

# PostgreSQL UNION

Query Editor Query History Explain Messages

```
1 --The following statements create two tables: top_rated_films and most_popular_films, and insert data into these tables:
2
3 DROP TABLE IF EXISTS top_rated_films;
4 CREATE TABLE top_rated_films(
5     title VARCHAR NOT NULL,
6     release_year SMALLINT
7 );
8
9 DROP TABLE IF EXISTS most_popular_films;
10 CREATE TABLE most_popular_films(
11     title VARCHAR NOT NULL,
12     release_year SMALLINT
13 );
14
15 INSERT INTO
16     top_rated_films(title,release_year)
17 VALUES
18     ('The Shawshank Redemption',1994),
19     ('The Godfather',1972),
20     ('12 Angry Men',1957);
21
22 INSERT INTO
23     most_popular_films(title,release_year)
24 VALUES
25     ('An American Pickle',2020),
26     ('The Godfather',1972);
```

Data Output Notifications

store_id	count
smallint	bigint
1	326

dvdrental/postgres@PostgreSQL

Query Editor Query History Explain Messages

```
1 --The following shows the data from the top_rated_films table:
2
3 SELECT * FROM top_rated_films;
```

Data Output Notifications

	title	release_year
	character varying	smallint
1	The Shawshank Redemption	1994
2	The Godfather	1972
3	12 Angry Men	1957

dvdrental/postgres@PostgreSQL

Query Editor Query History Explain Messages

```
1 --The following statement returns the data from the most_popular_films table:
2
3 SELECT * FROM most_popular_films;
```

Data Output Notifications

	title	release_year
	character varying	smallint
1	An American Pickle	2020
2	The Godfather	1972
3	Greyhound	2020

dvdrental/postgres@PostgreSQL

[Query Editor](#)
[Query History](#)
[Explain](#)
[Messages](#)

```

1  --The following statement uses the UNION operator to combine data from both tables:
2
3  SELECT * FROM topRatedFilms
4  UNION
5  SELECT * FROM mostPopularFilms;

```

Data Output

Notifications

	title	release_year
	character varying	smallint
1	An American Pickle	2020
2	Greyhound	2020
3	The Shawshank Redemption	1994
4	The Godfather	1972
5	12 Angry Men	1957

```

1  --The following statement uses the UNION ALL operator to combine result sets from the topRatedFilms and mostPopularFilms tables:
2
3  SELECT * FROM topRatedFilms
4  UNION ALL
5  SELECT * FROM mostPopularFilms;

```

Data Output

Notifications

	title	release_year
	character varying	smallint
1	12 Angry Men	1957
2	An American Pickle	2020
3	Greyhound	2020
4	The Godfather	1972
5	The Godfather	1972
6	The Shawshank Redemption	1994

dvdrental/postgres@PostgreSQL

[Query Editor](#)
[Query History](#)
[Explain](#)
[Messages](#)

```

1  --To sort the result returned by the UNION operator, you place the ORDER BY clause end of the last query like this:
2
3  SELECT * FROM topRatedFilms
4  UNION ALL
5  SELECT * FROM mostPopularFilms
6  ORDER BY title;

```

Data Output

Notifications

	title	release_year
	character varying	smallint
1	12 Angry Men	1957
2	An American Pickle	2020
3	Greyhound	2020
4	The Godfather	1972
5	The Godfather	1972
6	The Shawshank Redemption	1994