

-- Create database

CREATE DATABASE IF NOT EXISTS CrimeManagement;

USE CrimeManagement;

-- Create 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)

);

-- Insert sample data

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');

select * from Crime;

select * from Victim;

select * from Suspect;

Queries

-- 1. Select all open incidents.

```
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
*	NULL	NULL	NULL	NULL	NULL	NULL

-- 2. Find the total number of incidents.

```
SELECT COUNT(*) AS TotalIncidents FROM Crime;
```

	TotalIncidents
▶	3

-- 3. List all unique incident types.

```
SELECT DISTINCT IncidentType FROM Crime;
```

	IncidentType
▶	Robbery
	Homicide
	Theft

-- 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';
```

Result Grid Filter Rows: Edit: Export/Import: Wrap						
	CrimeID	IncidentType	IncidentDate	Location	Description	Status
▶	3	Theft	2023-09-10	789 Oak St, Villagetown	Shoplifting incident at a mall	Closed
*	NULL	NULL	NULL	NULL	NULL	NULL

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

```
SELECT * FROM Victim
```

```
UNION
```

```
SELECT * FROM Suspect
ORDER BY VictimID DESC, SuspectID DESC;
```

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

-- Assuming age is not provided, and the average is based on VictimID and SuspectID.

```
SELECT AVG(PersonAge) AS AverageAge
FROM (
    SELECT VictimID AS PersonID, VictimID AS PersonAge FROM Victim
    UNION
    SELECT SuspectID, SuspectID FROM Suspect
) AS Persons;
```

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

```
SELECT IncidentType, COUNT(*) AS IncidentCount
FROM Crime
WHERE Status = 'Open'
GROUP BY IncidentType;
```

	IncidentType	IncidentCount
►	Robbery	1

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

```
SELECT * FROM Victim WHERE Name LIKE '%Doe%'
UNION
SELECT * FROM Suspect WHERE Name LIKE '%Doe%';
```

	VictimID	CrimeID	Name	ContactInfo	Injuries
►	1	1	John Doe	johndoe@example.com	Minor injuries

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

```
SELECT Name, 'Victim' AS PersonType FROM Victim WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Open')
```

UNION

```
SELECT Name, 'Suspect' AS PersonType FROM Suspect WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Open')
```

UNION

```
SELECT Name, 'Victim' AS PersonType FROM Victim WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Closed')
```

UNION

```
SELECT Name, 'Suspect' AS PersonType FROM Suspect WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE Status = 'Closed');
```

	Name	PersonType
▶	John Doe	Victim
	Robber 1	Suspect
	Alice Johnson	Victim
	Suspect 1	Suspect

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

```
SELECT DISTINCT IncidentType
```

```
FROM (
```

```
    SELECT IncidentType, VictimID AS PersonID FROM Victim WHERE VictimID IN (30, 35)
```

```
    UNION
```

```
    SELECT IncidentType, SuspectID FROM Suspect WHERE SuspectID IN (30, 35)
```

```
) AS IncidentsWithPersons;
```

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

```
SELECT * FROM Victim WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE IncidentType = 'Robbery')
```

UNION

```
SELECT * FROM Suspect WHERE CrimeID IN (SELECT CrimeID FROM Crime WHERE IncidentType = 'Robbery');
```

	VictimID	CrimeID	Name	ContactInfo	Injuries
▶	1	1	John Doe	johndoe@example.com	Minor injuries
	1	1	Robber 1	Armed and masked robber	Previous robbery convictions

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

```
SELECT IncidentType
FROM Crime
WHERE Status = 'Open'
GROUP BY IncidentType
HAVING COUNT(*) > 1;
```

IncidentType

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

```
SELECT c.*, s.*
FROM Crime c
JOIN Suspect s ON c.CrimeID = s.CrimeID
WHERE s.Name IN (SELECT Name FROM Victim WHERE CrimeID <> c.CrimeID);
```

CrimeID	IncidentType	IncidentDate	Location	Description	Status	SuspectID	CrimeID	Name	Description	CriminalHistory
---------	--------------	--------------	----------	-------------	--------	-----------	---------	------	-------------	-----------------

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

```
SELECT c.*, v.*, s.*
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	CrimeID	Name	ContactInfo	Injuries	SuspectID	CrimeID	Name
▶ 1	Robbery	2023-09-15	123 Main St, Cityville	Armed robbery at a convenience store	Open	1	1	John Doe	johndoe@example.com	Minor injuries	1	1	Robber 1
2	Homicide	2023-09-20	456 Elm St, Townsville	Investigation into a murder case	Under Investigation	2	2	Jane Smith	janesmith@example.com	Deceased	2	2	Unknown
3	Theft	2023-09-10	789 Oak St, Villagetown	Shoplifting incident at a mall	Closed	3	3	Alice Johnson	alicejohnson@example.com	None	3	3	Suspect 1

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

-- Assuming age is not provided, and the age condition is based on VictimID and SuspectID.

```
SELECT c.*  
FROM Crime c  
JOIN Victim v ON c.CrimeID = v.CrimeID  
JOIN Suspect s ON c.CrimeID = s.CrimeID  
WHERE s.SuspectID > v.VictimID;
```

	CrimeID	IncidentType	IncidentDate	Location	Description	Status
--	---------	--------------	--------------	----------	-------------	--------

-- 16. Find suspects involved in multiple incidents.

```
SELECT SuspectID, COUNT(DISTINCT CrimeID) AS IncidentCount  
FROM Suspect  
GROUP BY SuspectID  
HAVING IncidentCount > 1;
```

	SuspectID	IncidentCount
--	-----------	---------------

-- 17. List incidents with no suspects involved.

```
SELECT c.*  
FROM Crime c  
LEFT JOIN Suspect s ON c.CrimeID = s.CrimeID  
WHERE s.SuspectID IS NULL;
```

	CrimeID	IncidentType	IncidentDate	Location	Description	Status
--	---------	--------------	--------------	----------	-------------	--------

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

```
SELECT DISTINCT c1.*  
FROM Crime c1  
JOIN Crime c2 ON c1.CrimeID <> c2.CrimeID
```

WHERE c1.IncidentType = 'Homicide' AND c2.IncidentType = 'Robbery';

	CrimeID	IncidentType	IncidentDate	Location	Description	Status
▶	2	Homicide	2023-09-20	456 Elm St, Townsville	Investigation into a murder case	Under Investigation

-- 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.*, 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

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

SELECT DISTINCT s.*

FROM Suspect s

JOIN Crime c ON s.CrimeID = c.CrimeID

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

	SuspectID	CrimeID	Name	Description	CriminalHistory
▶	1	1	Robber 1	Armed and masked robber	Previous robbery convictions