Independent Subquery - Scalar Subquery

22 February 2023   14:28

1. Find the movie with highest profit(vs order by)
2. Find how many movies have a rating > the avg of all the movie ratings(Find the count of above average movies)
3. Find the highest rated movie of 2000
4. Find the highest rated movie among all movies whose number of votes are > the dataset avg votes

```
2

SELECT * FROM movies
WHERE (gross - budget) = (SELECT MAX(gross - budget)
                                    FROM movies);

SELECT * FROM movies
ORDER BY (gross - budget) DESC LIMIT 1
```

Result Grid

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross	company	runtime
Titanic	PG-13	Drama	1997	December 19, 1997 (United States)	7.8	1100000.0	James Cameron	James Cameron	Leonardo DiCaprio	United States	200000000	2201647264.	Twentieth Century Fox	194.0

Action Output

#	Time	Action	Message
✓ 328	20:22:24	SELECT * FROM sql_cx_live.movies LIMIT 0, 1000	1000 row(s) returned
✓ 329	20:23:45	USE sql_cx_live	0 row(s) affected
✗ 330	20:24:18	SELECT * FROM movies WHERE (gross - budget) = (SELECT MAX(gross - profit) FROM movies) LIMIT 0, 1000	Error Code: 1054. Unknown column 'profit' in 'field list'
✓ 331	20:24:28	SELECT * FROM movies WHERE (gross - budget) = (SELECT MAX(gross - budget) FROM movies) LIMIT 0, 1...	1 row(s) returned
✓ 332	20:25:27	SELECT * FROM movies ORDER BY (gross - budget) DESC LIMIT 0, 1000	1000 row(s) returned
✓ 333	20:25:34	SELECT * FROM movies ORDER BY (gross - budget) DESC LIMIT 1	1 row(s) returned

campusx - Warning - not supp... ×

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

movies ×

Limit to 1000 rows

```sql
 6
 7 • SELECT * FROM movies
 8   ORDER BY (gross - budget) DESC LIMIT 1;
 9
10 • SELECT * FROM movies
11   WHERE score > (SELECT AVG(score)
12                  FROM movies)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross
The French Lieutenant's Woman	R	Drama	1981	October 16, 1981 (United States)	7.0	13000.0	Karel Reisz	John Fowles	Meryl Streep	United Kingdom	0	2689006
The Great Muppet Caper	G	Adventure	1981	June 26, 1981 (United States)	7.2	14000.0	Jim Henson	Tom Patchett	Jim Henson	United Kingdom	0	3120625
Nighthawks	R	Action	1981	April 10, 1981 (United States)	6.4	18000.0	Bruce Malmuth	David Shaber	Sylvester Stallone	United States	5000000	1990535
Eye of the Needle	R	Romance	1981	July 24, 1981 (United States)	7.1	9900.0	Richard Marquand	Ken Follett	Donald Sutherl...	United Kingdom	0	1758363
Prince of the City	R	Crime	1981	August 26, 1981 (United States)	7.5	7600.0	Sidney Lumet	Jay Presson Allen	Treat Williams	United States	8600000	8124257.
Diva	R	Music	1981	April 23, 1982 (United States)	7.2	12000.0	Jean-Jacques Beineix	Daniel Odier	W			
Fort Apache the Bronx	R	Crime	1981	February 6, 1981 (United States)	6.7	5800.0	Daniel Petrie	Heywood Gould	P			

movies 6 ×

Output

Action Output

#	Time	Action	Message
❌ 330	20:24:18	SELECT * FROM movies WHERE (gross - budget) = (SELECT MAX(gross - profit) FROM movies) LIMIT 0, 1000	Error Code: 1054. Unknown column 'profit' in 'field list'
✅ 331	20:24:28	SELECT * FROM movies WHERE (gross - budget) = (SELECT MAX(gross - budget) FROM movies) LIMIT 0, 1...	1 row(s) returned
✅ 332	20:25:27	SELECT * FROM movies ORDER BY (gross - budget) DESC LIMIT 0, 1000	1000 row(s) returned
✅ 333	20:25:34	SELECT * FROM movies ORDER BY (gross - budget) DESC LIMIT 1	1 row(s) returned
✅ 334	20:30:17	SELECT AVG(score) FROM movies LIMIT 0, 1000	1 row(s) returned
✅ 335	20:30:56	SELECT * FROM movies WHERE score > (SELECT AVG(score) FROM movies) LIMIT 0, 1000	1000 row(s) returned

Navigator

SCHEMAS

Filter objects

▶ campusx
▶ flipkart
▶ ipl
▶ maang
▶ mydb
▶ phpmyadmin
▶ sample
▼ sql_cx_live
  ▼ Tables
    ▶ class
    ▶ groups
    ▶ loyal_users
    ▶ membership
    ▶ movies
    ▶ person1
    ▶ person2
    ▶ students
    ▶ titanic
    ▶ users
  Views
  Stored Procedures
  Functions
▶ sql_tasks
▶ swiggy
▶ temp_db
▶ test
▶ tinder
▼ zomato

Administration   Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info   Session

campusx - Warning - not supp... ×

File Edit View Query Database Server Tools Scripting Help

movies ×

Limit to 1000 rows

```sql
11   WHERE score > (SELECT AVG(score)
12                        FROM movies);
13
14 • SELECT * FROM movies
15   WHERE year = 2000 AND score = (SELECT MAX(score)
16                                  FROM movies
17                                  WHERE year = 2000)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross	company	runtime
Gladiator	R	Action	2000	May 5, 2000 (United States)	8.5	1400000.0	Ridley Scott	David Franzoni	Russell Crowe	United States	103000000	465380802.0	Dreamworks Pictures	155.0

movies 9 ×

Output

Action Output

#	Time	Action	Message
333	20:25:34	SELECT * FROM movies ORDER BY (gross - budget) DESC LIMIT 1	1 row(s) returned
334	20:30:17	SELECT AVG(score) FROM movies LIMIT 0, 1000	1 row(s) returned
335	20:30:56	SELECT * FROM movies WHERE score > (SELECT AVG(score) FROM movies) LIMIT 0, 1000	1000 row(s) returned
336	20:31:13	SELECT COUNT(*) FROM movies WHERE score > (SELECT AVG(score) FROM movies) LIMIT 0, 1000	1 row(s) returned
337	20:32:25	SELECT MAX(score) FROM movies WHERE year = 2000 LIMIT 0, 1000	1 row(s) returned
338	20:33:09	SELECT * FROM movies WHERE year = 2000 AND score = (SELECT MAX(score) FROM movies WHERE ye...	1 row(s) returned

SCHEMAS

Filter objects

- campusx
- flipkart
- ipl
- maang
- mydb
- phpmyadmin
- sample
- ▼ sql_cx_live
  - ▼ Tables
    - class
    - groups
    - loyal_users
    - membership
    - movies
    - person1
    - person2
    - students
    - titanic
    - users
  - Views
  - Stored Procedures
  - Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- ▼ zomato

Administration Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info Session

```
17                                        WHERE year = 2000);

18

19 •  SELECT * FROM movies
20    WHERE score = (SELECT MAX(score) FROM movies
21                        WHERE votes > (SELECT AVG(votes)
22                                       FROM movies))

23
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: $\overline{IA}$

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross	company	runtime
The Shawshank Redemption	R	Drama	1994	October 14, 1994 (United States			abont	Stephen King	Tim Robbins	United States	25000000	28817291.0	Castle Rock Entertainment	142.0

October 14, 1994 (United States)

movies 13 ×

Output

Action Output

#	Time	Action	Message
337	20:32:25	SELECT MAX(score) FROM movies WHERE year = 2000 LIMIT 0, 1000	1 row(s) returned
338	20:33:09	SELECT * FROM movies WHERE year = 2000 AND score = (SELECT MAX(score) FROM movies WHERE ye...	1 row(s) returned
339	20:34:51	SELECT AVG(votes) FROM movies LIMIT 0, 1000	1 row(s) returned
340	20:35:33	SELECT * FROM movies WHERE votes > (SELECT AVG(votes) FROM movies) LIMIT 0, 1000	855 row(s) returned
341	20:35:51	SELECT MAX(score) FROM movies WHERE votes > (SELECT AVG(votes) FROM movies) LIMIT 0, 1000	1 row(s) returned
342	20:36:38	SELECT * FROM movies WHERE score = (SELECT MAX(score) FROM movies WHERE votes > (SELECT AV...	1 row(s) returned

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Independent Subquery - Row Subquery(One Col Multi Rows)

22 February 2023   14:29

1. Find all users who never ordered
2. Find all the movies made by top 3 directors(in terms of total gross income)
3. Find all movies of all those actors whose filmography's avg rating > 8.5(take 25000 votes as cutoff)

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

movies   users   orders   order_details   food   menu   delivery_partner ×

SCHEMAS

Filter objects

▶ campusx
▶ flipkart
▶ ipl
▶ maang
▶ mydb
▶ phpmyadmin
▶ sample
▶ sql_cx_live
▶ sql_tasks
▶ swiggy
▶ temp_db
▶ test
▶ tinder
▼ zomato
  ▼ Tables
    ▶ delivery_partner
    ▶ food
    ▶ loyal_users
    ▶ menu
    ▶ order_details
    ▶ orders
    ▶ restaurants
    ▶ users
  Views
  Stored Procedures
  Functions

Limit to 1000 rows

```
1    USE zomato;
2
3 •  SELECT * FROM users
4    WHERE user_id NOT IN (SELECT DISTINCT(user_id)
5                                      FROM orders)
6
```

Result Grid | Filter Rows: Export: Wrap Cell Content: ΙΑ

user_id	name	email	password
6	Anupama	anupama@gmail.com	46rdw2
7	Rishabh	Rishabh gmail.com	4sw123

Administration Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info Session

users 4 ×

Output

Action Output

#	Time	Action	Message
❌ 350	20:41:34	USE zomato SELECT user_id FROM zomato	Error Code: 1064. You have an error in your SQL syntax; check t...
✅ 351	20:41:38	USE zomato	0 row(s) affected
❌ 352	20:41:42	SELECT user_id FROM zomato LIMIT 0, 1000	Error Code: 1146. Table 'zomato.zomato' doesn't exist
✅ 353	20:41:48	SELECT user_id FROM orders LIMIT 0, 1000	25 row(s) returned
✅ 354	20:42:05	SELECT DISTINCT(user_id) FROM orders LIMIT 0, 1000	5 row(s) returned
✅ 355	20:43:01	SELECT * FROM users WHERE user_id NOT IN (SELECT DISTINCT(user_id) FROM orders) LIMIT 0, 1000	2 row(s) returned

Result Grid

Form Editor

```
22                                    FROM movies));

23

24 •  SELECT * FROM movies
25  ⊖ WHERE director IN (SELECT director
26                        FROM movies
27                        GROUP BY director
28                        ORDER BY SUM(gross) DESC
29                        LIMIT 3)

30

31

32

33
```

#	Time	Action	Message
✅	357 20:44:24	USE sql_cx_live	0 row(s) affected
✅	358 20:44:55	SELECT director,SUM(gross) FROM movies GROUP BY director LIMIT 0, 1000	1000 row(s) returned
✅	359 20:45:12	SELECT director,SUM(gross) FROM movies GROUP BY director ORDER BY SUM(gross) DESC LIMIT 0, 1000	1000 row(s) returned
✅	360 20:45:23	SELECT director,SUM(gross) FROM movies GROUP BY director ORDER BY SUM(gross) DESC LIMIT 3	3 row(s) returned
✅	361 20:45:34	SELECT director FROM movies GROUP BY director ORDER BY SUM(gross) DESC LIMIT 3	3 row(s) returned
❌	362 20:46:40	SELECT * FROM movies WHERE director IN (SELECT director FROM movies GROUP BY director ORDER B...	Error Code: 1235. This version of MariaDB doesn't yet support 'L...

campusx - Warning - not supp... ✕

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

movies ✕   users   orders   order_details   food   menu   delivery_partner

Limit to 1000 rows

```sql
24 •  WITH top_directors AS (SELECT director
25                           FROM movies
26                           GROUP BY director
27                           ORDER BY SUM(gross) DESC
28                           LIMIT 3)
29     SELECT * FROM movies
30     WHERE director IN (SELECT * FROM top_directors)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: 🔤

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross
Indiana Jones and the Raiders of the Lost Ark	PG	Action	1981	June 12, 1981 (United States)	8.4	905000.0	Steven Spielberg	Lawrence Kasdan	Harrison Ford	United States	18000000	389925
E.T. the Extra-Terrestrial	PG	Family	1982	June 11, 1982 (United States)	7.8	381000.0	Steven Spielberg	Melissa Mathison	Henry Thomas	United States	10500000	792910
The Terminator	R	Action	1984	October 26, 1984 (United States)	8.0	812000.0	James Cameron	James Cameron	Arnold Schwarzenegger	United Kingdom	6400000	783712
Indiana Jones and the Temple of Doom	PG	Action	1984	May 23, 1984 (United States)	7.5	459000.0	Steven Spielberg	Willard Huyck	Harrison Ford	United States	28000000	333107
Romancing the Stone	PG	Action	1984	March 30, 1984 (United States)	6.9	86000.0	Robert Zemeckis	Diane Thomas	Michael Douglas	Mexico	10000000	865722
Back to the Future	PG	Adventure	1985	July 3, 1985 (United States)	8.5	1100000.0	Robert Zemeckis	Robert Zemeckis	Michael			
The Color Purple	PG-13	Drama	1985	February 7, 1986 (United States)	7.8	81000.0	Steven Spielberg	Menno Meyjes	Danny			

Result 18 ✕

Output

Action Output

#	Time	Action	Message
✓	358 20:44:55	SELECT director,SUM(gross) FROM movies GROUP BY director LIMIT 0, 1000	1000 row(s) returned
✓	359 20:45:12	SELECT director,SUM(gross) FROM movies GROUP BY director ORDER BY SUM(gross) DESC LIMIT 0, 1000	1000 row(s) returned
✓	360 20:45:23	SELECT director,SUM(gross) FROM movies GROUP BY director ORDER BY SUM(gross) DESC LIMIT 3	3 row(s) returned
✓	361 20:45:34	SELECT director FROM movies GROUP BY director ORDER BY SUM(gross) DESC LIMIT 3	3 row(s) returned
✗	362 20:46:40	SELECT * FROM movies WHERE director IN (SELECT director FROM movies GROUP BY director ORDER B...	Error Code: 1235. This version of MariaDB doesn't yet support 'L...
✓	363 20:48:12	WITH top_directors AS (SELECT director FROM movies GROUP BY director ORDER BY SUM(gross) DESC L...	30 row(s) returned

SCHEMAS

Filter objects

▶ campusx
▶ flipkart
▶ ipl
▶ maang
▶ mydb
▶ phpmyadmin
▶ sample
▼ sql_cx_live
  ▼ Tables
    ▶ class
    ▶ groups
    ▶ loyal_users
    ▶ membership
    ▶ movies
    ▶ person1
    ▶ person2
    ▶ students
    ▶ titanic
    ▶ users
  Views
  Stored Procedures
  Functions
▶ sql_tasks
▶ swiggy
▶ temp_db
▶ test
▶ tinder
▼ zomato

Administration   Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info   Session

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

movies × | users | orders | order_details | food | menu | delivery_partner

Limit to 1000 rows

```
30        WHERE director IN (SELECT * FROM top_directors);

31

32 •     SELECT * FROM movies

33        WHERE star IN (SELECT star FROM movies

34        WHERE votes > 25000

35        GROUP BY star

36        HAVING AVG(score) > 8.5)

37
```

Navigator

SCHEMAS

Filter objects

▶ campusx
▶ flipkart
▶ ipl
▶ maang
▶ mydb
▶ phpmyadmin
▶ sample
▼ sql_cx_live
  ▼ Tables
    ▶ class
    ▶ groups
    ▶ loyal_users
    ▶ membership
    ▶ movies
    ▶ person1
    ▶ person2
    ▶ students
    ▶ titanic
    ▶ users
  Views
  Stored Procedures
  Functions
▶ sql_tasks
▶ swiggy
▶ temp_db
▶ test
▶ tinder
▼ zomato

Administration   Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info   Session

Result Grid | Filter Rows: | Export: | Wrap Cell Content: ĪA

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross
Johnny Stecchino	R	Comedy	1991	October 24, 1991 (Italy)	7.2	10000.0	Roberto Benigni	Roberto Benigni	Roberto Benigni	Italy	0	626057.0
The Adventures of Huck Finn	PG	Adventure	1993	April 2, 1993 (United States)	6.2	8000.0	Stephen Sommers	Mark Twain	Elijah Wood	United States	6500000	24103594.0
Son of the Pink Panther	PG	Comedy	1993	August 27, 1993 (United States)	3.9	5000.0	Blake Edwards	Maurice Richlin	Roberto Benigni	Italy	25000000	2438031.0
North	PG	Adventure	1994	July 22, 1994 (United States)	4.5	13000.0	Rob Reiner	Alan Zweibel	Elijah			
The War	PG-13	Drama	1994	November 4, 1994 (United States)	6.8	12000.0	Jon Avnet	Kathy McWorter	Elijah			

movies 27 ×

Output

Action Output

#	Time	Action	Message
367	20:52:25	SELECT star,AVG(score) FROM movies WHERE votes > 25000 GROUP BY star HAVING AVG(score) > 8.5 LI...	3 row(s) returned
368	20:52:46	SELECT star FROM movies WHERE votes > 25000 GROUP BY star HAVING AVG(score) > 8.5 LIMIT 0, 1000	3 row(s) returned
369	20:53:13	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	8 row(s) returned
370	20:53:34	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	5 row(s) returned
371	20:53:40	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	3 row(s) returned
372	20:54:12	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	8 row(s) returned

Result Grid

Independent Subquery - Table Subquery(Multi Col Multi Row)

22 February 2023   14:29

1. Find the most profitable movie of each year
2. Find the highest rated movie of each genre votes cutoff of 25000
3. Find the highest grossing movies of top 5 actor/director combo in terms of total gross income

```sql
36    HAVING AVG(score) > 8.5);

37

38 •  SELECT * FROM movies

39    WHERE (year,gross-budget) IN (SELECT year,MAX(gross - budget)
40                                         FROM movies
41                                         GROUP BY year)

42

43
```

Result Grid

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross
Star Wars: Episode V - The Empire Strikes Back	PG	Action	1980	June 20, 1980 (United States)	8.7	1200000.0	Irvin Kershner	Leigh Brackett	Mark Hamill	United States	18000000	538375067.0
Indiana Jones and the Raiders of the Lost Ark	PG	Action	1981	June 12, 1981 (United States)	8.4	905000.0	Steven Spielberg	Lawrence Kasdan	Harrison Ford	United States	18000000	389925971.0
E.T. the Extra-Terrestrial	PG	Family	1982	June 11, 1982 (United States)	7.8	381000.0	Steven Spielberg	Melissa Mathison	Henry Thomas	United States	10500000	792910554.0
Star Wars: Episode VI - Return of the Jedi	PG	Action	1983	May 25, 1983 (United States)	8.3	973000.0	Richard Marquand	Lawrence Kasdan	Mark			
Indiana Jones and the Temple of Doom	PG	Action	1984	May 23, 1984 (United States)	7.5	459000.0	Steven Spielberg	Willard Huyck	Harri			

movies 29

Output

Action Output

#	Time	Action	Message
369	20:53:13	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	8 row(s) returned
370	20:53:34	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	5 row(s) returned
371	20:53:40	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	3 row(s) returned
372	20:54:12	SELECT * FROM movies WHERE star IN (SELECT star FROM movies WHERE votes > 25000 GROUP BY st...	8 row(s) returned
373	21:00:47	SELECT year,MAX(gross - budget) FROM movies GROUP BY year LIMIT 0, 1000	22 row(s) returned
374	21:03:12	SELECT * FROM movies WHERE (year,gross-budget) IN (SELECT year,MAX(gross - budget) FROM movies G...	22 row(s) returned

Table: movies

Columns:
name varchar(72)
rating varchar(9)
genre varchar(9)
year int(4)
released varchar(45)
score decimal(2,1)
votes decimal(8,1)

```sql
41                              GROUP BY year);

42

43 •  SELECT * FROM movies

44    WHERE (genre,score) IN (SELECT genre,MAX(score)

45                            FROM movies

46                            WHERE votes > 25000

47                            GROUP BY genre)

48    AND votes > 25000
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross	company
The Thing	R	Horror	1982	June 25, 1982 (United States)	8.1	382000.0	John Carpenter	Bill Lancaster	Kurt Russell	United States	15000000	19632053.0	Universal Picture
E.T. the Extra-Terrestrial	PG	Family	1982	June 11, 1982 (United States)	7.b	381000.0	Steven Spielberg	Melissa Mathison	Henry Thomas	United States	10500000	792910554.0	Universal Picture
Starman	PG	Romance	1984	December 14, 1984 (United States)	7.0	44000.0	John Carpenter	Bruce A. Evans	Jeff Bridges	United States	22000000	28744356.0	Columbia Picture
Back to the Future	PG	Adventure	1985	July 3, 1985 (United States)	8.5	1100000.0	Robert Zemeckis	Robert Zemeckis	Michael J. Fox	Unit			
Schindler's List	R	Biography	1993	February 4, 1994 (United States)	8.9	1200000.0	Steven Spielberg	Thomas Keneally	Liam Neeson	Unit			

movies 33 ×

Output

Action Output

#	Time	Action	Message
373	21:00:47	SELECT year,MAX(gross - budget) FROM movies GROUP BY year LIMIT 0, 1000	22 row(s) returned
374	21:03:12	SELECT * FROM movies WHERE (year,gross-budget) IN (SELECT year,MAX(gross - budget) FROM movies G...	22 row(s) returned
375	21:05:18	SELECT genre,MAX(score) FROM movies GROUP BY genre LIMIT 0, 1000	17 row(s) returned
376	21:05:36	SELECT genre,MAX(score) FROM movies WHERE votes > 25000 GROUP BY genre LIMIT 0, 1000	12 row(s) returned
377	21:06:25	SELECT * FROM movies WHERE (genre,score) IN (SELECT genre,MAX(score) FROM movies WHERE votes ...	13 row(s) returned
378	21:07:16	SELECT * FROM movies WHERE (genre,score) IN (SELECT genre,MAX(score) FROM movies WHERE votes ...	12 row(s) returned

Navigator

SCHEMAS

Filter objects

- campusx
- flipkart
- ipl
- maang
- mydb
- phpmyadmin
- sample
- sql_cx_live
  - Tables
    - class
    - groups
    - loyal_users
    - membership
    - movies
    - person1
    - person2
    - students
    - titanic
    - users
  - Views
  - Stored Procedures
  - Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- zomato

Administration | Schemas

Information

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info | Session

campusx - Warning - not supp... ×

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

movies ×   users   orders   order_details   food   menu   delivery_partner

Limit to 1000 rows

```sql
51        SELECT star,director,MAX(gross)
52        FROM movies
53        GROUP BY star,director
54        ORDER BY SUM(gross) DESC LIMIT 5
55    )
56
57    SELECT * FROM movies
58    WHERE (star,director,gross) IN (SELECT * FROM top_duos)
```

SCHEMAS

- ▶ campusx
- ▶ flipkart
- ▶ ipl
- ▶ maang
- ▶ mydb
- ▶ phpmyadmin
- ▶ sample
- ▼ sql_cx_live
  - ▼ Tables
    - ▶ class
    - ▶ groups
    - ▶ loyal_users
    - ▶ membership
    - ▶ movies
    - ▶ person1
    - ▶ person2
    - ▶ students
    - ▶ titanic
    - ▶ users
  - Views
  - Stored Procedures
  - Functions
- ▶ sql_tasks
- ▶ swiggy
- ▶ temp_db
- ▶ test
- ▶ tinder
- ▼ zomato

Administration   Schemas

Information

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info   Session

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

name	rating	genre	year	released	score	votes	director	writer	star	country	budget	gross	company
Indiana Jones and the Last Crusade	PG-13	Action	1989	May 24, 1989 (United States)	8.2	707000.0	Steven Spielberg	Jeffrey Boam	Harrison Ford	United States	48000000	474171806.0	Paramou
Lethal Weapon 3	R	Action	1992	May 15, 1992 (United States)	6.7	160000.0	Richard ... Steven Spielberg Joam		Mel Gibson	United States	35000000	321731527.0	Warner E
Forrest Gump	PG-13	Drama	1994	July 6, 1994 (United States)	8.8	1900000.0	Robert Zemeckis	Winston Groom	Tom Hanks	United States	55000000	678226133.0	Paramou
The Lion King	G	Animation	1994	June 24, 1994 (United States)	8.5	970000.0	Roger Allers	Irene Mecchi	Matthew Broderick				
Titanic	PG-13	Drama	1997	December 19, 1997 (United States)	7.8	1100000.0	James Cameron	James Cameron	Leonardo DiCaprio				

Result 37 ×

Output

Action Output

#	Time	Action	Message
✓	377	21:06:25	SELECT * FROM movies WHERE (genre,score) IN (SELECT genre,MAX(score) FROM movies WHERE votes ...	13 row(s) returned
✓	378	21:07:16	SELECT * FROM movies WHERE (genre,score) IN (SELECT genre,MAX(score) FROM movies WHERE votes ...	12 row(s) returned
✓	379	21:10:27	SELECT star,director,SUM(gross),MAX(gross) FROM movies GROUP BY star,director LIMIT 0, 1000	1000 row(s) returned
✓	380	21:10:43	SELECT star,director,SUM(gross),MAX(gross) FROM movies GROUP BY star,director ORDER BY SUM(gross) ...	5 row(s) returned
✓	381	21:11:07	SELECT star,director,MAX(gross) FROM movies GROUP BY star,director ORDER BY SUM(gross) DESC LIMI...	5 row(s) returned
✓	382	21:12:19	WITH top_duos AS (SELECT star,director,MAX(gross) FROM movies GROUP BY star,director ORDER BY SU...	5 row(s) returned

Result Grid

Correlated Subquery

22 February 2023   14:29

1. Find all the movies that have a rating higher than the average rating of movies in the same genre.[Animation]
2. Find the favorite food of each customer.

```sql
59
60 • SELECT * FROM movies;

61

62 • SELECT * FROM movies m1
63    WHERE score > (SELECT AVG(score) FROM movies m2 WHERE m2.genre = m1.genre)

64

65

66

67

68

69

70

71
```

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Output — Action Output

#	Time	Action	Message
✓	380 21:10:43	SELECT star,director,SUM(gross),MAX(gross) FROM movies GROUP BY star,director ORDER BY SUM(gross) ...	5 row(s) returned
✓	381 21:11:07	SELECT star,director,MAX(gross) FROM movies GROUP BY star,director ORDER BY SUM(gross) DESC LIMI...	5 row(s) returned
✓	382 21:12:19	WITH top_duos AS (SELECT star,director,MAX(gross) FROM movies GROUP BY star,director ORDER BY SU...	5 row(s) returned
✓	383 21:16:13	SELECT * FROM movies LIMIT 0, 1000	1000 row(s) returned
✓	384 21:20:20	SELECT * FROM movies m1 WHERE score > (SELECT AVG(score) FROM movies m2 WHERE m2.genre = m...	1000 row(s) returned
✗	385 21:20:51	SELECT AVG(score) FROM movies m2 WHERE m2.genre = m1.genre LIMIT 0, 1000	Error Code: 1054. Unknown column 'm1.genre' in 'where clause'

campusx - Warning - not supp... ×

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

Navigator

SCHEMAS

Filter objects

- campusx
- flipkart
- ipl
- maang
- mydb
- phpmyadmin
- sample
- sql_cx_live
  - Tables
    - class
    - groups
    - loyal_users
    - membership
    - movies
    - person1
    - person2
    - students
    - titanic
    - users
  - Views
  - Stored Procedures
  - Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- zomato

Administration   Schemas

Information

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info   Session

movies   users   orders   order_details   food   menu   delivery_partner ×

Limit to 1000 rows

```
 5        (
                                      FROM orders);
 6
 7 •  WITH fav_food AS (
 8        SELECT t2.user_id,name,f_name,COUNT(*) AS 'frequency' FROM users t1
 9        JOIN orders t2 ON t1.user_id = t2.user_id
10        JOIN order_details t3 ON t2.order_id = t3.order_id
11        JOIN food t4 ON t3.f_id = t4.f_id
12        GROUP BY t2.user_id,t3.f_id
```

Result Grid   Filter Rows: Export: Wrap Cell Content: IA

user_id	name	f_name	frequency
1	Nitish	Non-veg Pizza	1
1	Nitish	Choco Lava cake	5
1	Nitish	Chicken Wings	1
1	Nitish	Rice Meal	2
1	Nitish	Roti meal	1
2	Khushboo	Non-veg Pizza	1

Result 13 ×

Output

Action Output

| #	Time	Action	Message
389	21:29:01	SELECT * FROM users t1 JOIN orders t2 ON t1.user_id = t2.user_id JOIN order_details t3 ON t2.order_id = t3....	50 row(s) returned
390	21:29:13	SELECT name,f_name FROM users t1 JOIN orders t2 ON t1.user_id = t2.user_id JOIN order_details t3 ON t2.o...	50 row(s) returned
391	21:30:29	SELECT name,f_name,COUNT(*) FROM users t1 JOIN orders t2 ON t1.user_id = t2.user_id JOIN order_details...	25 row(s) returned
392	21:31:40	WITH fav_food AS (SELECT name,f_name,COUNT(*) FROM users t1 JOIN orders t2 ON t1.user_id = t2.user_...	25 row(s) returned
393	21:32:02	WITH fav_food AS (SELECT name,f_name,COUNT(*) AS 'frequency' FROM users t1 JOIN orders t2 ON t1.us...	25 row(s) returned
394	21:32:13	WITH fav_food AS (SELECT t2.user_id,name,f_name,COUNT(*) AS 'frequency' FROM users t1 JOIN orders t...	25 row(s) returned

```
 8      SELECT t2.user_id,name,f_name,COUNT(*) AS 'frequency' FROM users t1
 9      JOIN orders t2 ON t1.user_id = t2.user_id
10      JOIN order_details t3 ON t2.order_id = t3.order_id
11      JOIN food t4 ON t3.f_id = t4.f_id
12      GROUP BY t2.user_id,t3.f_id
13   )
14   SELECT * FROM fav_food f1
15   WHERE frequency = (SELECT MAX(frequency)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

user_id	name	f_name	frequency
1	Nitish	Choco Lava cake	5
2	Khushboo	Choco Lava cake	3
3	Vartika	Chicken Wings	3
4	Ankit	Schezwan Noodles	3
4	Ankit	Veg Manchurian	3
5	Neha	Choco Lava cake	5

Result 14 ×

Output

Action Output

#	Time	Action	Message
✓ 391	21:30:29	SELECT name,f_name,COUNT(*) FROM users t1 JOIN orders t2 ON t1.user_id = t2.user_id JOIN order_details...	25 row(s) returned
✓ 392	21:31:40	WITH fav_food AS (SELECT name,f_name,COUNT(*) FROM users t1 JOIN orders t2 ON t1.user_...	25 row(s) returned
✓ 393	21:32:02	WITH fav_food AS (SELECT name,f_name,COUNT(*) AS 'frequency' FROM users t1 JOIN orders t2 ON t1.us...	25 row(s) returned
✓ 394	21:32:13	WITH fav_food AS (SELECT t2.user_id,name,f_name,COUNT(*) AS 'frequency' FROM users t1 JOIN orders t...	25 row(s) returned
✗ 395	21:34:18	WITH fav_food AS (SELECT t2.user_id,name,f_name,COUNT(*) AS 'frequency' FROM users t1 JOIN orders t...	Error Code: 1054. Unknown column 'frequency' in 'field list'
✓ 396	21:34:28	WITH fav_food AS (SELECT t2.user_id,name,f_name,COUNT(*) AS 'frequency' FROM users t1 JOIN orders t...	6 row(s) returned

Usage with SELECT

22 February 2023   14:29

1. Get the percentage of votes for each movie compared to the total number of votes.
2. Display all movie names ,genre, score and avg(score) of genre

-> Why this is inefficient?

$20 + 2$

name | percen of votes

```sql
64
65 •  SELECT name, (votes/(SELECT SUM(votes) FROM movies))*100 FROM movies;

66

67 •  SELECT name,genre,score,
68     (SELECT AVG(score) FROM movies m2 WHERE m2.genre = m1.genre)
69     FROM movies m1

70

71
```

Result Grid

name	genre	score	(SELECT AVG(score) FROM movies m2 WHERE m2.genre = m1.genre)
The Shining	Drama	8.4	6.67361
The Blue Lagoon	Adventure	5.8	6.15195
Star Wars: Episode V - The Empire Strikes Back	Action	8.7	6.01497
Airplane!	Comedy	7.7	6.17941
Caddyshack	Comedy	7.3	6.17941

Result 46

Output

Action Output

#	Time	Action	Message
399	21:40:01	SELECT name, (votes/(SELECT SUM(votes) FROM movies))*0.1 FROM movies LIMIT 0, 1000	1000 row(s) returned
400	21:40:10	SELECT name, (votes/(SELECT SUM(votes) FROM movies))*100 FROM movies LIMIT 0, 1000	1000 row(s) returned
401	21:41:31	SELECT name,genre,score FROM movies LIMIT 0, 1000	1000 row(s) returned
402	21:42:21	SELECT name,genre,score,(SELECT AVG(genre) FROM movies m2 WHERE m2.genre = m1.genre) FROM m...	1000 row(s) returned
403	21:43:02	SELECT name,genre,score, (SELECT AVG(genre) FROM movies m2 WHERE m2.genre = m1.genre) FROM m...	1000 row(s) returned
404	21:43:22	SELECT name,genre,score, (SELECT AVG(score) FROM movies m2 WHERE m2.genre = m1.genre) FROM m...	1000 row(s) returned

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Usage with FROM

22 February 2023   14:30

1. Display average rating of all the restaurants

File Edit View Query Database Server Tools Scripting Help

movies   users   orders   order_details   food   menu   **delivery_partner** × restaurants

Limit to 1000 rows

```sql
16                              FROM fav_food f2
17                              WHERE f2.user_id = f1.user_id);
18
19 •  SELECT r_name,avg_rating
20    FROM  (SELECT r_id,AVG(restaurant_rating) AS 'avg_rating'
21              FROM orders
22              GROUP BY r_id) t1 JOIN restaurants t2
23              ON t1.r_id = t2.r_id
```

SCHEMAS

- ▶ class
- ▶ groups
- ▶ loyal_users
- ▶ membership
- ▶ movies
- ▶ person1
- ▶ person2
- ▶ students
- ▶ titanic
- ▶ users
- Views
- Stored Procedures
- Functions
- ▶ sql_tasks
- ▶ swiggy
- ▶ temp_db
- ▶ test
- ▶ tinder
- ▼ zomato
  - ▼ Tables
    - ▶ delivery_partner
    - ▶ food
    - ▶ loyal_users
    - ▶ menu
    - ▶ order_details
    - ▶ orders
    - ▶ restaurants
    - ▶ users

Administration Schemas

Information

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info Session

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

r_name	avg_rating
dominos	1.6666666666666667
kfc	2.2
box8	4.666666666666667
Dosa Plaza	3.6666666666666665
China Town	3.6666666666666665

Result 16 ×

Output

Action Output

#	Time	Action	Message
✔ 403	21:43:02	SELECT name,genre,score, (SELECT AVG(genre) FROM movies m2 WHERE m2.genre = m1.genre) FROM m...	1000 row(s) returned
✔ 404	21:43:22	SELECT name,genre,score, (SELECT AVG(score) FROM movies m2 WHERE m2.genre = m1.genre) FROM m...	1000 row(s) returned
✔ 405	21:44:36	USE zomato	0 row(s) affected
✔ 406	21:44:46	SELECT * FROM zomato.restaurants LIMIT 0, 1000	5 row(s) returned
✔ 407	21:46:47	SELECT r_id,AVG(restaurant_rating) FROM orders GROUP BY r_id LIMIT 0, 1000	5 row(s) returned
✔ 408	21:48:00	SELECT r_name,avg_rating FROM (SELECT r_id,AVG(restaurant_rating) AS 'avg_rating' FROM orders GROU...	5 row(s) returned

Usage with HAVING

22 February 2023   14:30

1. Find genres having avg score > avg score of all the movies

Navigator

SCHEMAS

movies × users orders order_details food menu delivery_partner restaurants

Limit to 1000 rows

```sql
69    FROM movies m1;

70

71 •  SELECT genre,AVG(score)

72    FROM movies

73    GROUP BY genre

74    HAVING AVG(score) > (SELECT AVG(score) FROM movies)

75

76
```

SCHEMAS

- class
- groups
- loyal_users
- membership
- movies
- person1
- person2
- students
- titanic
- users
- Views
- Stored Procedures
- Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- zomato
  - Tables
    - delivery_partner
    - food
    - loyal_users
    - menu
    - order_details
    - orders
    - restaurants
    - users

Administration Schemas

Information

Result Grid | Filter Rows: | Export: | Wrap Cell Content: ⓘ

genre	AVG(score)
Family	6.55000
History	8.30000
Music	7.20000
Mystery	6.50000
Romance	6.40000

Result 48 ×

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info Session

Output

Action Output

#	Time	Action	Message
✓	408 21:48:00	SELECT r_name,avg_rating FROM (SELECT r_id,AVG(restaurant_rating) AS 'avg_rating' FROM orders GROU...	5 row(s) returned
✗	409 21:50:08	SELECT genre,AVG(score) FROM movies GROUP BY genres LIMIT 0, 1000	Error Code: 1146. Table 'zomato.movies' doesn't exist
✓	410 21:50:15	USE sql_cx_live	0 row(s) affected
✗	411 21:50:18	SELECT genre,AVG(score) FROM movies GROUP BY genres LIMIT 0, 1000	Error Code: 1054. Unknown column 'genres' in 'group statement'
✓	412 21:50:23	SELECT genre,AVG(score) FROM movies GROUP BY genre LIMIT 0, 1000	17 row(s) returned
✓	413 21:51:05	SELECT genre,AVG(score) FROM movies GROUP BY genre HAVING AVG(score) > (SELECT AVG(score) FR...	9 row(s) returned

Subquery In INSERT

Populate a already created loyal_customers table with records of only those customers who have ordered food more than 3 times.

```sql
1•  SELECT * FROM zomato.loyal_users;

2•  USE zomato;

3

4•  INSERT INTO loyal_users

5   (user_id, name)

6   SELECT t1.user_id, name

7   FROM orders t1

8   JOIN users t2 ON t1.user_id = t2.user_id

9   GROUP BY user_id

10  HAVING COUNT(*) > 3

11
```

Navigator: movies | users | orders | order_details | food | menu | delivery_partner | restaurants | loyal_users

SCHEMAS

Filter objects

- membership
- movies
- person1
- person2
- students
- titanic
- users
- Views
- Stored Procedures
- Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- zomato
  - Tables
    - delivery_partner
    - food
    - loyal_users
    - menu
    - order_details
    - orders
    - restaurants
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info Session

Output

Action Output

#	Time	Action	Message
✓ 418	21:56:22	SELECT user_id,COUNT(*) FROM orders GROUP BY user_id LIMIT 0, 1000	5 row(s) returned
✗ 419	21:57:03	SELECT user_id,name,COUNT(*) FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY user_i...	Error Code: 1052. Column 'user_id' in field list is ambiguous
✓ 420	21:57:13	SELECT t1.user_id,name,COUNT(*) FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY us...	5 row(s) returned
✓ 421	21:57:28	SELECT t1.user_id,name,COUNT(*) FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY us...	5 row(s) returned
✓ 422	21:57:35	SELECT t1.user_id,name FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY user_id HAVI...	5 row(s) returned
✓ 423	21:58:26	INSERT INTO loyal_users (user_id,name) SELECT t1.user_id,name FROM orders t1 JOIN users t2 ON t1.user...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0

MySQL Workbench

campusx - Warning - not supp... ×

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

Navigator

SCHEMAS

Filter objects

- membership
- movies
- person1
- person2
- students
- titanic
- users
- Views
- Stored Procedures
- Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- zomato
  - Tables
    - delivery_partner
    - food
    - loyal_users
    - menu
    - order_details
    - orders
    - restaurants
    - users
  - Views
  - Stored Procedures
  - Functions

Administration   Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
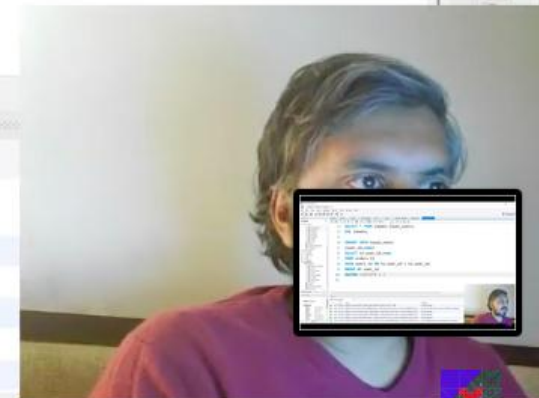score   decimal(2,1)
votes   decimal(8,1)

Object Info   Session

movies   users   orders   order_details   food   menu   delivery_partner   restaurants   loyal_users ×   loyal_users

```
1 • SELECT * FROM zomato.loyal_users;
```

Result Grid | Filter Rows: | Edit: Export/Import: | Wrap Cell Content:

user_id	name	money
1	Nitish	NULL
2	Khushboo	NULL
3	Vartika	NULL
4	Ankit	NULL
5	Neha	NULL
NULL	NULL	NULL

Output

Action Output

#	Time	Action	Message
❌ 419	21:57:03	SELECT user_id,name,COUNT(*) FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY user_i...	Error Code: 1052. Column 'user_id' in field list is ambiguous
✅ 420	21:57:13	SELECT t1.user_id,name,COUNT(*) FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY us...	5 row(s) returned
✅ 421	21:57:28	SELECT t1.user_id,name,COUNT(*) FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY us...	5 row(s) returned
✅ 422	21:57:35	SELECT t1.user_id,name FROM orders t1 JOIN users t2 ON t1.user_id = t2.user_id GROUP BY user_id HAVI...	5 row(s) returned
✅ 423	21:58:26	INSERT INTO loyal_users (user_id,name) SELECT t1.user_id,name FROM orders t1 JOIN users t2 ON t1.user...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
✅ 424	21:58:33	SELECT * FROM zomato.loyal_users LIMIT 0, 1000	5 row(s) returned

Subquery in UPDATE

22 February 2023　　14:31

Populate the money col of loyal_cutomer table using the orders table. Provide a 10% app money to all customers based on their order value.

Subquery in DELETE

22 February 2023　　14:31

Delete all the customers record who have never ordered.

campusx - Warning - not supp... ×

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ▶ ▦ membership
- ▶ ▦ movies
- ▶ ▦ person1
- ▶ ▦ person2
- ▶ ▦ students
- ▶ ▦ titanic
- ▶ ▦ users
- ▦ Views
- ▦ Stored Procedures
- ▦ Functions
- ▶ ▤ sql_tasks
- ▶ ▤ swiggy
- ▶ ▤ temp_db
- ▶ ▤ test
- ▶ ▤ tinder
- ▼ ▤ zomato
  - ▼ ▦ Tables
    - ▶ ▦ delivery_partner
    - ▶ ▦ food
    - ▶ ▦ loyal_users
    - ▶ ▦ menu
    - ▶ ▦ order_details
    - ▶ ▦ orders
    - ▶ ▦ restaurants
    - ▶ ▦ users
  - ▦ Views
  - ▦ Stored Procedures
  - ▦ Functions

Administration Schemas

Information

Table: movies

Columns:
name   varchar(72)
rating   varchar(9)
genre   varchar(9)
year   int(4)
released   varchar(45)
score   decimal(2,1)
votes   decimal(8,1)

Object Info Session

movies users × orders order_details food menu delivery_partner restaurants loyal_users loyal_users

Limit to 1000 rows

```sql
1 • SELECT * FROM zomato.users;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: IA

user_id	name	email	password
1	Nitish	nitish@gmail.com	p252h
2	Khushboo	khushboo@gmail.com	hxn9b
3	Vartika	vartika@gmail.com	9hu7j
4	Ankit	ankit@gmail.com	lkko3
5	Neha	neha@gmail.com	3i7qm
6	Anupama	anupama@gmail.com	46rdw2
7	Rishabh	rishabh@gmail.com	4sw123

users 1 ×

Output

Action Output

#	Time	Action	Message
✓	428	22:03:42	SELECT SUM(amount)*0.1 FROM orders GROUP BY user_id LIMIT 0, 1000	5 row(s) returned
✓	429	22:03:51	SELECT * FROM zomato.loyal_users LIMIT 0, 1000	5 row(s) returned
✓	430	22:03:51	USE zomato	0 row(s) affected
✗	431	22:03:51	INSERT INTO loyal_users (user_id,name) SELECT t1.user_id,name FROM orders t1 JOIN users t2 ON t1.user...	Error Code: 1062. Duplicate entry '1' for key 'PRIMARY'
✗	432	22:04:43	UPDATE loyal_users SET money = (SELECT SUM(amount)*0.1 FROM orders GROUP BY user_id)	Error Code: 1242. Subquery returns more than 1 row
✓	433	22:07:00	SELECT user_id FROM users WHERE user_id NOT IN (SELECT DISTINCT(user_id) FROM orders) LIMIT 0, 1...	2 row(s) returned

Result Grid

Form Editor

```sql
20    FROM   (SELECT r_id,AVG(restaurant_rating) AS 'avg_rating'
21              FROM orders
22              GROUP BY r_id) t1 JOIN restaurants t2
23              ON t1.r_id = t2.r_id;
24
25
26 ·  DELETE FROM users
27    WHERE user_id IN (SELECT user_id FROM users
28    WHERE user_id NOT IN (SELECT DISTINCT(user_id) FROM orders))
29
30
31
```

Navigator: movies | users | orders | order_details | food | menu | delivery_partner | restaurants | loyal_users | loyal_users | users

Limit to 1000 rows

```
1 •  SELECT * FROM zomato.users;
```

SCHEMAS

- membership
- movies
- person1
- person2
- students
- titanic
- users
- Views
- Stored Procedures
- Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- **zomato**
  - Tables
    - delivery_partner
    - food
    - loyal_users
    - menu
    - order_details
    - orders
    - restaurants
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info Session

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

user_id	name	email	password
1	Nitish	nitish@gmail.com	p252h
2	Khushboo	khushboo@gmail.com	hxn9b
3	Vartika	vartika@gmail.com	9hu7j
4	Ankit	ankit@gmail.com	lkko3
5	Neha	neha@gmail.com	3i7qm

users 1

Output

Action Output

#	Time	Action	Message
✓ 430	22:03:51	USE zomato	0 row(s) affected
✗ 431	22:03:51	INSERT INTO loyal_users (user_id,name) SELECT t1.user_id,name FROM orders t1 JOIN users t2 ON t1.user...	Error Code: 1062. Duplicate entry '1' for key 'PRIMARY'
✗ 432	22:04:43	UPDATE loyal_users SET money = (SELECT SUM(amount)*0.1 FROM orders GROUP BY user_id)	Error Code: 1242. Subquery returns more than 1 row
✓ 433	22:07:00	SELECT user_id FROM users WHERE user_id NOT IN (SELECT DISTINCT(user_id) FROM orders) LIMIT 0, 1...	2 row(s) returned
✓ 434	22:07:58	DELETE FROM users WHERE user_id IN (SELECT user_id FROM users WHERE user_id NOT IN (SELECT ...	2 row(s) affected
✓ 435	22:08:02	SELECT * FROM zomato.users LIMIT 0, 1000	5 row(s) returned

```sql
  7    FROM orders t1
  8    JOIN users t2 ON t1.user_id = t2.user_id
  9    GROUP BY user_id
 10    HAVING COUNT(*) > 3;
 11
 12 •  UPDATE loyal_users
 13    SET money = (SELECT SUM(amount)*0.1
 14                 FROM orders
 15                 WHERE orders.user_id = loyal_users.user_id)
 16
 17
 18
 19
```

campusx - Warning - not supp... ×

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- membership
- movies
- person1
- person2
- students
- titanic
- users
- Views
- Stored Procedures
- Functions
- sql_tasks
- swiggy
- temp_db
- test
- tinder
- zomato
  - Tables
    - delivery_partner
    - food
    - loyal_users
    - menu
    - order_details
    - orders
    - restaurants
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

Table: movies

Columns:
name	varchar(72)
rating	varchar(9)
genre	varchar(9)
year	int(4)
released	varchar(45)
score	decimal(2,1)
votes	decimal(8,1)

Object Info Session

movies | users | orders | order_details | food | menu | delivery_partner | restaurants | loyal_users | loyal_users | users | loyal_users | loyal_users ×

Limit to 1000 rows

```
1 •  SELECT * FROM zomato.loyal_users;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: ‍

user_id	name	money
1	Nitish	167
2	Khushboo	267
3	Vartika	132
4	Ankit	180
5	Neha	304
NULL	NULL	304

loyal_users 1 ×

Output

Action Output

#	Time	Action	Message
434	22:07:58	DELETE FROM users WHERE user_id IN (SELECT user_id FROM users WHERE user_id NOT IN (SELECT ...	2 row(s) affected
435	22:08:02	SELECT * FROM zomato.users LIMIT 0, 1000	5 row(s) returned
436	22:11:56	UPDATE loyal_users SET money = (SELECT SUM(amount) FROM orders WHERE orders.user_id = loy...	5 row(s) affected Rows matched: 5 Changed: 5 Warnings: 0
437	22:12:01	SELECT * FROM zomato.loyal_users LIMIT 0, 1000	5 row(s) returned
438	22:12:19	UPDATE loyal_users SET money = (SELECT SUM(amount)*0.1 FROM orders WHERE orders.user_id ...	5 row(s) affected Rows matched: 5 Changed: 5 Warnings: 0
439	22:12:24	SELECT * FROM zomato.loyal_users LIMIT 0, 1000	5 row(s) returned