Crime Management Code Challenge

# SQL Queries 25/03/2025

-**Database Creation**

CREATE DATABASE CrimeManagement;

USE CrimeManagement;

-**Table Creation**

CREATE TABLE Crime (

CrimeID INT PRIMARY KEY,

IncidentType VARCHAR(255),

IncidentDate DATE,

Location VARCHAR(255),

Description TEXT,

Status VARCHAR(20)

);

CREATE TABLE Victim (

VictimID INT PRIMARY KEY,

CrimeID INT,

Name VARCHAR(255),

ContactInfo VARCHAR(255),

Injuries VARCHAR(255),

FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)

);

CREATE TABLE Suspect (

SuspectID INT PRIMARY KEY,

CrimeID INT,

Name VARCHAR(255),

Description TEXT,

CriminalHistory TEXT,

FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)

);

-**Values Insertion**

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');

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');

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');

--Adding 'Age' Attribute into Victim & Suspect

ALTER TABLE Victim ADD Age INT;

ALTER TABLE Suspect ADD Age INT;

UPDATE Victim

SET Age = CASE

WHEN VictimID = 1 THEN 35

WHEN VictimID = 2 THEN 40

WHEN VictimID = 3 THEN 28

ELSE NULL

END;

UPDATE Suspect

SET Age = CASE

WHEN SuspectID = 1 THEN 31

WHEN SuspectID = 2 THEN 45

WHEN SuspectID = 3 THEN 25

ELSE NULL

END;

Select \* from Crime;

Select \* from Victim;

Select \* from Suspect;

----------------------------------Tasks------------------------------------------

--1. Select all open incidents

SELECT \* FROM Crime WHERE Status = 'Open';

--2. Find the total number of incidents

SELECT COUNT(\*) as TotalNumOfIncidents from Crime;

--3. List all unique incident types

SELECT DISTINCT(IncidentType) from Crime;

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

SELECT CrimeID, IncidentType, IncidentDate from Crime

WHERE IncidentDate BETWEEN '2023-09-01' AND '2023-09-10';

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

SELECT v.Name AS Involvedpersons FROM Victim v

UNION ALL

SELECT s.Name FROM Suspect s

ORDER BY Involvedpersons DESC;

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

SELECT AVG(Age) AS AvgAge FROM (

SELECT Age FROM Victim

UNION ALL

SELECT Age FROM Suspect

) AS VictimAndSuspectAge;

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

SELECT IncidentType, COUNT(\*) as IncidentCount

FROM Crime

WHERE Status = 'Open'

GROUP BY IncidentType;

--8. Find persons with name containing 'Doe'

SELECT \* FROM Victim WHERE Name LIKE '%Doe%';

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

SELECT v.VictimID , v.Name AS VictimName, v.ContactInfo, s.Name AS SuspectName, c.Status

FROM Victim v

JOIN Crime c ON v.CrimeID = c.CrimeID

Join Suspect s ON c.CrimeID = s.CrimeID

WHERE c.Status IN ('Open', 'Closed');

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

SELECT DISTINCT c.IncidentType, v.Age

FROM Crime c

JOIN Victim v ON c.CrimeID = v.CrimeID

WHERE v.Age IN (30, 35)

UNION

SELECT DISTINCT c.IncidentType, s.Age

FROM Crime c

JOIN Suspect s ON c.CrimeID = s.CrimeID

WHERE s.Age IN (30, 35);

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

SELECT c.CrimeId, s.Name as SuspectName, v.Name AS VictimName

FROM Suspect s

Join Crime c ON s.CrimeID = c.CrimeID

Join Victim v ON c.CrimeID = v.CrimeID

WHERE c.IncidentType = 'Robbery';

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

SELECT IncidentType, COUNT(\*) AS OpenCases

FROM Crime

WHERE Status = 'Open'

GROUP BY IncidentType

HAVING COUNT(\*) > 1;

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

SELECT c.\*

from Crime c

Join Suspect s ON c.CrimeID = s.CrimeID

Join Victim v ON s.Name = v.Name;

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

SELECT c.\*, v.\*, s.\*

FROM Crime c

Join Victim v ON c.CrimeID = v.CrimeID

Join Suspect s ON c.CrimeID = s.CrimeID;

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

SELECT v.Age as VictimAge, s.Age as SuspectAge, c.CrimeID, c.IncidentType FROM Crime c

JOIN Suspect s ON c.CrimeID = s.CrimeID

JOIN Victim v ON c.CrimeID = v.CrimeID

WHERE s.Age > v.Age;

--16. Find suspects involved in multiple incidents

SELECT Name, COUNT(CrimeID) As IncidentCounts

FROM Suspect

GROUP BY Name

HAVING COUNT(CrimeID) > 1;

--17. List incidents with no suspects involved

SELECT \* FROM Crime c

LEFT JOIN Suspect s ON c.CrimeID = s.CrimeID

WHERE s.SuspectID IS NULL;

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

SELECT \*

FROM Crime

WHERE IncidentType = 'Homicide'

OR (IncidentType = 'Robbery' AND CrimeID NOT IN (

SELECT CrimeID FROM Crime WHERE IncidentType = 'Homicide')); --Outputs cases containing homicide exists & all are Robbery

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

SELECT c.CrimeID, c.IncidentType, c.IncidentDate, c.Status, s.Name AS SuspectName

FROM Crime c

JOIN Suspect s ON c.CrimeID = s.CrimeID;

-- All the incidents have associated suspects

-- Query 2

SELECT c.CrimeID, c.IncidentType,

CASE

WHEN s.Name IS NULL THEN 'No Suspect'

ELSE s.Name

END AS SuspectName

FROM Crime c

LEFT JOIN Suspect s ON c.CrimeID = s.CrimeID;

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

SELECT c.CrimeId, s.SuspectID, s.Name

FROM Suspect s

JOIN Crime c ON c.CrimeID = s.CrimeID

WHERE c.IncidentType = 'Robbery' OR c.IncidentType = 'Assault';