Y. Asritha

Coding Challenge 3_RDBMS_CrimeManagement

1. Creating Database and Tables :

```
mysql> create database CrimeManagement;
Query OK, 1 row affected (0.02 sec)

mysql> use CrimeManagement;
Database changed
mysql> CREATE TABLE Crime (
    ->   CrimeID INT PRIMARY KEY,
    ->   IncidentType VARCHAR(255),
    ->   IncidentDate DATE,
    ->   Location VARCHAR(255),
    ->   Description TEXT,
    ->   Status VARCHAR(20)
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql> CREATE TABLE Victim (
    ->   VictimID INT PRIMARY KEY,
    ->   CrimeID INT,
    ->   Name VARCHAR(255),
    ->   Age INT,
    ->   ContactInfo VARCHAR(255),
    ->   Injuries VARCHAR(255),
    ->   FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)
    -> );
Query OK, 0 rows affected (0.05 sec)

mysql> CREATE TABLE Suspect (
    ->   SuspectID INT PRIMARY KEY,
    ->   CrimeID INT,
    ->   Name VARCHAR(255),
    ->   Age INT,
    ->   Description TEXT,
    ->   CriminalHistory TEXT,
    ->   FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)
    -> );
Query OK, 0 rows affected (0.06 sec)
```

2. Inserting values into tables

```
mysql> INSERT INTO Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)
    -> VALUES
    ->   (1, 'Robbery', '2023-09-15', '123 Main St, Cityville', 'Armed robbery at a convenience store', 'Open'),
    ->   (2, 'Homicide', '2023-09-20', '456 Elm St, Townsville', 'Investigation into a murder case', 'Under Investigation'),
    ->   (3, 'Theft', '2023-09-10', '789 Oak St, Villagetown', 'Shoplifting incident at a mall', 'Closed');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Victim (VictimID, CrimeID, Name, Age, ContactInfo, Injuries)
    -> VALUES
    ->   (1, 1, 'John Doe', 27, 'johndoe@example.com',  'Minor injuries'),
    ->   (2, 2, 'Jane Smith', 29, 'janesmith@example.com',  'Deceased'),
    ->   (3, 3, 'Alice Johnson', 26, 'alicejohnson@example.com',  'None');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Suspect (SuspectID, CrimeID, Name,Age, Description, CriminalHistory)
    -> VALUES
    ->   (1, 1, 'Robber 1', 32, 'Armed and masked robber', 'Previous robbery convictions'),
    ->   (2, 2, 'Unknown', 27, 'Investigation ongoing', NULL),
    ->   (3, 3, 'Suspect 1', 30, 'Shoplifting suspect', 'Prior shoplifting arrests');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

Solve the below queries:

1. Select all open incidents.

```
mysql> SELECT * FROM Crime WHERE Status = 'Open';
+---------+-------------+-------------+----------------------+-----------------------------------+--------+
| CrimeID | IncidentType | IncidentDate | Location             | Description                       | Status |
+---------+-------------+-------------+----------------------+-----------------------------------+--------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open   |
+---------+-------------+-------------+----------------------+-----------------------------------+--------+
1 row in set (0.00 sec)
```

2. Find the total number of incidents

```
mysql> SELECT COUNT(*) AS TotalIncidents FROM Crime;
+----------------+
| TotalIncidents |
+----------------+
|              3 |
+----------------+
1 row in set (0.00 sec)
```

3. List all unique incident types.

```
mysql> SELECT DISTINCT IncidentType FROM Crime;
+--------------+
| IncidentType |
+--------------+
| Robbery      |
| Homicide     |
| Theft        |
+--------------+
3 rows in set (0.00 sec)
```

4. Retrieve incidents that occurred between '2023-09-01' and '2023-09-10'.

```
mysql> SELECT * FROM Crime WHERE IncidentDate >= '2023-09-01' AND IncidentDate <= '2023-09-10';
+---------+-------------+-------------+-----------------------+------------------------------+--------+
| CrimeID | IncidentType | IncidentDate | Location              | Description                  | Status |
+---------+-------------+-------------+-----------------------+------------------------------+--------+
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall | Closed |
+---------+-------------+-------------+-----------------------+------------------------------+--------+
1 row in set (0.00 sec)
```

5. List persons involved in incidents in descending order of age.

```
mysql> SELECT Name, Age FROM Victim
    -> UNION ALL
    -> SELECT Name, Age FROM Suspect
    -> ORDER BY Age DESC;
+---------------+------+
| Name          | Age  |
+---------------+------+
| Robber 1      |   32 |
| Suspect 1     |   30 |
| Jane Smith    |   29 |
| John Doe      |   27 |
| Unknown       |   27 |
| Alice Johnson |   26 |
+---------------+------+
6 rows in set (0.00 sec)
```

6. Find the average age of persons involved in incidents.

```
mysql> SELECT AVG(Age) AS AverageAge
    -> FROM (
    ->   SELECT Age FROM Victim
    ->   UNION ALL
    ->   SELECT Age FROM Suspect
    -> ) AS PersonsInvolved;
+-------------+
| AverageAge |
+-------------+
|    28.5000 |
+-------------+
1 row in set (0.01 sec)
```

7. List incident types and their counts, only for open cases.

```
mysql> SELECT IncidentType, COUNT(*) AS IncidentCount
    -> FROM Crime
    -> WHERE Status = 'Open'
    -> GROUP BY IncidentType;
+--------------+---------------+
| IncidentType | IncidentCount |
+--------------+---------------+
| Robbery      |             1 |
+--------------+---------------+
1 row in set (0.00 sec)
```

8. Find persons with names containing 'Doe'.

```
mysql> SELECT *
    -> FROM (
    ->   SELECT Name FROM Victim
    ->   UNION ALL
    ->   SELECT Name FROM Suspect
    -> ) AS Persons
    -> WHERE Persons.Name LIKE '%Doe%';
+----------+
| Name     |
+----------+
| John Doe |
+----------+
1 row in set (0.01 sec)
```

9. Retrieve the names of persons involved in open cases and closed cases.

```
mysql> SELECT Victim.Name AS Person
    -> FROM Crime
    -> INNER JOIN Victim ON Crime.CrimeID = Victim.CrimeID
    -> WHERE Crime.Status = 'Open'
    -> UNION ALL
    -> SELECT Suspect.Name AS Person
    -> FROM Crime
    -> INNER JOIN Suspect ON Crime.CrimeID = Suspect.CrimeID
    -> WHERE Crime.Status = 'Closed';
+------------+
| Person     |
+------------+
| John Doe   |
| Suspect 1  |
+------------+
2 rows in set (0.01 sec)
```

10. List incident types where there are persons aged 30 or 35 involved.

```
mysql> SELECT c.IncidentType
    -> FROM Crime c
    -> INNER JOIN Victim v ON c.CrimeID = v.CrimeID
    -> WHERE v.Age IN (30, 35)
    -> UNION ALL
    -> SELECT c.IncidentType
    -> FROM Crime c
    -> INNER JOIN Suspect s ON c.CrimeID = s.CrimeID
    -> WHERE s.Age IN (30, 35);
+--------------+
| IncidentType |
+--------------+
| Theft        |
+--------------+
1 row in set (0.01 sec)
```

11. Find persons involved in incidents of the same type as 'Robbery'.

```
mysql> SELECT p.Name
    -> FROM Crime c
    -> INNER JOIN Victim p ON c.CrimeID = p.CrimeID
    -> WHERE c.IncidentType = 'Robbery'
    -> UNION ALL
    -> SELECT s.Name
    -> FROM Crime c
    -> INNER JOIN Suspect s ON c.CrimeID = s.CrimeID
    -> WHERE c.IncidentType = 'Robbery';
+----------+
| Name     |
+----------+
| John Doe |
| Robber 1 |
+----------+
2 rows in set (0.00 sec)
```

12. List incident types with more than one open case.

```
mysql> SELECT c.IncidentType
    -> FROM Crime c
    -> WHERE c.Status = 'Open'
    -> GROUP BY c.IncidentType
    -> HAVING COUNT(*) > 1;
Empty set (0.00 sec)
```

13. List all incidents with suspects whose names also appear as victims in other incidents.

```
mysql> SELECT c1.IncidentType, c1.IncidentDate, c1.Location
    -> FROM Crime c1
    -> INNER JOIN Suspect s ON c1.CrimeID = s.CrimeID
    -> WHERE EXISTS (
    ->   SELECT 1
    ->   FROM Crime c2
    ->   INNER JOIN Victim v ON c2.CrimeID = v.CrimeID
    ->   WHERE c2.CrimeID != c1.CrimeID AND v.Name = s.Name
    -> );
Empty set (0.01 sec)
```

14. Retrieve all incidents along with victim and suspect details.

```
mysql> SELECT c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status,
    ->        v.VictimID, v.Name AS VictimName, v.Age, v.ContactInfo, v.Injuries,
    ->        s.SuspectID, s.Name AS SuspectName, s.Age, s.Description, s.CriminalHistory
    -> FROM Crime c
    -> LEFT JOIN Victim v ON c.CrimeID = v.CrimeID
    -> LEFT JOIN Suspect s ON c.CrimeID = s.CrimeID;
+---------+-------------+------------+----------------------+-----------------------------------+--------------------+----------+--------------+------+----------
---------------+---------------+-----------+-------------+------+-----------------------+-----------------------------+
| CrimeID | IncidentType | IncidentDate | Location             | Description                       | Status             | VictimID | VictimName   | Age  | ContactIn
fo             | Injuries      | SuspectID | SuspectName | Age  | Description           | CriminalHistory             |
+---------+-------------+------------+----------------------+-----------------------------------+--------------------+----------+--------------+------+----------
---------------+---------------+-----------+-------------+------+-----------------------+-----------------------------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open               |        1 | John Doe     |   27 | johndoe@e
xample.com     | Minor injuries |         1 | Robber 1    |   32 | Armed and masked robber | Previous robbery convictions |
|       2 | Homicide     | 2023-09-20   | 456 Elm St, Townsville | Investigation into a murder case     | Under Investigation |        2 | Jane Smith   |   29 | janesmith
@example.com   | Deceased      |         2 | Unknown     |   27 | Investigation ongoing | NULL                        |
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall       | Closed             |        3 | Alice Johnson |   26 | alicejohn
son@example.com | None          |         3 | Suspect 1   |   30 | Shoplifting suspect   | Prior shoplifting arrests   |
+---------+-------------+------------+----------------------+-----------------------------------+--------------------+----------+--------------+------+----------
---------------+---------------+-----------+-------------+------+-----------------------+-----------------------------+
3 rows in set (0.00 sec)
```

15. Find incidents where the suspect is older than any victim.

```
mysql> SELECT c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status
    -> FROM Crime c
    -> INNER JOIN Suspect s ON c.CrimeID = s.CrimeID
    -> WHERE NOT EXISTS (
    ->   SELECT 1
    ->   FROM Victim v
    ->   WHERE c.CrimeID = v.CrimeID AND v.Age >= s.Age
    -> );
+---------+-------------+-------------+----------------------+-----------------------------------+--------+
| CrimeID | IncidentType | IncidentDate | Location             | Description                       | Status |
+---------+-------------+-------------+----------------------+-----------------------------------+--------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open   |
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall       | Closed |
+---------+-------------+-------------+----------------------+-----------------------------------+--------+
2 rows in set (0.00 sec)
```

16. Find suspects involved in multiple incidents:

```
mysql> SELECT s.Name, count(*) AS InvolvedInCount
    -> FROM Suspect s
    -> GROUP BY s.Name
    -> HAVING InvolvedInCount > 1;
Empty set (0.00 sec)
```

17. List incidents with no suspects involved.

```
mysql> SELECT c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status
    -> FROM Crime c
    -> LEFT JOIN Suspect s ON c.CrimeID = s.CrimeID
    -> WHERE s.SuspectID IS NULL;
Empty set (0.00 sec)
```

18. List all cases where at least one incident is of type 'Homicide' and all other incidents are of type 'Robbery'.

```
mysql> SELECT *
    -> FROM Crime c
    -> WHERE c.IncidentType = 'Homicide'
    -> OR (NOT EXISTS (
    ->   SELECT 1
    ->   FROM Crime c2
    ->   WHERE c2.CrimeID != c.CrimeID AND c2.IncidentType != 'Robbery'
    -> ) AND c.IncidentType = 'Robbery');
+---------+-------------+-------------+----------------------+-------------------------------+--------------------+
| CrimeID | IncidentType | IncidentDate | Location             | Description                   | Status             |
+---------+-------------+-------------+----------------------+-------------------------------+--------------------+
|       2 | Homicide     | 2023-09-20   | 456 Elm St, Townsville | Investigation into a murder case | Under Investigation |
+---------+-------------+-------------+----------------------+-------------------------------+--------------------+
1 row in set (0.00 sec)
```

19. Retrieve a list of all incidents and the associated suspects, showing suspects for each incident, or 'No Suspect' if there are none.

```
mysql> SELECT c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status,
    ->        COALESCE(s.Name, 'No Suspect') AS SuspectName
    -> FROM Crime c
    -> LEFT JOIN Suspect s ON c.CrimeID = s.CrimeID;
+---------+-------------+-------------+----------------------+-----------------------------------+--------------------+-------------+
| CrimeID | IncidentType | IncidentDate | Location             | Description                       | Status             | SuspectName |
+---------+-------------+-------------+----------------------+-----------------------------------+--------------------+-------------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open               | Robber 1    |
|       2 | Homicide     | 2023-09-20   | 456 Elm St, Townsville | Investigation into a murder case     | Under Investigation | Unknown     |
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall       | Closed             | Suspect 1   |
+---------+-------------+-------------+----------------------+-----------------------------------+--------------------+-------------+
3 rows in set (0.00 sec)
```

20.List all suspects who have been involved in incidents with incident types 'Robbery' or 'Assault

```
mysql> SELECT DISTINCT s.Name AS SuspectName
    -> FROM Suspect s
    -> INNER JOIN Crime c ON s.CrimeID = c.CrimeID
    -> WHERE c.IncidentType IN ('Robbery', 'Assault');
+-------------+
| SuspectName |
+-------------+
| Robber 1    |
+-------------+
1 row in set (0.00 sec)
```