

Assignment on SQL concepts

Divyansh khatri

Q1. Create tables and keep up the good practices.

A.

Customer Table:

```
CREATE TABLE Hamen.Customer (
    CustomerId varchar(100) NOT NULL,
    FirstName varchar(100) NOT NULL,
    LastName varchar(100) NOT NULL,
    DOB DATE NOT NULL,
    Gender varchar(100) NOT NULL,
    MobileNo INT NOT NULL,
    CONSTRAINT Customer_PK PRIMARY KEY (CustomerId)
)
```

The screenshot shows the DBeaver Database Navigator interface. On the left, the Database Navigator pane displays a tree view of databases, schemas, and tables. In the center, the main workspace shows a grid view of the 'Customer' table. The table has columns: CustomerId, FirstName, LastName, DOB, Gender, MobileNo, and ExecutiveId. The data grid contains 10 rows of customer information. At the bottom, there are various navigation and search tools.

CustomerID	FirstName	LastName	DOB	Gender	MobileNo	ExecutiveID
1	Customer	A	1999-02-20	female	9,338,868,299	10
2	Customer	B	1999-02-23	male	9,338,328,299	7
3	Customer	B	1999-02-23	male	9,338,328,299	7
4	Customer	C	1989-06-30	female	9,338,328,299	5
5	Customer	D	1992-05-03	female	9,338,988,299	3
6	Customer	E	1995-08-21	male	9,338,327,699	2
7	Customer	F	1988-10-18	female	8,938,328,299	8
8	Customer	G	1979-11-19	female	9,338,868,299	1
9	Customer	H	1994-01-14	male	9,338,324,599	9
10	Customer	I	1997-03-17	male	9,338,328,299	4

Purchases Table:

CREATE TABLE Hamen.Purchases (

 PurchaseName **varchar(100) NOT NULL**,

 PurchaseDate **DATE NOT NULL**,

 NoOfUnits **INT NOT NULL**,

 CustomerId **varchar(100) NOT NULL**,

CONSTRAINT Purchases_FK **FOREIGN KEY** (CustomerId) **REFERENCES**

Hamen.Customer(CustomerId)

)

The screenshot shows the DBeaver Database Navigator interface. On the left, the Database Navigator pane displays the schema structure of the 'Hamen' database, including tables like Category, Customer, ExecutiveLocation, Location, ProductLocation, Products, and Purchases. The Purchases table is selected, showing its data grid. The grid contains the following data:

	ProductName	PurchaseDate	NoOfUnits	CustomerId
1	BookB	2021-02-10	6	2
2	TelevisionA	2021-02-10	6	2
3	ClothingA	2021-02-11	2	3
4	FridgeB	2021-02-09	3	4
5	GameA	2021-02-13	1	5
6	ClothingB	2021-02-02	10	6
7	PhoneA	2021-02-06	9	7
8	LifestyleB	2021-02-12	12	8

The 'CustomerId' column for row 6 is highlighted with a yellow background. The 'Category' column for row 6 is also highlighted with a yellow background. The bottom status bar indicates 'Inserted: 0 / Deleted: 0 / Updated: 1'.

Category Table:

```
CREATE TABLE Hamen.Category (
    CategoryCode varchar(100) NOT NULL,
    CategoryName varchar(100) NOT NULL,
    CONSTRAINT Category_PK PRIMARY KEY (CategoryCode)
)
```

The screenshot shows the DBeaver interface with the 'Category' table selected in the central workspace. The table has two columns: 'CategoryCode' and 'CategoryName'. The data is as follows:

CategoryCode	CategoryName
1	Books
2	Furniture
3	Games
4	Phones
5	Clothing
6	Lifestyle
7	Televisions
8	Fridge
9	Accessories

The left sidebar shows the database structure, including databases like 'DBeaver Sample', 'localhost - localhost:3306', and 'Hamen'. The 'Tables' section under 'Hamen' lists 'Category', 'Customer', 'ExecutiveLocation', 'Location', 'ProductLocation', 'Products', 'Purchases', and 'Sales Executive'.

ProductLocation Table:

```
CREATE TABLE Hamen.ProductLocation (
    ProductId varchar(100) NULL,
    LocationCode varchar(100) NULL,
    CONSTRAINT ProductLocation_FK FOREIGN KEY (LocationCode)
        REFERENCES Hamen.Location(LocationCode),
    CONSTRAINT ProductLocation_FK_1 FOREIGN KEY (ProductId)
        REFERENCES Hamen.Products(ProductId)
)
```

DBeaver Sample Database (SQLite) - localhost:3306

ProductLocation

	ProductId	LocationCode
1	1	B
2	2	B
3	3	A
4	4	F
5	5	E
6	6	C
7	7	D
8	8	A
9	9	B
10	10	B
11	11	C
12	12	A
13	13	B
14	14	C
15	15	D
16	16	E
17	17	F
18	18	D
19	19	D

LocationTable:

CREATE TABLE Hamen.Location (

LocationCode **varchar(100)** **NOT NULL**,

LocationName **varchar(100)** **NOT NULL**,

CONSTRAINT Location_PK **PRIMARY KEY** (LocationCode)

)

DBeaver Sample Database (SQLite) - localhost:3306

Location

	LocationCode	LocationName
1	A	Delhi
2	B	Mumbai
3	C	Hyderabad
4	D	Chennai
5	E	Bangalore
6	F	Pune

Sales Executive Table:

```
CREATE TABLE Hamen.`Sales Executive` (
    Executiveld varchar(100) NOT NULL,
    FirstName varchar(100) NOT NULL,
    LastName varchar(100) NOT NULL,
    DOB DATE NOT NULL,
    Gender varchar(100) NOT NULL,
    MobileNo INT NOT NULL,
    CONSTRAINT Sales_Executive_PK PRIMARY KEY (Executiveld)
)
```

The screenshot shows the DBeaver Database Navigator interface. On the left, the database structure is visible, including the Hamen schema which contains the Sales Executive table. The table has columns: Executiveld, FirstName, LastName, DOB, Gender, and MobileNo. The data grid displays 10 rows of sample data.

	Executiveld	FirstName	LastName	DOB	Gender	MobileNo
1	1	Sales	A	1997-10-12	male	9,868,228,970
2	10	Sales	J	1993-02-27	female	9,868,228,353
3	2	Sales	B	1997-10-12	male	9,868,228,971
4	3	Sales	C	1995-11-11	male	9,868,228,671
5	4	Sales	D	1997-01-11	male	4,982,323,983
6	5	Sales	E	1999-03-05	female	9,244,256,543
7	6	Sales	F	1986-05-08	female	9,244,345,543
8	7	Sales	G	1998-05-23	female	9,933,853,243
9	8	Sales	H	1998-09-30	male	9,338,753,243
10	9	Sales	I	1998-11-24	male	9,338,798,243

ExecutiveLocation Table:

```
CREATE TABLE Hamen.ExecutiveLocation (
    Executiveld varchar(100) NOT NULL,
    LocationCode varchar(100) NULL,
    CONSTRAINT ExecutiveLocation_FK FOREIGN KEY (Executiveld)
        REFERENCES Hamen.`Sales Executive`(Executiveld),
    CONSTRAINT ExecutiveLocation_FK_1 FOREIGN KEY (LocationCode)
        REFERENCES Hamen.Location(LocationCode)
)
```

The screenshot shows the DBeaver Database Navigator interface. The left sidebar displays a tree view of databases, tables, and schema. The main area shows the data grid for the 'ExecutiveLocation' table.

Table Data Grid:

Record	ExecutiveID	LocationCode
1	1	B
2	2	A
3	3	C
4	4	B
5	5	D
6	6	F
7	7	C
8	8	F
9	9	E
10	10	B

Toolbar: Save, Cancel, Script, Undo, Redo, Previous, Next, Last, First, Find, Replace, Sort, Filter, Refresh, 200, 10, Rows: 1, 10 row(s) fetched - 3ms.

Products Table:

CREATE TABLE Hamen.Products (

ProductId varchar(100) NOT NULL,

ProductName **varchar(100) NOT NULL**,

UnitPrice DECIMAL NOT NULL,

CategoryCode varchar(100) NOT NULL,

CONSTRAINT Products PK **PRIMARY KEY** (ProductId),

CONSTRAINT Products FK FOREIGN KEY (CategoryCode) REFERENCES

Hamen.Category(CategoryCode)

)

The screenshot shows the DBeaver Database Navigator interface. The left sidebar displays the database structure for 'DBeaver Sample Database (SQLITE)'. The 'Products' table is selected in the central grid, which shows the following data:

ProductID	ProductName	UnitPrice	CategoryCode
1	BookA	124	1
2	ClothingA	3,000	5
3	ClothingB	2,000	5
4	LifestyleA	300	6
5	LifestyleB	1,000	6
6	TelevisionA	35,000	7
7	TelevisionB	100,000	7
8	FridgeA	600,000	8
9	FridgeB	40,000	8
10	AccessoryA	100	9
11	AccessoryB	200	9
12	BookB	250	1
13	FurnitureA	10,000	2
14	FurnitureB	12,000	2
15	BookA	124	1
16	GameA	1,500	3
17	GameB	3,500	3
18	PhoneA	70,000	4
19	PhoneB	100,000	4

The bottom status bar indicates '19 row(s) fetched - 1ms (+1ms)'.

Q2. Write a query to retrieve the most sold product per day in a specific location in the last week. You can pick the location of your choice.

A. Query:

```
SELECT MAX(NoOfUnits) from Hamen.Purchases P, Hamen.Customer C,
Hamen.ExecutiveLocation EL
```

```
WHERE PurchaseDate >= curdate() - INTERVAL DAYOFWEEK(curdate())+6 DAY
AND PurchaseDate < curdate() - INTERVAL DAYOFWEEK(curdate())-1 DAY
AND P.CustomerId = C.CustomerId
AND C.Executiveld = EL.Executiveld
AND EL.LocationCode = "C"
```

Screenshot:

The screenshot shows the DBeaver Database Navigator interface. On the left, the Database Navigator pane displays the database schema, including tables like Category, Customer, ExecutiveLocation, Location, ProductLocation, Products, Purchases, and SalesExecutive. In the center, the SQL Editor pane contains the following query:

```
SELECT MAX(NoOfUnits), PurchaseDate
FROM Hamen.Purchases P, Hamen.Customer C, Hamen.ExecutiveLocation EL
WHERE PurchaseDate >= curdate() - INTERVAL DAYOFWEEK(curdate())+6 DAY
AND PurchaseDate < curdate() - INTERVAL DAYOFWEEK(curdate())-1 DAY
AND P.CustomerId = C.CustomerId
AND C.Executiveld = EL.Executiveld
AND EL.LocationCode = "C"
GROUP BY PurchaseDate;
```

Below the SQL editor, the Results pane shows the output of the query:

	MAX(NoOfUnits)	PurchaseDate
1	6	2021-02-10
2	3	2021-02-09

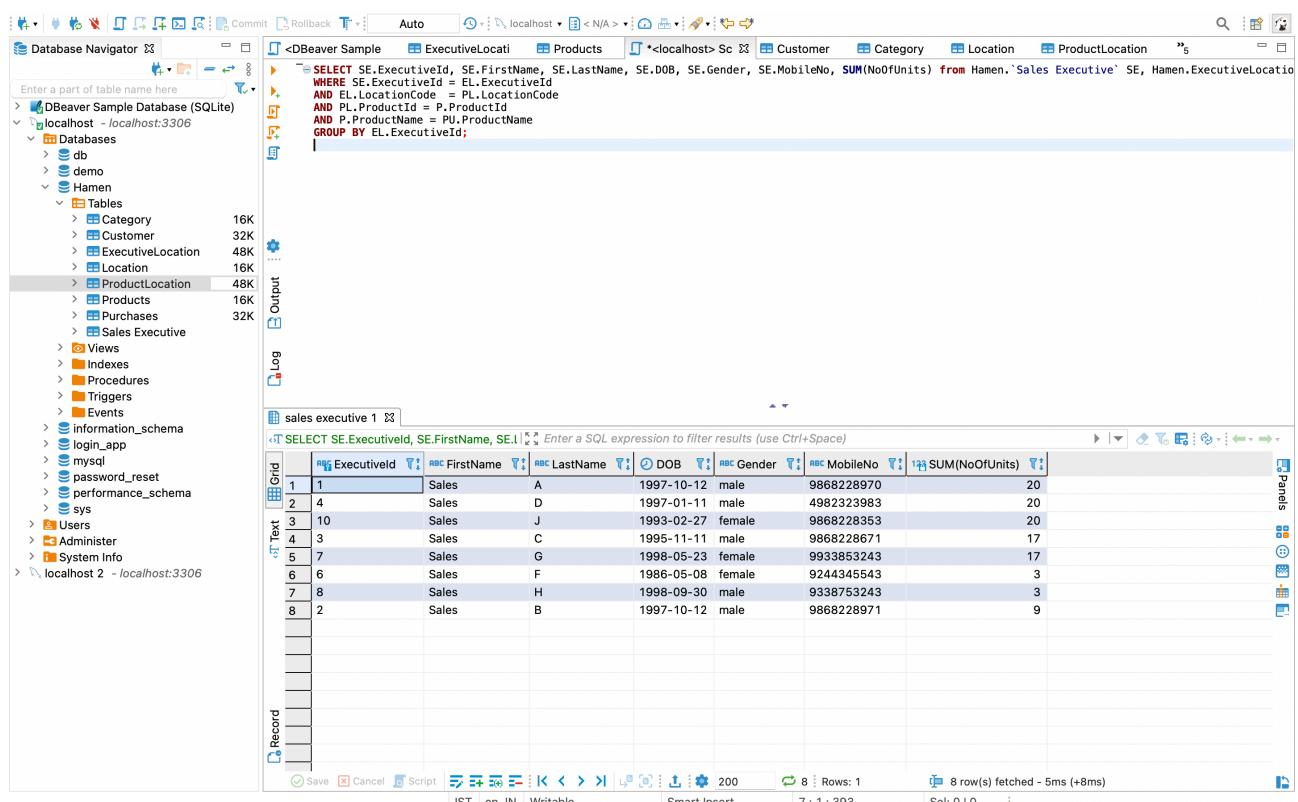
Q3. Write a query to list all the salesperson's details along with the count of products sold by them (if any) till the current date.

A. Query:

```
SELECT SE.Executiveld, SE.FirstName, SE.LastName, SE.DOB, SE.Gender,  
SE.MobileNo, SUM(NoOfUnits) from Hamen.`Sales Executive` SE,  
Hamen.ExecutiveLocation EL, Hamen.ProductLocation PL, Hamen.Products P,  
Hamen.Purchases PU
```

```
WHERE SE.Executiveld = EL.Executiveld  
AND EL.LocationCode = PL.LocationCode  
AND PL.ProductId = P.ProductId  
AND P.ProductName = PU.ProductName  
GROUP BY EL.Executiveld
```

Screenshot:



The screenshot shows the DBeaver SQL interface. On the left is the Database Navigator with a tree view of databases, tables, and views. In the center, a SQL editor window displays the query:

```
<DBeaver Sample> SELECT SE.Executiveld, SE.FirstName, SE.LastName, SE.DOB, SE.Gender,  
SE.MobileNo, SUM(NoOfUnits) from Hamen.`Sales Executive` SE,  
Hamen.ExecutiveLocation EL, Hamen.ProductLocation PL, Hamen.Products P,  
Hamen.Purchases PU  
WHERE SE.Executiveld = EL.Executiveld  
AND EL.LocationCode = PL.LocationCode  
AND PL.ProductId = P.ProductId  
AND P.ProductName = PU.ProductName  
GROUP BY EL.Executiveld;
```

Below the editor is a results grid titled "sales executive 1". The grid shows the following data:

	Executiveld	FirstName	LastName	DOB	Gender	MobileNo	SUM(NoOfUnits)
1	1	Sales	A	1997-10-12	male	9868228970	20
2	4	Sales	D	1997-01-11	male	4982323983	20
3	10	Sales	J	1993-02-27	female	9868228353	20
4	3	Sales	C	1995-11-11	male	9868228671	17
5	7	Sales	G	1998-05-23	female	9933853243	17
6	6	Sales	F	1986-05-08	female	9244345543	3
7	8	Sales	H	1998-09-30	male	9338753243	3
8	2	Sales	B	1997-10-12	male	9868228971	9