DBMS internal prep

1. **Database Schema for a Customer-Sale Scenario**

   a) **Create Tables:**

```
CREATE TABLE Customer (
    Cust_id INT PRIMARY KEY,
    Cust_name VARCHAR(50) NOT NULL
);

CREATE TABLE Item (
    Item_id INT PRIMARY KEY,
    Item_name VARCHAR(50) NOT NULL,
    Price INT CHECK (Price > 0)
);

CREATE TABLE Sale (
    Bill_no INT PRIMARY KEY,
    Bill_date DATE NOT NULL,
    Cust_id INT,
    Item_id INT,
    Qty_sold INT CHECK (Qty_sold > 0),
    FOREIGN KEY (Cust_id) REFERENCES Customer(Cust_id),
    FOREIGN KEY (Item_id) REFERENCES Item(Item_id)
);
```

   b) **Insert Records:**

```
INSERT INTO Customer VALUES (1, 'Alice'), (2, 'Bob'), (3,
'Charlie'), (4, 'David'), (5, 'Eve'), (6, 'Frank'), (7, 'G
race');
INSERT INTO Item VALUES (101, 'Laptop', 800), (102, 'Phon
e', 600), (103, 'Tablet', 300), (104, 'Monitor', 200), (10
5, 'Keyboard', 50), (106, 'Mouse', 25), (107, 'Printer', 1
```

```
50);
INSERT INTO Sale VALUES (201, CURDATE(), 1, 101, 1), (202,
CURDATE(), 2, 102, 2), (203, CURDATE(), 3, 103, 1), (204,
CURDATE(), 4, 104, 3), (205, CURDATE(), 5, 105, 4), (206,
CURDATE(), 6, 106, 5), (207, CURDATE(), 7, 107, 2);
```

c) **List Bills for Current Date:**

```
SELECT Sale.Bill_no, Customer.Cust_name, Sale.Item_id
FROM Sale
JOIN Customer ON Sale.Cust_id = Customer.Cust_id
WHERE Sale.Bill_date = CURDATE();
```

d) **Customers Who Bought Products Priced > $200:**

```
SELECT DISTINCT Customer.*
FROM Customer
JOIN Sale ON Customer.Cust_id = Sale.Cust_id
JOIN Item ON Sale.Item_id = Item.Item_id
WHERE Item.Price > 200;
```

e) **Count of Products Bought by Each Customer:**

```
SELECT Cust_id, COUNT(Item_id) AS Product_Count
FROM Sale
GROUP BY Cust_id;
```

f) **Create View for Bill Details:**

```
CREATE VIEW Bill_Details AS
SELECT Bill_no, Bill_date, Cust_id, Item_id, Price, Qty_so
ld,
       (Qty_sold * Price) AS Amount
FROM Sale
JOIN Item ON Sale.Item_id = Item.Item_id;
```

2. **Database Schema for a Student Library Scenario**

a) **Create Tables:**

```
CREATE TABLE Student (
    Stud_no INT PRIMARY KEY,
    Stud_name VARCHAR(50) NOT NULL
);

CREATE TABLE Membership (
    Mem_no INT PRIMARY KEY,
    Stud_no INT,
    FOREIGN KEY (Stud_no) REFERENCES Student(Stud_no)
);

CREATE TABLE Book (
    Book_no INT PRIMARY KEY,
    Book_name VARCHAR(50) NOT NULL,
    Author VARCHAR(50) NOT NULL
);

CREATE TABLE Iss_rec (
    Iss_no INT PRIMARY KEY,
    Iss_date DATE NOT NULL,
    Mem_no INT,
    Book_no INT,
    FOREIGN KEY (Mem_no) REFERENCES Membership(Mem_no),
    FOREIGN KEY (Book_no) REFERENCES Book(Book_no)
);
```

b) **Insert Records:**

```
INSERT INTO Student VALUES (1,'Alice'),(2,'Bob'),(3,'Charl
ie'),(4,'David'),(5,'Eve'),(6,'Frank'),(7,'Grace');
INSERT INTO Membership VALUES (101,1),(102,2),(103,3),(10
4,4),(105,5),(106,6),(107,7);
```

```
INSERT INTO Book VALUES (201,'DBMS','Rama Krishna'),(20
2,'Java','Robett Wilkins'),(203,'Harry Potter','JK Rowlin
g'),(204,'C++','Bjarne Stroustrup'),(205,'Python','Guido v
an Rossum'),(206,'Web Development','John Doe'),(207,'Machi
ne Learning','Andrew Ng');
INSERT INTO Iss_rec VALUES (301,CURDATE(),101,201),(302,CU
RDATE(),102,202),(303,CURDATE(),103,203),(304,CURDATE(),10
4,204),(305,CURDATE(),105,205),(306,CURDATE(),106,206),(30
7,CURDATE(),107,207);
```

c) **List Issues for Current Date:**

```
SELECT Student.Stud_name, Book.Book_name
FROM Iss_rec
JOIN Membership ON Iss_rec.Mem_no = Membership.Mem_no
JOIN Student ON Membership.Stud_no = Student.Stud_no
JOIN Book ON Iss_rec.Book_no = Book.Book_no
WHERE Iss_rec.Iss_date = CURDATE();
```

d) **Count of Books Borrowed by Each Student:**

```
SELECT Stud_no, COUNT(Book_no) AS Book_Count
FROM Iss_rec
GROUP BY Stud_no;
```

e) **List Books Issued as of Today:**

```
SELECT * FROM Book
WHERE Book_no IN (
    SELECT Book_no FROM Iss_rec WHERE Iss_date <= CURDATE
()
);
```

f) **Create View for Issue Records:**

```
CREATE VIEW Issue_Details AS
SELECT Iss_no, Iss_date, Student.Stud_name AS Stud_Name ,
Book.Book_name AS Book_Name
FROM Iss_rec
JOIN Membership ON Iss_rec.Mem_no = Membership.Mem_no
JOIN Student ON Membership.Stud_no = Student.Stud_no
JOIN Book ON Iss_rec.Book_no = Book.Book_no;
```

3. **Database Schema for a Video Library Scenario**

   a) **Create Tables:**

```
CREATE TABLE Customer (
    Cust_no INT PRIMARY KEY,
    Cust_name VARCHAR(50) NOT NULL
);

CREATE TABLE Membership (
    Mem_no INT PRIMARY KEY,
    Cust_no INT,
    FOREIGN KEY (Cust_no) REFERENCES Customer(Cust_no)
);

CREATE TABLE Cassette (
    Cass_no INT PRIMARY KEY,
    Cass_name VARCHAR(50),
    Language VARCHAR(30)
);

CREATE TABLE Iss_rec (
    Iss_no INT PRIMARY KEY,
    Iss_date DATE NOT NULL,
    Mem_no INT,
    Cass_no INT,
    FOREIGN KEY (Mem_no) REFERENCES Membership(Mem_no),
```

```
    FOREIGN KEY (Cass_no) REFERENCES Cassette(Cass_no)
);
```

b) **Insert Records:**

```
INSERT INTO Customer VALUES (1,'Alice'),(2,'Bob'),(3,'Char
lie'),(4,'David'),(5,'Eve'),(6,'Frank'),(7,'Grace');
INSERT INTO Membership VALUES (101 ,1),(102 ,2),(103 ,3),
(104 ,4),(105 ,5),(106 ,6),(107 ,7);
INSERT INTO Cassette VALUES (201 ,'The Legend','English'),
(202 ,'Titanic','English'),
                            (203 ,'Avatar','English'),(20
4 ,'Inception','English'),
                            (205 ,'Frozen','English'),(20
6 ,'Coco','Spanish'),
                            (207 ,'Spirited Away','Japane
se');
INSERT INTO Iss_rec VALUES (301,CURDATE(),101 ,201),(302,C
URDATE(),102 ,202),
                            (303,CURDATE(),103 ,203),(30
4,CURDATE(),104 ,204),
                            (305,CURDATE(),105 ,205),(30
6,CURDATE(),106 ,206),
                            (307,CURDATE(),107 ,207);
```

c) **List Issues for Current Date:**

```
SELECT Customer.Cust_name ,Cassette.Cass_name
FROM Iss_rec
JOIN Membership ON Iss_rec.Mem_no = Membership.Mem_no
JOIN Customer ON Membership.Cust_no = Customer.Cust_no
JOIN Cassette ON Iss_rec.Cass_no = Cassette.Cass_no
WHERE Iss_rec.Iss_date = CURDATE();
```

d) **Details of Customers Who Borrowed "The Legend":**

```
SELECT * FROM Customer
WHERE Cust_no IN (
    SELECT Membership.Cust_no
    FROM Membership
    JOIN Iss_rec ON Membership.Mem_no = Iss_rec.Mem_no
    WHERE Iss_rec.Cass_no IN (
        SELECT Cassette.Cass_no
        FROM Cassette
        WHERE Cass_name = 'The Legend'
    )
);
```

e) **Count of Cassettes Borrowed by Each Customer:**

```
SELECT Cust_no , COUNT(Cassette.Cass_name ) AS Cassette_Co
unt
FROM Membership
JOIN Iss_rec ON Membership.Mem_no = Iss_rec.Mem_no
GROUP BY Cust_no;
```

f) **Create View for Issue Records:**

```
CREATE VIEW Video_Issues AS
SELECT Iss_rec.Iss_no ,Iss_rec.Iss_date ,Customer.Cust_nam
e ,Cassette.Cass_name
FROM Iss_rec
JOIN Membership ON Iss_rec.Mem_no = Membership.Mem_no
JOIN Customer ON Membership.Cust_No = Customer.Cust_No
JOIN Cassette ON Iss_rec.Cass_No = Cassette.Cass_No;
```

4. **Database Schema for an Employee-Pay Scenario**

   a) **Create Tables:**

```
CREATE TABLE Employee (
    Emp_id INT PRIMARY KEY,
    Emp_name VARCHAR(50)
);

CREATE TABLE Department (
    Dept_id INT PRIMARY KEY,
    Dept_name VARCHAR(50)
);

CREATE TABLE PayDetails (
    Emp_id INT,
    Dept_id INT,
    Basic DECIMAL CHECK(Basic >=0),
    Deductions DECIMAL CHECK(Deductions >=0),
    Additions DECIMAL CHECK(Additions >=0),
    DOJ DATE,
    FOREIGN KEY(Emp_id) REFERENCES Employee(Emp_id),
    FOREIGN KEY(Dept_id) REFERENCES Department(Dept_id)
);

CREATE TABLE Payroll (
    Emp_id INT,
    Pay_date DATE NOT NULL,
    FOREIGN KEY(Emp_id) REFERENCES Employee(Emp_id)
);
```

b) **Insert Records:**

```
INSERT INTO Employee VALUES (1,'Alice'),(2,'Bob'),(3,'Char
lie'),
                            (4,'David'),(5,'Eve'),(6,'Fra
nk'),
                            (7,'Grace');
```

```
INSERT INTO Department VALUES (101 ,'HR'),
                              (102 ,'Finance'),
                              (103 ,'IT'),
                              (104 ,'Sales'),
                              (105 ,'Marketing'),
                              (106 ,'Admin'),
                              (107 ,'Support');

INSERT INTO PayDetails VALUES
            (1 ,101 ,15000 ,2000 ,1000 ,'2020-01-01'),
            (2 ,102 ,18000 ,2500 ,1500 ,'2020-01-02'),
            (3 ,103 ,12000 ,1000 ,500 ,'2020-01-03'),
            (4 ,104 ,22000 ,3000 ,2000 ,'2020-01-04'),
            (5 ,105 ,16000 ,1500 ,700 ,'2020-01-05'),
            (6 ,106 ,13000 ,1200 ,600 ,'2020-01-06'),
            (7 ,107 ,17000 ,1800 ,800 ,'2020-01-07');

INSERT INTO Payroll VALUES
            (1 ,'2024-11-01'),
            (2 ,'2024-11-01'),
            (3 ,'2024-11-01'),
            (4 ,'2024-11-01'),
            (5 ,'2024-11-01'),
            (6 ,'2024-11-01'),
            (7 ,'2024-11-01');
```

c) **List Employee Details Department Wise:**

```
SELECT Employee.Emp_name ,
       Department.Dept_name ,
       PayDetails.Basic ,
       PayDetails.Deductions ,
       PayDetails.Additions
FROM Employee
```

```
JOIN PayDetails ON Employee.Emp_id = PayDetails.Emp_id
JOIN Department ON PayDetails.Dept_Id=Department.Dept_Id;
```

d) **Employees with Basic Salary Between $10K and $20K:**

```
SELECT *
FROM Employee
JOIN PayDetails ON Employee.Emp_ID=PayDetails.Emp_ID
WHERE Basic BETWEEN 10000 AND 20000;
```

e) **Count of Employees in Each Department:**

```
  SELECT Dept_Id ,
         COUNT(Emp_Id ) AS Emp_Count
  FROM PayDetails
  GROUP BY Dept_Id;
  ```

  f)** Create View for Net Salary of Employees:**
  ```sql
  CREATE VIEW Net_Salary AS
  SELECT Emp_Name ,
         Basic - Deductions + Additions AS Net_Salary
  FROM Employee
  JOIN PayDetails ON Employee.Emp_ID=PayDetails.Emp_ID;
  ```
```

5. **Database Schema for Sailors-Boats Scenario**

```
a)** Create Tables:**
```sql
CREATE TABLE Sailors (
    Sid INT PRIMARY KEY ,
    Sname VARCHAR(50 ) ,
```

```
    Rating INT CHECK(Rating BETWEEN 1 AND 10 ) ,
    Age REAL CHECK(Age >=18 )
);

CREATE TABLE Boats (
    Bid INT PRIMARY KEY ,
    Bname VARCHAR(50 ) ,
    Color VARCHAR(20 )
);

CREATE TABLE Reserves (
    Sid INT ,
    Bid INT ,
    Day DATE ,
    FOREIGN KEY(Sid ) REFERENCES Sailors(Sid ) ,
    FOREIGN KEY(Bid ) REFERENCES Boats(Bid )
);
```

b)** Insert Records:**

```sql
INSERT INTO Sailors VALUES
        (1 ,'Alice' ,8 ,25),
        (2 ,'Bob' ,9 ,22),
        (3 ,'Charlie' ,10 ,30),
        (4 ,'David' ,7 ,20),
        (5 ,'Eve' ,6 ,19),
        (6 ,'Frank' ,5 ,24),
        (7 ,'Grace' ,8.5 ,-23);

INSERT INTO Boats VALUES
        (101 ,'Boat A' ,'Red' ),
        (102 ,'Boat B' ,'Green' ),
        (103 ,'Boat C' ,'Blue' ),
        (104 ,'Boat D' ,'Yellow' ),
        (105 ,'Boat E' ,'Red' );
```

```
INSERT INTO Reserves VALUES
        ((1 ),101,CURDATE()),
        ((2 ),102,CURDATE()),
        ((3 ),103,CURDATE()),
        ((4 ),104,CURDATE()),
        ((5 ),105,CURDATE()),
        ((6 ),101,CURDATE()),
        ((7 ),102,CURDATE());
```

c)** Find Names of Sailors Who Reserved Boat Number 103:**

```sql
SELECT Sname
FROM Sailors
WHERE Sid IN
      ((SELECT Sid
       FROM Reserves
       WHERE Bid=103));
```

d)** Ages of Sailors Whose Name Begins and Ends with B:**

```sql
SELECT Age
FROM Sailors
WHERE Sname LIKE 'B%' AND Sname LIKE '%B';
```

e)** Names of Sailors Who Reserved Red or Green Boat:**

```sql
SELECT DISTINCT Sname
FROM Sailors s
      JOIN Reserves r ON s.Sid=r.Sid
      JOIN Boats b ON r.Bid=b.Bid
      WHERE b.Color IN ('Red', 'Green');
```

```
 ```


 f)** Sailor ID of Those with Rating of 10 or Reserved Boa
t Number 104:**
 ```sql
 SELECT DISTINCT Sid
 FROM Sailors
       WHERE Rating=10 OR Sid IN
             ((SELECT Sid
               FROM Reserves
               WHERE Bid=104));
  ```
```


Citations:
[1]
https://www.studocu.com/in/document/vellore-institute-of-technology/database-management-systems/dbms-manual/36190592
[2]
https://www.transtutors.com/questions/database-schema-for-a-student-lab-scenario-student-stud-no-integer-stud-name-string--838421.htm
[3]
https://vertabelo.com/blog/database-for-library-system/
[4]
https://www.mycompiler.io/view/7Bm73T3
[5]
https://dbmslabnmit.wordpress.com/2016/10/20/2-database-schema-for-a-student-library-scenario/
[6]
https://www.youtube.com/watch?v=gS_E0dqFUgo
[7]
https://www.studocu.com/in/document/mahatma-gandhi-university/dbms-and-no-sql/lib-record-vishal-v-nair124/45317785
```