CRIME MANAGEMENT SQL CODE CHALLLENGE ( J531 VINAYAK SONI )

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

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

SELECT AVG(Age) AS AverageAge FROM (

SELECT Age FROM Victim

UNION ALL

SELECT Age FROM Suspect

) AS AllPersons;

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

SELECT IncidentType, COUNT(\*) AS CountOfIncidents

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' OR Status = 'Closed')

UNION

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

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

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 Victim.Name FROM Victim WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE IncidentType = 'Robbery')

UNION

SELECT Suspect.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 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 \* FROM Crime WHERE CrimeID IN (

SELECT S.CrimeID FROM Suspect S

JOIN Victim V ON S.Name = V.Name

);

14. Retrieve all incidents along with victim and suspect details

SELECT C.\*, V.Name AS VictimName, V.ContactInfo, V.Age AS VictimAge,

S.Name AS SuspectName, S.Description AS SuspectDescription, S.Age AS SuspectAge

FROM Crime C

LEFT JOIN Victim V ON C.CrimeID = V.CrimeID

LEFT JOIN Suspect S ON C.CrimeID = S.CrimeID;

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

SELECT \* FROM Crime C WHERE C.CrimeID IN (

SELECT CrimeID FROM Suspect S

WHERE S.Age > (SELECT (V.Age) FROM Victim V WHERE V.CrimeID = S.CrimeID)

);

//couldn’t solve fully The right answers were not coming //

16. Find suspects involved in multiple incidents

SELECT Name, COUNT(\*) AS IncidentCount

FROM Suspect

GROUP BY Name

HAVING COUNT(\*) > 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 \* FROM Crime WHERE IncidentType = 'Homicide'

UNION

SELECT \* FROM Crime WHERE IncidentType = '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, S.name //cant solve properly as Null values showed//

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 S.Name, C.IncidentType

FROM Suspect S

JOIN Crime C ON S.CrimeID = C.CrimeID

WHERE C.IncidentType IN ('Robbery', 'Assault');