Problem Solving on SQL queries

1. Problem Statement

Create a database ecommerce and then create a table users, products, orders with the following constraints.

Users table contains below columns

Column	Data type	Constraints
id	BIGINT	AUTO_INCREMENT, NOT NULL, PRIMARY KEY
name	VARCHAR(255)	NOT NULL
category	VARCHAR(255)	NOT NULL
price	BIGINT	NOT NULL
createdAt	DATETIME	DEFAULT NOW()

Products table contains below

Column	Data type	Constraints
id	BIGINT	AUTO_INCREMENT, NOT NULL, PRIMARY KEY
email	VARCHAR(255)	NOT NULL
password	VARCHAR(255)	NOT NULL
createdAt	DATETIME	DEFAULT NOW()

Orders table contains below columns

Column	Data type	Constraints
id	BIGINT	AUTO_INCREMENT, NOT NULL, PRIMARY KEY
productId	BIGINT	NOT NULL, FOREIGN KEY
userId	BIGINT	NOT NULL, FOREIGN KEY
quantity	INT	NOT NULL
createdAt	DATETIME	DEFAULT NOW()
payment	BIT	NOT NULL, DEFAULT 0

Solution:

```
CREATE DATABASE ecommerce;
```

Explanation:

Create Database <**Database name**> is the syntax to create the Database

a) user table query

```
CREATE TABLE users (
        id BIGINT NOT NULL AUTO_INCREMENT,
        email VARCHAR(255) NOT NULL,
        password VARCHAR(255) NOT NULL,
         createdAt DATETIME DEFAULT NOW(),
        PRIMARY KEY(id)
);
```

b) products table query

```
CREATE TABLE products (
        id BIGINT NOT NULL AUTO_INCREMENT,
        name VARCHAR(255) NOT NULL,
        category VARCHAR(255) NOT NULL,
        price BIGINT NOT NULL,
        createdAt DATETIME DEFAULT NOW(),
        PRIMARY KEY(id)
);
```

c) orders table query

```
CREATE TABLE orders (
        id BIGINT NOT NULL AUTO_INCREMENT,
        productId BIGINT NOT NULL,
        userId BIGINT NOT NULL,
        quantity INT NOT NULL,
        createdAt DATETIME DEFAULT NOW(),
        payment BIT DEFAULT 0,
        PRIMARY KEY(id)
)
```

2. Problem Statement

a) To Insert the below data in the users table

email	password
abc@relevel.com	abcPassWord
xyz@relevel.com	xyzPassWord
mno@relevel.com	mnoPassWord
srk@relevel.com	srkPassWord
relevel@relevel.com	relevelPassWord

b) To Insert the below data in the products table

name	category	price
Pixel 4a	MobilePhone	30000
Pixel 4	MobilePhone	34000
Sony Bravia	Television	40000
Dell	Laptop	50000
Lenovo	Laptop	35000
Samsung s7	MobilePhone	70000

c) To Insert the below data in the orders table

productId	userId	Quantity
1	1	2
3	1	3
4	3	2
5	2	1
2	4	2

Solution:

a) user table insertion query

```
INSERT INTO users (email, password) VALUES
("abc@relevel.com", "abcPassWord"),
("xyz@relevel.com", "xyzPassWord"),
("mno@relevel.com", "mnoPassWord"),
("srk@relevel.com", "srkPassWord"),
("relevel@relevel.com", "relevelPassWord");
```

b) product table insertion query

```
    INSERT INTO products (name, category, price) VALUES
("Pixel 4a", "MobilePhone", 30000),
("Pixel 4", "MobilePhone", 34000),
("Sony Bravia", " Television ", 40000),
("Dell", "Laptop", 50000),
("Lenovo", "Laptop", 35000),
("Samsung s7", "MobilePhone", 70000);
```

c) orders table insertion query

```
    INSERT INTO orders (productId, userId, quantity) VALUES
(1, 1, 2),
(3, 1, 3),
(4, 3, 2),
(5, 2, 1),
(2, 4, 2);
```

3. Problem Statement

Login user

Task is to write a query to check the email and password in the user table and give custom message based on the below condition

Constraints	Message
Both email and password match	Logged in successfully
email match but password doesn’t match	Incorrect password
email not match	User not found

Input and output:

Input		Output
email	password	Message
relevel@relevel.com	relevelPassWord	Logged in successfully
relevel@relevel.com	relevelPasssWord	Incorrect credential
relevel1@relevel.com	relevelPasssWord	User not found

Solution:

a) Logged in successfully

```
SELECT CASE WHEN email IS NOT NULL THEN "Logged in
successfully" ELSE "" END AS message FROM users WHERE email =
"abc@relevel.com" AND password = "abcPassWord";
```

b) Incorrect credential

```
SELECT IF((SELECT email FROM users WHERE email = "
relevel@relevel.com” AND password = "abcPassWodrd") IS NULL,
"Incorrect password", "Logged in successfully") as result;
```

c) User not found

```
SELECT IF((SELECT email FROM users WHERE email =
"abc@relevel1.com") IS NULL, "User not found", "") as result;
```

4. Problem Statement

Write a query to get all the products from the products table and the products should be group by category and the products should be sorted based on the price in ascending order.

Expected Output:

1	Pixel 4a	MobilePhone	30000	2022-03-08 09:09:03
2	Pixel 4	MobilePhone	34000	2022-03-08 09:09:03
5	Lenovo	Laptop	35000	2022-03-08 09:09:03
3	Sony Bravia	Television	40000	2022-03-08 09:09:03
4	Dell	Laptop	50000	2022-03-08 09:09:03

Solution:

SELECT id, name, category, price, createdAt FROM products
GROUP BY category, name order by price;

Explanation:

GROUP BY – will group based on the category but that will give only one result for each group so again we are grouping based on the name.

ORDER BY – will order based on the price and by default it should sort in ascending order.

5. Problem Statement

Write a query to get the costliest Mobile phone from the product table

Expected Output:

6	Samsung s7	MobilePhone	70000	2022-03-08 21:29:17

Solution:

```
SELECT id, name, category, price, createdAt FROM ecommerce.products
WHERE category = "MobilePhone" AND price IN (SELECT MAX(price)
FROM products WHERE category = "MobilePhone");
```

Explanation:

In the query WHERE clause should be provided the category as “MobilePhone” to filter the data from the all data.

Sub query to get the Maximum price from the product list and the inner query result is passed to the outer query condition

6. Problem Statement

Write a query to get all the products and their details, user email which are added in the cart (using joins).

Expected Output:

name	email	category	quantity
Pixel 4a	abc@relevel.com	MobilePhone	2
Pixel 4	xyz@relevel.com	MobilePhone	3
Sony Bravia	mno@relevel.com	Television	2
Dell	srk@relevel.com	Laptop	1
Lenovo	relevel@relevel.com	Laptop	2

Solution:

```
select name, email, category, quantity from orders o join products p
on o.id = p.id
join users u on u.id = o.id
```

Explanation:

In the orders table we are only storing the id's from products and users table, Using joins will join the three table and allow to get the data from all 3 tables in a single query

Syntax: <source table> Join <destination table> on common key (table 1) = common key (table 2)

7. Problem Statement

Write a query to get the user and total price who has added the products which total cost is highest of all of them (using joins and sub query).

Expected Output:

email	total_price
xyz@relevel.com	102000

Solution:

```
select email, price*quantity as total_price  from orders o join products p on o.id = p.id
join users u on u.id = o.id WHERE price*quantity IN (SELECT MAX(price*quantity)  from orders o join products p on o.id = p.id join users u on u.id = o.id);
```

Explanation:

In the orders table we are only storing the id's from products and users table, Using joins will join the three table and allow to get the data from all 3 tables in a single query

In the query WHERE clause should be provided the price*quantity to filter the data from the all data.

Sub query to get the Maximum of price*quantity from the product list and the inner query result is passed to the outer query condition

Syntax: <source table> Join <destination table> on common key (table 1) = common key (table 2)

8. Problem Statement

Write a query to update all the quantity in the orders table by adding 1 with the existing quantity.

Then run a select query to see the updated value.

Expected Output:

id	productId	userId	quantity	createdAt	payment
1	1	1	3	2022-03-08 09:38:29	0
2	3	1	4	2022-03-08 09:38:29	0
3	4	3	3	2022-03-08 09:38:29	0
4	5	2	2	2022-03-08 09:38:29	0
5	2	4	3	2022-03-08 09:38:29	0

Solution:

```
    SET SQL_SAFE_UPDATES=0;
update orders set quantity = quantity + 1;
select * from orders;
```

Explanation:

In the above query we are setting the SQL_SAFE_UPDATES to 0 which means while updating we will get error in the sql.

Error - You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column To disable safe mode, toggle the option in Preferences -> SQL Editor -> Query Editor and reconnect.

To disabling the safe sql update will allow us to update the columns without where condition.

9. Problem Statement

Write a query to get the count of products in all the category and the result should be sorted by count in the ascending order. (use Aggregate function)

Expected Output:

category	count
Television	1
Laptop	2
MobilePhone	3

Solution:

select category, count(category) as count from products group by category order by count

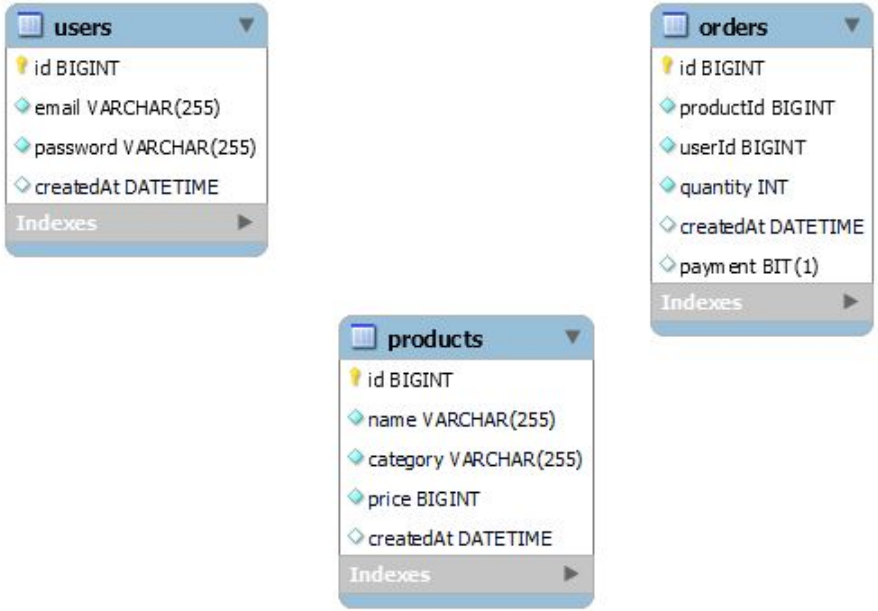
Explanation:

Group by – while using the count aggregate function the result should be grouped by some values so we are grouped by category and order the count in the ascending.

10. Problem Statement

Create ER diagram based on the table (Schema)

Expected Output:



Solution:

Go to database.

Database -> Reverse Engineer -> Next (Follow the steps) it will generate an ER Diagram for the schema

Explanation:

ER Diagram will plays an major role in finding the relationship between two tables and the data types of the columns

11. Problem Statement

Write a query to delete all the data in the orders columns without deleting the table structure.

Then run a select query to see the value in the columns.

Expected Output:

id	productId	userId	quantity	createdAt	payment

Solution:

```
Truncate table orders;
select * from orders;
```

Explanation:

Truncate - In the above query we are only deleting the values in the table without disturbing the table schema

12. Problem Statement (10mins)

Write a query to delete the table the orders without maintaining any schema

Expected Output:

Check in the database after deletion

Solution:

Delete table orders;

Explanation:

Delete - In the above query we are deleting the table

MCQ

1. Select the syntax of Single Line Comment.

A. .
B. !
C. --
D. #

Answer: C. –

2. Select the Aggregate function(s) among the following.

A. AVG()
B. FIRST()
C. LAST()
D. All of the above

Answer: D. All of the above

3. Which function returns the largest value of the column?

A. MIN()
B. MAX()
C. LARGE()
D. AVG()

MCQ

Answer: B. MAX()

4. Which of the following is not a valid SQL type?
A. DECIMAL
B. NUMERIC
C. FLOAT
D. CHARACTER
Answer: A. DECIMAL

5. Which operator is used to compare a value to a specified list of values?
A. BETWEEN
B. ANY
C. IN
D. ALL
Answer: C. IN

Thank You!