

CODING CHALLENGE MY SQL

NAME :- SUSHANT KUMAR SINGH

EMAIL:- ssushant886@gmail.com

DATABASE NAME: - CRIME_MANAGEMENT (TASK4)

TABLE CREATED FORM GIVEN DATA

```
Type 'help;' or '\h' for help. Type '\c' to clear the current in
mysql> Create Database Crime_Management;
Query OK, 1 row affected (0.03 sec)

mysql> USE CRIME_MANAGEMENT;
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 (0.07 sec)

mysql> DESC CRIME;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CrimeID    | int           | NO   | PRI | NULL    |       |
| IncidentType | varchar(255)  | YES  |     | NULL    |       |
| IncidentDate | date          | YES  |     | NULL    |       |
| Location    | varchar(255)  | YES  |     | NULL    |       |
| Description | text          | YES  |     | NULL    |       |
| Status      | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)
```

```
mysql> CREATE TABLE Victim (
  -> VictimID INT PRIMARY KEY,
  -> CrimeID INT,
  -> Name VARCHAR(255),
  -> ContactInfo VARCHAR(255),
  -> Injuries VARCHAR(255),
  -> FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql> DESC VICTIM;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| VictimID   | int           | NO   | PRI | NULL    |       |
| CrimeID    | int           | YES  | MUL | NULL    |       |
| Name       | varchar(255)  | YES  |     | NULL    |       |
| ContactInfo | varchar(255)  | YES  |     | NULL    |       |
| Injuries   | varchar(255)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> CREATE TABLE Suspect (
```

```
mysql> CREATE TABLE Suspect (
  -> SuspectID INT PRIMARY KEY,
  -> CrimeID INT,
  -> Name VARCHAR(255),
  -> Description TEXT,
  -> CriminalHistory TEXT,
  -> FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)
  -> );
Query OK, 0 rows affected (0.05 sec)

mysql> DESC SUSPECT;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SuspectID   | int           | NO   | PRI | NULL    |       |
| CrimeID     | int           | YES  | MUL | NULL    |       |
| Name        | varchar(255)  | YES  |     | NULL    |       |
| Description  | text          | YES  |     | NULL    |       |
| CriminalHistory | text        | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

INSERTED VALUES INTO THE TABLES FROM DATABASE GIVEN IN QUESTIONS

```
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 (0.00 sec)

Records: 3 Duplicates: 0 Warnings: 0

```
mysql> select * from crime;
```

CrimeID	IncidentType	IncidentDate	Location	Description	Status
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

3 rows in set (0.00 sec)

```
mysql> INSERT INTO Victim (VictimID, CrimeID, Name, ContactInfo, Injuries)
```

```
mysql> INSERT INTO Victim (VictimID, CrimeID, Name, ContactInfo, Injuries)
-> VALUES (1, 1, 'John Doe', 'johndoe@example.com', 'Minor injuries'),
-> (2, 2, 'Jane Smith', 'janesmith@example.com', 'Deceased'),
-> (3, 3, 'Alice Johnson', 'alicejohnson@example.com', 'None');
```

Query OK, 3 rows affected (0.01 sec)

Records: 3 Duplicates: 0 Warnings: 0

```
mysql> select * from victim;
```

VictimID	CrimeID	Name	ContactInfo	Injuries
1	1	John Doe	johndoe@example.com	Minor injuries
2	2	Jane Smith	janesmith@example.com	Deceased
3	3	Alice Johnson	alicejohnson@example.com	None

3 rows in set (0.00 sec)

```
mysql> INSERT INTO Suspect (SuspectID, CrimeID, Name, Description, CriminalHistory)
-> VALUES
-> (1, 1, 'Robber 1', 'Armed and masked robber', 'Previous robbery convictions'),
-> (2, 2, 'Unknown', 'Investigation ongoing', NULL),
-> (3, 3, 'Suspect 1', 'Shoplifting suspect', 'Prior shoplifting arrests');
```

Query OK, 3 rows affected (0.01 sec)

Records: 3 Duplicates: 0 Warnings: 0

```
mysql> select *from suspect;
```

SuspectID	CrimeID	Name	Description	CriminalHistory
1	1	Robber 1	Armed and masked robber	Previous robbery convictions
2	2	Unknown	Investigation ongoing	NULL
3	3	Suspect 1	Shoplifting suspect	Prior shoplifting arrests

3 rows in set (0.00 sec)

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)  
  
mysql> _
```

2. Find the total number of incidents.

```
mysql> SELECT COUNT(*) AS Total_Incidents  
-> FROM CRIME;  
+-----+  
| Total_Incidents |  
+-----+  
| 3 |  
+-----+  
1 row in set (0.01 sec)  
  
mysql> _
```

3. List all unique incident types.

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

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

```
mysql> SELECT *
-> FROM CRIME
-> WHERE IncidentDate BETWEEN '2023-09-01' AND '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)

mysql>
```

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

```
mysql> Select * from victim order by Age DESC;
+-----+-----+-----+-----+-----+-----+
| VictimID | CrimeID | Name | ContactInfo | Injuries | Age |
+-----+-----+-----+-----+-----+-----+
| 3 | 3 | Alice Johnson | alicejohnson@example.com | None | 71 |
| 1 | 1 | John Doe | johndoe@example.com | Minor injuries | 56 |
| 2 | 2 | Jane Smith | janesmith@example.com | Deceased | 30 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> SELECT * FROM VICTIM;
+-----+-----+-----+-----+-----+-----+
| VictimID | CrimeID | Name | ContactInfo | Injuries | Age |
+-----+-----+-----+-----+-----+-----+
| 1 | 1 | John Doe | johndoe@example.com | Minor injuries | 56 |
| 2 | 2 | Jane Smith | janesmith@example.com | Deceased | 30 |
| 3 | 3 | Alice Johnson | alicejohnson@example.com | None | 71 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> _
```

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

```
ERROR 1054 (42S22): Unknown column 'V.CrimeID' in 'FROM clause'
mysql> SELECT AVG(V.Age) AS Average_age
-> FROM Victim V
-> JOIN Crime C ON V.CrimeID = C.CrimeID;
+-----+
| Average_age |
+-----+
| 52.3333 |
+-----+
1 row in set (0.00 sec)

mysql> _
```

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

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

mysql>
```

8. Find persons with names containing 'Doe'.

```
mysql> SELECT VictimID , Name
-> FROM Victim
-> WHERE Name LIKE '%Doe%'
-> UNION
-> SELECT SuspectID , Name
-> FROM Suspect
-> WHERE Name LIKE '%Doe%';
+-----+-----+
| VictimID | Name      |
+-----+-----+
| 1        | John Doe |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

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

```
mysql> SELECT DISTINCT v.name,C.status
-> from victim v
-> join crime c ON v.crimeID = C.CrimeID
-> WHERE c.status IN ('open','closed');
+-----+-----+
| name          | status |
+-----+-----+
| John Doe      | Open   |
| Alice Johnson | Closed |
+-----+-----+
2 rows in set (0.01 sec)

mysql>
```

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

```
-> WHERE V.Age IN (30, 35);
ERROR 1054 (42S22): Unknown column 'C.Incident_Type' in
mysql> SELECT DISTINCT C.IncidentType
-> FROM Victim V
-> JOIN Crime C ON V.CrimeID = C.CrimeID
-> WHERE V.Age IN (30, 35);
+-----+
| IncidentType |
+-----+
| Homicide     |
+-----+
1 row in set (0.00 sec)

mysql>
```

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

```
mysql> SELECT
->     Victim.Name AS VictimName,
->     Suspect.Name AS SuspectName,
->     Suspect.Description AS SuspectDescription,
->     Suspect.CriminalHistory AS SuspectCriminalHistory
-> FROM
->     Crime
-> JOIN
->     Victim ON Crime.CrimeID = Victim.CrimeID
-> JOIN
->     Suspect ON Crime.CrimeID = Suspect.CrimeID
-> WHERE
->     Crime.IncidentType = 'Robbery';
```

VictimName	SuspectName	SuspectDescription	SuspectCriminalHistory
John Doe	Robber 1	Armed and masked robber	Previous robbery convictions

1 row in set (0.00 sec)

```
mysql> █
```

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

```
Empty set (0.00 sec)

mysql> SELECT IncidentType,
->     COUNT(*) AS OpenCasesCount
-> FROM Crime
-> WHERE Status = 'Open'
-> HAVING
->     COUNT(*) > 1;
Empty set (0.00 sec)

mysql>
```

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

```
mysql> SELECT C.*, S.SuspectID, S.Name AS SuspectName, V.VictimID AS MatchingVictimID
-> FROM Crime C
-> JOIN Suspect S ON C.CrimeID = S.CrimeID
-> JOIN Victim V ON S.Name = V.Name AND S.CrimeID <> V.CrimeID;
Empty set (0.00 sec)

mysql>
```

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

```
mysql> SELECT C.CrimeID, C.IncidentType, C.Description,  
->      V.VictimID, V.Name AS VictimName,  
->      S.SuspectID, S.Name AS SuspectName, S.Description AS SuspectDescription  
-> FROM Crime C  
-> LEFT JOIN Victim V ON C.CrimeID = V.CrimeID  
-> LEFT JOIN Suspect S ON C.CrimeID = S.CrimeID;
```

CrimeID	IncidentType	Description	VictimID	VictimName	SuspectID	SuspectName	SuspectDescription
1	Robbery	Armed robbery at a convenience store	1	John Doe	1	Robber 1	Armed and masked robber
2	Homicide	Investigation into a murder case	2	Jane Smith	2	Unknown	Investigation ongoing
3	Theft	Shoplifting incident at a mall	3	Alice Johnson	3	Suspect 1	Shoplifting suspect

3 rows in set (0.01 sec)

```
mysql>
```

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

```
mysql> SELECT  
->      C.CrimeID,  
->      S.SuspectID,  
->      S.Name AS SuspectName,  
->      S.Age AS SuspectAge,  
->      V.VictimID,  
->      V.Name AS VictimName,  
->      V.Age AS VictimAge  
-> FROM  
->      Crime C  
-> JOIN  
->      Suspect S ON C.CrimeID = S.CrimeID  
-> JOIN  
->      Victim V ON C.CrimeID = V.CrimeID  
-> WHERE  
->      S.Age > ANY (SELECT Age FROM Victim WHERE CrimeID = C.CrimeID);
```

CrimeID	SuspectID	SuspectName	SuspectAge	VictimID	VictimName	VictimAge
3	3	Suspect 1	91	3	Alice Johnson	71

1 row in set (0.01 sec)

```
mysql>
```


16. Find suspects involved in multiple incidents:

```
mysql> SELECT
->     S.SuspectID,
->     S.Name AS SuspectName,
->     COUNT(DISTINCT C.CrimeID) AS NumIncidents
-> FROM
->     Suspect S
-> JOIN
->     Crime C ON S.CrimeID = C.CrimeID
-> GROUP BY
->     S.SuspectID, S.Name
-> HAVING
->     COUNT(DISTINCT C.CrimeID) > 1;
Empty set (0.00 sec)

mysql>
```

17. List incidents with no suspects involved.

```
mysql> SELECT
->     C.CrimeID,
->     S.SuspectID,
->     S.Name AS SuspectName,
->     S.Description AS SuspectDescription,
->     S.CriminalHistory AS SuspectCriminalHistory
-> FROM
->     Crime C
-> LEFT JOIN
->     Suspect S ON C.CrimeID = S.CrimeID
-> WHERE
->     S.SuspectID IS NULL OR S.Name = 'Unknown';
+-----+-----+-----+-----+-----+
| CrimeID | SuspectID | SuspectName | SuspectDescription | SuspectCriminalHistory |
+-----+-----+-----+-----+-----+
|      2 |      2 | Unknown    | Investigation ongoing | NULL                    |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

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

```
mysql> SELECT
->     C.CrimeID,
->     C.IncidentType
-> FROM
->     Crime C
-> WHERE
->     C.IncidentType = 'Homicide'
->     OR (C.IncidentType = 'Robbery' AND NOT EXISTS (
->         SELECT *
->         FROM Crime AS SubCrime
->         WHERE SubCrime.CrimeID = C.CrimeID
->             AND SubCrime.IncidentType = 'Homicide'
->     ));
+-----+-----+
| CrimeID | IncidentType |
+-----+-----+
|      1  | Robbery      |
|      2  | Homicide     |
+-----+-----+
2 rows in set (0.00 sec)

mysql> _
```

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.IncidentType,
->     S.Name AS SuspectName
-> FROM
->     Crime C
-> JOIN
->     Suspect S ON C.CrimeID = S.CrimeID;
+-----+-----+
| IncidentType | SuspectName |
+-----+-----+
| Robbery      | Robber 1    |
| Homicide     | Unknown     |
| Theft        | Suspect 1   |
+-----+-----+
3 rows in set (0.00 sec)

mysql> _
```

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

```
mysql> SELECT
->     S.SuspectID,
->     S.Name AS SuspectName
-> FROM
->     Suspect S
-> JOIN
->     Crime C ON S.CrimeID = C.CrimeID
-> WHERE
->     C.IncidentType IN ('Robbery', 'Assault');
```

```
+-----+-----+
| SuspectID | SuspectName |
+-----+-----+
|          1 | Robber 1    |
+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT
```