שאילתות:

login and register:

SELECT \*

FROM users

WHERE(email = [www@gmail.com](mailto:www@gmail.com));

השאילתה הזו מחזירה את כל פרטי המשתמש

אשר כתובת המייל שלו זהה לכתובת המייל המופיעה.

INSERT INTO users set ?;

השאילתה הזו יוצרת משתמש חדש בטבלת users

ומכניסה את כל הפרטים המועברים אליה בסט.

data = {

idusers: users.idusers,

username: users.username,

password: users.password,

email: users.email,

age: users.age

};

data – דוגמה למבנה הסט שאותו נשלח.

שאילתות עדכון:

השאילתה הזו מעדכנת דירוג של משתמש אם כבר קיים דירוג של המשתמש עבור המתכון הנ"ל

INSERT INTO rating

VALUES("34", "3000", "5")

ON DUPLICATE KEY UPDATE rate="5"

אחרת, היא מכניסה דירוג חדש.

שאילתות מחיקה:

DELETE

FROM rating

WHERE(movieid=20000 AND userid=20)

השאילתה הזו מוחקת דירוג של משתמש מסוים (המתאים לiduser)

עבור סרט מסוים (המתאים movieid).

שאילתות חיפוש:

השאילתה הזו מחזירה את הid ושם הסרט שבו שם הסרט מכיל את המילה המופיעה.

למשל בדוגמה שהבאנו השאילתה מחזירה את כל הסרטים שבשם שלהם יש את המילה the.

SELECT id, name

FROM movie

WHERE(name LIKE '% the %' OR name LIKE '% the' OR name LIKE 'the %' OR name LIKE 'the");

השאילתה הזו מחזירה את הid ושם הסרט שהמשתמש דירג בדירוג 5 .

SELECT id, name

FROM movie, rating

WHERE(movie.id=rating.movieid AND iduser = 45 AND rate=5);

כלומר, מחזיר את הסרטים האהובים על המשתמש.

השאילתה הזו מחזירה את הid ושם הסרט שממוצע הדירוגים שלהם גדול מ4.

SELECT id, name

FROM rating, movie

WHERE(rating.movieid = movie.id)

Group by(movieid)

HAVING avg(rate) > 4;

כלומר, מחזירה את הסרטים המומלצים.