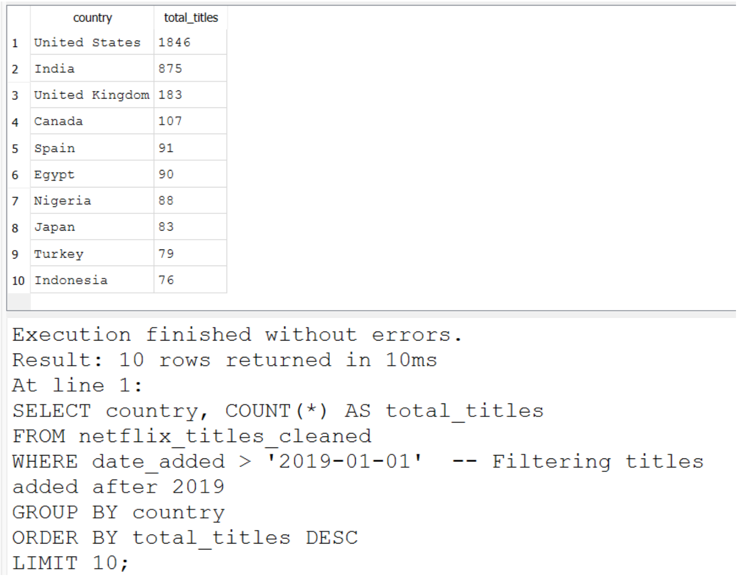
1. **Query using SELECT, WHERE, ORDER BY, GROUP BY**

```
SELECT country, COUNT(*) AS total_titles
FROM netflix_titles_cleaned
WHERE date_added > '2019-01-01'  -- Filtering titles added after 2019
GROUP BY country
ORDER BY total_titles DESC
LIMIT 10;
```

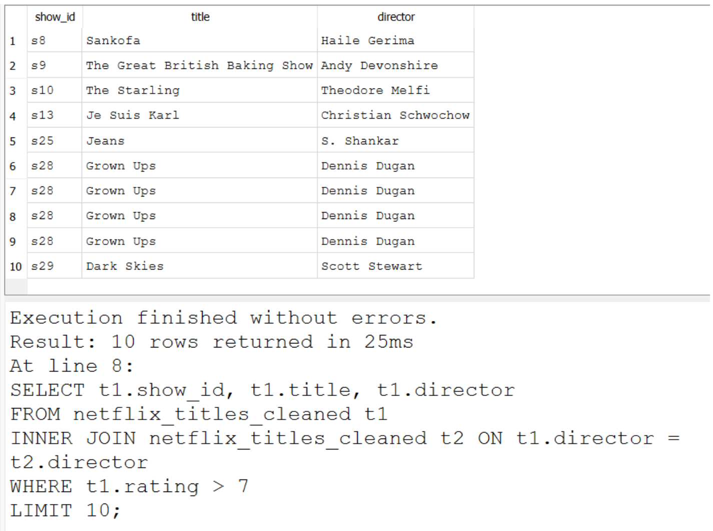


	country	total_titles
1	United States	1846
2	India	875
3	United Kingdom	183
4	Canada	107
5	Spain	91
6	Egypt	90
7	Nigeria	88
8	Japan	83
9	Turkey	79
10	Indonesia	76

```
Execution finished without errors.
Result: 10 rows returned in 10ms
At line 1:
SELECT country, COUNT(*) AS total_titles
FROM netflix_titles_cleaned
WHERE date_added > '2019-01-01'  -- Filtering titles
added after 2019
GROUP BY country
ORDER BY total_titles DESC
LIMIT 10;
```

2. **Query using JOINS (INNER JOIN, LEFT JOIN, RIGHT JOIN)**

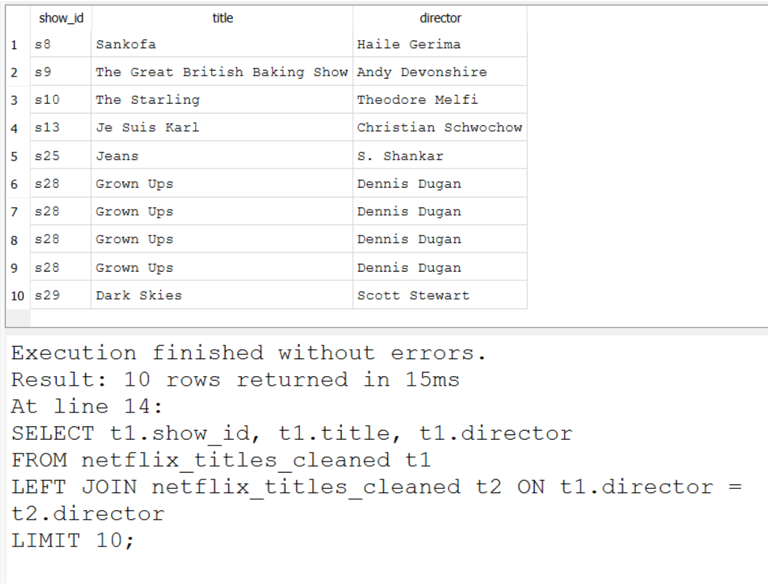
```
SELECT t1.show_id, t1.title, t1.director
FROM netflix_titles_cleaned t1
INNER JOIN netflix_titles_cleaned t2 ON t1.director = t2.director
WHERE t1.rating > 7
LIMIT 10;
```



	show_id	title	director
1	s8	Sankofa	Haile Gerima
2	s9	The Great British Baking Show	Andy Devonshire
3	s10	The Starling	Theodore Melfi
4	s13	Je Suis Karl	Christian Schwochow
5	s25	Jeans	S. Shankar
6	s28	Grown Ups	Dennis Dugan
7	s28	Grown Ups	Dennis Dugan
8	s28	Grown Ups	Dennis Dugan
9	s28	Grown Ups	Dennis Dugan
10	s29	Dark Skies	Scott Stewart

```
Execution finished without errors.
Result: 10 rows returned in 25ms
At line 8:
SELECT t1.show_id, t1.title, t1.director
FROM netflix_titles_cleaned t1
INNER JOIN netflix_titles_cleaned t2 ON t1.director =
t2.director
WHERE t1.rating > 7
LIMIT 10;
```

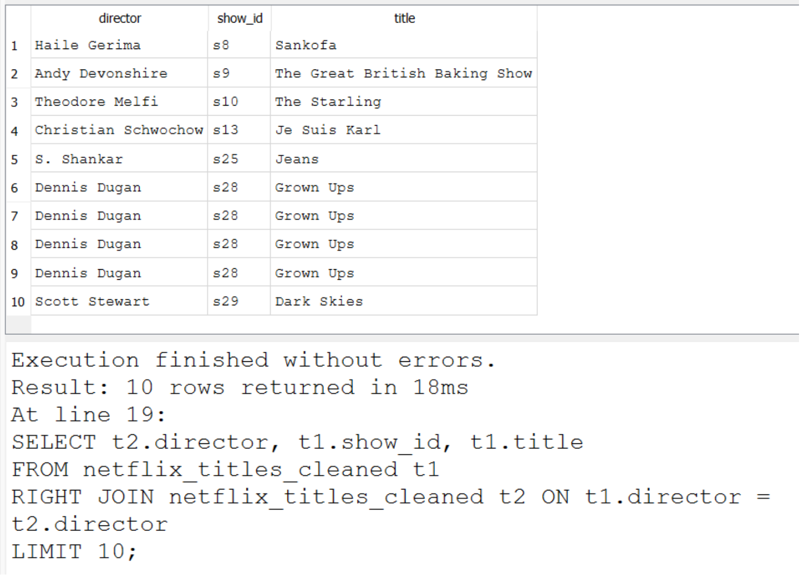
```
SELECT t1.show_id, t1.title, t1.director
FROM netflix_titles_cleaned t1
LEFT JOIN netflix_titles_cleaned t2 ON t1.director = t2.director
LIMIT 10;
```



	show_id	title	director
1	s8	Sankofa	Haile Gerima
2	s9	The Great British Baking Show	Andy Devonshire
3	s10	The Starling	Theodore Melfi
4	s13	Je Suis Karl	Christian Schwochow
5	s25	Jeans	S. Shankar
6	s28	Grown Ups	Dennis Dugan
7	s28	Grown Ups	Dennis Dugan
8	s28	Grown Ups	Dennis Dugan
9	s28	Grown Ups	Dennis Dugan
10	s29	Dark Skies	Scott Stewart

```
Execution finished without errors.
Result: 10 rows returned in 15ms
At line 14:
SELECT t1.show_id, t1.title, t1.director
FROM netflix_titles_cleaned t1
LEFT JOIN netflix_titles_cleaned t2 ON t1.director =
t2.director
LIMIT 10;
```

```
SELECT t2.director, t1.show_id, t1.title
FROM netflix_titles_cleaned t1
RIGHT JOIN netflix_titles_cleaned t2 ON t1.director = t2.director
LIMIT 10;
```



	director	show_id	title
1	Haile Gerima	s8	Sankofa
2	Andy Devonshire	s9	The Great British Baking Show
3	Theodore Melfi	s10	The Starling
4	Christian Schwochow	s13	Je Suis Karl
5	S. Shankar	s25	Jeans
6	Dennis Dugan	s28	Grown Ups
7	Dennis Dugan	s28	Grown Ups
8	Dennis Dugan	s28	Grown Ups
9	Dennis Dugan	s28	Grown Ups
10	Scott Stewart	s29	Dark Skies

```
Execution finished without errors.
Result: 10 rows returned in 18ms
At line 19:
SELECT t2.director, t1.show_id, t1.title
FROM netflix_titles_cleaned t1
RIGHT JOIN netflix_titles_cleaned t2 ON t1.director =
t2.director
LIMIT 10;
```

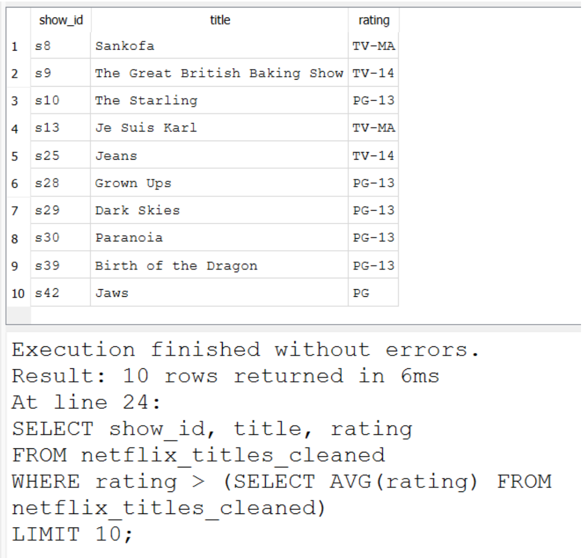
3. Query using Subqueries

```
SELECT show_id, title, rating
```

FROM netflix_titles_cleaned

WHERE rating > (SELECT AVG(rating) FROM netflix_titles_cleaned)

LIMIT 10;



	show_id	title	rating
1	s8	Sankofa	TV-MA
2	s9	The Great British Baking Show	TV-14
3	s10	The Starling	PG-13
4	s13	Je Suis Karl	TV-MA
5	s25	Jeans	TV-14
6	s28	Grown Ups	PG-13
7	s29	Dark Skies	PG-13
8	s30	Paranoia	PG-13
9	s39	Birth of the Dragon	PG-13
10	s42	Jaws	PG

```
Execution finished without errors.
Result: 10 rows returned in 6ms
At line 24:
SELECT show_id, title, rating
FROM netflix_titles_cleaned
WHERE rating > (SELECT AVG(rating) FROM
netflix_titles_cleaned)
LIMIT 10;
```

4. **Query using Aggregate Functions (SUM, AVG)**

SELECT type, AVG(duration) AS avg_duration

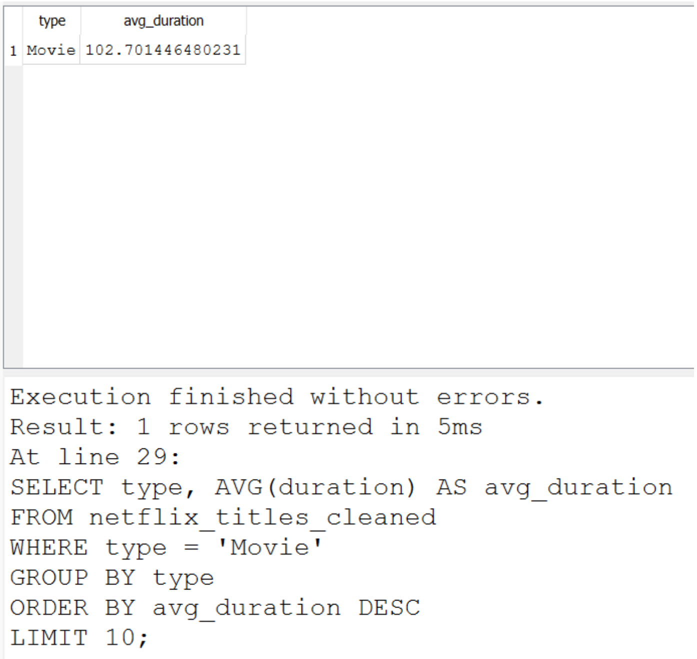
FROM netflix_titles_cleaned

WHERE type = 'Movie'

GROUP BY type

ORDER BY avg_duration DESC

LIMIT 10;



	type	avg_duration
1	Movie	102.701446480231

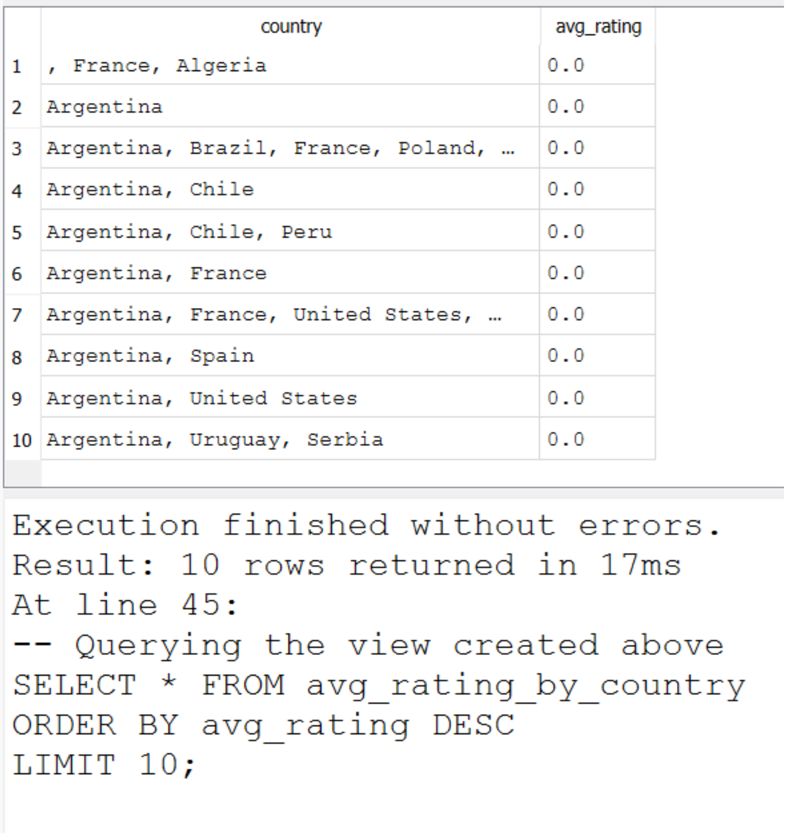
```
Execution finished without errors.
Result: 1 rows returned in 5ms
At line 29:
SELECT type, AVG(duration) AS avg_duration
FROM netflix_titles_cleaned
WHERE type = 'Movie'
GROUP BY type
ORDER BY avg_duration DESC
LIMIT 10;
```

5. **Create a View for Analysis & Optimize Queries with Indexes**

```
-- Create a view for average rating by country
CREATE VIEW avg_rating_by_country AS
SELECT country, AVG(rating) AS avg_rating
FROM netflix_titles_cleaned
GROUP BY country;

-- Optimizing queries with an index on the 'release_year' column
CREATE INDEX idx_release_year ON netflix_titles_cleaned(release_year);

-- Querying the view created above
SELECT * FROM avg_rating_by_country
ORDER BY avg_rating DESC
LIMIT 10;
```



	country	avg_rating
1	, France, Algeria	0.0
2	Argentina	0.0
3	Argentina, Brazil, France, Poland, …	0.0
4	Argentina, Chile	0.0
5	Argentina, Chile, Peru	0.0
6	Argentina, France	0.0
7	Argentina, France, United States, …	0.0
8	Argentina, Spain	0.0
9	Argentina, United States	0.0
10	Argentina, Uruguay, Serbia	0.0

```
Execution finished without errors.
Result: 10 rows returned in 17ms
At line 45:
-- Querying the view created above
SELECT * FROM avg_rating_by_country
ORDER BY avg_rating DESC
LIMIT 10;
```