/\*

Aggregate Queries

REQUIREMENT - Use a multi-line comment to paste the first 5 or fewer results under your query

THEN records returned.

\*/

USE orderbook\_activity\_db;

-- #1: How many users do we have?

select count(\*)

from user;

Obraz zawierający tekst, Czcionka, linia, zrzut ekranu

Opis wygenerowany automatycznie

-- #2: List the username, userid, and number of orders each user has placed.

select u.uname, u.userid, count(o.orderid)

from user u

left join `order` o on u.userid = o.userid

group by u.userid;

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

-- #3: List the username, symbol, and number of orders placed for each user and for each symbol.

-- Sort results in alphabetical order by symbol.

select u.uname, o.symbol, count(o.orderid)

from user u

join `order` o on u.userid = o.userid

group by u.userid, o.symbol

order by o.symbol;

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Opis wygenerowany automatycznie

-- #4: Perform the same query as the one above, but only include admin users.

select u.uname, o.symbol, count(o.orderid)

from user u

join `order` o on u.userid = o.userid

join userroles ur on u.userid = ur.userid

join role r on ur.roleid = r.roleid

where r.name= 'admin'

group by u.userid, o.symbol

order by o.symbol;

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

-- #5: List the username and the average absolute net order amount for each user with an order.

-- Round the result to the nearest hundredth and use an alias (averageTradePrice).

-- Sort the results by averageTradePrice with the largest value at the top.

select u.uname, round(avg(abs(o.shares\*o.price)), 2) as averageTradePrice

from user u

join `order` o on u.userid = o.userid

group by u.uname

order by averageTradePrice desc;

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

-- #6: How many shares for each symbol does each user have?

-- Display the username and symbol with number of shares.

select u.uname, o.symbol, sum(o.shares)

from user u

join `order` o on u.userid = o.userid

group by u.uname, o.symbol;

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Opis wygenerowany automatycznie

-- #7: What symbols have at least 3 orders?

select o.symbol, count(o.orderid)

from `order` o

group by o.symbol

having count(o.orderid) >=3;

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

-- #8: List all the symbols and absolute net fills that have fills exceeding $100.

-- Do not include the WLY symbol in the results.

-- Sort the results by highest net with the largest value at the top.

select f.symbol, abs(sum(f.share \* f.price))

from fill f

where abs(f.share \* f.price) > 100 and f.symbol != 'WLY'

group by f.symbol

order by abs(sum(f.share \* f.price)) desc;

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

-- #9: List the top five users with the greatest amount of outstanding orders.

-- Display the absolute amount filled, absolute amount ordered, and net outstanding.

-- Sort the results by the net outstanding amount with the largest value at the top.

select u.uname, abs(sum(f.share)), abs(sum(o.shares)), abs(sum(o.shares)-sum(f.share))

from user u

join `order` o on u.userid = o.userid

left join fill f on o.orderid = f.orderid

group by u.uname

order by abs(sum(o.shares)-sum(f.share))

limit 5;

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie