2019-04-07	2019-04-07	Hypertension	21
	70	2018-07-17	2018-07-17	Migraine	20
	78	2018-06-17	2018-06-17	Hypertension	17

### Insights:

- Finds cases discharged the same day.
- Useful for outpatient vs inpatient analysis.
- 481 row(s) returned.

### Query 13 – Admissions for patient 579

```
SELECT COUNT(*) AS total_admissions FROM admissions WHERE patient_id = 579;
```



The screenshot shows a database query result grid. At the top, there is a tab labeled 'Result Grid' with a grid icon, a refresh icon, and a 'Filter Rows:' input field. Below this, the query result is displayed in a table with one column named 'total\_admissions' and one row containing the value '2'.

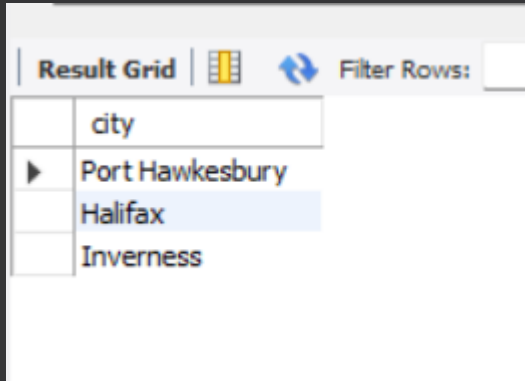
	total_admissions
▶	2

#### Insights:

- Measures one patient's admission frequency.

## Query 14 – Unique NS cities

```
SELECT DISTINCT city FROM patients WHERE province_id = 'NS';
```



The screenshot shows a database query result grid. At the top, there is a tab labeled 'Result Grid' with a grid icon, a refresh icon, and a 'Filter Rows:' input field. Below this, a table displays the results of the query. The table has one column labeled 'city'. The data rows are 'Port Hawkesbury', 'Halifax', and 'Inverness'. The 'Halifax' row is currently selected, highlighted in blue.

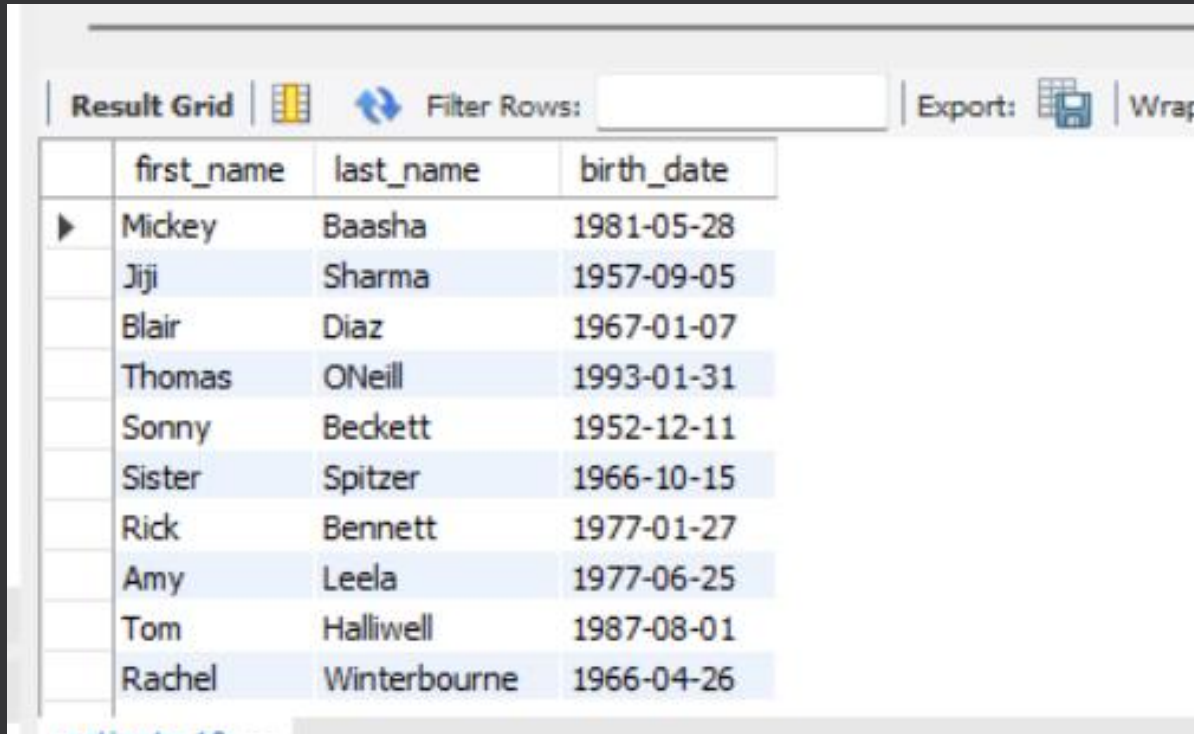
city
Port Hawkesbury
Halifax
Inverness

### Insights:

- Shows geographical spread in Nova Scotia.
- Useful for regional resource allocation.

## Query 15 – Height >160cm & weight >70kg

SELECT first\_name, last\_name, birth\_date FROM patients WHERE height > 160 AND weight > 70;



The screenshot shows a database interface with a 'Result Grid' tab. Above the grid, there are icons for a grid, a refresh button, a 'Filter Rows' input field, an 'Export' button with a download icon, and a 'Wrap' button. The grid itself contains a table with four columns: 'first\_name', 'last\_name', and 'birth\_date'. There are 12 rows of data, each with a small blue triangle icon to its left. The data is as follows:

	first_name	last_name	birth_date
▶	Mickey	Baasha	1981-05-28
	Jiji	Sharma	1957-09-05
	Blair	Diaz	1967-01-07
	Thomas	ONeill	1993-01-31
	Sonny	Beckett	1952-12-11
	Sister	Spitzer	1966-10-15
	Rick	Bennett	1977-01-27
	Amy	Leela	1977-06-25
	Tom	Halliwel	1987-08-01
	Rachel	Winterbourne	1966-04-26

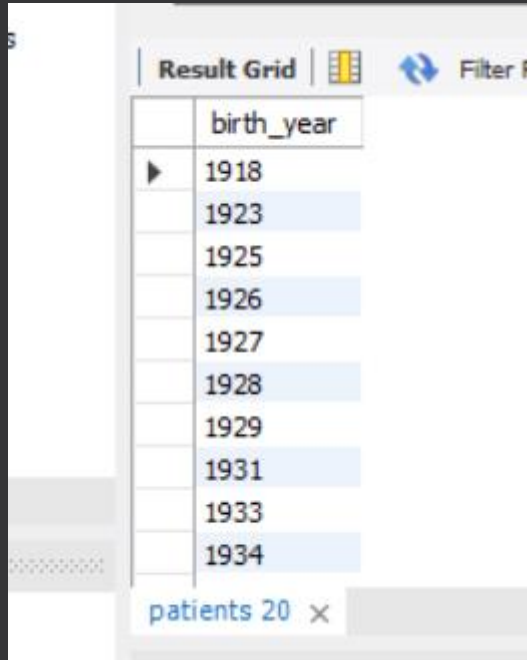
### Insights:

- Identifies heavier/taller patients.
- Can help target specific health programs.
- 2091 row(s) returned.



## Query 16 – Unique birth years

```
SELECT DISTINCT YEAR(birth_date) AS birth_year FROM patients ORDER BY birth_year ASC;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid displays the results of the query, showing unique birth years from the 'patients' table, ordered ascending. The first row is the header 'birth\_year'. The data rows are: 1918, 1923, 1925, 1926, 1927, 1928, 1929, 1931, 1933, and 1934. The years 1923, 1926, 1928, 1931, and 1934 are highlighted in blue. At the bottom of the grid, there is a status bar that says 'patients 20' with a close button (X).

birth_year
1918
1923
1925
1926
1927
1928
1929
1931
1933
1934

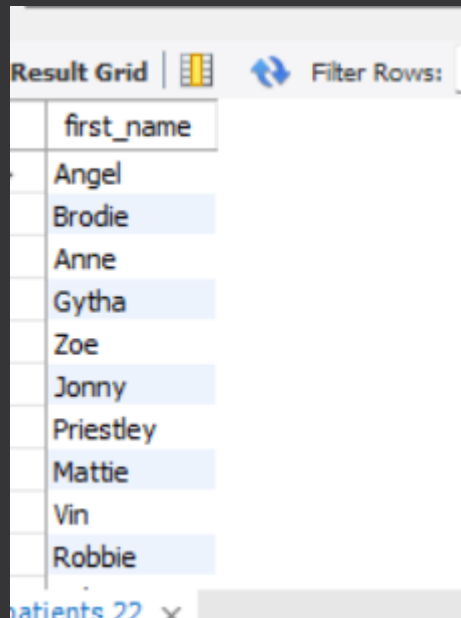
patients 20 X

### Insights:

- Gives range of birth years in dataset.
- Useful for demographic profiling.

### Query 17 – Unique first names (occur once)

```
SELECT first_name FROM patients GROUP BY first_name HAVING COUNT(*) = 1 ;
```



The screenshot shows a 'Result Grid' window with a 'Filter Rows' button. The grid displays a single column titled 'first\_name' with the following values: Angel, Brodie, Anne, Gytha, Zoe, Jonny, Priestley, Mattie, Vin, and Robbie. The rows are alternatingly highlighted in white and light blue. At the bottom left, a tab labeled 'patients.22' is visible.

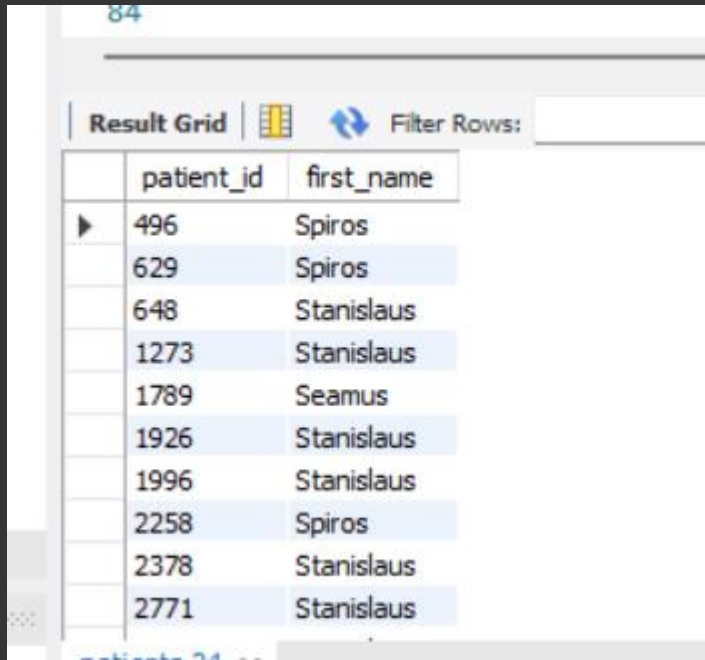
first_name
Angel
Brodie
Anne
Gytha
Zoe
Jonny
Priestley
Mattie
Vin
Robbie

#### Insights:

- Finds rare names in the database.
- Could be used for identity verification.

## Query 18 – First name starts/ends with S, length ≥6

SELECT patient\_id, first\_name FROM patients WHERE first\_name LIKE 'S%s' AND LENGTH(first\_name) >= 6;



The screenshot shows a database query result grid with two columns: patient\_id and first\_name. The grid displays 11 rows of data, all of which have a first name starting with 'S' and a length of 6 or more characters. The rows are: 496 Spiros, 629 Spiros, 648 Stanislaus, 1273 Stanislaus, 1789 Seamus, 1926 Stanislaus, 1996 Stanislaus, 2258 Spiros, 2378 Stanislaus, and 2771 Stanislaus. The grid is titled 'Result Grid' and has a 'Filter Rows' button.

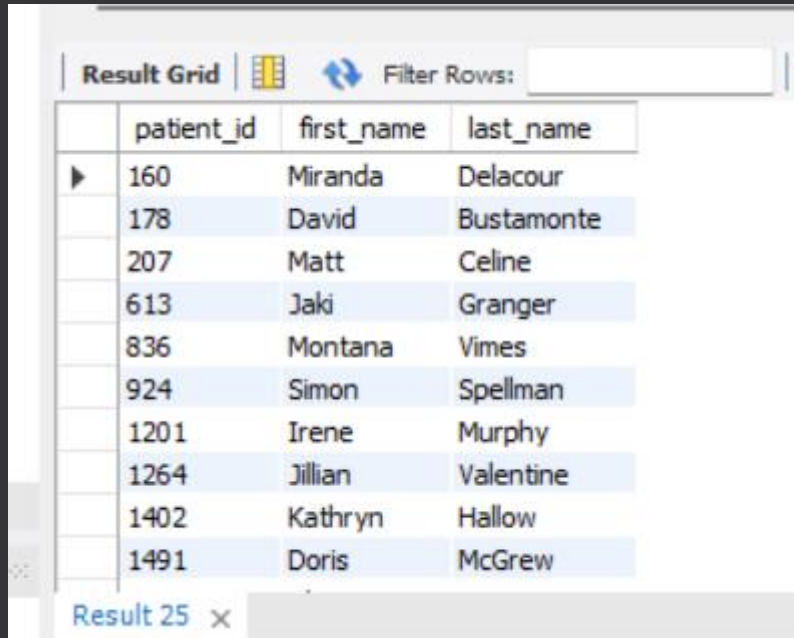
patient_id	first_name
496	Spiros
629	Spiros
648	Stanislaus
1273	Stanislaus
1789	Seamus
1926	Stanislaus
1996	Stanislaus
2258	Spiros
2378	Stanislaus
2771	Stanislaus

### Insights:

- Very specific name pattern search.
- Example of using multiple string filters.
- 11 row(s) returned.

## Query 19 – Dementia patients

```
SELECT p.patient_id, p.first_name, p.last_name FROM patients  
p JOIN admissions a ON p.patient_id = a.patient_id WHERE a.diagnosis = 'Dementia';
```



The screenshot shows a database interface with a 'Result Grid' tab. It displays 12 rows of data from a query. The columns are 'patient\_id', 'first\_name', and 'last\_name'. The rows are numbered 1 through 12 on the left. At the bottom, it says 'Result 25' with a close button.

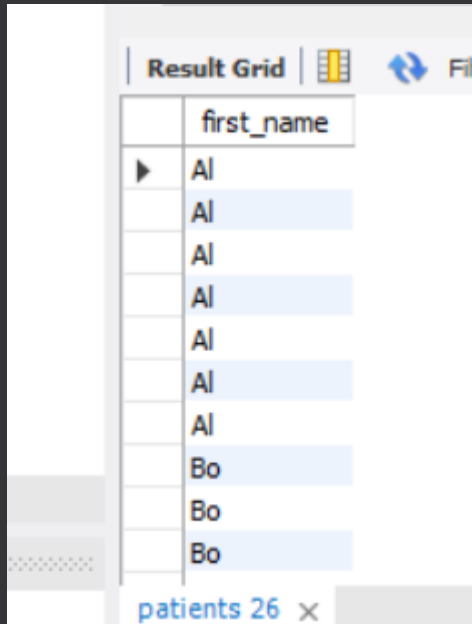
	patient_id	first_name	last_name
▶	160	Miranda	Delacour
	178	David	Bustamonte
	207	Matt	Celine
	613	Jaki	Granger
	836	Montana	Vimes
	924	Simon	Spellman
	1201	Irene	Murphy
	1264	Jillian	Valentine
	1402	Kathryn	Hallow
	1491	Doris	McGrew

### Insights:

- Identifies patients with dementia diagnoses.
- Critical for tracking chronic illness cases.
- 26 row(s) returned.

## Query 20 – Order first names by length then alphabet

```
SELECT first_name FROM patients ORDER BY LENGTH(first_name) asc ,  
first_name;
```



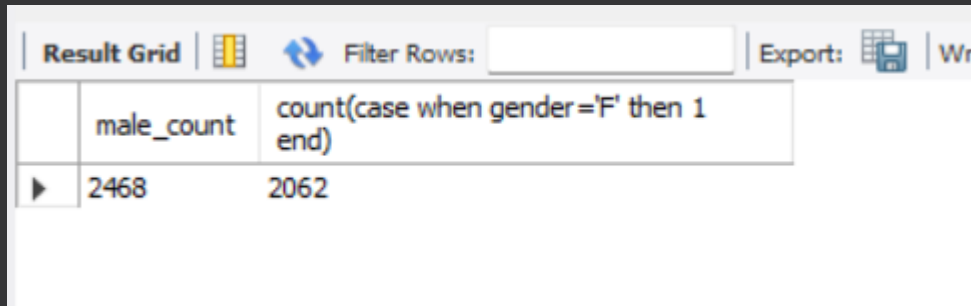
first_name
Al
Al
Al
Al
Al
Al
Al
Al
Al
Bo
Bo
Bo

### Insights:

- Sorts names first by length, then alphabetically.
- Helpful for formatting or name length studies.

### Query 21 – Male/Female totals in one row

```
SELECT SUM(CASE WHEN gender = 'M' THEN 1 ELSE 0 END) AS male_count,  
SUM(CASE WHEN gender = 'F' THEN 1 ELSE 0 END) AS female_count FROM patients;
```



The screenshot shows a database query result grid. At the top, there is a toolbar with 'Result Grid', 'Filter Rows', and 'Export' buttons. The grid has two columns: 'male\_count' and 'count(case when gender='F' then 1 end)'. The first row shows the values 2468 and 2062 respectively.

	male_count	count(case when gender='F' then 1 end)
▶	2468	2062

#### Insights:

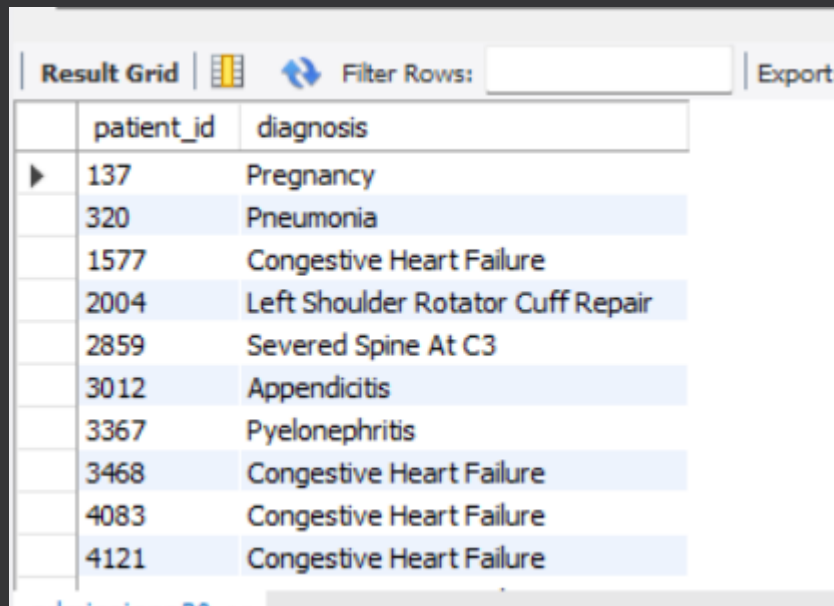
- Gives gender distribution at a glance.
- Useful for demographic reporting.

## Query 22 – Male/Female totals (duplicate)

*(Same SQL and insights as Query 21)*

## Query 23 – Multiple admissions same diagnosis

SELECT patient\_id, diagnosis FROM admissions GROUP BY patient\_id, diagnosis HAVING COUNT(\*) > 1;



The screenshot shows a database query result grid with two columns: patient\_id and diagnosis. The grid contains 11 rows of data. The first row is highlighted with a blue background. The interface includes a 'Result Grid' tab, a 'Filter Rows' input field, and an 'Export' button.

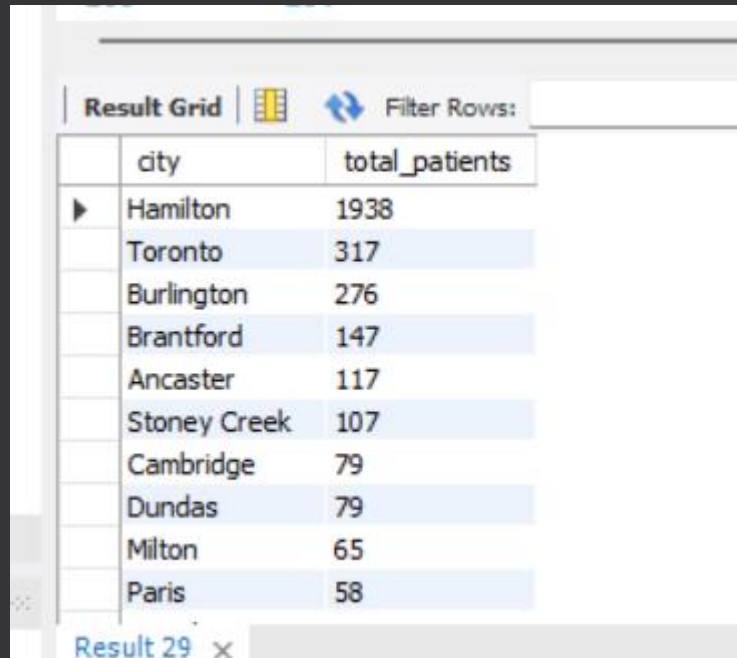
patient_id	diagnosis
137	Pregnancy
320	Pneumonia
1577	Congestive Heart Failure
2004	Left Shoulder Rotator Cuff Repair
2859	Severed Spine At C3
3012	Appendicitis
3367	Pyelonephritis
3468	Congestive Heart Failure
4083	Congestive Heart Failure
4121	Congestive Heart Failure

### Insights:

- Shows recurring diagnosis per patient.
- Helps identify chronic or recurring illnesses.

## Query 24 – City and total patients

```
SELECT city, COUNT(*) AS patient_count FROM patients  
GROUP BY city ORDER BY patient_count DESC, city ASC;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid displays the results of a SQL query, ordered by patient count in descending order. The columns are 'city' and 'total\_patients'. The data is as follows:

city	total_patients
Hamilton	1938
Toronto	317
Burlington	276
Brantford	147
Ancaster	117
Stoney Creek	107
Cambridge	79
Dundas	79
Milton	65
Paris	58

At the bottom of the window, it says 'Result 29' with a close button.

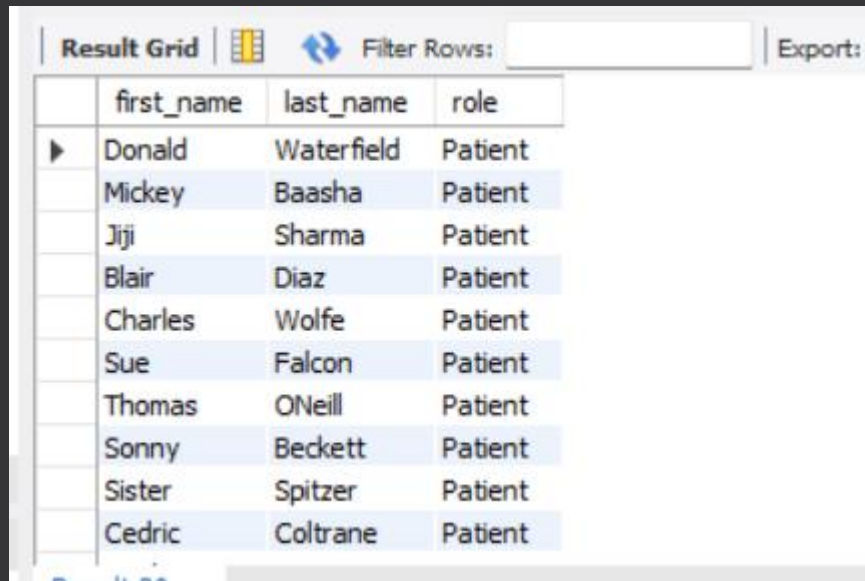
### Insights:

- Shows patient distribution by city.
- Highlights high-demand locations.



## Query 25 – Patients and doctors list with role

```
SELECT first_name, last_name, 'Patient' AS role FROM patients UNION  
SELECT first_name, last_name, 'Doctor' AS role FROM doctors;
```



The screenshot shows a database query result grid. At the top, there is a toolbar with a 'Result Grid' tab, a 'Filter Rows' button, and an 'Export' button. Below the toolbar is a table with four columns: 'first\_name', 'last\_name', and 'role'. The table contains 12 rows of data, all of which are patients. The first name of the first row is truncated in the image.

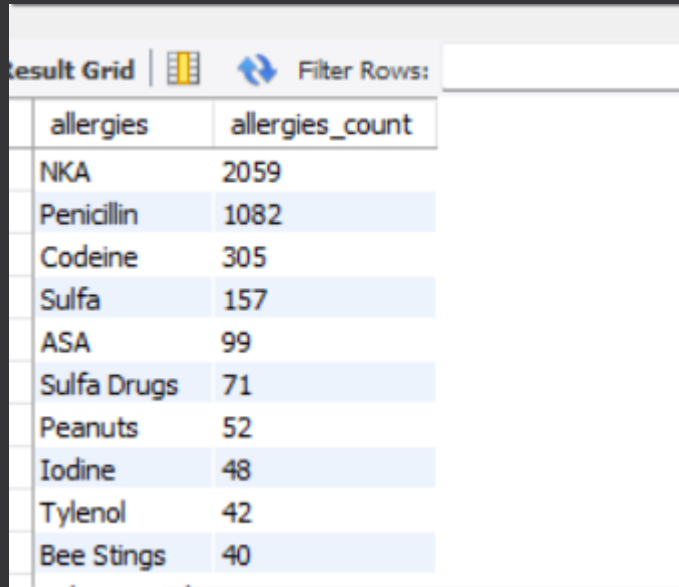
	first_name	last_name	role
▶	Donald	Waterfield	Patient
	Mickey	Baasha	Patient
	Jiji	Sharma	Patient
	Blair	Diaz	Patient
	Charles	Wolfe	Patient
	Sue	Falcon	Patient
	Thomas	ONeill	Patient
	Sonny	Beckett	Patient
	Sister	Spitzer	Patient
	Cedric	Coltrane	Patient

### Insights:

- Combines both groups into one list.
- Useful for all-personnel directories.
- 4538 row(s) returned.

## Query 26 – Allergies by popularity

```
SELECT allergies, COUNT(*) AS count FROM patients WHERE  
allergies IS NOT NULL GROUP BY allergies ORDER BY count DESC, allergies ASC;
```



The screenshot shows a database query result grid. At the top, there is a tab labeled 'Result Grid' and a 'Filter Rows' button. Below this, a table displays the results of the query. The table has two columns: 'allergies' and 'allergies\_count'. The data is sorted by 'allergies\_count' in descending order. The rows are as follows:

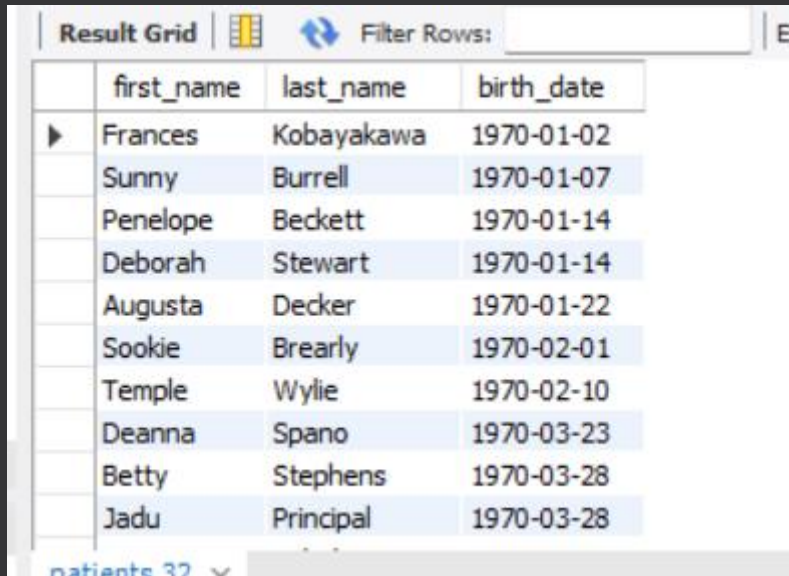
allergies	allergies_count
NKA	2059
Penicillin	1082
Codeine	305
Sulfa	157
ASA	99
Sulfa Drugs	71
Peanuts	52
Iodine	48
Tylenol	42
Bee Stings	40

### Insights:

- Ranks most common allergies.
- Guides allergy-related precautions.

## Query 27 – Born in 1970s decade

```
SELECT first_name, last_name, birth_date FROM patients  
WHERE birth_date BETWEEN '1970-01-01' AND '1979-12-31' ORDER BY birth_date ASC;
```



The screenshot shows a database interface with a 'Result Grid' tab. It displays a table with three columns: 'first\_name', 'last\_name', and 'birth\_date'. The data is sorted by birth date in ascending order. The first 11 rows are visible, showing patients born between January 2, 1970, and March 28, 1970. The interface includes a 'Filter Rows' button and a search bar.

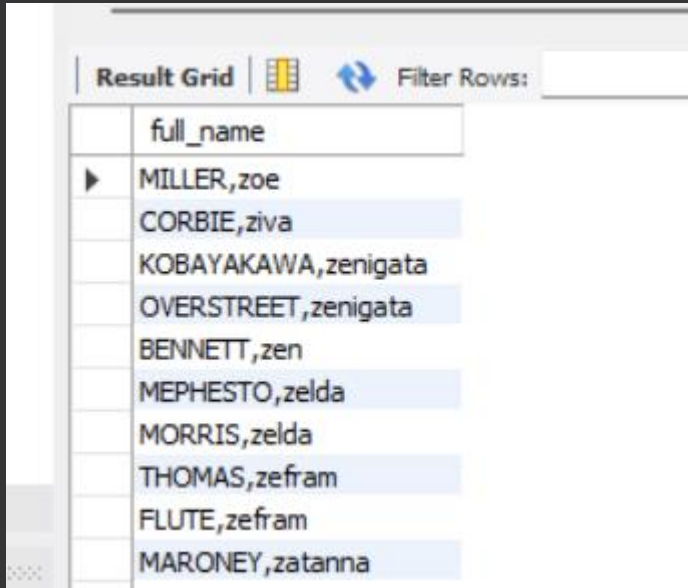
	first_name	last_name	birth_date
▶	Frances	Kobayakawa	1970-01-02
	Sunny	Burrell	1970-01-07
	Penelope	Beckett	1970-01-14
	Deborah	Stewart	1970-01-14
	Augusta	Decker	1970-01-22
	Sookie	Brearly	1970-02-01
	Temple	Wylie	1970-02-10
	Deanna	Spano	1970-03-23
	Betty	Stephens	1970-03-28
	Jadu	Principal	1970-03-28

### Insights:

- Filters for a specific decade of births.
- Useful for generation-based studies.
- 623 row(s) returned.

## Query 28 – Full name format LAST,first

```
SELECT CONCAT(UPPER(last_name), ', ', LOWER(first_name)) AS  
full_name FROM patients ORDER BY first_name DESC;
```



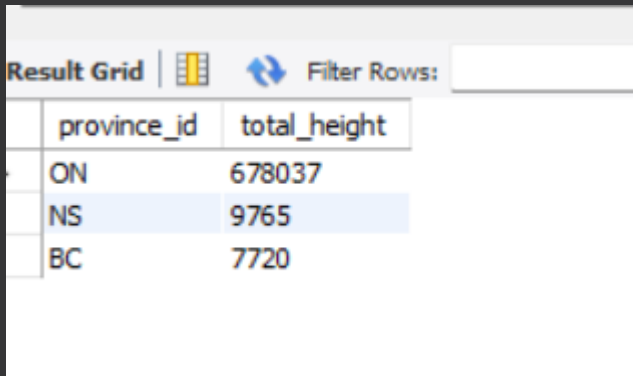
full_name
MILLER,zoe
CORBIE,ziva
KOBAYAKAWA,zenigata
OVERSTREET,zenigata
BENNETT,zen
MEPHESTO,zelda
MORRIS,zelda
THOMAS,zefram
FLUTE,zefram
MARONEY,zatanna

### Insights:

- Formats names in a consistent style.
- Useful for directories or mailing lists.

### Query 29 – Province height total $\geq 7000$

```
SELECT province_id, SUM(height) AS total_height FROM patients GROUP BY  
province_id HAVING SUM(height) >= 7000 ;
```



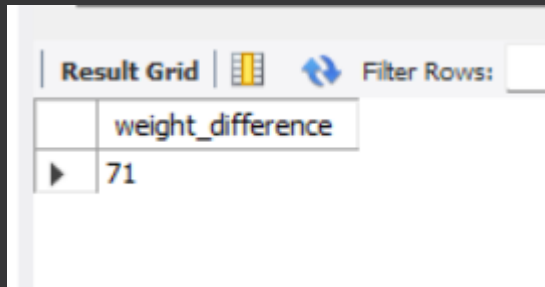
province_id	total_height
ON	678037
NS	9765
BC	7720

#### Insights:

- Sums height per province.
- Arbitrary threshold used for filtering.

### Query 30 – Weight range for 'Maroni'

```
SELECT MAX(weight) - MIN(weight) AS weight_diff FROM patients WHERE last_name = 'Maroni';
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid has one column labeled 'weight\_difference' and one row with the value '71'. There are icons for 'Filter Rows' and a 'Filter Rows' input field.

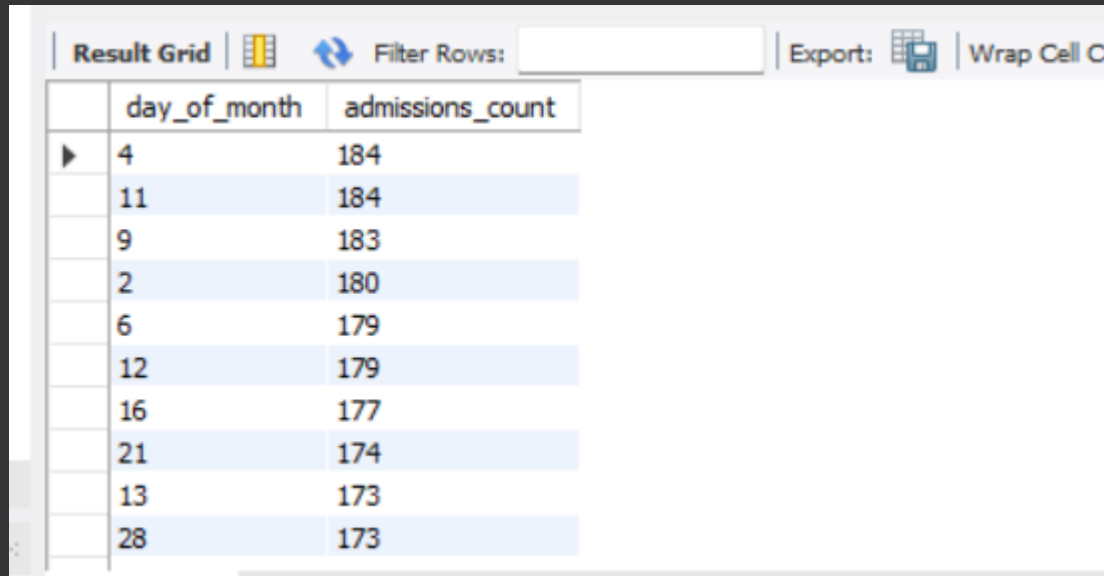
	weight_difference
▶	71

#### Insights:

- Measures variation in weight for a family name.
- Useful for identifying inconsistencies.

### Query 31 – Admissions per day of month

```
SELECT DAY(admission_date) AS day_of_month, COUNT(*) AS admissions_count FROM  
admissions GROUP BY day_of_month ORDER BY admissions_count DESC, day_of_month ASC;
```



The screenshot shows a database query result grid with two columns: 'day\_of\_month' and 'admissions\_count'. The results are ordered by 'admissions\_count' in descending order, and then by 'day\_of\_month' in ascending order. The data shows that the 4th and 11th of the month have the highest number of admissions (184 each), followed by the 9th (183), 2nd (180), 6th (179), 12th (179), 16th (177), 21st (174), 13th (173), and 28th (173).

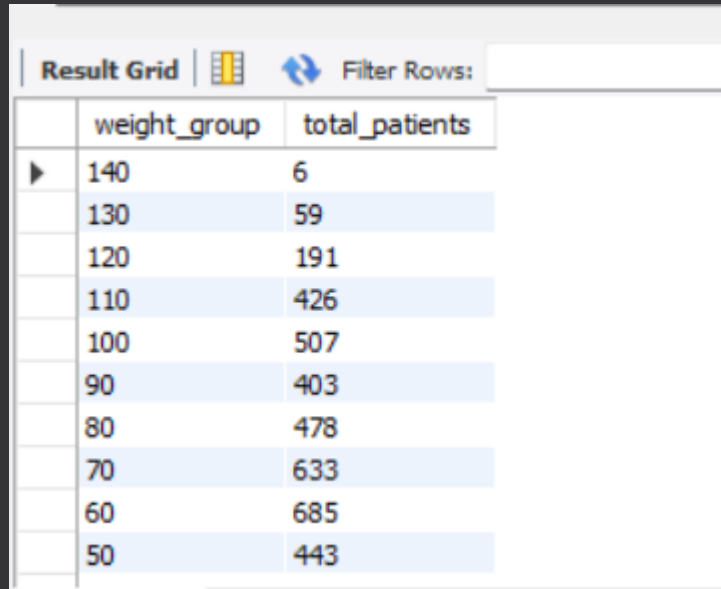
day_of_month	admissions_count
4	184
11	184
9	183
2	180
6	179
12	179
16	177
21	174
13	173
28	173

#### Insights:

- Shows daily admission patterns.
- Can help spot peak days.

### Query 32 – Weight groups of 10kg

```
SELECT FLOOR(weight / 10) * 10 AS weight_group, COUNT(*) AS total_patients FROM  
patients GROUP BY weight_group ORDER BY weight_group DESC;
```



The screenshot shows a SQL query result grid with two columns: 'weight\_group' and 'total\_patients'. The data is ordered by weight\_group in descending order. The weight groups range from 140 down to 50 in increments of 10. The total number of patients for each group is listed in the second column.

	weight_group	total_patients
▶	140	6
	130	59
	120	191
	110	426
	100	507
	90	403
	80	478
	70	633
	60	685
	50	443

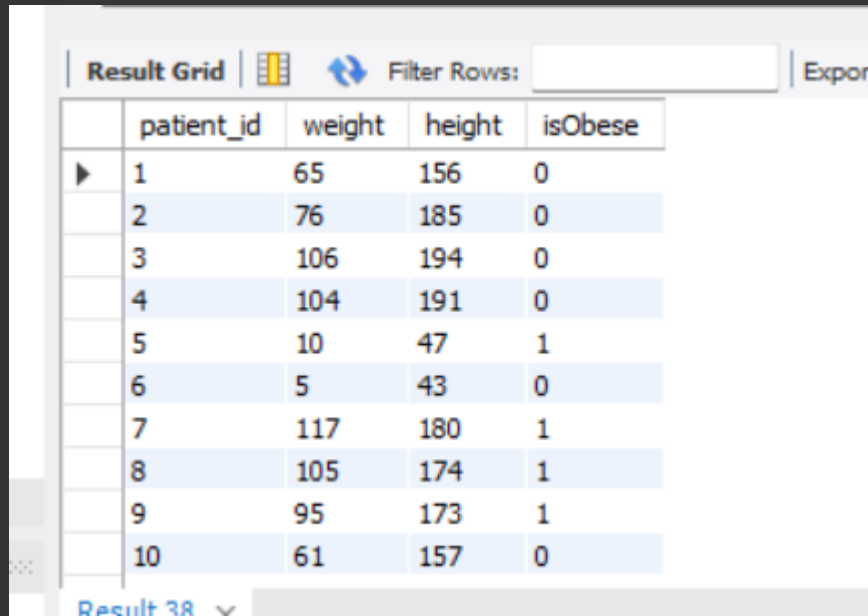
#### Insights:

- Groups patients into 10kg ranges.
- Useful for BMI/weight distribution studies.



### Query 33 – Obesity flag (BMI $\geq 30$ )

```
SELECT patient_id, weight, height, CASE WHEN weight / POWER(height/100, 2) >= 30  
THEN 1 ELSE 0 END AS isObese FROM patients;
```



The screenshot shows a database query result grid with the following data:

	patient_id	weight	height	isObese
▶	1	65	156	0
	2	76	185	0
	3	106	194	0
	4	104	191	0
	5	10	47	1
	6	5	43	0
	7	117	180	1
	8	105	174	1
	9	95	173	1
	10	61	157	0

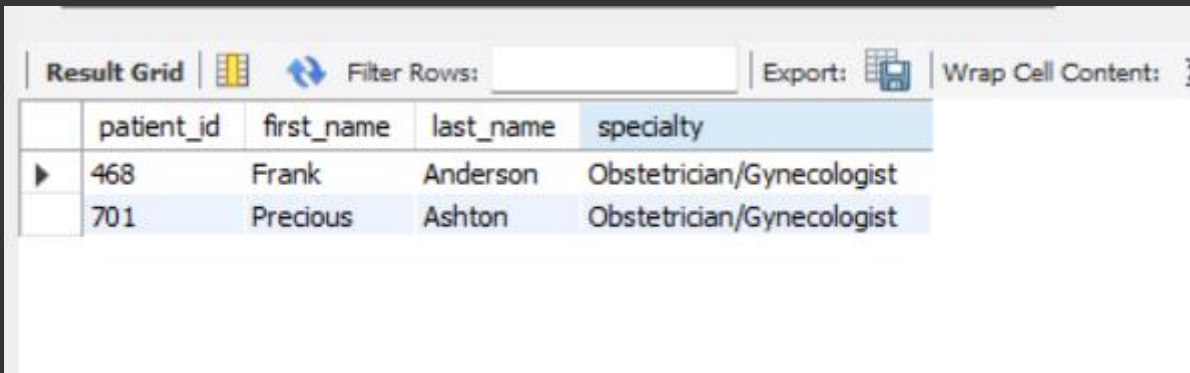
Result 38

#### Insights:

- Flags obese patients using BMI formula.
- Critical for health risk tracking.

### Query 34 – Epilepsy patients with doctor Lisa

```
SELECT p.patient_id, p.first_name, p.last_name, d.specialty
FROM patients p JOIN admissions a ON p.patient_id = a.patient_id JOIN doctors d ON a.doctor_id = d.doctor_id
WHERE a.diagnosis = 'Epilepsy' AND d.first_name = 'Lisa';
```



The screenshot shows a database query result grid. At the top, there are controls: 'Result Grid' with a grid icon, 'Filter Rows:' with a search box, 'Export:' with a document icon, and 'Wrap Cell Content:' with a text wrap icon. The table has five columns: 'patient\_id', 'first\_name', 'last\_name', and 'specialty'. There are two data rows. The first row has patient\_id 468, first\_name Frank, last\_name Anderson, and specialty Obstetrician/Gynecologist. The second row has patient\_id 701, first\_name Precious, last\_name Ashton, and specialty Obstetrician/Gynecologist.

	patient_id	first_name	last_name	specialty
▶	468	Frank	Anderson	Obstetrician/Gynecologist
	701	Precious	Ashton	Obstetrician/Gynecologist

#### Insights:

- Filters patients with epilepsy treated by Dr. Lisa.
- Useful for targeted case reviews.