**Coding Challenge SQL**

**Name: Dhruv Umesh Sompura**

**ID: JA311**

**Topic: Crime Management Schema DDL and DML**

**-- creating tables**

**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)**

**);**

**-- inserting values**

**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');**

**-- altering tables**

**-- as some questions require age as a parameter (q5,q6,q10,q15) i am adding it as a column**

**ALTER TABLE Victim ADD Age INT;**

**ALTER TABLE Suspect ADD Age INT;**

**UPDATE Victim SET Age = 35 WHERE VictimID = 1;**

**UPDATE Victim SET Age = 40 WHERE VictimID = 2;**

**UPDATE Victim SET Age = 25 WHERE VictimID = 3;**

**UPDATE Suspect SET Age = 30 WHERE SuspectID = 1;**

**UPDATE Suspect SET Age = 45 WHERE SuspectID = 2;**

**UPDATE Suspect SET Age = 27 WHERE SuspectID = 3;**

**-- 1. Select all open incidents**

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

**-- 2. Find the total number of incidents**

**SELECT COUNT(\*) AS TotalIncidents 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 \* FROM Crime WHERE IncidentDate BETWEEN '2023-09-01' AND '2023-09-10';**

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

**-- using new column age**

**SELECT Name, Age FROM Victim**

**UNION ALL**

**SELECT Name, Age FROM Suspect**

**ORDER BY Age DESC;**

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

**-- using new column age**

**SELECT AVG(Age) AS AverageAge FROM (**

**SELECT Age FROM Victim**

**UNION ALL**

**SELECT Age FROM Suspect**

**) AS q5;**

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

**SELECT IncidentType, COUNT(\*) AS Count**

**FROM Crime**

**WHERE Status = 'Open'**

**GROUP BY IncidentType;**

**-- 8. Find persons with names containing 'Doe'**

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

**UNION**

**SELECT Name FROM Suspect WHERE Name LIKE '%Doe%';**

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

**SELECT Name FROM Victim WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Open')**

**UNION**

**SELECT Name FROM Suspect WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Open')**

**UNION**

**SELECT Name FROM Victim WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Closed')**

**UNION**

**SELECT Name FROM Suspect WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Closed');**

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

**-- using new column age**

**SELECT DISTINCT IncidentType**

**FROM Crime**

**WHERE CrimeID IN (**

**SELECT CrimeID FROM Victim WHERE Age IN (30, 35)**

**UNION**

**SELECT CrimeID FROM Suspect WHERE Age IN (30, 35)**

**);**

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

**SELECT CrimeID, Name FROM Victim WHERE CrimeID IN (**

**SELECT CrimeID FROM Crime WHERE IncidentType = 'Robbery'**

**)**

**UNION**

**SELECT CrimeID, Name FROM Suspect WHERE CrimeID IN (**

**SELECT CrimeID FROM Crime WHERE IncidentType = 'Robbery'**

**);**

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

**SELECT IncidentType, COUNT(\*) AS Count**

**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 Crime.\***

**FROM Crime**

**JOIN Suspect ON Crime.CrimeID = Suspect.CrimeID**

**WHERE Suspect.Name IN (SELECT Name FROM Victim WHERE Victim.CrimeID <> Suspect.CrimeID);**

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

**SELECT Crime.\*, Victim.\*, Suspect.\***

**FROM Crime**

**LEFT JOIN Victim ON Crime.CrimeID = Victim.CrimeID**

**LEFT JOIN Suspect ON Crime.CrimeID = Suspect.CrimeID;**

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

**-- using new column age**

**SELECT Crime.\***

**FROM Crime**

**JOIN Victim ON Crime.CrimeID = Victim.CrimeID**

**JOIN Suspect ON Crime.CrimeID = Suspect.CrimeID**

**WHERE Suspect.Age > Victim.Age;**

**-- 16. Find suspects involved in multiple incidents**

**SELECT**

**Suspect.SuspectID,**

**Suspect.Name,**

**Suspect.Description,**

**Suspect.CriminalHistory,**

**COUNT(DISTINCT Suspect.CrimeID) AS Count**

**FROM**

**Suspect**

**GROUP BY**

**Suspect.SuspectID,**

**Suspect.Name,**

**Suspect.Description,**

**Suspect.CriminalHistory**

**HAVING**

**COUNT(DISTINCT Suspect.CrimeID) > 1;**

**-- 17. List incidents with no suspects involved**

**SELECT \* FROM Crime WHERE CrimeID NOT IN (SELECT CrimeID FROM Suspect);**

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

**SELECT CrimeID, IncidentType, IncidentDate, Location, Description, Status**

**FROM Crime**

**WHERE CrimeID IN (**

**SELECT CrimeID**

**FROM Crime**

**GROUP BY CrimeID**

**HAVING SUM(CASE WHEN IncidentType = 'Homicide' THEN 1 ELSE 0 END) > 0**

**)**

**OR IncidentType = 'Robbery'**

**ORDER BY CrimeID;**

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

**SELECT Crime.CrimeID, Crime.IncidentType, IFNULL(Suspect.Name, 'No Suspect') AS SuspectName**

**FROM Crime**

**LEFT JOIN Suspect ON Crime.CrimeID = Suspect.CrimeID;**

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

**SELECT Suspect.Name**

**FROM Suspect**

**JOIN Crime ON Suspect.CrimeID = Crime.CrimeID**

**WHERE Crime.IncidentType IN ('Robbery', 'Assault');**

**Thank You**