

Query Query History

```
1 ---1. Display the customers name that share the same address (eg husband and wife)---
2 v SELECT c.first_name, c.last_name, a.address
3 FROM address AS a
4 JOIN customer AS c ON a.address_id = c.address_id
5 GROUP BY c.first_name, c.last_name, a.address
6 HAVING COUNT (a.address)>1;
7 ---There are no customers with the same address---
8
```

Data Output Messages Notifications



first_name character varying (45) 🔒	last_name character varying (45) 🔒	address character varying (50) 🔒
--	---------------------------------------	-------------------------------------

Query Query History

```
9 ---2. What is the name of the customer with the highest total payments---
10 ✓ SELECT p.customer_id, c.first_name, c.last_name, SUM(p.amount) AS highest_payments
11 FROM payment p
12 LEFT JOIN customer c ON p.customer_id = c.customer_id
13 GROUP BY p.customer_id, c.first_name, c.last_name
14 ORDER BY highest_payments DESC LIMIT 1;
15 ---Eleanor hunt is the customer with the highest total payments = 211.55---
16
```

Data Output Messages Notifications



	customer_id smallint	first_name character varying (45)	last_name character varying (45)	highest_payments numeric
1	148	Eleanor	Hunt	211.55




Query Query History

```
17 ---3. What is the movie(s) that was rented the most---
18 ✓ SELECT i.film_id, f.title, COUNT (i.film_id) AS rents
19 FROM inventory i
20 JOIN rental AS r ON i.inventory_id = r.inventory_id
21 JOIN film AS f ON i.film_id = f.film_id
22 GROUP BY i.film_id, f.title
23 HAVING COUNT (i.film_id)> 1
24 ORDER BY rents DESC LIMIT 2;
25 ---Bucket Brotherhood is the movie with the most rental at 34--
```

Data Output Messages Notifications



SQL

	film_id smallint 	title character varying (255) 	rents bigint 
1	103	Bucket Brotherhood	34
2	738	Rocketeer Mother	33

35

36

37

38

39

40

41

42

43

5. Which movies have not been rented so far--

SELECT f.film_id, f.title

FROM film AS f

LEFT JOIN inventory AS i ON f.film_id = i.film_id

LEFT JOIN rental AS r ON i.inventory_id = r.inventory_id

WHERE r.rental_id IS null

ORDER BY f.title;

43 movies have not been rented so far---



SQL

	film_id	title
	[PK] integer	character varying (255)
1	1	Academy Dinosaur
2	14	Alice Fantasia
3	33	Apollo Teen
4	36	Argonauts Town
5	38	Ark Ridgemont
6	41	Arsenic Independence
7	87	Boondock Ballroom
8	108	Butch Panther

Query Query History

```
43
44 ---6. Which customers have not rented any movies so far--
45 ✓ SELECT c.first_name, c.last_name, c.customer_id, r.rental_id
46 FROM customer AS c
47 LEFT JOIN rental AS r ON c.customer_id = r.customer_id
48 WHERE r.rental_id IS null;
49 ---There are no customers that have not rented any movies so far---
50
```

Data Output Messages Notifications



first_name
character varying (45) 🔒

last_name
character varying (45) 🔒

customer_id
integer 🔒

rental_id
integer 🔒




Query Query History

```
51 ---7. Display each movies and the number of times it got rented---
52 ✓ SELECT i.film_id, f.title, COUNT(f.film_id) AS no_of_times_rented
53 FROM inventory AS i
54 JOIN rental AS r ON i.inventory_id = r.inventory_id
55 JOIN film AS f ON i.film_id = f.film_id
56 GROUP BY i.film_id, f.title
57 HAVING COUNT (i.film_id) >= 1
58 ORDER BY no_of_times_rented DESC;
59 ---The movies and number of times rented are shown below---
```

Data Output Messages Notifications



SQL

	film_id smallint 	title character varying (255) 	no_of_times_rented bigint 
1	103	Bucket Brotherhood	34
2	738	Rocketeer Mother	33
3	767	Scalawag Duck	32
4	730	Ridgemont Submarine	32
5	331	Forward Temple	32
6	382	Grit Clockwork	32
7	489	Juggler Hardly	32
8	418	Hobbit Alien	31


```
60
61 ---8. Show the first and last names and the number of films each actor acted in---
62 ✓ SELECT a.actor_id, a.first_name, a.last_name, COUNT(fa.film_id) AS no_of_films_acted
63 FROM film_actor AS fa
64 JOIN actor AS a ON fa.actor_id = a.actor_id
65 JOIN film AS f ON fa.film_id =f.film_id
66 GROUP BY a.actor_id, a.first_name, a.last_name
67 ORDER BY no_of_films_acted DESC;
68 --- There are over 200 actors and Gina acted in the highest number of films (42)---
```



SQL

	actor_id [PK] integer	first_name character varying (45)	last_name character varying (45)	no_of_films_acted bigint
1	107	Gina	Degeneres	42
2	102	Walter	Torn	41
3	198	Mary	Keitel	40
4	181	Matthew	Carrey	39
5	23	Sandra	Kilmer	37
6	81	Scarlett	Damon	36
7	158	Vivien	Basinger	35
8	60	Henry	Berry	35


```
70 ---9. Display the names of the actors that acted in more than 20 movies---
71 SELECT a.actor_id, a.first_name, a.last_name, COUNT(fa.film_id) AS no_of_films_acted
72 FROM film_actor AS fa
73 JOIN actor AS a ON fa.actor_id = a.actor_id
74 JOIN film AS f ON fa.film_id =f.film_id
75 GROUP BY a.actor_id, a.first_name, a.last_name
76 HAVING COUNT (fa.film_id) >20
77 ORDER BY no_of_films_acted DESC;
78 ---181 actors have acted in more than 20 movies---
```



SQL

	actor_id [PK] integer	first_name character varying (45)	last_name character varying (45)	no_of_films_acted bigint
1	107	Gina	Degeneres	42
2	102	Walter	Torn	41
3	198	Mary	Keitel	40
4	181	Matthew	Carrey	39
5	23	Sandra	Kilmer	37
6	81	Scarlett	Damon	36
7	60	Henry	Berry	35
8	37	Val	Bolger	35

Query Query History

```
80 ---10. For all the movies rated "PG" show the movie and the number of times it got rented---
81 v SELECT i.film_id, f.title, COUNT(f.film_id) AS no_of_times_rented
82 FROM rental As r
83 JOIN inventory AS i ON r.inventory_id = i.inventory_id
84 JOIN film AS f ON i.film_id = f.film_id
85 WHERE rating = 'PG'
86 GROUP BY i.film_id, f.title
87 ORDER BY no_of_times_rented DESC;
88 ---183 movies were rated 'PG'---
```

Data Output Messages Notifications

≡+ 📄 ▼ 📋 ▼ 🗑️ 🗄️ ⬇️ 📈 SQL

	film_id smallint 🔒	title character varying (255) 🔒	no_of_times_rented bigint 🔒
1	103	Bucket Brotherhood	34
2	382	Grit Clockwork	32
3	753	Rush Goodfellas	31
4	369	Goodfellas Salute	31
5	869	Suspects Quills	30
6	374	Graffiti Love	30
7	341	Frost Head	30
8	595	Moon Bunch	29

Total rows: 183 of 183 Query complete 00:00:00.076

Activate W
Go to Settings

```
89
90 ---11. Display the movies offered for rent in store_id 1 and not store_id 2---
91 ✓ SELECT i.film_id, f.title, i.store_id
92 FROM inventory AS i
93 JOIN film AS f ON i.film_id = f.film_id
94 GROUP BY i.film_id, f.title, i.store_id
95 HAVING i.store_id = 1 AND i.store_id != 2;
96 --- There are 759 films offered for rent in store_id 1 and not in store_id 2---
97
```



SQL

	film_id smallint	title character varying (255)	store_id smallint
1	391	Half Outfield	1
2	73	Bingo Talented	1
3	98	Bright Encounters	1
4	681	Pirates Roxanne	1
5	437	House Dynamite	1
6	344	Fury Murder	1
7	655	Panther Reds	1
8	140	Cheaper Clyde	1

Query Query History

```
97  
98 ---12. Display the movies offered for rent in any of the 2 stores---  
99 ✓ SELECT i.film_id, f.title, i.store_id  
100 FROM inventory AS i  
101 JOIN film AS f ON i.film_id = f.film_id  
102 GROUP BY i.film_id, f.title, i.store_id  
103 HAVING i.store_id = 1 OR i.store_id = 2;  
104 ---There are 1521 movies offered in any of the 2 stores---  
105
```

Data Output Messages Notifications








	film_id smallint	title character varying (255)	store_id smallint
1	696	Pride Alamo	1
2	374	Graffiti Love	2
3	188	Crazy Home	1
4	182	Control Anthem	1
5	413	Hedwig Alter	1
6	983	Won Dares	1
7	460	Innocent Usual	2
8	623	Newton Labyrinth	1

Total rows: 1521 of 1521 Query complete 00:00:00.097

Query Query History

```
105  
106 ---13. Display the movies titles of those movies offered in both stores at the same time  
107 ✓ SELECT i.film_id, f.title, i.store_id  
108 FROM inventory AS i  
109 JOIN film AS f ON i.film_id = f.film_id  
110 GROUP BY i.film_id, f.title, i.store_id  
111 HAVING i.store_id = 1 AND i.store_id = 2;  
112 ---There are no movies offered in both stores at the same time---  
113
```

Data Output Messages Notifications

									SQL
film_id		title		store_id					
smallint		character varying (255)		smallint					
									

Query Query History

```
114 ---14. Display the movie title for the most rented movie in the store with store_id 1---
115 v SELECT f.film_id, f.title, i.store_id, COUNT(r.customer_id) AS no_of_rents
116 FROM film AS f
117 JOIN inventory AS i ON f.film_id = i.film_id
118 JOIN rental AS r ON i.inventory_id = r.inventory_id
119 GROUP BY f.film_id, f.title, i.store_id
120 HAVING i.store_id = 1 AND COUNT(r.customer_id) >1
121 ORDER BY no_of_rents DESC LIMIT 1;
122 ---Love suicides is the most rented in store 1
```

Data Output Messages Notifications



SQL

	film_id integer	title character varying (255)	store_id smallint	no_of_rents bigint
1	535	Love Suicides	1	20




Query Query History

```
123
124 ---15. How many movies are not offered for rent in the stores yet. There are two stores only 1 and 2---
125 v SELECT i.film_id, f.title, i.store_id
126 FROM inventory AS i
127 JOIN film AS f ON i.film_id = f.film_id
128 GROUP BY i.film_id, f.title, i.store_id
129 HAVING i.store_id != 1 AND i.store_id != 2;
130 ---There are no movies not offered for rent in either store---
131
```

Data Output Messages Notifications






SQL



film_id	title	store_id
smallint	character varying (255)	smallint
		

Query Query History

```
132 ---16. Show the number of rented movies under each rating---
133 ✓ SELECT f.rating, COUNT(r.rental_id) AS no_of_rented_movies
134 FROM film AS f
135 JOIN inventory AS i ON f.film_id = i.film_id
136 JOIN rental AS r ON i.inventory_id = r.inventory_id
137 GROUP BY f.rating
138 ORDER BY no_of_rented_movies DESC;
139 ---PG-13 movies had the highest number of rental at 3585--
140
```

Data Output Messages Notifications



	rating mpaa_rating 	no_of_rented_movies bigint 
1	PG-13	3585
2	NC-17	3293
3	PG	3212
4	R	3181
5	G	2773

Query Query History

```
141 ---17. Show the profit of each of the stores 1 and 2---
142 ✓ SELECT i.store_id, SUM(p.amount)
143 FROM payment AS p
144 JOIN rental AS r ON p.rental_id = r.rental_id
145 JOIN inventory AS i ON r.inventory_id = i.inventory_id
146 GROUP BY i.store_id
147 ORDER BY SUM DESC;
148 ---Store_id 2 has the highest amount in profits at 30683.13---
149
```

Data Output Messages Notifications



	store_id smallint	sum numeric
1	2	30683.13
2	1	30628.91