COVID DATABASE ANALYSIS

1 INTRODUCTION:

COVID-19 analysis is needed to guide public health responses, allocate resources effectively, identify trends, plan vaccination campaigns, promote public awareness, conduct research, monitor the virus, foster global collaboration, learn from the pandemic for future preparedness, understand economic impact, address mental health issues, facilitate a return to normal life, and maintain public trust in healthcare systems. It is crucial for saving lives and mitigating the pandemic's impact.

1. covid_cases_india Table:

This table holds data on COVID-19 cases reported in India. It includes details like the date of reporting, the number of confirmed cases, deaths, recoveries, and the specific district where each case is reported.

2. covid_testing_india Table:

This table records information related to COVID-19 testing conducted in India. It includes data such as the date of testing, the total number of tests performed, the number of positive test results, and the district where testing took place.

3. covid_vaccination_india Table:

This table tracks COVID-19 vaccination data for India. It includes records for each vaccination administered, including the date of vaccination, the total number of vaccinations given, the count of first and second doses administered, and the district where vaccinations are provided.

4. covid_hospitalizations_india Table:

This table contains data about COVID-19 hospitalizations in India. It includes information such as the date of admission, the total number of patients hospitalized, the number of patients in the intensive care unit (ICU), and the district where hospitalization is happening.

5. covid_demographics_india Table:

This table stores demographic information for individuals affected by COVID-19 in India. It includes details like age, gender, and a link to the specific COVID-19 case with which each demographic record is associated.

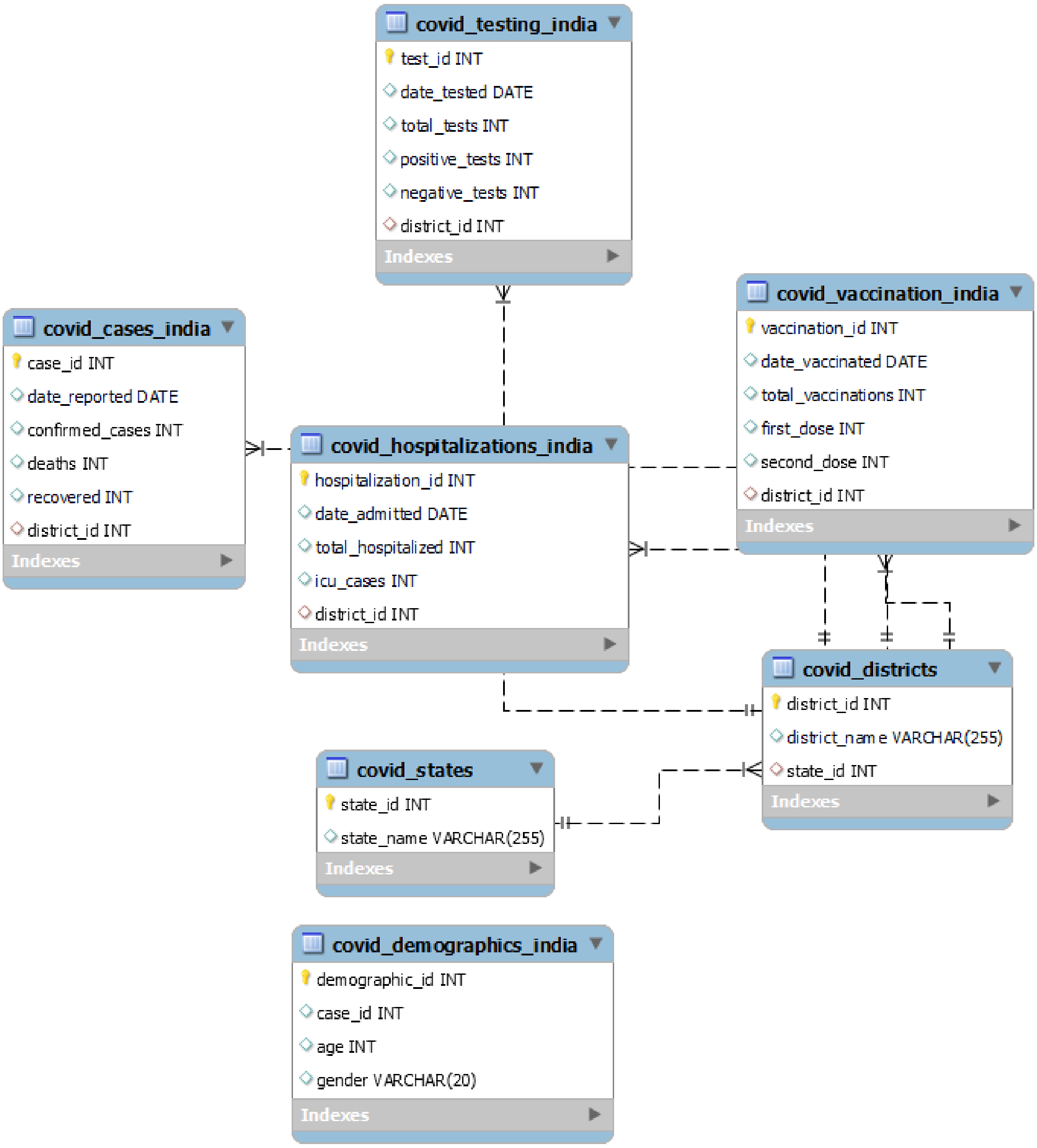
6. covid_states Table:

This table provides information about Indian states, including a unique identifier and the name of each state.

7. covid_districts Table:

This table contains data about Indian districts or regions. It includes a unique identifier for each district, the name of the district, and a reference to the state to which the district belongs.
These tables collectively help in organizing, storing, and analyzing crucial COVID-19 data, enabling efficient management and data-driven decision-making to address the pandemic's impact in different regions of India.

ER DIAGRAM


covid_testing_india
test_id INT
date_tested DATE
total_tests INT
positive_tests INT
negative_tests INT
district_id INT
Indexes
covid_cases_india
case_id INT
date_reported DATE
confirmed_cases INT
deaths INT
recovered INT
district_id INT
Indexes
covid_hospitalizations_india
hospitalization_id INT
date_admitted DATE
total_hospitalized INT
icu_cases INT
district_id INT
Indexes
covid_vaccination_india
vaccination_id INT
date_vaccinated DATE
total_vaccinations INT
first_dose INT
second_dose INT
district_id INT
Indexes
covid_districts
district_id INT
district_name VARCHAR(255)
state_id INT
Indexes
covid_states
state_id INT
state_name VARCHAR(255)
Indexes
covid_demographics_india
demographic_id INT
case_id INT
age INT
gender VARCHAR(20)
Indexes

3. Database Design

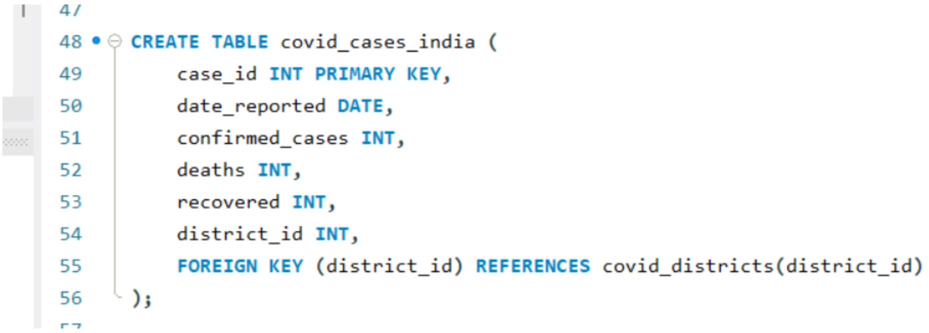
Databases: COVID DATABASE

TABLES

1. **covid_cases_india Table**
2. **covid_testing_india Table**
3. **covid_vaccination_india Table**
4. **covid_hospitalizations_india Table**
5. **covid_demographics_india Table**
6. **covid_states Table**
7. **covid_districts Table**

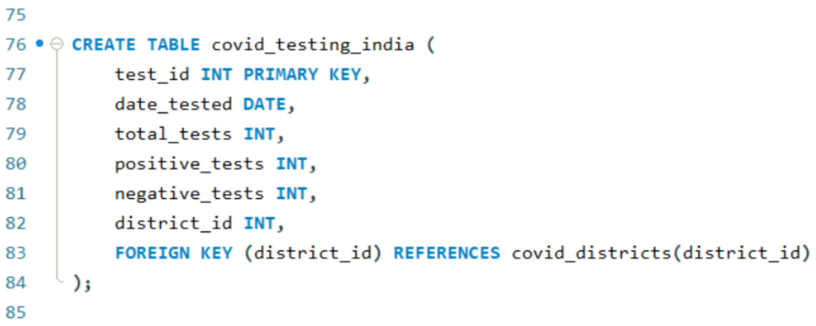
4. Creating Table

1 covid_cases_india Table



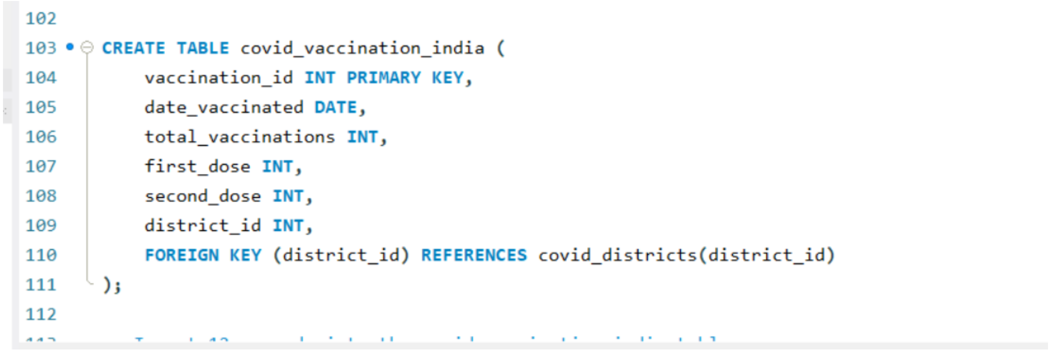
```
CREATE TABLE covid_cases_india (
    case_id INT PRIMARY KEY,
    date_reported DATE,
    confirmed_cases INT,
    deaths INT,
    recovered INT,
    district_id INT,
    FOREIGN KEY (district_id) REFERENCES covid_districts(district_id)
);
```

2 covid_testing_india Table



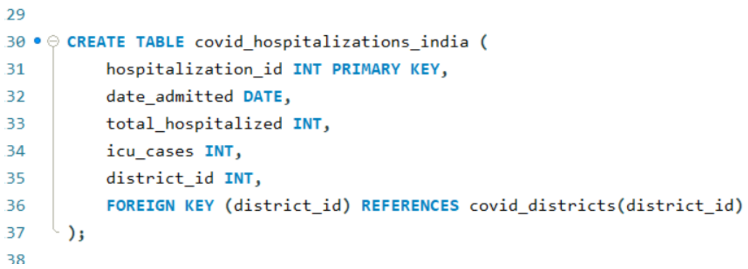
```
CREATE TABLE covid_testing_india (
    test_id INT PRIMARY KEY,
    date_tested DATE,
    total_tests INT,
    positive_tests INT,
    negative_tests INT,
    district_id INT,
    FOREIGN KEY (district_id) REFERENCES covid_districts(district_id)
);
```

3 covid_vaccination_india Table



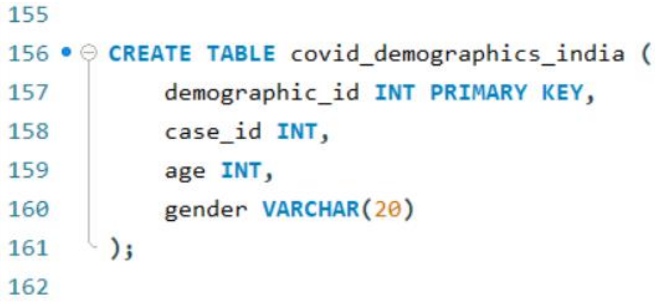
```
CREATE TABLE covid_vaccination_india (
    vaccination_id INT PRIMARY KEY,
    date_vaccinated DATE,
    total_vaccinations INT,
    first_dose INT,
    second_dose INT,
    district_id INT,
    FOREIGN KEY (district_id) REFERENCES covid_districts(district_id)
);
```

4 covid_hospitalizations_india Table



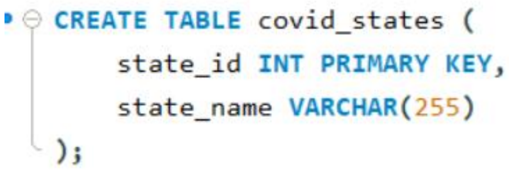
```sql
CREATE TABLE covid_hospitalizations_india (
    hospitalization_id INT PRIMARY KEY,
    date_admitted DATE,
    total_hospitalized INT,
    icu_cases INT,
    district_id INT,
    FOREIGN KEY (district_id) REFERENCES covid_districts(district_id)
);
```

5 covid_demographics_india Table



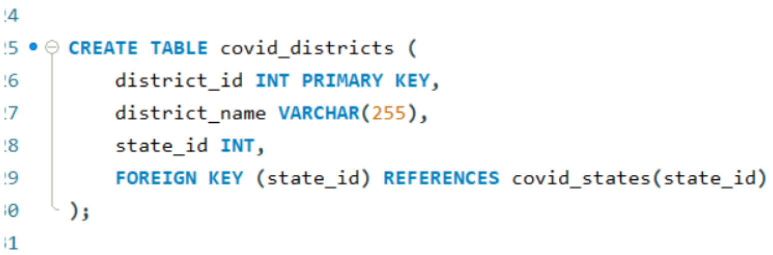
```sql
CREATE TABLE covid_demographics_india (
    demographic_id INT PRIMARY KEY,
    case_id INT,
    age INT,
    gender VARCHAR(20)
);
```

6 covid_states Table



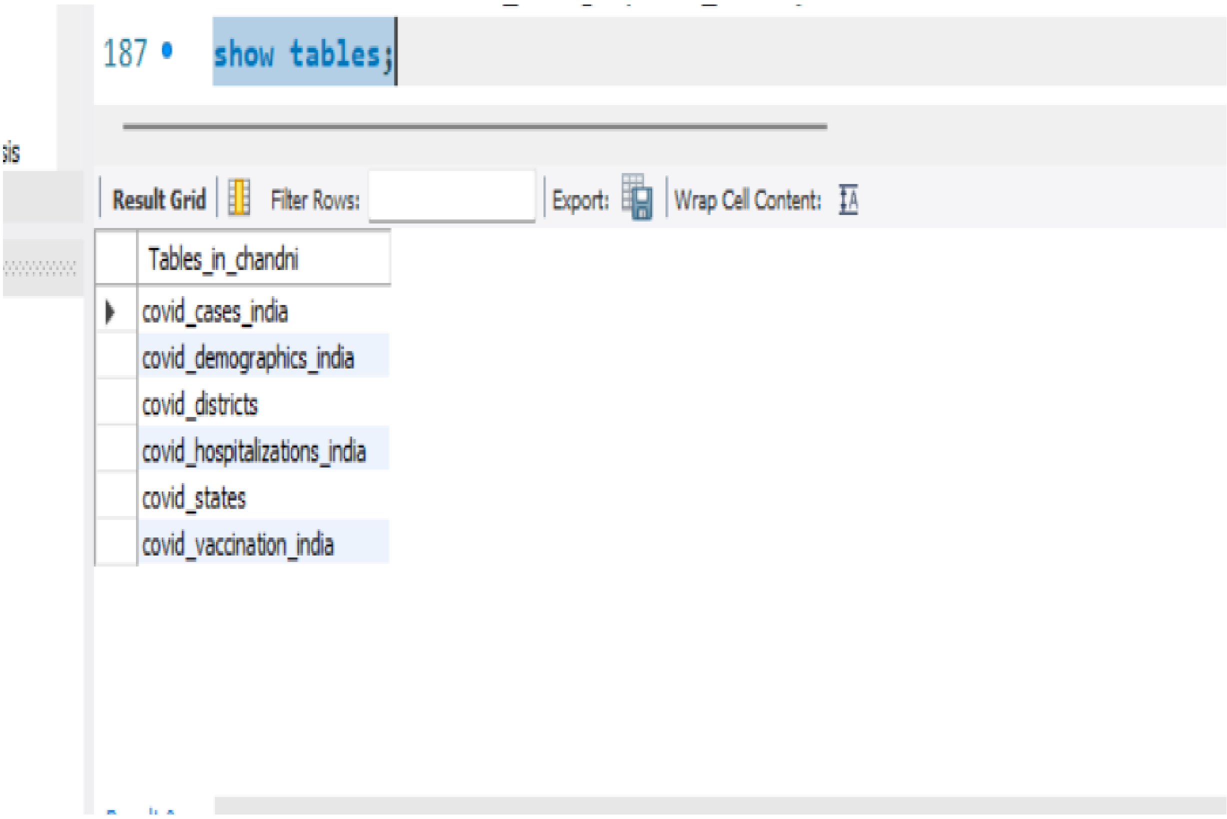
```sql
CREATE TABLE covid_states (
    state_id INT PRIMARY KEY,
    state_name VARCHAR(255)
);
```

7 covid_districts Table



```sql
CREATE TABLE covid_districts (
    district_id INT PRIMARY KEY,
    district_name VARCHAR(255),
    state_id INT,
    FOREIGN KEY (state_id) REFERENCES covid_states(state_id)
);
```

5. Tables in databases



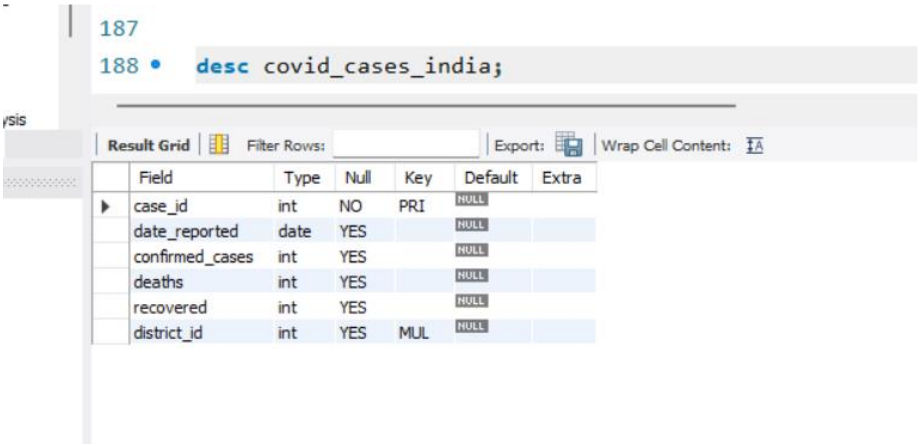
```
show tables;
```

Tables_in_chandni
covid_cases_india
covid_demographics_india
covid_districts
covid_hospitalizations_india
covid_states
covid_vaccination_india

6. Data Definition language (DDL)

a) Creating Tables:

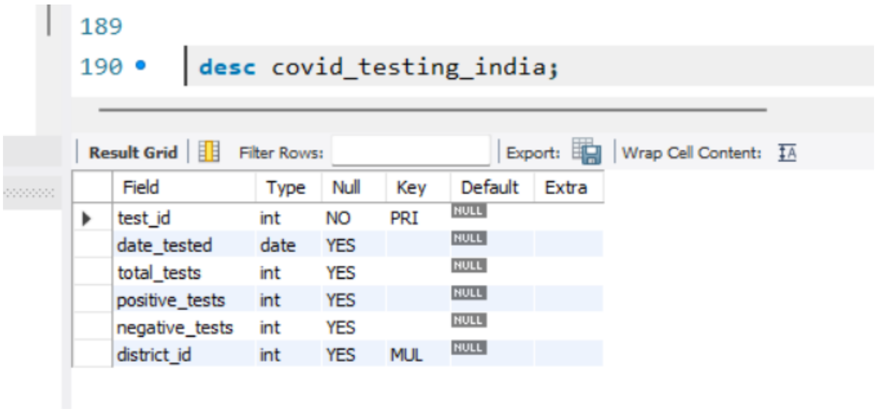
1 covid_cases india



```
desc covid_cases_india;
```

Field	Type	Null	Key	Default	Extra
case_id	int	NO	PRI	NULL	
date_reported	date	YES		NULL	
confirmed_cases	int	YES		NULL	
deaths	int	YES		NULL	
recovered	int	YES		NULL	
district_id	int	YES	MUL	NULL	

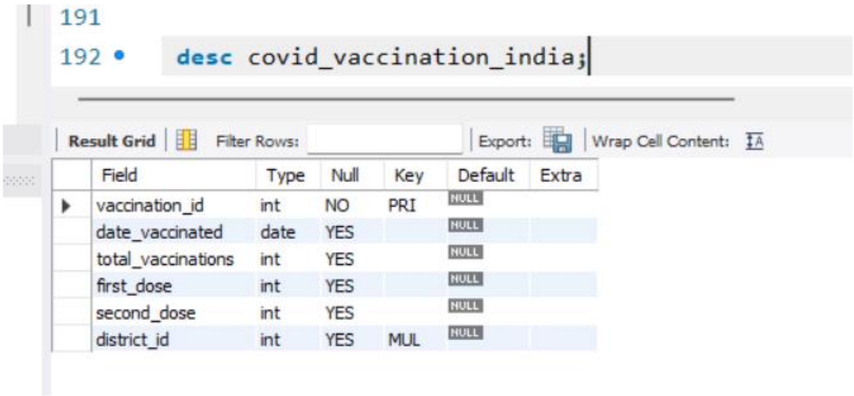
2 covid_testing_india



```
desc covid_testing_india;
```

Field	Type	Null	Key	Default	Extra
test_id	int	NO	PRI	NULL	
date_tested	date	YES		NULL	
total_tests	int	YES		NULL	
positive_tests	int	YES		NULL	
negative_tests	int	YES		NULL	
district_id	int	YES	MUL	NULL	

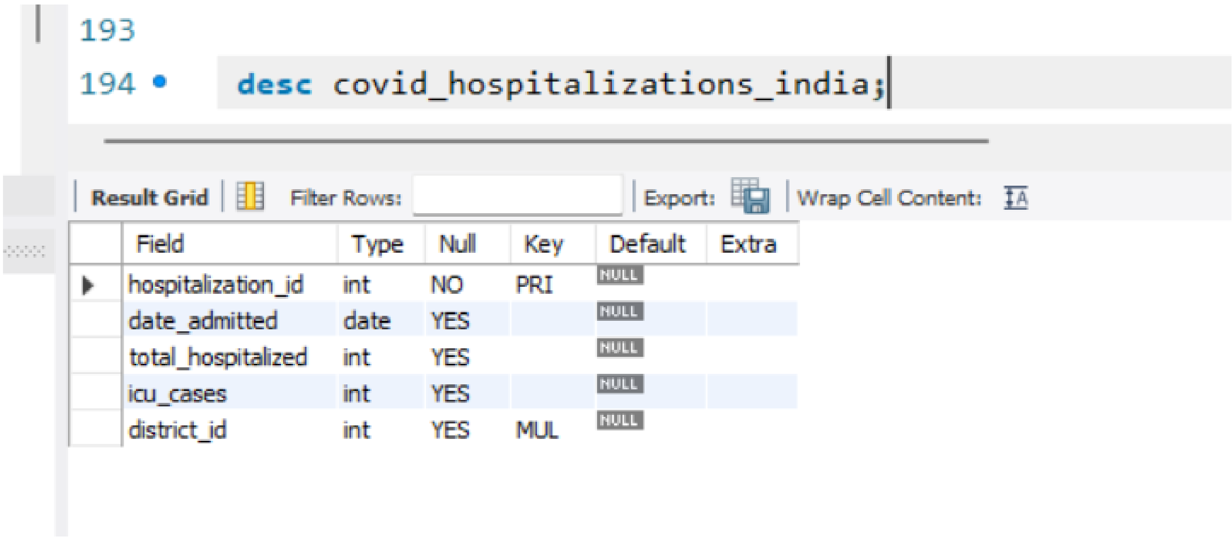
3 covid_vaccination_india



```
desc covid_vaccination_india;
```

Field	Type	Null	Key	Default	Extra
vaccination_id	int	NO	PRI	NULL	
date_vaccinated	date	YES		NULL	
total_vaccinations	int	YES		NULL	
first_dose	int	YES		NULL	
second_dose	int	YES		NULL	
district_id	int	YES	MUL	NULL	

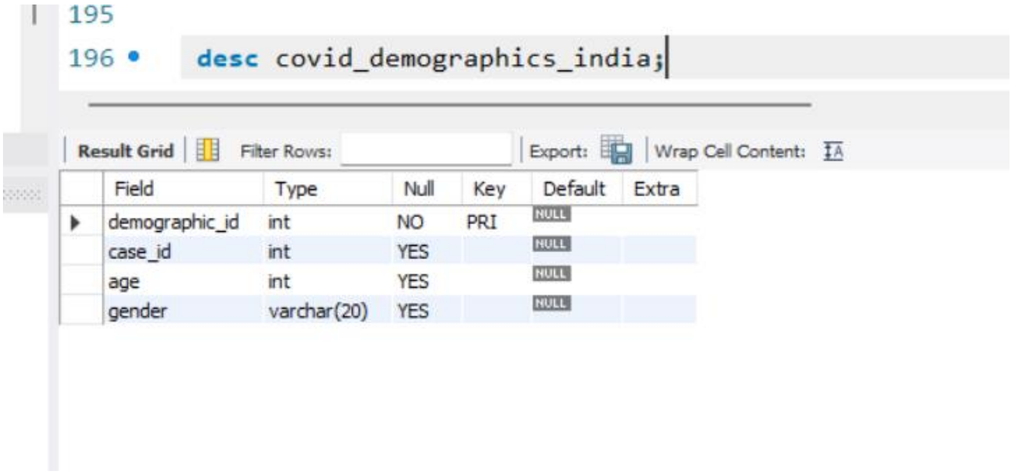
4 covid_hospitalization_india



```
desc covid_hospitalizations_india;
```

Field	Type	Null	Key	Default	Extra
hospitalization_id	int	NO	PRI	NULL	
date_admitted	date	YES		NULL	
total_hospitalized	int	YES		NULL	
icu_cases	int	YES		NULL	
district_id	int	YES	MUL	NULL	

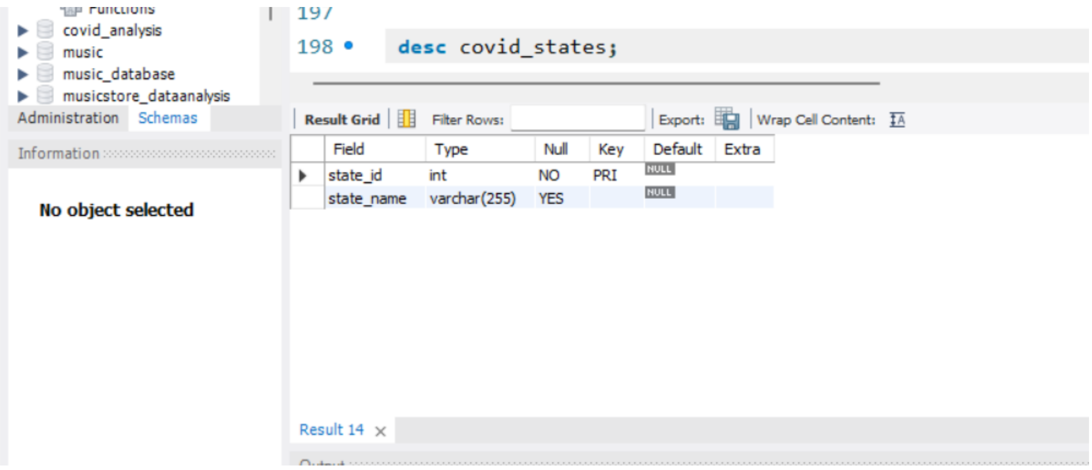
5 covid_demographics_india



```
desc covid_demographics_india;
```

Field	Type	Null	Key	Default	Extra
demographic_id	int	NO	PRI	NULL	
case_id	int	YES		NULL	
age	int	YES		NULL	
gender	varchar(20)	YES		NULL	

6 covid _states

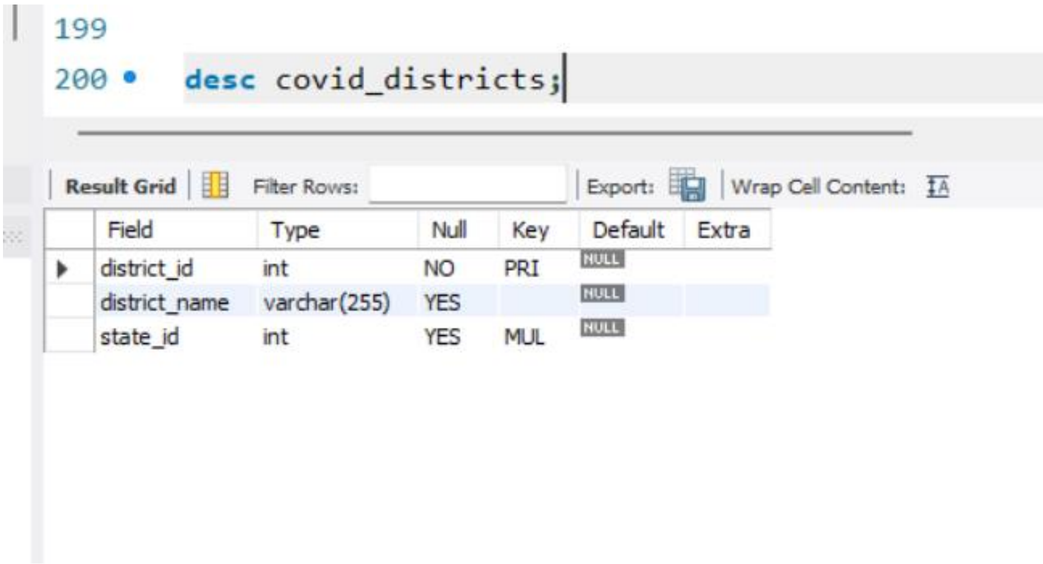


No object selected

```
desc covid_states;
```

Field	Type	Null	Key	Default	Extra
state_id	int	NO	PRI	NULL	
state_name	varchar(255)	YES		NULL	

7 covid_districts

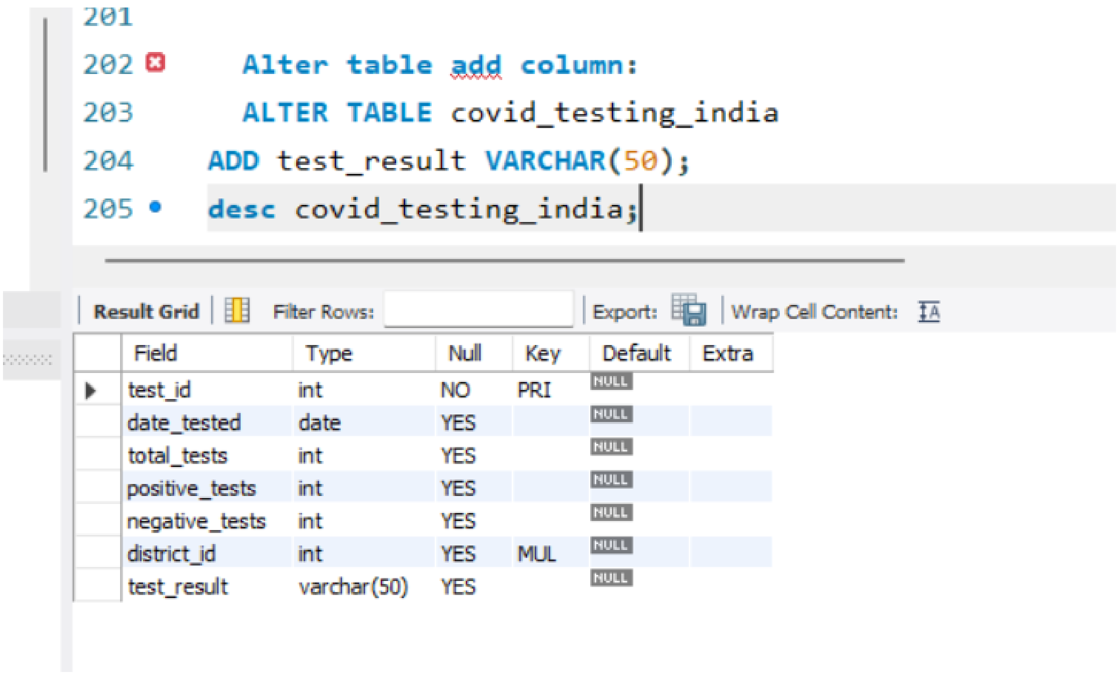


```
desc covid_districts;
```

Field	Type	Null	Key	Default	Extra
district_id	int	NO	PRI	NULL	
district_name	varchar(255)	YES		NULL	
state_id	int	YES	MUL	NULL	

b) Alter table

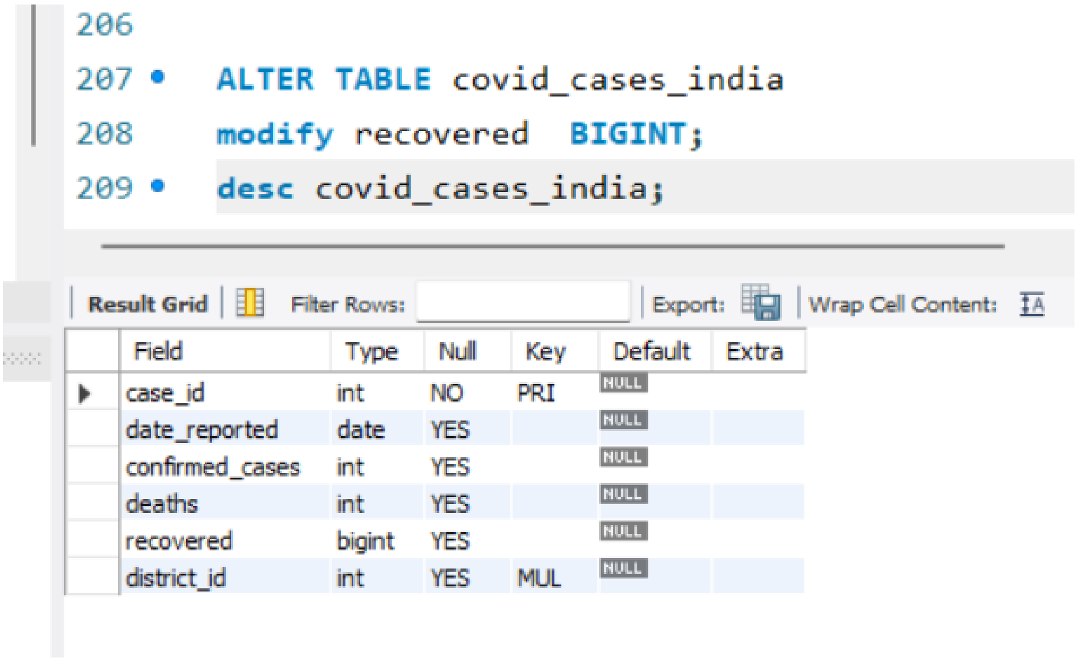
1) Alter table add column:



```
    Alter table add column:
    ALTER TABLE covid_testing_india
ADD test_result VARCHAR(50);
desc covid_testing_india;
```

	Field	Type	Null	Key	Default	Extra
▶	test_id	int	NO	PRI	NULL	
	date_tested	date	YES		NULL	
	total_tests	int	YES		NULL	
	positive_tests	int	YES		NULL	
	negative_tests	int	YES		NULL	
	district_id	int	YES	MUL	NULL	
	test_result	varchar(50)	YES		NULL	

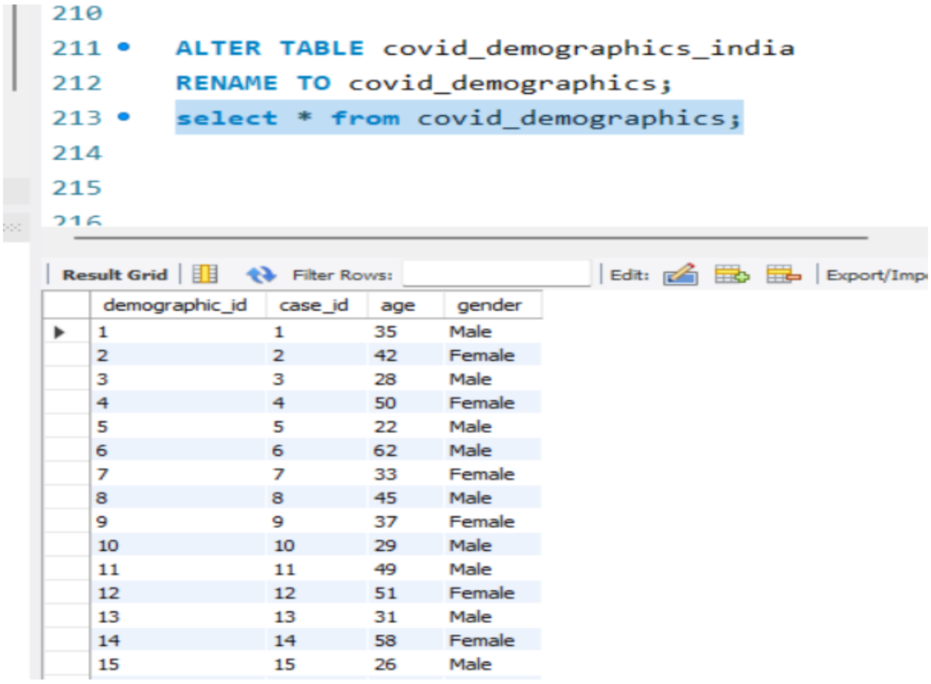
2) Alter table modify column:



```
ALTER TABLE covid_cases_india
modify recovered  BIGINT;
desc covid_cases_india;
```

	Field	Type	Null	Key	Default	Extra
▶	case_id	int	NO	PRI	NULL	
	date_reported	date	YES		NULL	
	confirmed_cases	int	YES		NULL	
	deaths	int	YES		NULL	
	recovered	bigint	YES		NULL	
	district_id	int	YES	MUL	NULL	

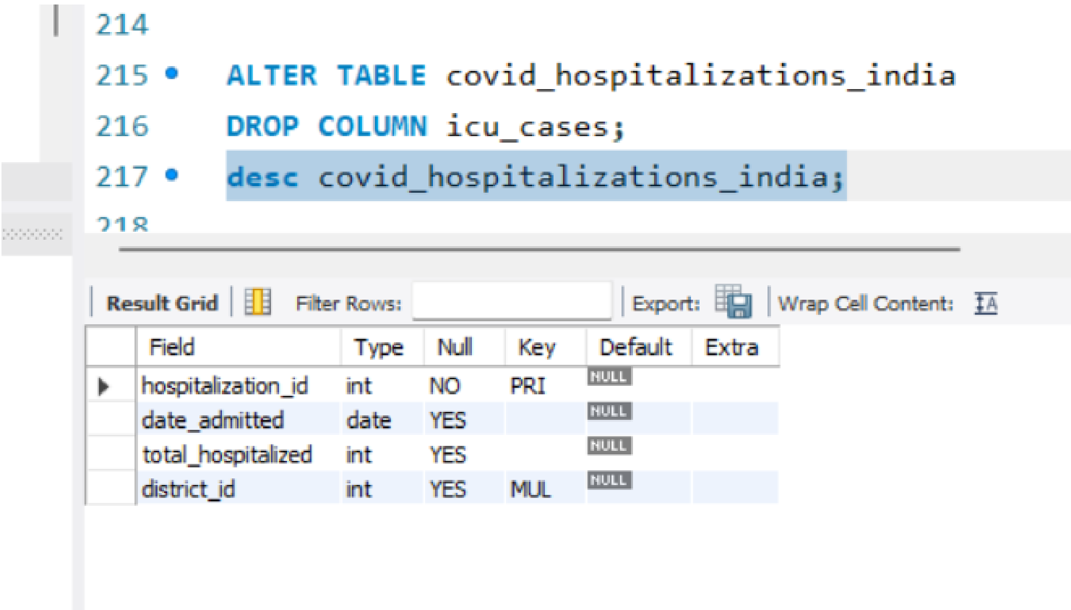
3) Alter table rename column:



```
ALTER TABLE covid_demographics_india
RENAME TO covid_demographics;
select * from covid_demographics;
```

demographic_id	case_id	age	gender
1	1	35	Male
2	2	42	Female
3	3	28	Male
4	4	50	Female
5	5	22	Male
6	6	62	Male
7	7	33	Female
8	8	45	Male
9	9	37	Female
10	10	29	Male
11	11	49	Male
12	12	51	Female
13	13	31	Male
14	14	58	Female
15	15	26	Male

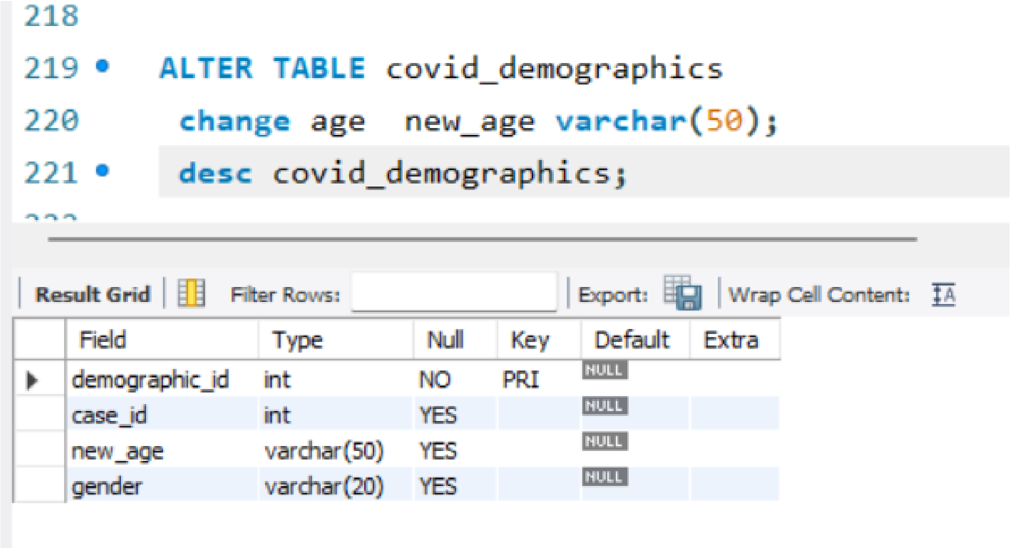
4) Alter table drop column:



```
ALTER TABLE covid_hospitalizations_india
DROP COLUMN icu_cases;
desc covid_hospitalizations_india;
```

Field	Type	Null	Key	Default	Extra
hospitalization_id	int	NO	PRI	NULL	
date_admitted	date	YES		NULL	
total_hospitalized	int	YES		NULL	
district_id	int	YES	MUL	NULL	

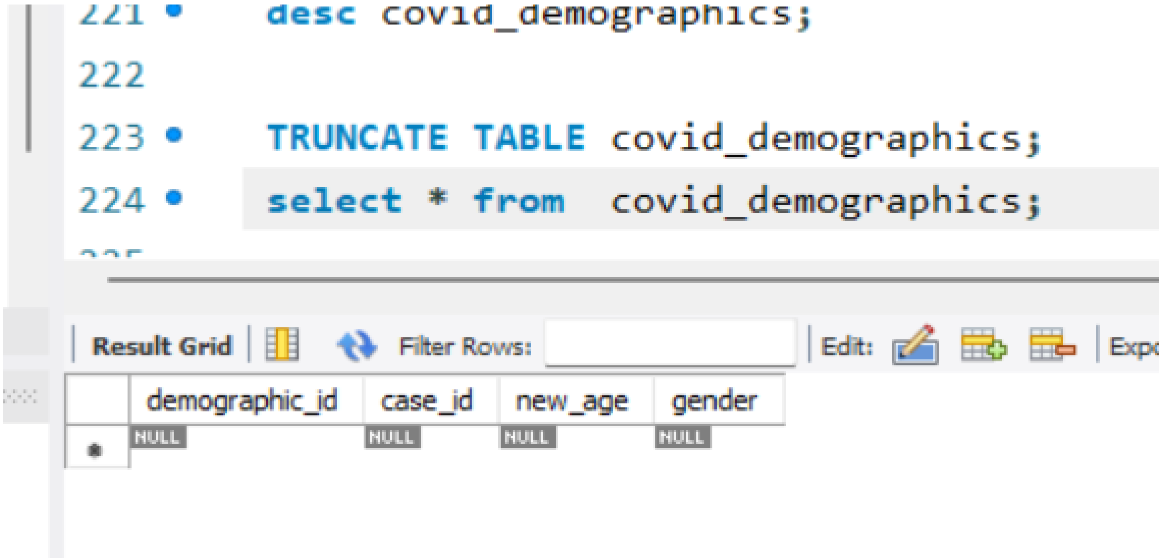
c) Rename table



```
ALTER TABLE covid_demographics
 change age  new_age varchar(50);
 desc covid_demographics;
```

	Field	Type	Null	Key	Default	Extra
▶	demographic_id	int	NO	PRI	NULL	
	case_id	int	YES		NULL	
	new_age	varchar(50)	YES		NULL	
	gender	varchar(20)	YES		NULL	

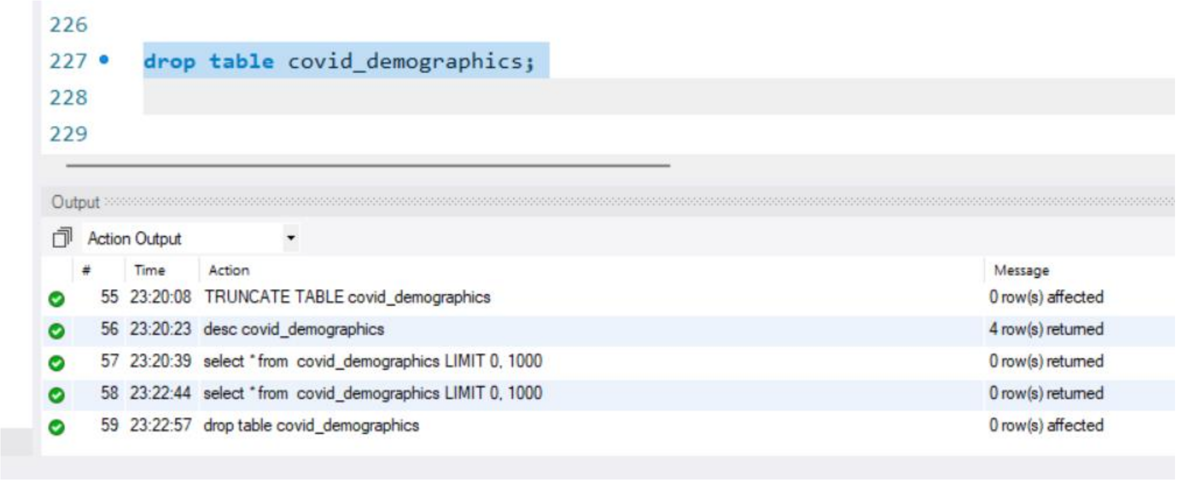
d) Truncate table



```
desc covid_demographics;

TRUNCATE TABLE covid_demographics;
select * from  covid_demographics;
```

	demographic_id	case_id	new_age	gender
*	NULL	NULL	NULL	NULL

e) Drop table

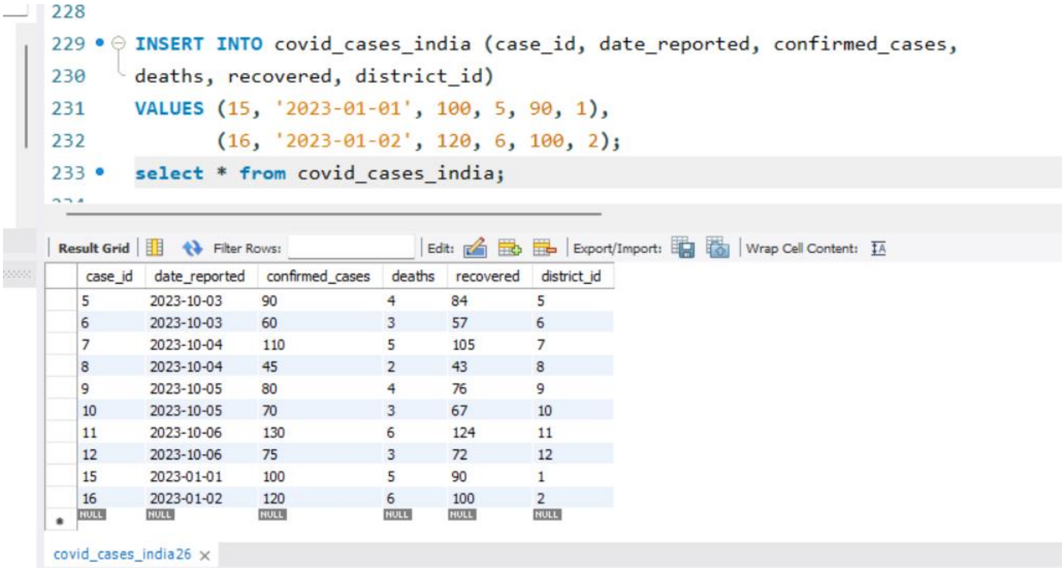


```sql
drop table covid_demographics;
```

```
   #   Time      Action                                           Message
   55  23:20:08  TRUNCATE TABLE covid_demographics                0 row(s) affected
   56  23:20:23  desc covid_demographics                          4 row(s) returned
   57  23:20:39  select * from  covid_demographics LIMIT 0, 1000  0 row(s) returned
   58  23:22:44  select * from  covid_demographics LIMIT 0, 1000  0 row(s) returned
   59  23:22:57  drop table covid_demographics                    0 row(s) affected
```

7. Data Manipulation language (DMl)

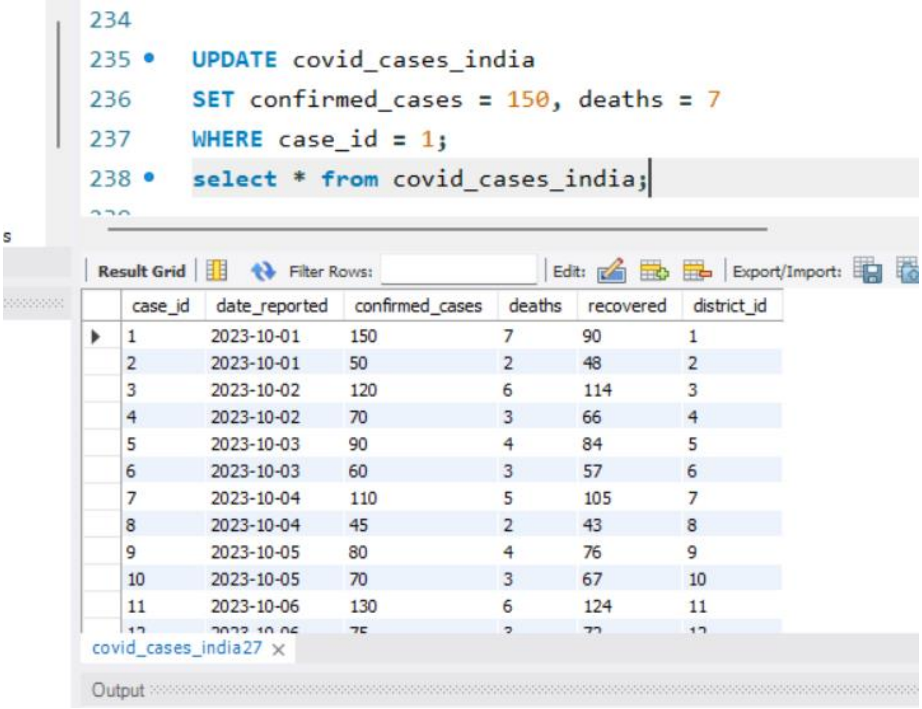
a. Insert into table



```
INSERT INTO covid_cases_india (case_id, date_reported, confirmed_cases,
deaths, recovered, district_id)
VALUES (15, '2023-01-01', 100, 5, 90, 1),
       (16, '2023-01-02', 120, 6, 100, 2);
select * from covid_cases_india;
```

case_id	date_reported	confirmed_cases	deaths	recovered	district_id
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12
15	2023-01-01	100	5	90	1
16	2023-01-02	120	6	100	2
NULL	NULL	NULL	NULL	NULL	NULL

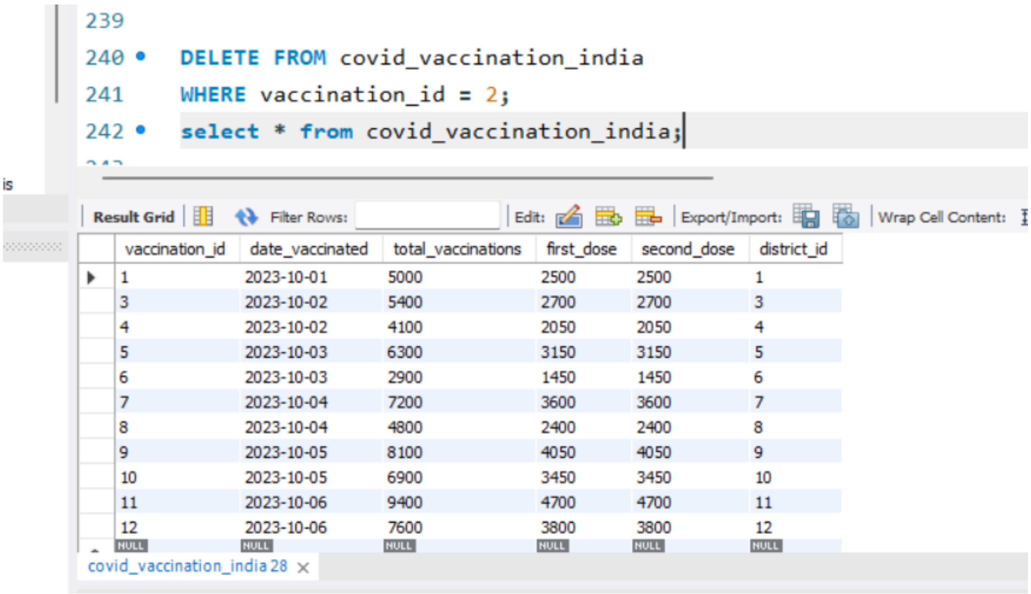
b. Update into table



```
UPDATE covid_cases_india
SET confirmed_cases = 150, deaths = 7
WHERE case_id = 1;
select * from covid_cases_india;
```

case_id	date_reported	confirmed_cases	deaths	recovered	district_id
1	2023-10-01	150	7	90	1
2	2023-10-01	50	2	48	2
3	2023-10-02	120	6	114	3
4	2023-10-02	70	3	66	4
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12

c. Delete into table

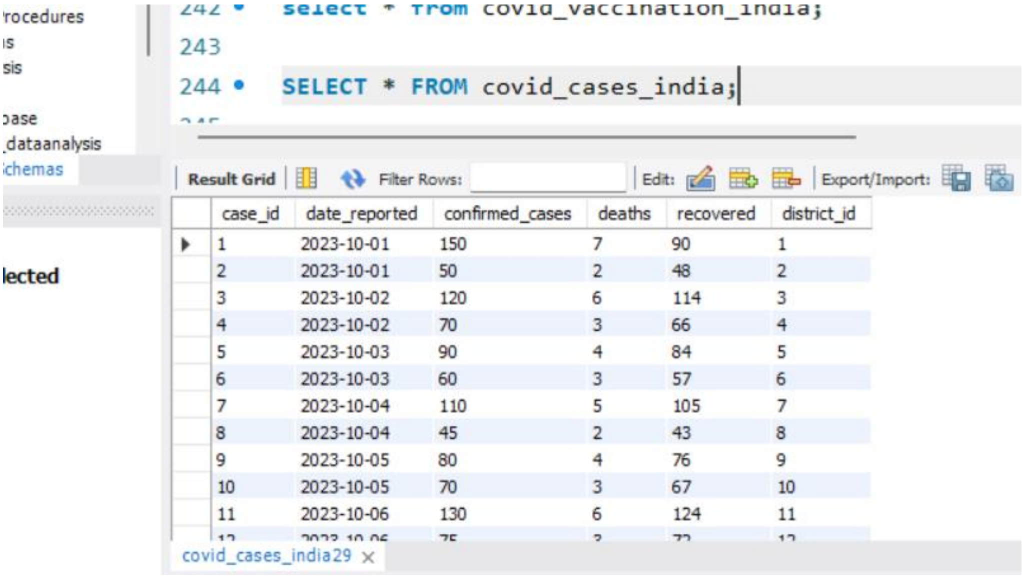


```
DELETE FROM covid_vaccination_india
WHERE vaccination_id = 2;
select * from covid_vaccination_india;
```

vaccination_id	date_vaccinated	total_vaccinations	first_dose	second_dose	district_id
1	2023-10-01	5000	2500	2500	1
3	2023-10-02	5400	2700	2700	3
4	2023-10-02	4100	2050	2050	4
5	2023-10-03	6300	3150	3150	5
6	2023-10-03	2900	1450	1450	6
7	2023-10-04	7200	3600	3600	7
8	2023-10-04	4800	2400	2400	8
9	2023-10-05	8100	4050	4050	9
10	2023-10-05	6900	3450	3450	10
11	2023-10-06	9400	4700	4700	11
12	2023-10-06	7600	3800	3800	12
NULL	NULL	NULL	NULL	NULL	NULL

8. Data Query Language (DQL)

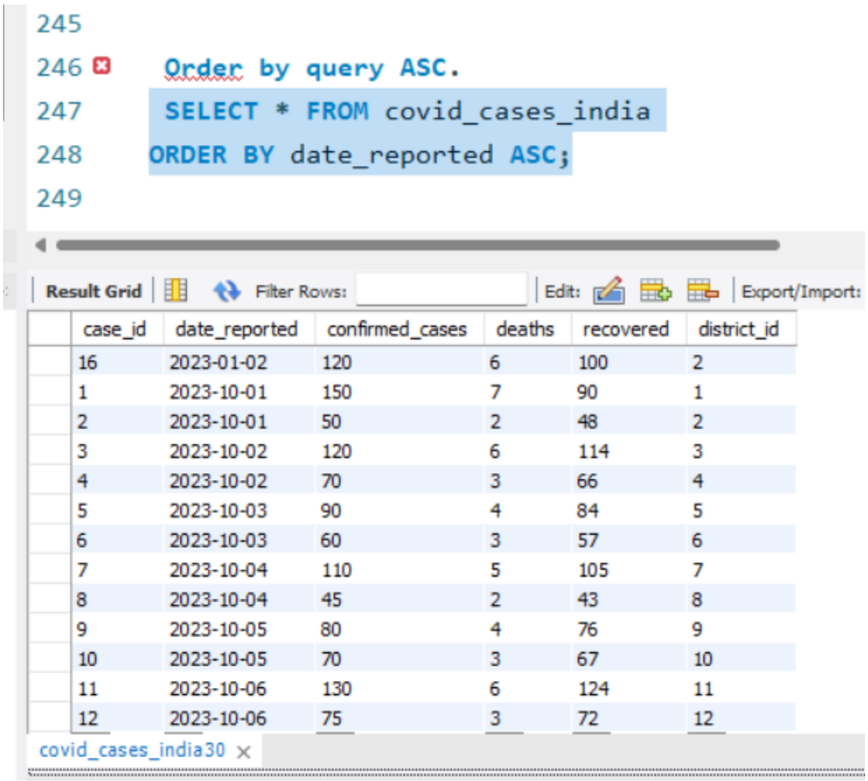
a. Select query



```
select * from covid_vaccination_india;

SELECT * FROM covid_cases_india;
```

case_id	date_reported	confirmed_cases	deaths	recovered	district_id
1	2023-10-01	150	7	90	1
2	2023-10-01	50	2	48	2
3	2023-10-02	120	6	114	3
4	2023-10-02	70	3	66	4
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12

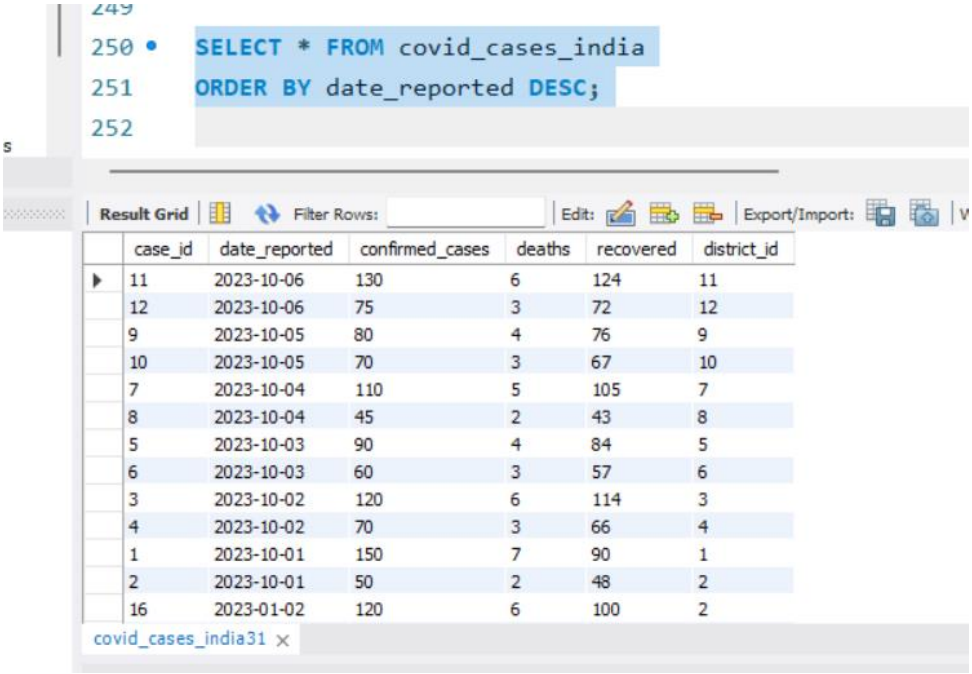
b. Order by query ASC.



```

  Order by query ASC.
  SELECT * FROM covid_cases_india
ORDER BY date_reported ASC;

```

case_id	date_reported	confirmed_cases	deaths	recovered	district_id
16	2023-01-02	120	6	100	2
1	2023-10-01	150	7	90	1
2	2023-10-01	50	2	48	2
3	2023-10-02	120	6	114	3
4	2023-10-02	70	3	66	4
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12

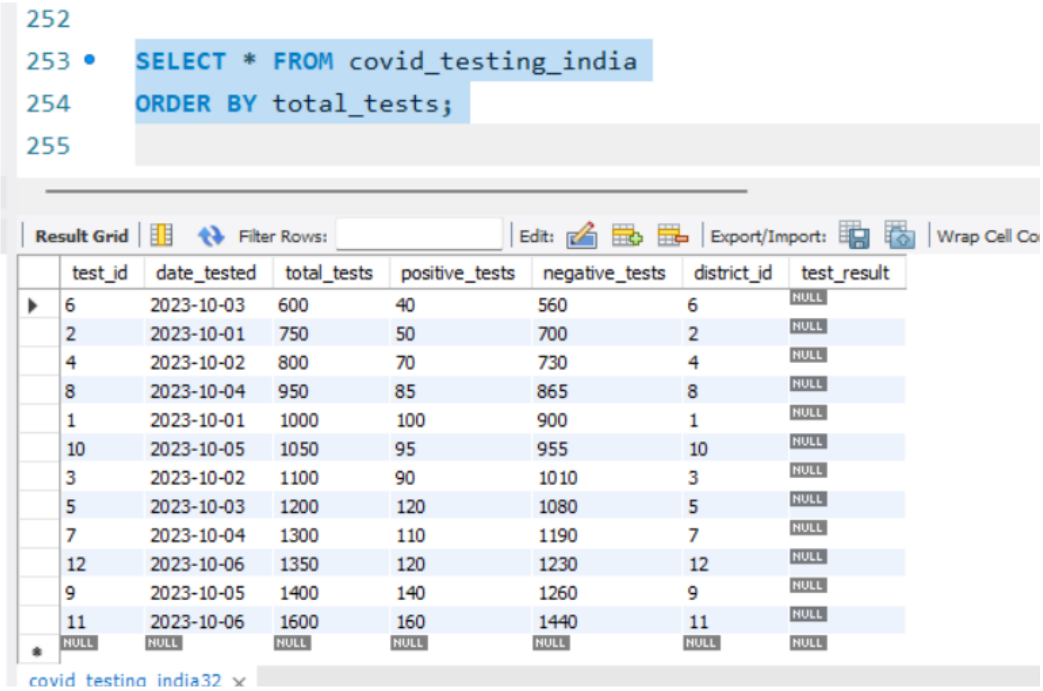
c. Order by query DESC



```sql
SELECT * FROM covid_cases_india
ORDER BY date_reported DESC;
```

case_id	date_reported	confirmed_cases	deaths	recovered	district_id
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
3	2023-10-02	120	6	114	3
4	2023-10-02	70	3	66	4
1	2023-10-01	150	7	90	1
2	2023-10-01	50	2	48	2
16	2023-01-02	120	6	100	2

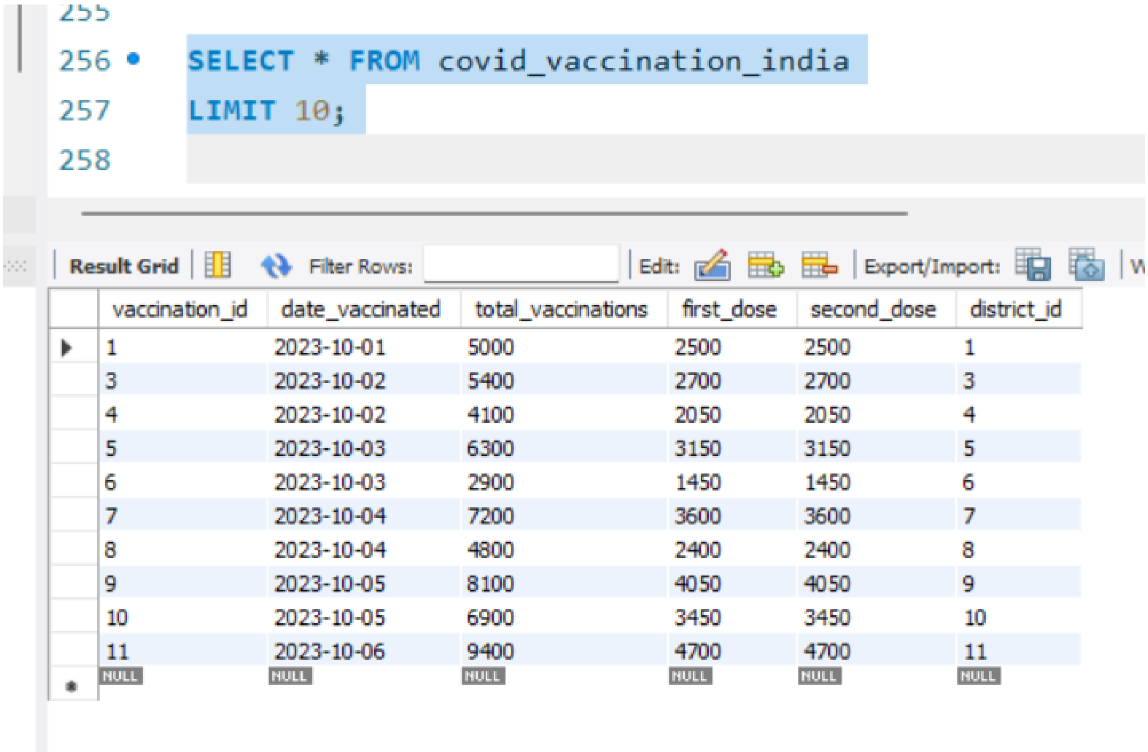
d. Order by column



```sql
SELECT * FROM covid_testing_india
ORDER BY total_tests;
```

test_id	date_tested	total_tests	positive_tests	negative_tests	district_id	test_result
6	2023-10-03	600	40	560	6	NULL
2	2023-10-01	750	50	700	2	NULL
4	2023-10-02	800	70	730	4	NULL
8	2023-10-04	950	85	865	8	NULL
1	2023-10-01	1000	100	900	1	NULL
10	2023-10-05	1050	95	955	10	NULL
3	2023-10-02	1100	90	1010	3	NULL
5	2023-10-03	1200	120	1080	5	NULL
7	2023-10-04	1300	110	1190	7	NULL
12	2023-10-06	1350	120	1230	12	NULL
9	2023-10-05	1400	140	1260	9	NULL
11	2023-10-06	1600	160	1440	11	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

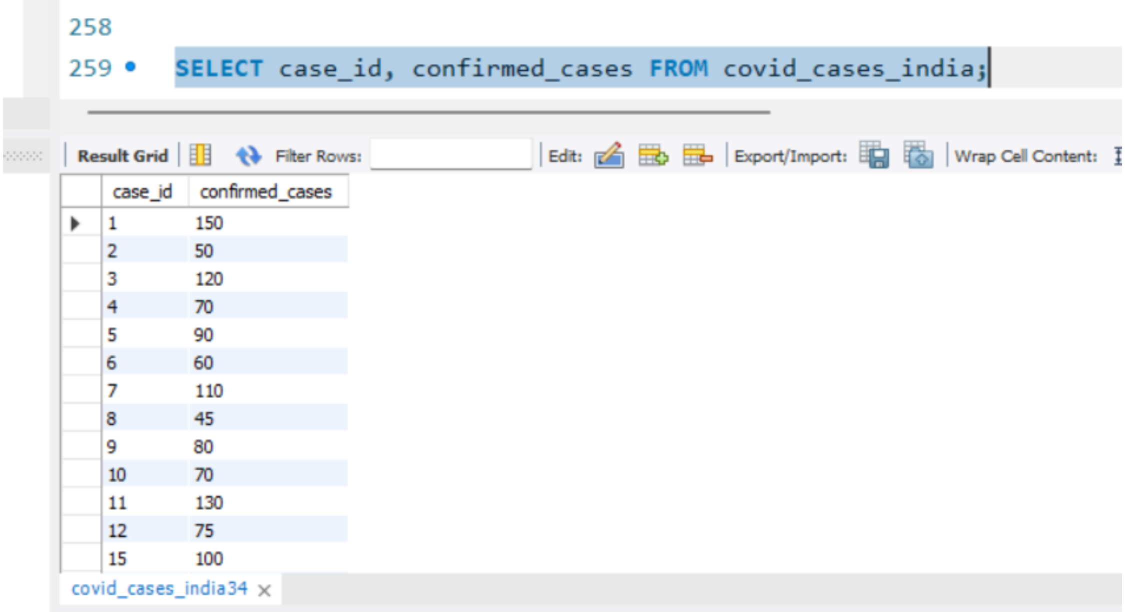
e. Limit query



```
SELECT * FROM covid_vaccination_india
LIMIT 10;
```

vaccination_id	date_vaccinated	total_vaccinations	first_dose	second_dose	district_id
1	2023-10-01	5000	2500	2500	1
3	2023-10-02	5400	2700	2700	3
4	2023-10-02	4100	2050	2050	4
5	2023-10-03	6300	3150	3150	5
6	2023-10-03	2900	1450	1450	6
7	2023-10-04	7200	3600	3600	7
8	2023-10-04	4800	2400	2400	8
9	2023-10-05	8100	4050	4050	9
10	2023-10-05	6900	3450	3450	10
11	2023-10-06	9400	4700	4700	11
NULL	NULL	NULL	NULL	NULL	NULL

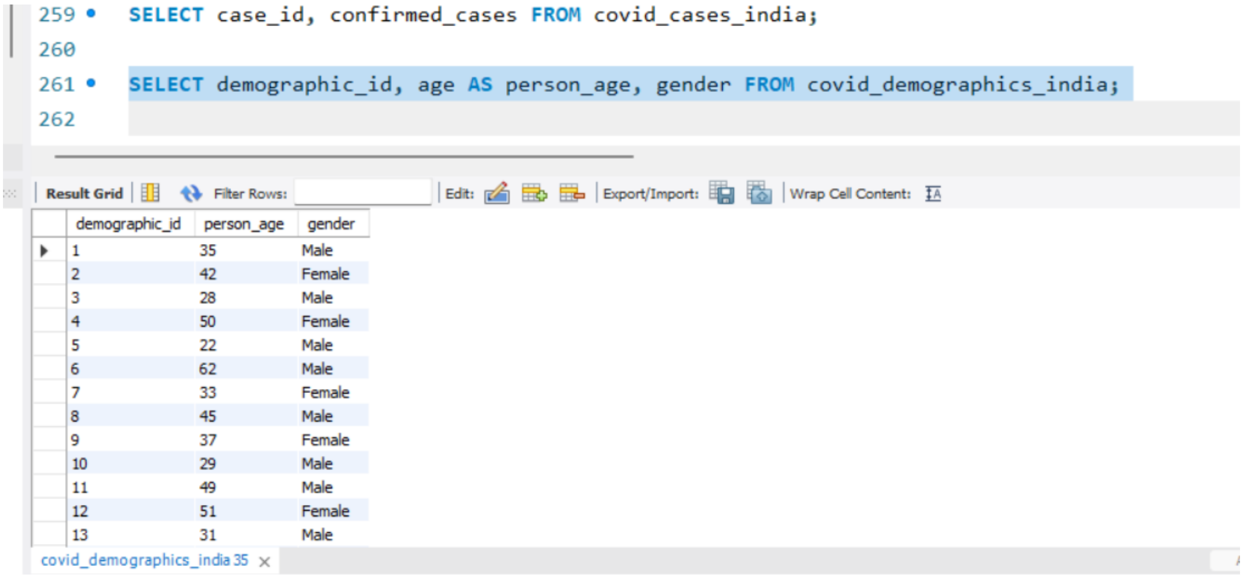
f. Select query with specific column



```
SELECT case_id, confirmed_cases FROM covid_cases_india;
```

case_id	confirmed_cases
1	150
2	50
3	120
4	70
5	90
6	60
7	110
8	45
9	80
10	70
11	130
12	75
15	100

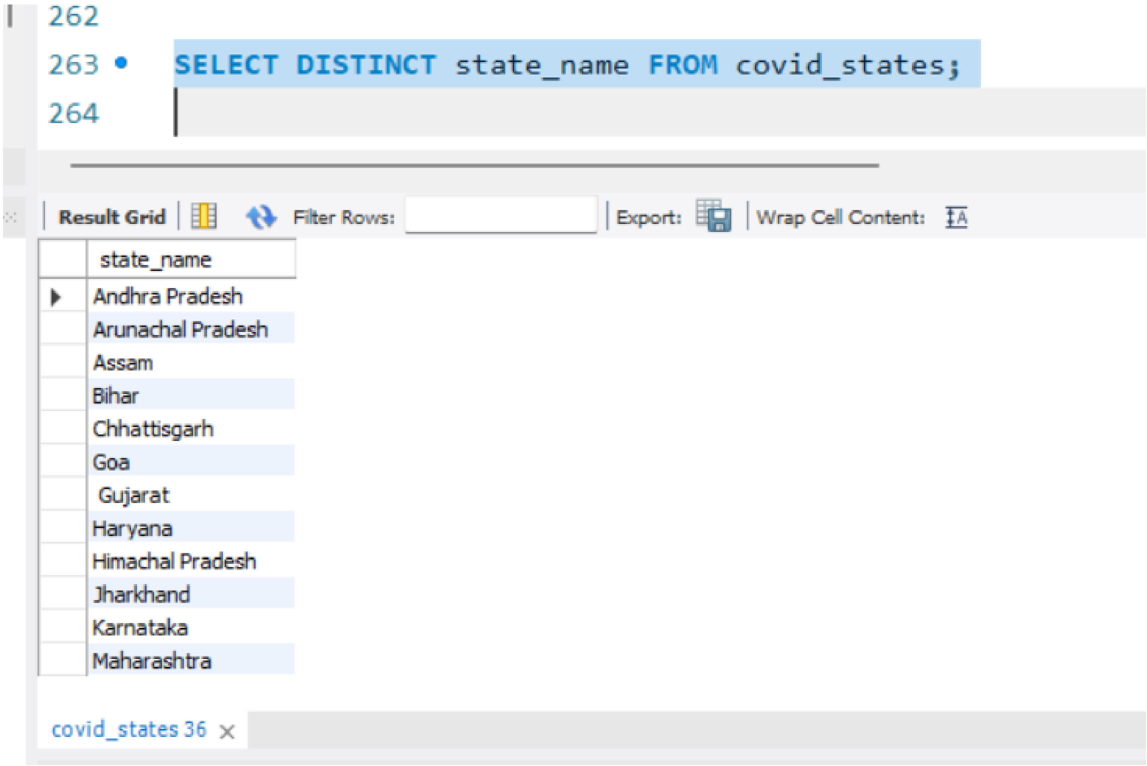
g. Select query with column name change



```
SELECT case_id, confirmed_cases FROM covid_cases_india;

SELECT demographic_id, age AS person_age, gender FROM covid_demographics_india;

```

demographic_id	person_age	gender
1	35	Male
2	42	Female
3	28	Male
4	50	Female
5	22	Male
6	62	Male
7	33	Female
8	45	Male
9	37	Female
10	29	Male
11	49	Male
12	51	Female
13	31	Male

h. Distinct query

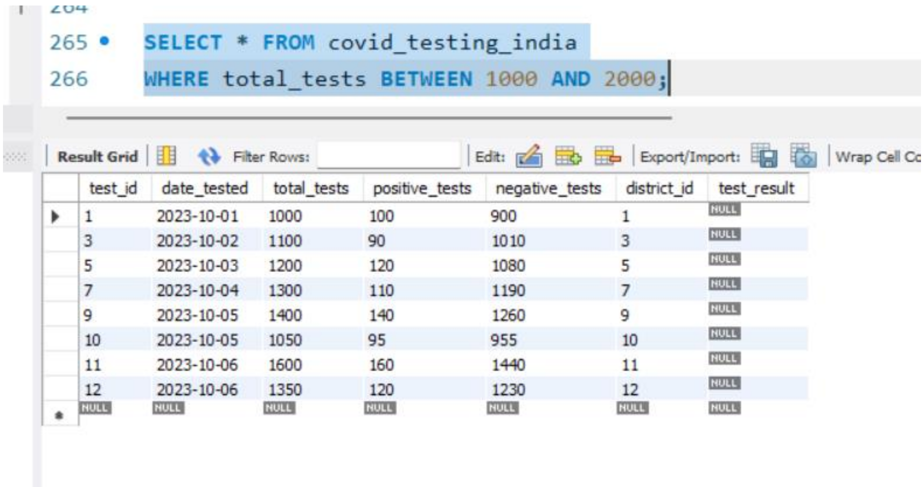


```

SELECT DISTINCT state_name FROM covid_states;

```

state_name
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
Goa
Gujarat
Haryana
Himachal Pradesh
Jharkhand
Karnataka
Maharashtra

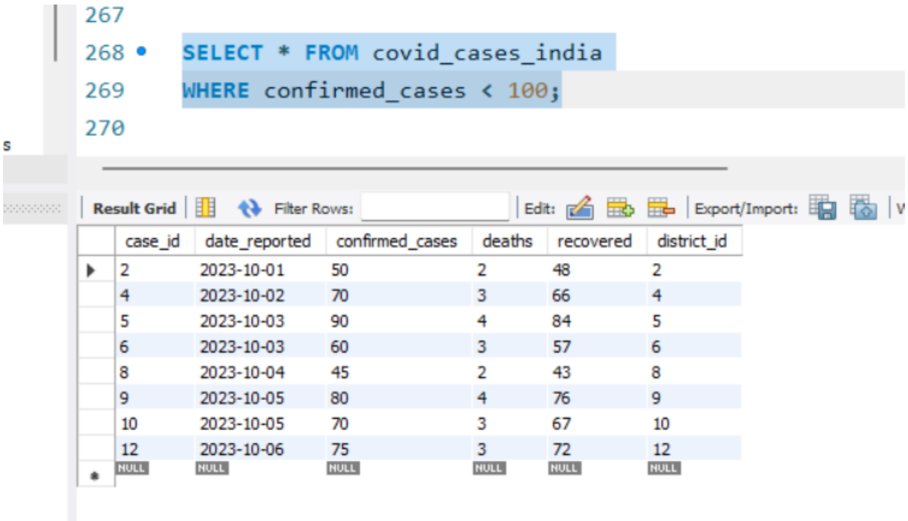
9. Using where clause

a. With Comparision Operator



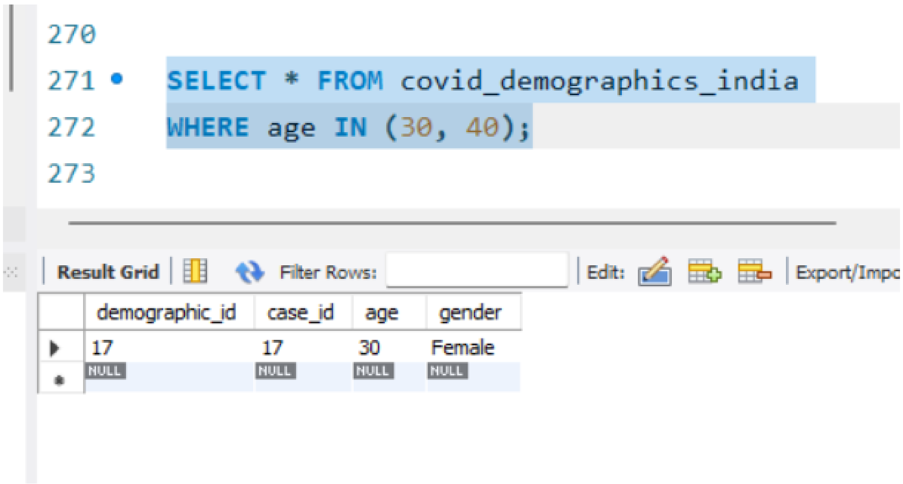
```
SELECT * FROM covid_testing_india
WHERE total_tests BETWEEN 1000 AND 2000;
```

	test_id	date_tested	total_tests	positive_tests	negative_tests	district_id	test_result
▶	1	2023-10-01	1000	100	900	1	NULL
	3	2023-10-02	1100	90	1010	3	NULL
	5	2023-10-03	1200	120	1080	5	NULL
	7	2023-10-04	1300	110	1190	7	NULL
	9	2023-10-05	1400	140	1260	9	NULL
	10	2023-10-05	1050	95	955	10	NULL
	11	2023-10-06	1600	160	1440	11	NULL
	12	2023-10-06	1350	120	1230	12	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL



```
SELECT * FROM covid_cases_india
WHERE confirmed_cases < 100;
```

	case_id	date_reported	confirmed_cases	deaths	recovered	district_id
▶	2	2023-10-01	50	2	48	2
	4	2023-10-02	70	3	66	4
	5	2023-10-03	90	4	84	5
	6	2023-10-03	60	3	57	6
	8	2023-10-04	45	2	43	8
	9	2023-10-05	80	4	76	9
	10	2023-10-05	70	3	67	10
	12	2023-10-06	75	3	72	12
*	NULL	NULL	NULL	NULL	NULL	NULL

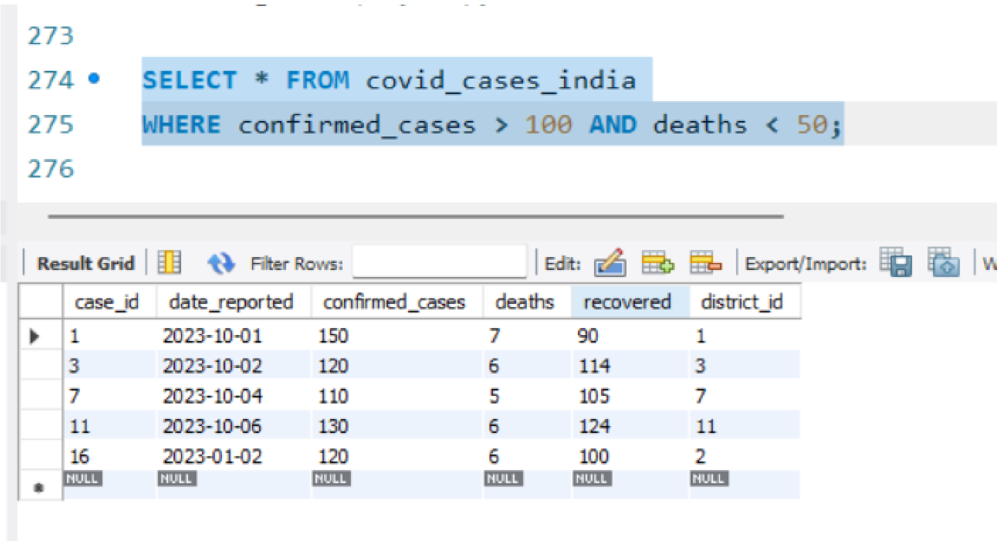


```
SELECT * FROM covid_demographics_india
WHERE age IN (30, 40);
```

	demographic_id	case_id	age	gender
▶	17	17	30	Female
*	NULL	NULL	NULL	NULL

10. Using Logical Operator

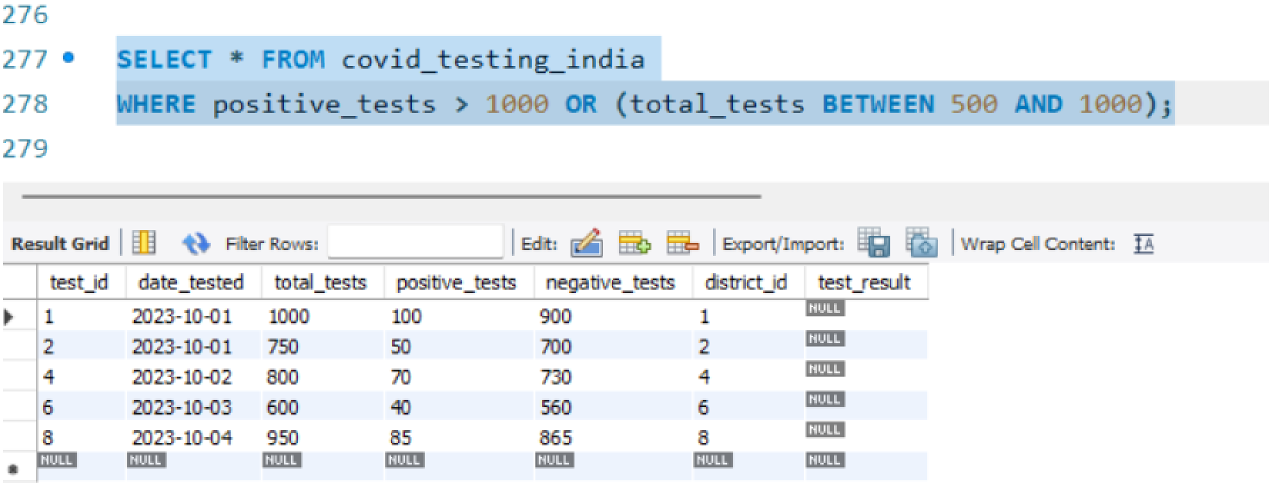
a. Using AND operator



```sql
SELECT * FROM covid_cases_india
WHERE confirmed_cases > 100 AND deaths < 50;
```

case_id	date_reported	confirmed_cases	deaths	recovered	district_id
1	2023-10-01	150	7	90	1
3	2023-10-02	120	6	114	3
7	2023-10-04	110	5	105	7
11	2023-10-06	130	6	124	11
16	2023-01-02	120	6	100	2
NULL	NULL	NULL	NULL	NULL	NULL

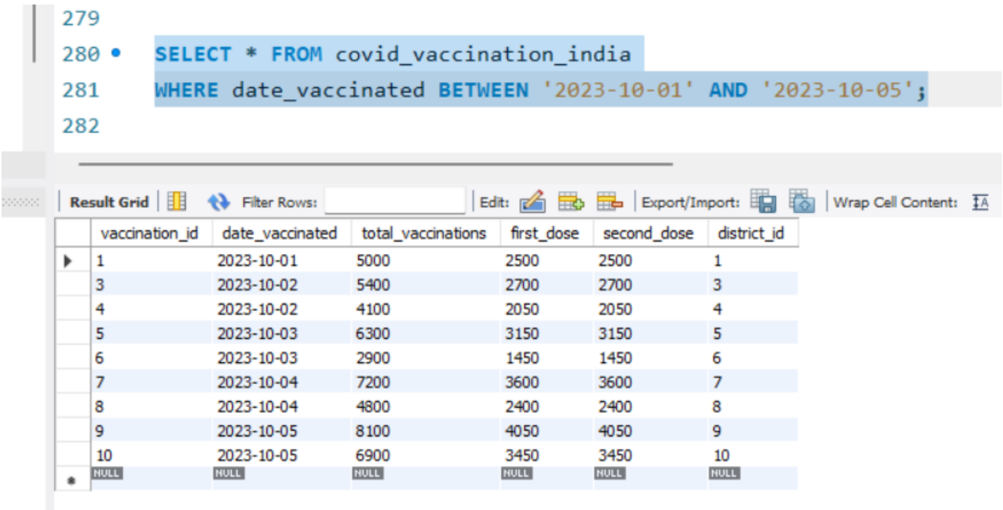
b. Using AND/ OR operator



```sql
SELECT * FROM covid_testing_india
WHERE positive_tests > 1000 OR (total_tests BETWEEN 500 AND 1000);
```

test_id	date_tested	total_tests	positive_tests	negative_tests	district_id	test_result
1	2023-10-01	1000	100	900	1	NULL
2	2023-10-01	750	50	700	2	NULL
4	2023-10-02	800	70	730	4	NULL
6	2023-10-03	600	40	560	6	NULL
8	2023-10-04	950	85	865	8	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

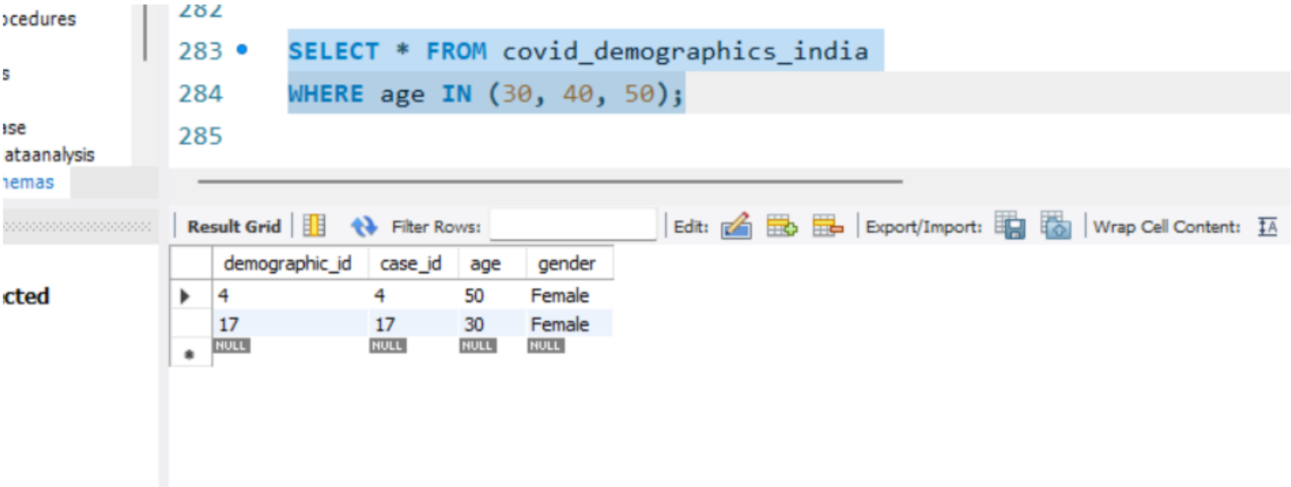
c. Using between clause



```sql
SELECT * FROM covid_vaccination_india
WHERE date_vaccinated BETWEEN '2023-10-01' AND '2023-10-05';
```

vaccination_id	date_vaccinated	total_vaccinations	first_dose	second_dose	district_id
1	2023-10-01	5000	2500	2500	1
3	2023-10-02	5400	2700	2700	3
4	2023-10-02	4100	2050	2050	4
5	2023-10-03	6300	3150	3150	5
6	2023-10-03	2900	1450	1450	6
7	2023-10-04	7200	3600	3600	7
8	2023-10-04	4800	2400	2400	8
9	2023-10-05	8100	4050	4050	9
10	2023-10-05	6900	3450	3450	10
NULL	NULL	NULL	NULL	NULL	NULL

d. Using IN clause

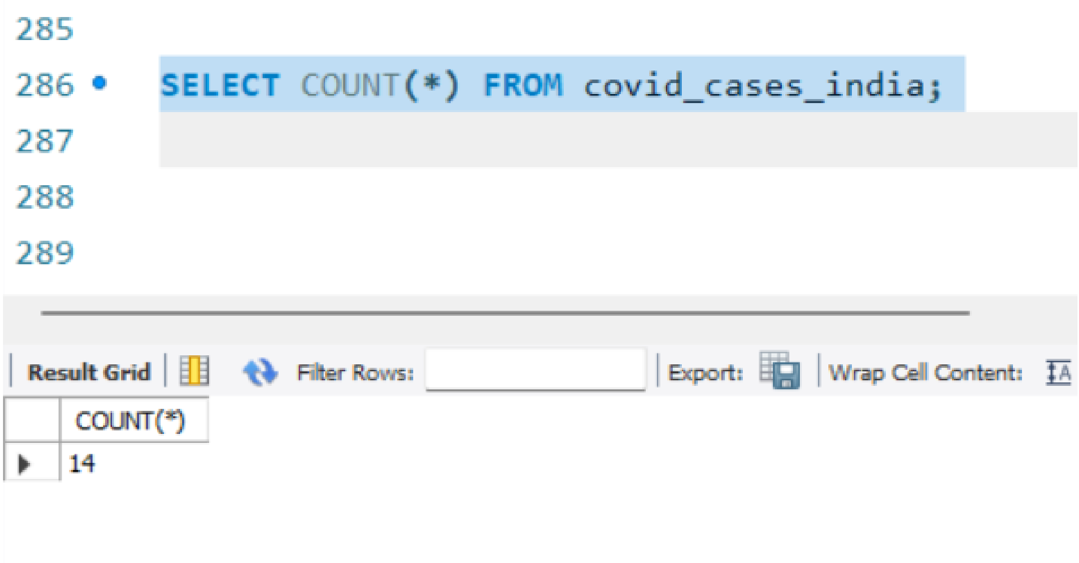


```sql
SELECT * FROM covid_demographics_india
WHERE age IN (30, 40, 50);
```

demographic_id	case_id	age	gender
4	4	50	Female
17	17	30	Female
NULL	NULL	NULL	NULL

11. Aggregate function

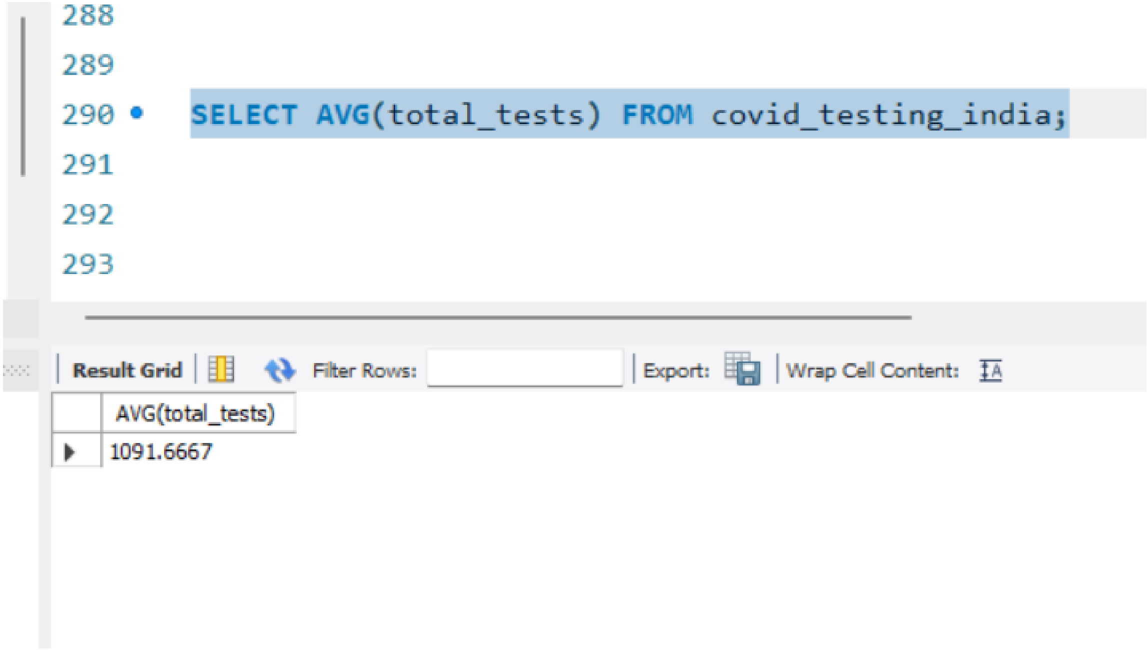
a. Count function



```sql
SELECT COUNT(*) FROM covid_cases_india;
```

```
COUNT(*)
14
```

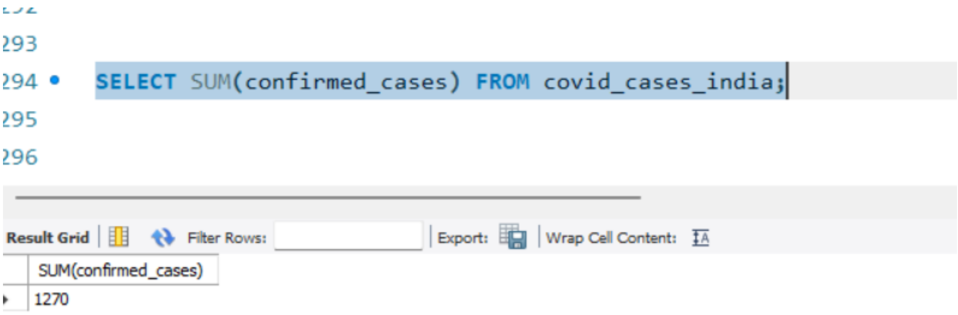
b. Average function



```sql
SELECT AVG(total_tests) FROM covid_testing_india;
```

```
AVG(total_tests)
1091.6667
```

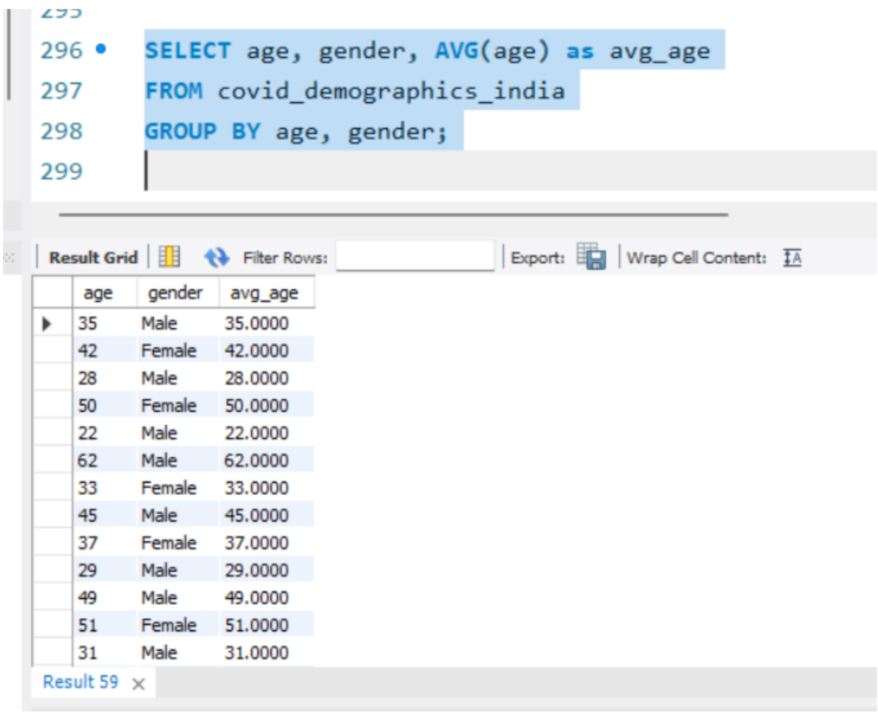
C .Sum Function



```sql
SELECT SUM(confirmed_cases) FROM covid_cases_india;
```

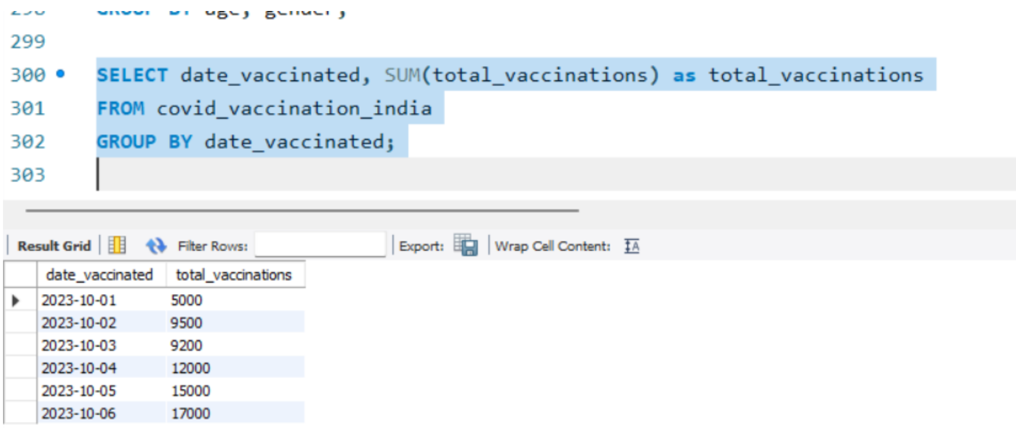
SUM(confirmed_cases)
1270

12. Group By clause



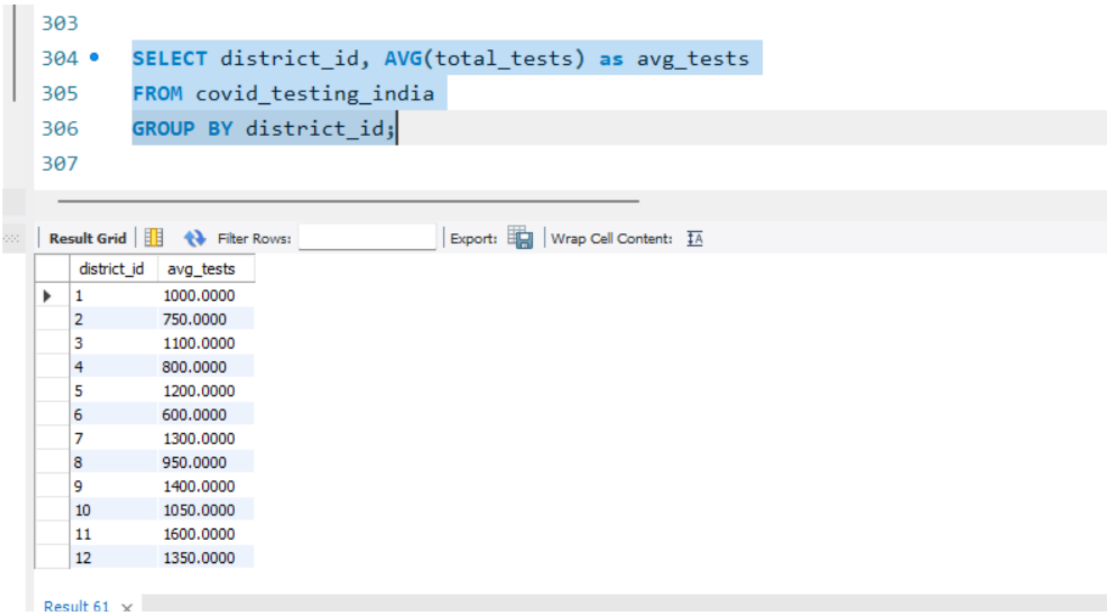
```sql
SELECT age, gender, AVG(age) as avg_age
FROM covid_demographics_india
GROUP BY age, gender;
```

age	gender	avg_age
35	Male	35.0000
42	Female	42.0000
28	Male	28.0000
50	Female	50.0000
22	Male	22.0000
62	Male	62.0000
33	Female	33.0000
45	Male	45.0000
37	Female	37.0000
29	Male	29.0000
49	Male	49.0000
51	Female	51.0000
31	Male	31.0000



```sql
GROUP BY age, gender;

SELECT date_vaccinated, SUM(total_vaccinations) as total_vaccinations
FROM covid_vaccination_india
GROUP BY date_vaccinated;
```

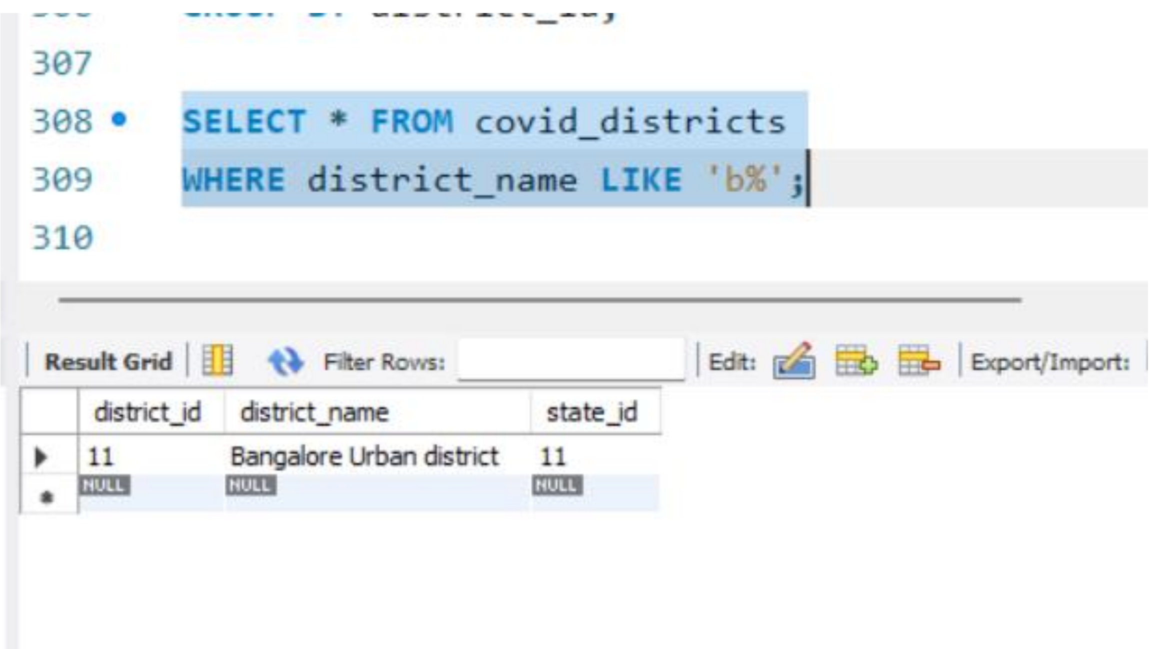
date_vaccinated	total_vaccinations
2023-10-01	5000
2023-10-02	9500
2023-10-03	9200
2023-10-04	12000
2023-10-05	15000
2023-10-06	17000



```sql
SELECT district_id, AVG(total_tests) as avg_tests
FROM covid_testing_india
GROUP BY district_id;
```

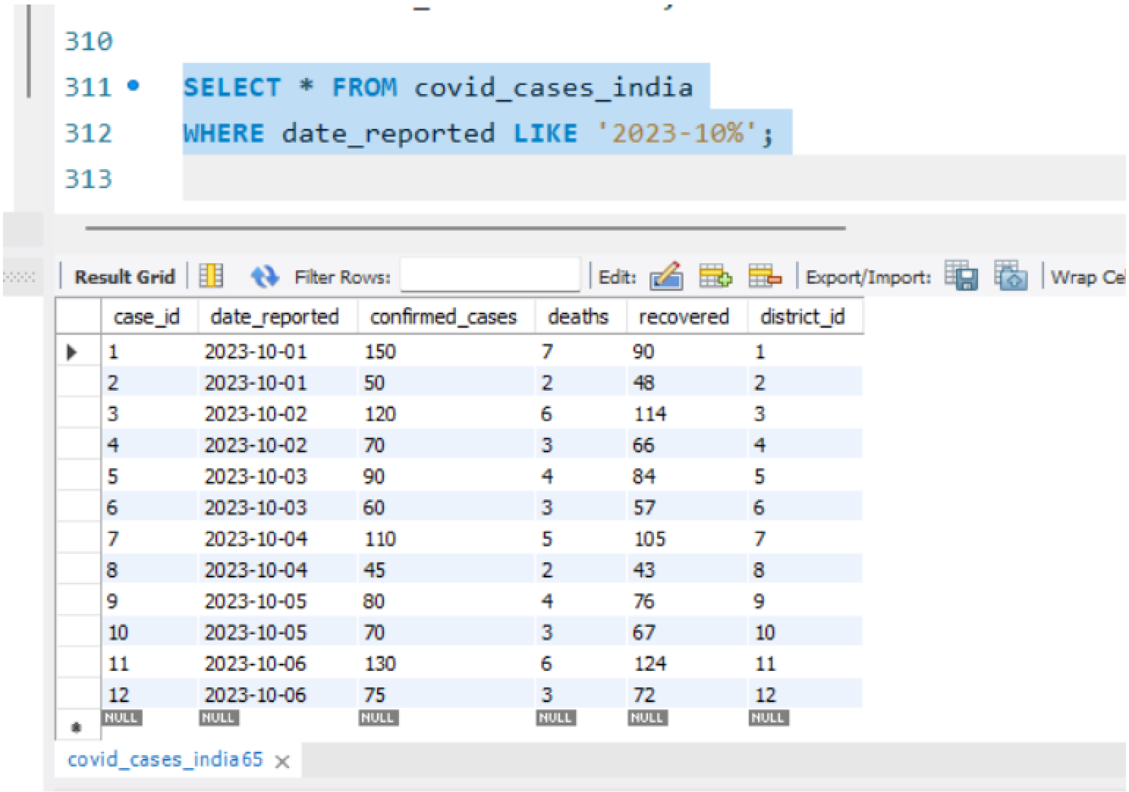
district_id	avg_tests
1	1000.0000
2	750.0000
3	1100.0000
4	800.0000
5	1200.0000
6	600.0000
7	1300.0000
8	950.0000
9	1400.0000
10	1050.0000
11	1600.0000
12	1350.0000

Like operator



```sql
SELECT * FROM covid_districts
WHERE district_name LIKE 'b%';
```

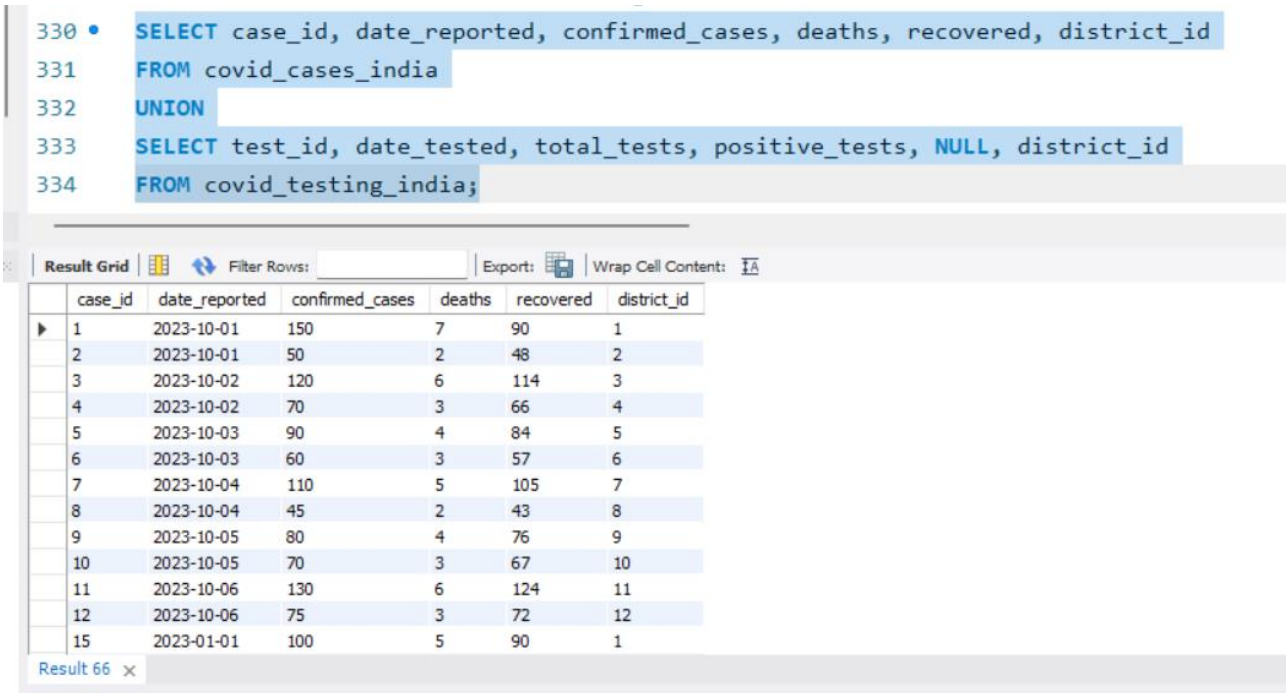
district_id	district_name	state_id
11	Bangalore Urban district	11
NULL	NULL	NULL



```sql
SELECT * FROM covid_cases_india
WHERE date_reported LIKE '2023-10%';
```

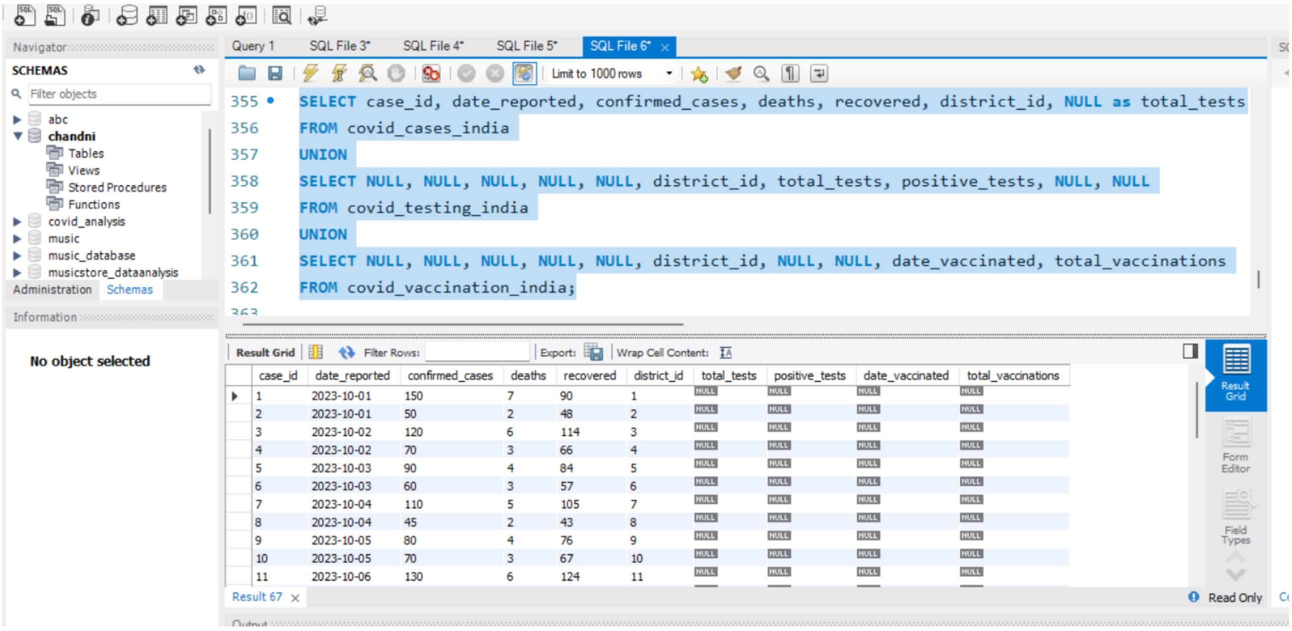
case_id	date_reported	confirmed_cases	deaths	recovered	district_id
1	2023-10-01	150	7	90	1
2	2023-10-01	50	2	48	2
3	2023-10-02	120	6	114	3
4	2023-10-02	70	3	66	4
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12
NULL	NULL	NULL	NULL	NULL	NULL

Union



```sql
SELECT case_id, date_reported, confirmed_cases, deaths, recovered, district_id
FROM covid_cases_india
UNION
SELECT test_id, date_tested, total_tests, positive_tests, NULL, district_id
FROM covid_testing_india;
```

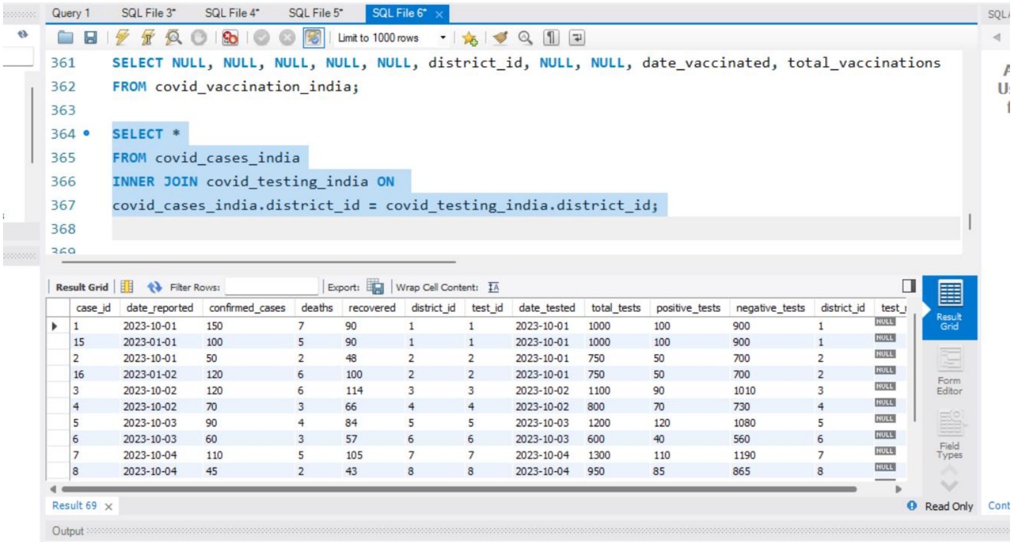
case_id	date_reported	confirmed_cases	deaths	recovered	district_id
1	2023-10-01	150	7	90	1
2	2023-10-01	50	2	48	2
3	2023-10-02	120	6	114	3
4	2023-10-02	70	3	66	4
5	2023-10-03	90	4	84	5
6	2023-10-03	60	3	57	6
7	2023-10-04	110	5	105	7
8	2023-10-04	45	2	43	8
9	2023-10-05	80	4	76	9
10	2023-10-05	70	3	67	10
11	2023-10-06	130	6	124	11
12	2023-10-06	75	3	72	12
15	2023-01-01	100	5	90	1



```sql
SELECT case_id, date_reported, confirmed_cases, deaths, recovered, district_id, NULL as total_tests
FROM covid_cases_india
UNION
SELECT NULL, NULL, NULL, NULL, NULL, district_id, total_tests, positive_tests, NULL, NULL
FROM covid_testing_india
UNION
SELECT NULL, NULL, NULL, NULL, NULL, district_id, NULL, NULL, date_vaccinated, total_vaccinations
FROM covid_vaccination_india;
```

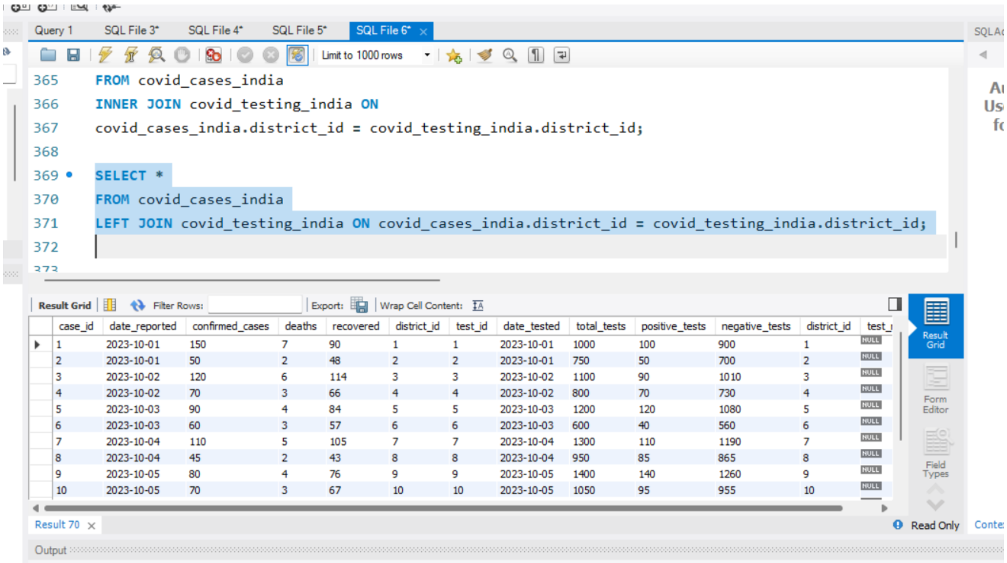
case_id	date_reported	confirmed_cases	deaths	recovered	district_id	total_tests	positive_tests	date_vaccinated	total_vaccinations
1	2023-10-01	150	7	90	1	NULL	NULL	NULL	NULL
2	2023-10-01	50	2	48	2	NULL	NULL	NULL	NULL
3	2023-10-02	120	6	114	3	NULL	NULL	NULL	NULL
4	2023-10-02	70	3	66	4	NULL	NULL	NULL	NULL
5	2023-10-03	90	4	84	5	NULL	NULL	NULL	NULL
6	2023-10-03	60	3	57	6	NULL	NULL	NULL	NULL
7	2023-10-04	110	5	105	7	NULL	NULL	NULL	NULL
8	2023-10-04	45	2	43	8	NULL	NULL	NULL	NULL
9	2023-10-05	80	4	76	9	NULL	NULL	NULL	NULL
10	2023-10-05	70	3	67	10	NULL	NULL	NULL	NULL
11	2023-10-06	130	6	124	11	NULL	NULL	NULL	NULL

Joins



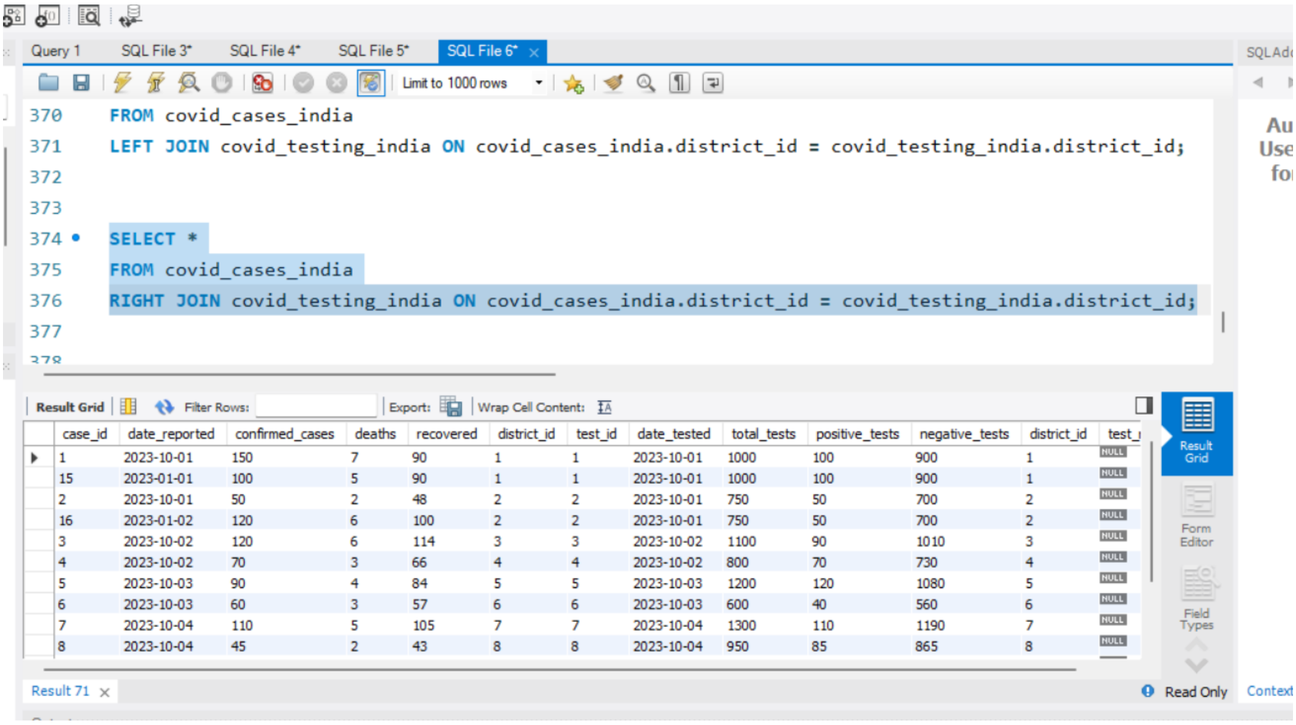
```sql
SELECT NULL, NULL, NULL, NULL, NULL, district_id, NULL, NULL, date_vaccinated, total_vaccinations
FROM covid_vaccination_india;

SELECT *
FROM covid_cases_india
INNER JOIN covid_testing_india ON
covid_cases_india.district_id = covid_testing_india.district_id;
```

	case_id	date_reported	confirmed_cases	deaths	recovered	district_id	test_id	date_tested	total_tests	positive_tests	negative_tests	district_id	test_
▶	1	2023-10-01	150	7	90	1	1	2023-10-01	1000	100	900	1	NULL
	15	2023-01-01	100	5	90	1	1	2023-10-01	1000	100	900	1	NULL
	2	2023-10-01	50	2	48	2	2	2023-10-01	750	50	700	2	NULL
	16	2023-01-02	120	6	100	2	2	2023-10-01	750	50	700	2	NULL
	3	2023-10-02	120	6	114	3	3	2023-10-02	1100	90	1010	3	NULL
	4	2023-10-02	70	3	66	4	4	2023-10-02	800	70	730	4	NULL
	5	2023-10-03	90	4	84	5	5	2023-10-03	1200	120	1080	5	NULL
	6	2023-10-03	60	3	57	6	6	2023-10-03	600	40	560	6	NULL
	7	2023-10-04	110	5	105	7	7	2023-10-04	1300	110	1190	7	NULL
	8	2023-10-04	45	2	43	8	8	2023-10-04	950	85	865	8	NULL



```sql
FROM covid_cases_india
INNER JOIN covid_testing_india ON
covid_cases_india.district_id = covid_testing_india.district_id;

SELECT *
FROM covid_cases_india
LEFT JOIN covid_testing_india ON covid_cases_india.district_id = covid_testing_india.district_id;
```

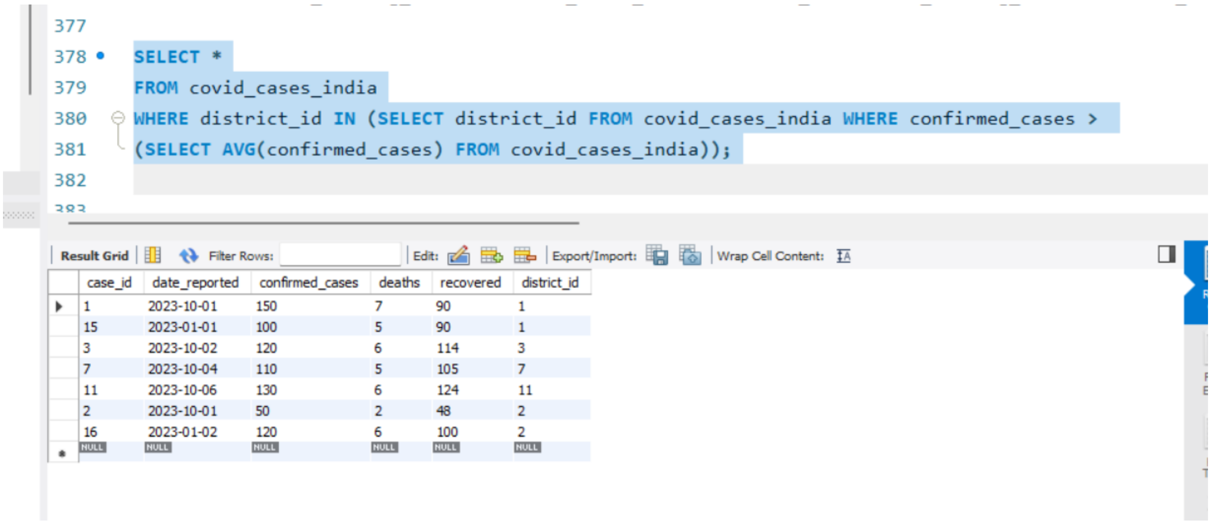
	case_id	date_reported	confirmed_cases	deaths	recovered	district_id	test_id	date_tested	total_tests	positive_tests	negative_tests	district_id	test_
▶	1	2023-10-01	150	7	90	1	1	2023-10-01	1000	100	900	1	NULL
	2	2023-10-01	50	2	48	2	2	2023-10-01	750	50	700	2	NULL
	3	2023-10-02	120	6	114	3	3	2023-10-02	1100	90	1010	3	NULL
	4	2023-10-02	70	3	66	4	4	2023-10-02	800	70	730	4	NULL
	5	2023-10-03	90	4	84	5	5	2023-10-03	1200	120	1080	5	NULL
	6	2023-10-03	60	3	57	6	6	2023-10-03	600	40	560	6	NULL
	7	2023-10-04	110	5	105	7	7	2023-10-04	1300	110	1190	7	NULL
	8	2023-10-04	45	2	43	8	8	2023-10-04	950	85	865	8	NULL
	9	2023-10-05	80	4	76	9	9	2023-10-05	1400	140	1260	9	NULL
	10	2023-10-05	70	3	67	10	10	2023-10-05	1050	95	955	10	NULL



```
FROM covid_cases_india
LEFT JOIN covid_testing_india ON covid_cases_india.district_id = covid_testing_india.district_id;


SELECT *
FROM covid_cases_india
RIGHT JOIN covid_testing_india ON covid_cases_india.district_id = covid_testing_india.district_id;
```

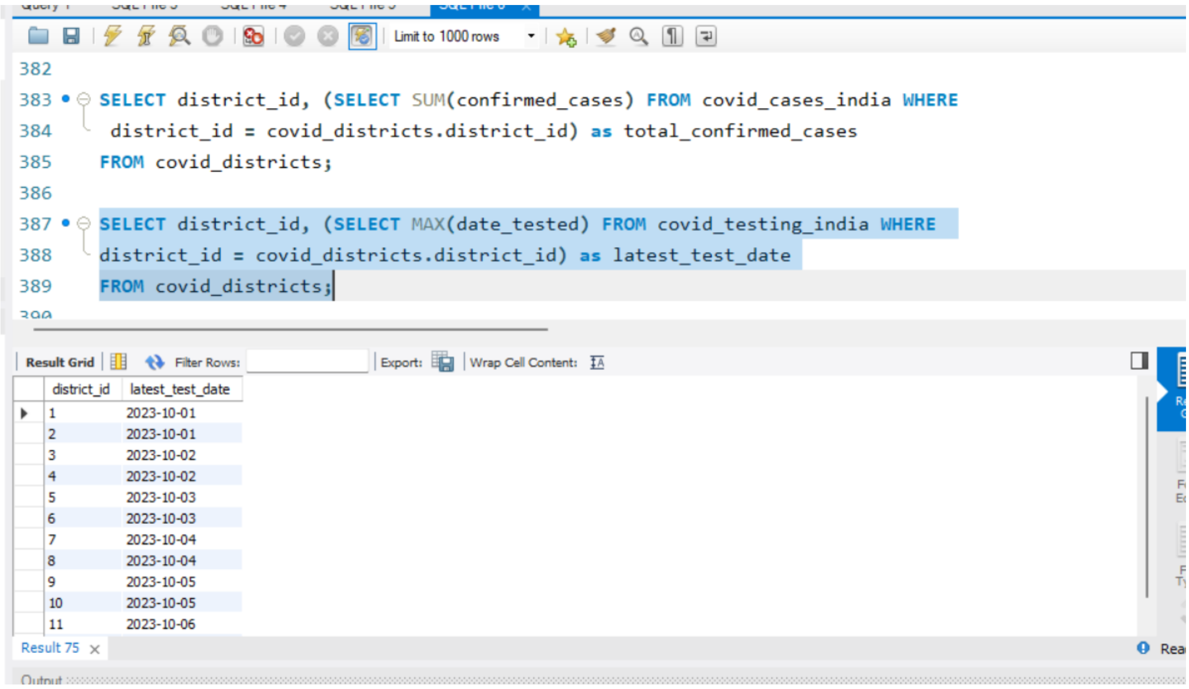
case_id	date_reported	confirmed_cases	deaths	recovered	district_id	test_id	date_tested	total_tests	positive_tests	negative_tests	district_id	test_
1	2023-10-01	150	7	90	1	1	2023-10-01	1000	100	900	1	NULL
15	2023-01-01	100	5	90	1	1	2023-10-01	1000	100	900	1	NULL
2	2023-10-01	50	2	48	2	2	2023-10-01	750	50	700	2	NULL
16	2023-01-02	120	6	100	2	2	2023-10-01	750	50	700	2	NULL
3	2023-10-02	120	6	114	3	3	2023-10-02	1100	90	1010	3	NULL
4	2023-10-02	70	3	66	4	4	2023-10-02	800	70	730	4	NULL
5	2023-10-03	90	4	84	5	5	2023-10-03	1200	120	1080	5	NULL
6	2023-10-03	60	3	57	6	6	2023-10-03	600	40	560	6	NULL
7	2023-10-04	110	5	105	7	7	2023-10-04	1300	110	1190	7	NULL
8	2023-10-04	45	2	43	8	8	2023-10-04	950	85	865	8	NULL

Subquery



```sql
SELECT *
FROM covid_cases_india
WHERE district_id IN (SELECT district_id FROM covid_cases_india WHERE confirmed_cases >
(SELECT AVG(confirmed_cases) FROM covid_cases_india));
```

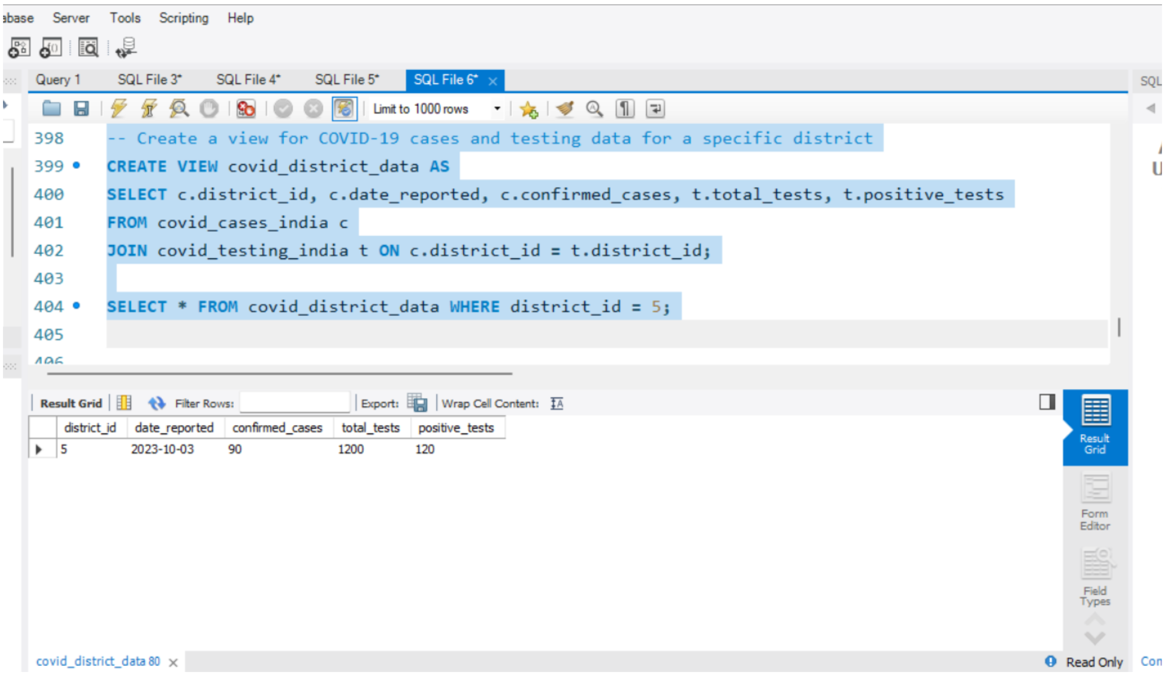
case_id	date_reported	confirmed_cases	deaths	recovered	district_id
1	2023-10-01	150	7	90	1
15	2023-01-01	100	5	90	1
3	2023-10-02	120	6	114	3
7	2023-10-04	110	5	105	7
11	2023-10-06	130	6	124	11
2	2023-10-01	50	2	48	2
16	2023-01-02	120	6	100	2
NULL	NULL	NULL	NULL	NULL	NULL



```sql
SELECT district_id, (SELECT SUM(confirmed_cases) FROM covid_cases_india WHERE
 district_id = covid_districts.district_id) as total_confirmed_cases
FROM covid_districts;

SELECT district_id, (SELECT MAX(date_tested) FROM covid_testing_india WHERE
district_id = covid_districts.district_id) as latest_test_date
FROM covid_districts;
```

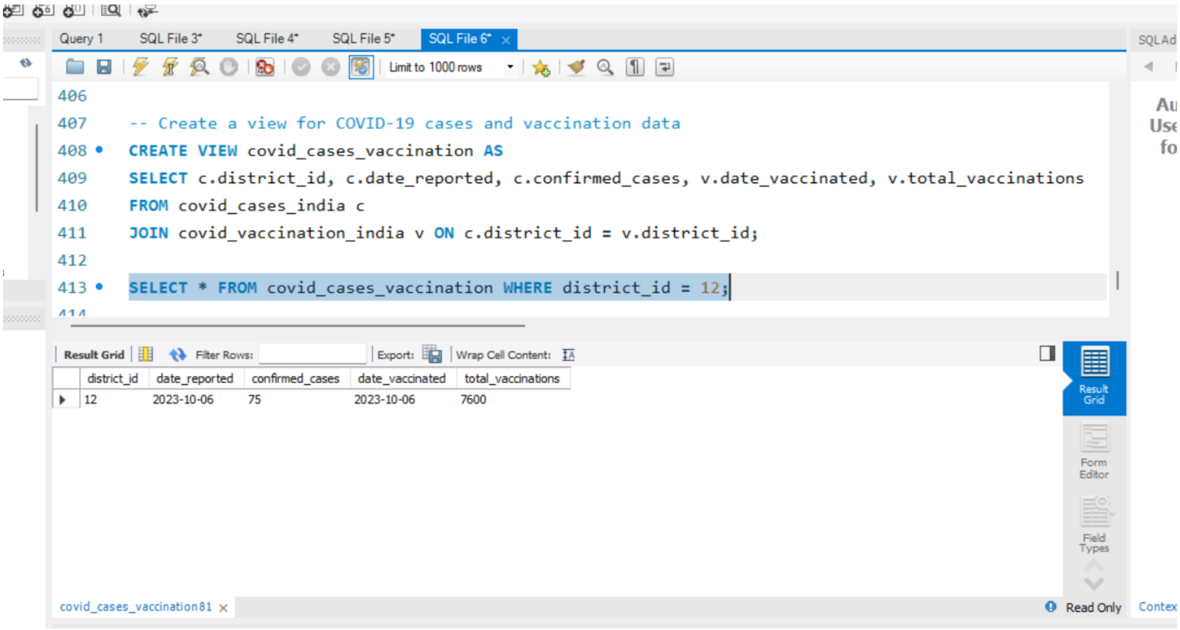
district_id	latest_test_date
1	2023-10-01
2	2023-10-01
3	2023-10-02
4	2023-10-02
5	2023-10-03
6	2023-10-03
7	2023-10-04
8	2023-10-04
9	2023-10-05
10	2023-10-05
11	2023-10-06

Views



```
-- Create a view for COVID-19 cases and testing data for a specific district
CREATE VIEW covid_district_data AS
SELECT c.district_id, c.date_reported, c.confirmed_cases, t.total_tests, t.positive_tests
FROM covid_cases_india c
JOIN covid_testing_india t ON c.district_id = t.district_id;

SELECT * FROM covid_district_data WHERE district_id = 5;
```

district_id	date_reported	confirmed_cases	total_tests	positive_tests
5	2023-10-03	90	1200	120



```
-- Create a view for COVID-19 cases and vaccination data
CREATE VIEW covid_cases_vaccination AS
SELECT c.district_id, c.date_reported, c.confirmed_cases, v.date_vaccinated, v.total_vaccinations
FROM covid_cases_india c
JOIN covid_vaccination_india v ON c.district_id = v.district_id;

SELECT * FROM covid_cases_vaccination WHERE district_id = 12;
```

district_id	date_reported	confirmed_cases	date_vaccinated	total_vaccinations
12	2023-10-06	75	2023-10-06	7600