31.03.2025 Coding Challenge SQL

Crime Management Shema DDL and DML

create database crime;

use crime;

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 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');

ALTER TABLE Victim

ADD Age INT;

ALTER TABLE Suspect

ADD Age INT;

UPDATE Victim

SET Age = 32 WHERE VictimID = 1;

UPDATE Victim

SET Age = 22 WHERE VictimID = 2;

UPDATE Victim

SET Age = 42 WHERE VictimID = 3;

UPDATE Suspect

SET Age = 34 WHERE SuspectID = 1;

UPDATE Suspect

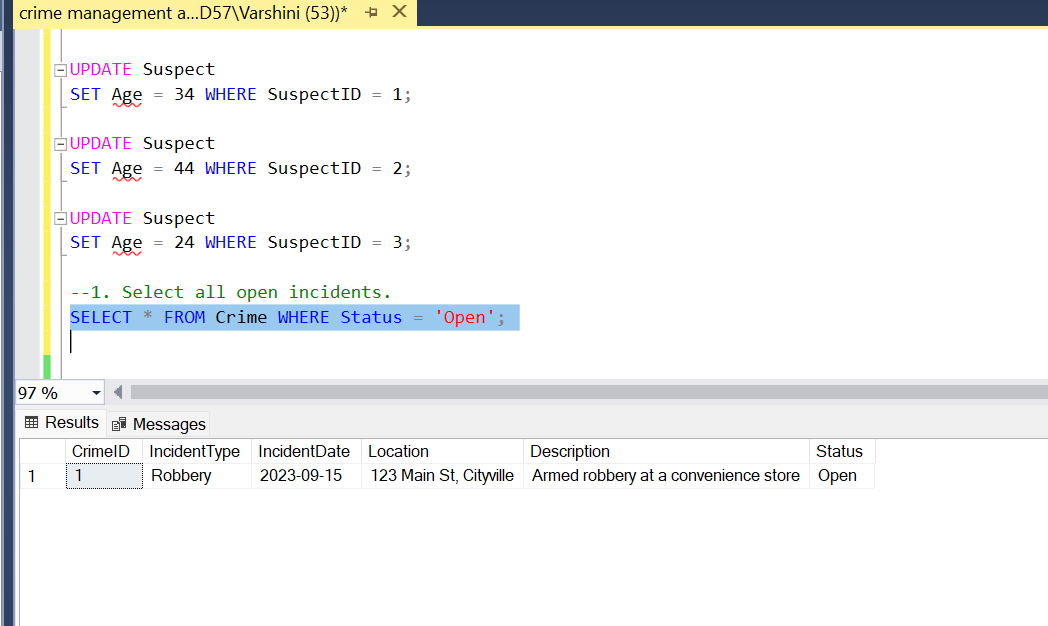
SET Age = 44 WHERE SuspectID = 2;

UPDATE Suspect

SET Age = 24 WHERE SuspectID = 3;

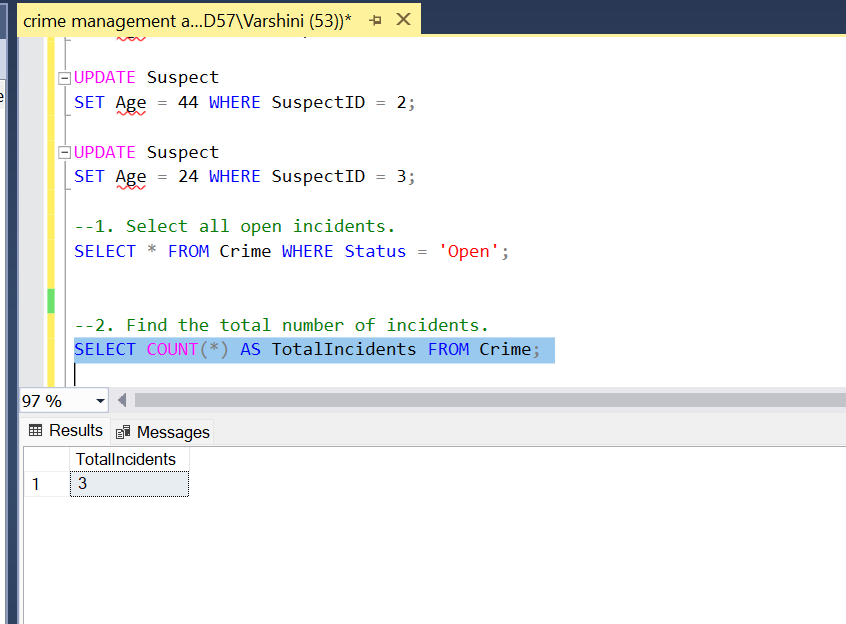
--1. Select all open incidents.

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



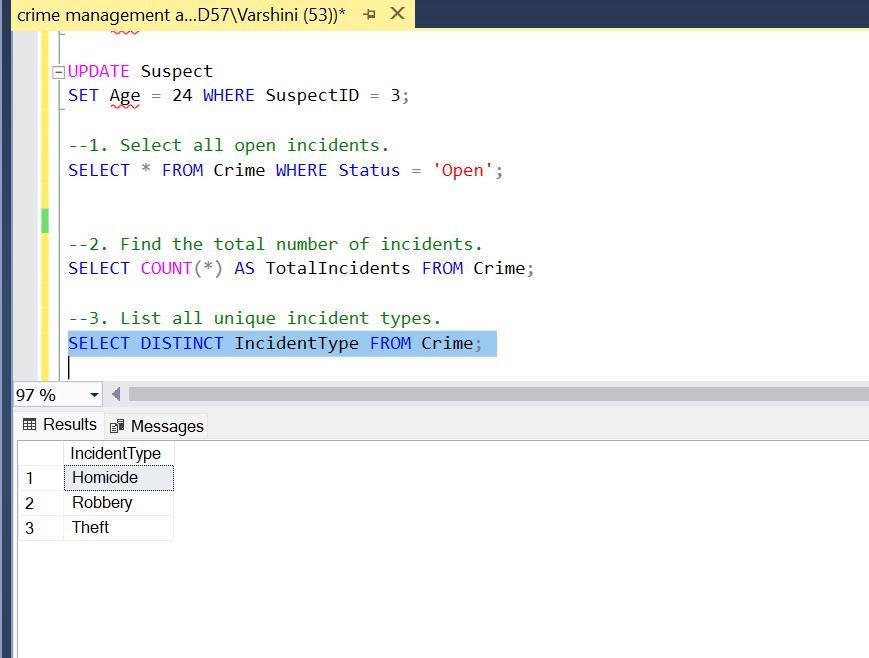
--2. Find the total number of incidents.

SELECT COUNT(\*) AS TotalIncidents FROM Crime;



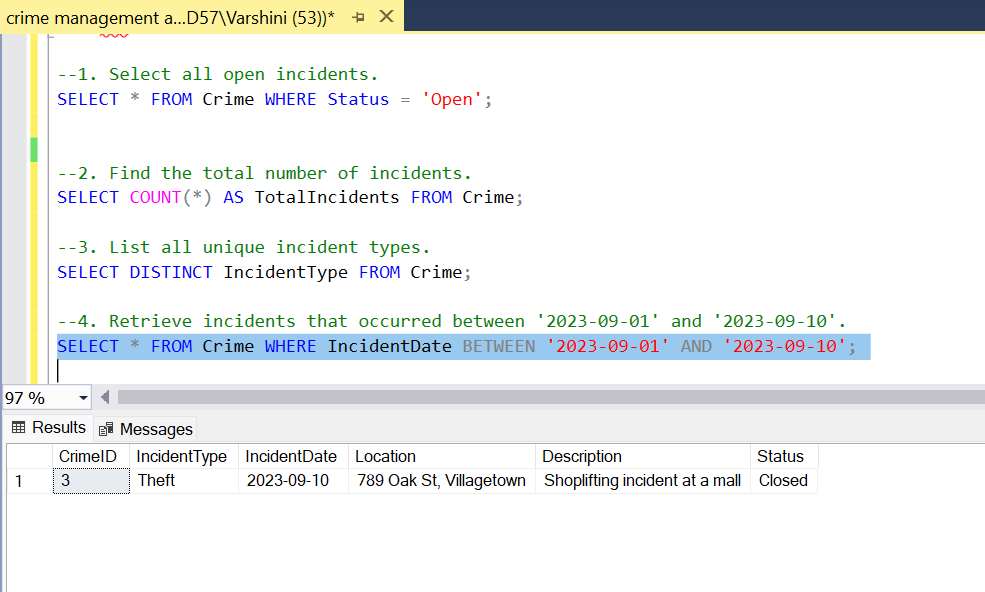
--3. List all unique incident types.

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';



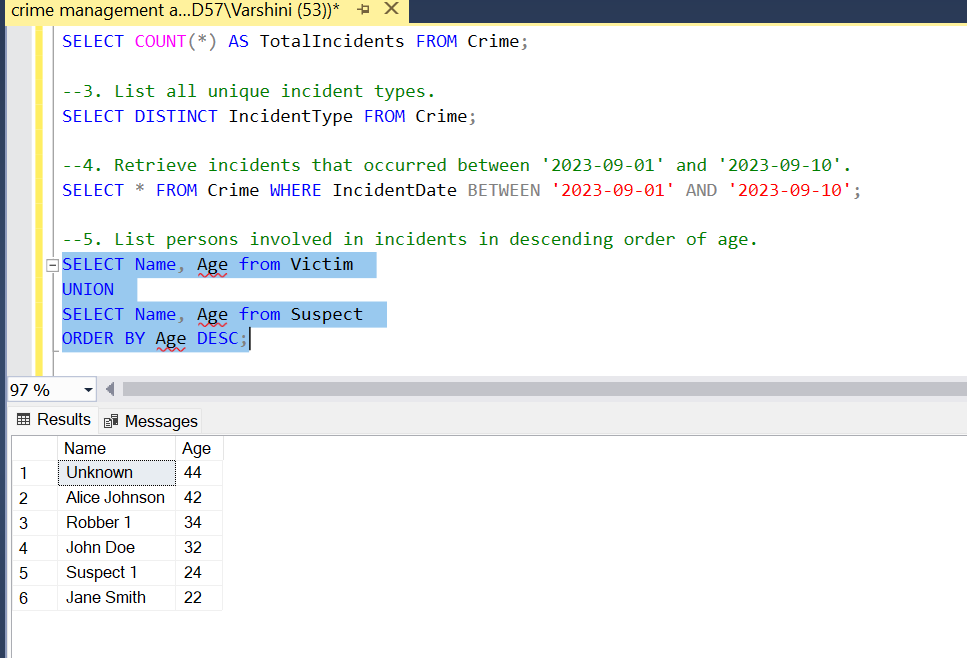
--5. List persons involved in incidents in descending order of age.

SELECT Name, Age from Victim

UNION

SELECT Name, Age from Suspect

ORDER BY Age DESC;



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

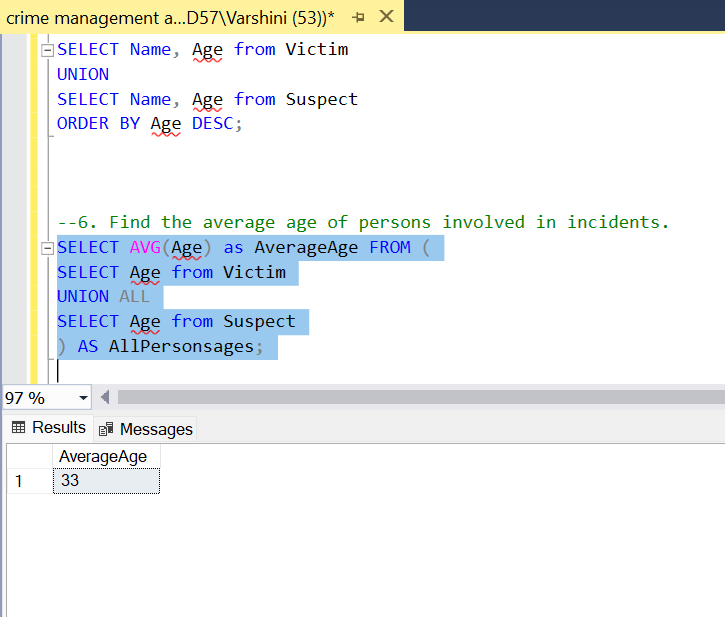
SELECT AVG(Age) as AverageAge FROM (

SELECT Age from Victim

UNION ALL

SELECT Age from Suspect

) AS AllPersonsages;



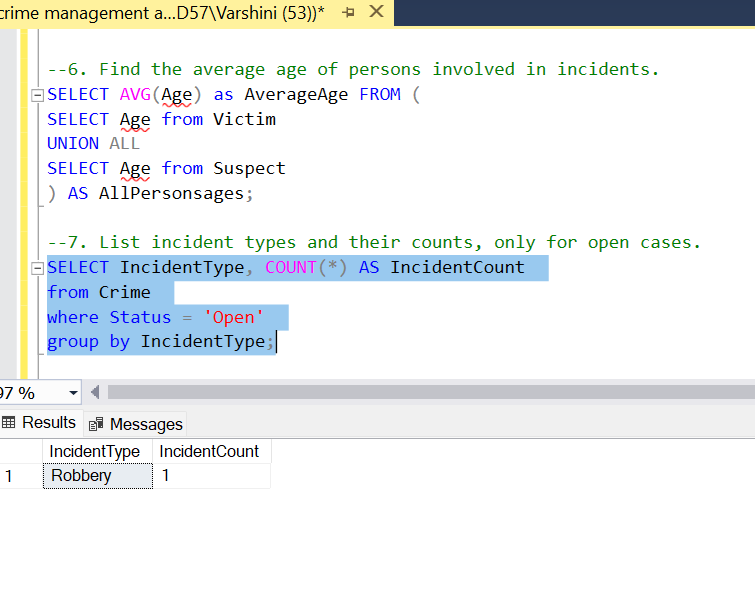
--7. List incident types and their counts, only for open cases.

SELECT IncidentType, COUNT(\*) AS IncidentCount

from Crime

where Status = 'Open'

group by IncidentType;

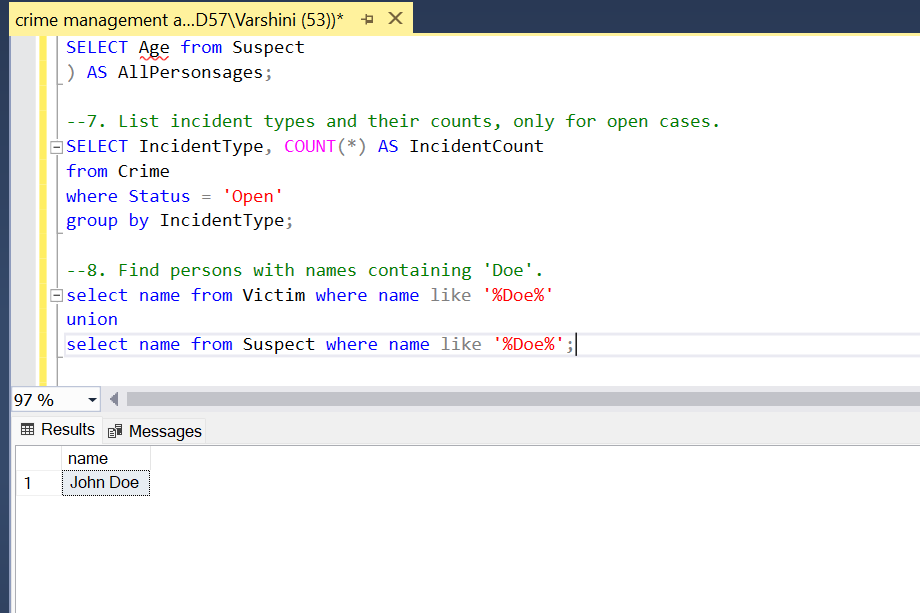


--8. Find persons with names containing 'Doe'.

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 v.Name FROM Victim v

join Crime c ON v.CrimeID = C.CrimeID

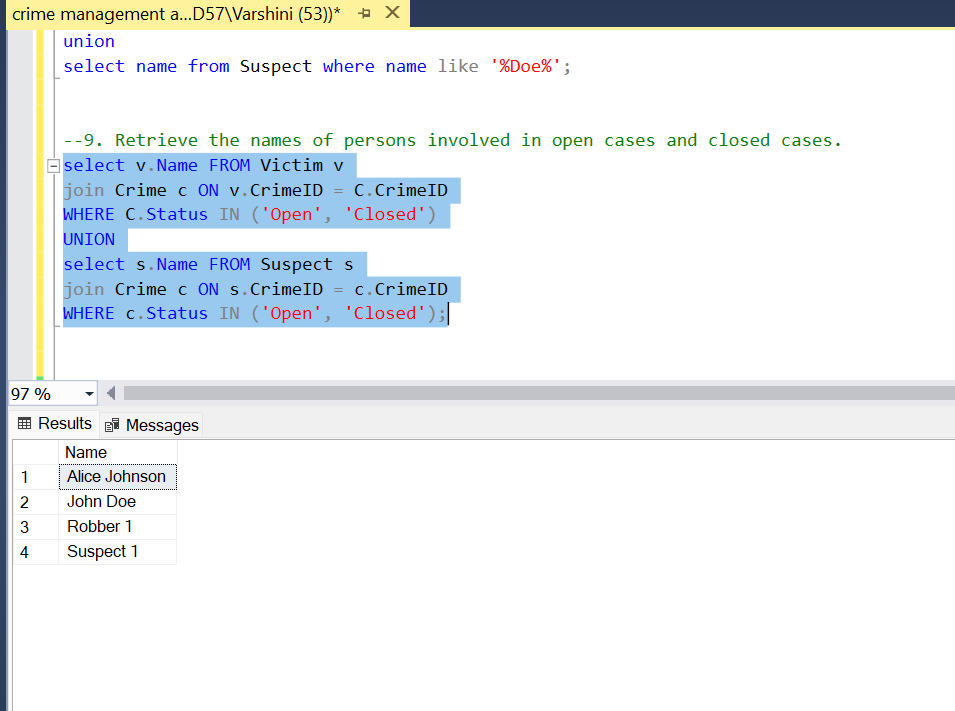
WHERE C.Status IN ('Open', 'Closed')

UNION

select s.Name FROM Suspect s

join Crime c ON s.CrimeID = c.CrimeID

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



UPDATE Victim

SET Age = 35 WHERE VictimID = 1;

UPDATE Victim

SET Age = 28 WHERE VictimID = 2;

UPDATE Victim

SET Age = 40 WHERE VictimID = 3;

UPDATE Suspect

SET Age = 30 WHERE SuspectID = 1;

UPDATE Suspect

SET Age = 45 WHERE SuspectID = 2;

UPDATE Suspect

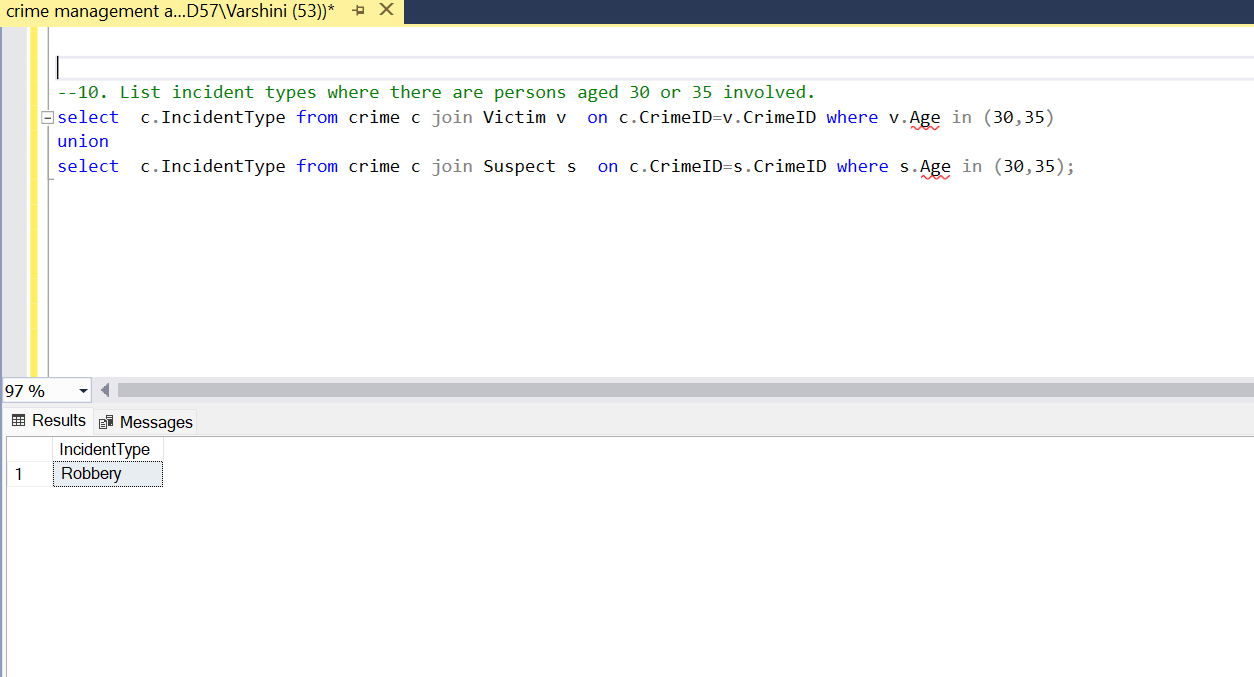
SET Age = 25 WHERE SuspectID = 3;

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

select c.IncidentType from crime c join Victim v on c.CrimeID=v.CrimeID where v.Age in (30,35)

union

select c.IncidentType from crime c join Suspect s on c.CrimeID=s.CrimeID where s.Age in (30,35);

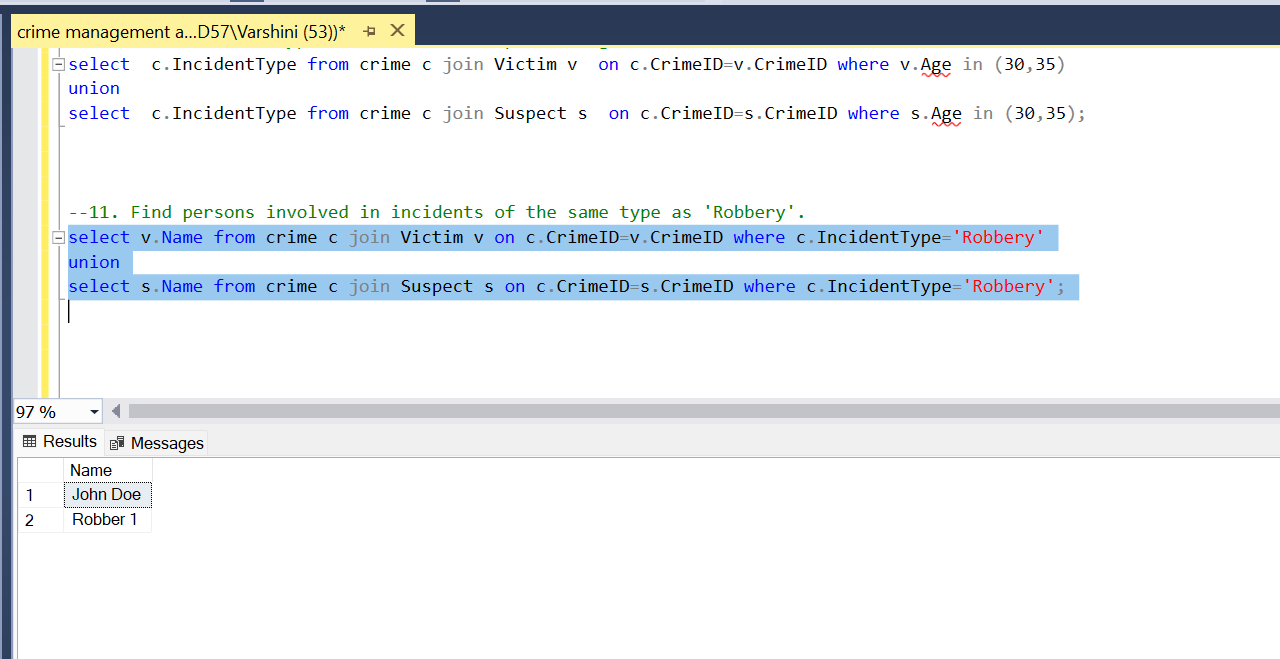


--11. Find persons involved in incidents of the same type as 'Robbery'.

select v.Name from crime c join Victim v on c.CrimeID=v.CrimeID where c.IncidentType='Robbery'

union

select s.Name from crime c join Suspect s on c.CrimeID=s.CrimeID where c.IncidentType='Robbery';



--insert values in crime

insert into Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)

values

(6, 'Robbery', '2023-10-01', '500 King St, Metro City', 'Bank robbery', 'Open'),

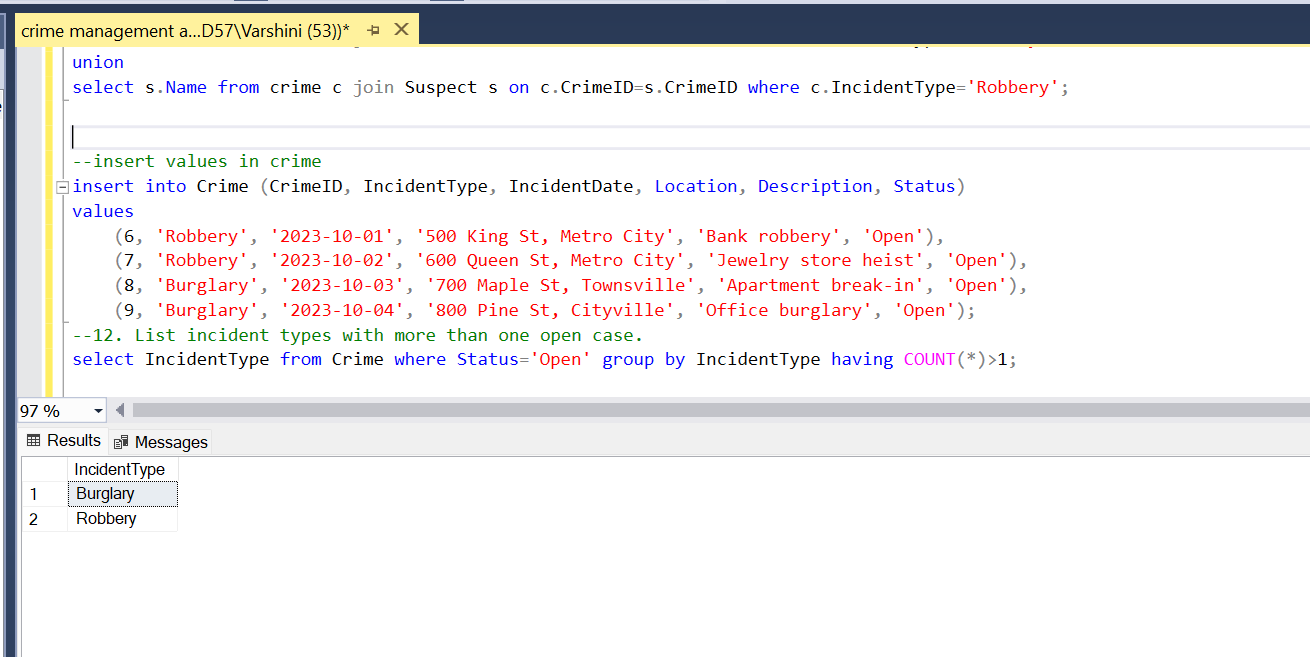
(7, 'Robbery', '2023-10-02', '600 Queen St, Metro City', 'Jewelry store heist', 'Open'),

(8, 'Burglary', '2023-10-03', '700 Maple St, Townsville', 'Apartment break-in', 'Open'),

(9, 'Burglary', '2023-10-04', '800 Pine St, Cityville', 'Office burglary', 'Open');

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

select IncidentType from Crime where Status='Open' group by IncidentType having COUNT(\*)>1;



--insert into crime

insert into Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)

values (4, 'Burglary', '2023-09-25', '101 Pine St, Metro City', 'House break-in reported', 'Open');

-- insert a victim who will also be a suspect in another incident

insert into Victim (VictimID, CrimeID, Name, ContactInfo, Injuries)

values (4, 4, 'Michael', 'michael@example.com', 'Minor injuries');

-- insert a new suspect with the same name as a victim in another incident

insert into Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)

values (5, 'Assault', '2023-09-26', '222 Birch St, Metro City', 'Street fight reported', 'Under Investigation');

insert into Suspect (SuspectID, CrimeID, Name, Description, CriminalHistory)

values (4, 5, 'Michael Brown', 'Involved in a street fight', 'Prior altercation reports');

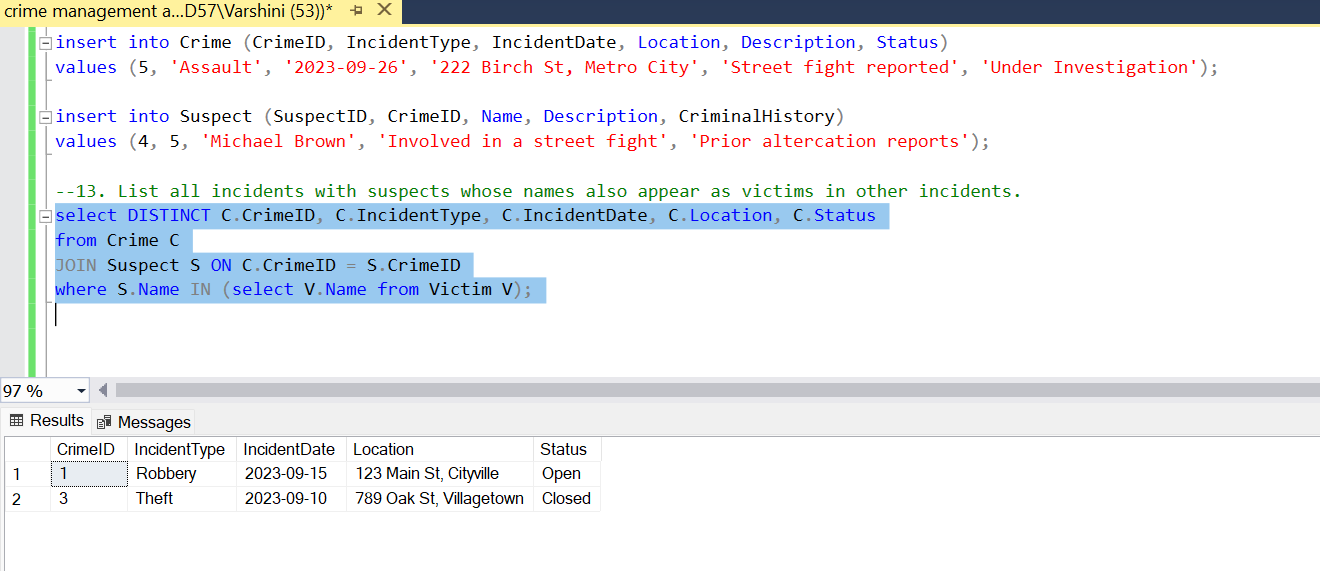
--13. List all incidents with suspects whose names also appear as victims in other incidents.

select DISTINCT C.CrimeID, C.IncidentType, C.IncidentDate, C.Location, C.Status

from Crime C

JOIN Suspect S ON C.CrimeID = S.CrimeID

where S.Name IN (select V.Name from Victim V);



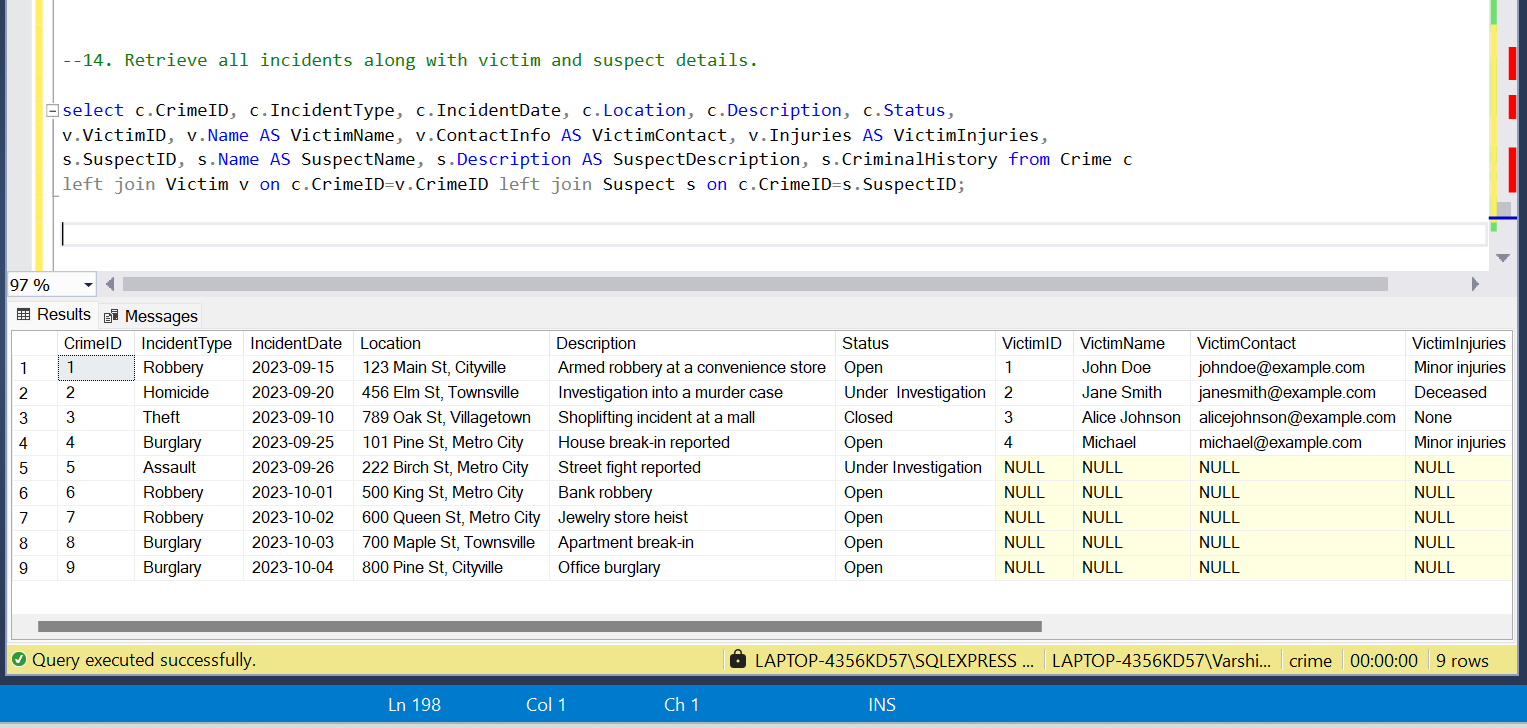
--14. Retrieve all incidents along with victim and suspect details.

select c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status,

v.VictimID, v.Name AS VictimName, v.ContactInfo AS VictimContact, v.Injuries AS VictimInjuries,

s.SuspectID, s.Name AS SuspectName, s.Description AS SuspectDescription, s.CriminalHistory from Crime c

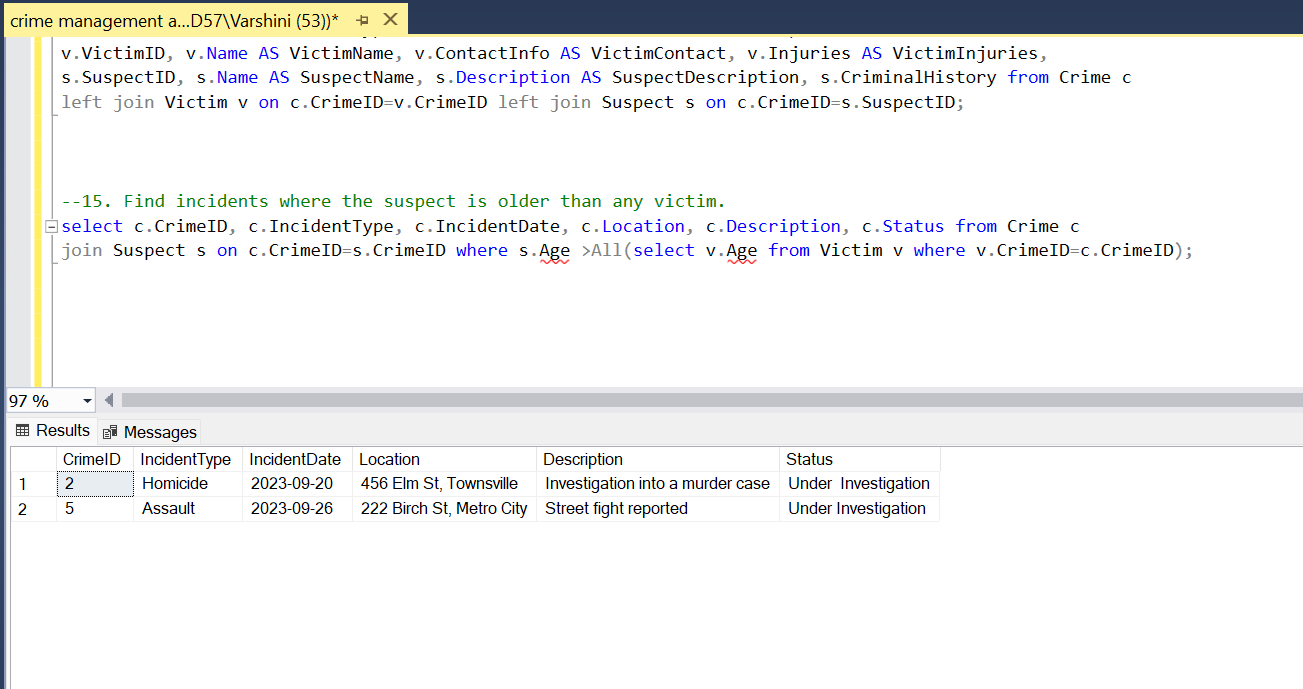
left join Victim v on c.CrimeID=v.CrimeID left join Suspect s on c.CrimeID=s.SuspectID;



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

select c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status from Crime c

join Suspect s on c.CrimeID=s.CrimeID where s.Age >All(select v.Age from Victim v where v.CrimeID=c.CrimeID);



-- Adding a suspect linked to multiple crimes

insert into Suspect (SuspectID, CrimeID, Name, Description, CriminalHistory)

values

(5, 1, 'John Doe', 'Repeat offender involved in multiple incidents', 'Previous theft and robbery cases'),

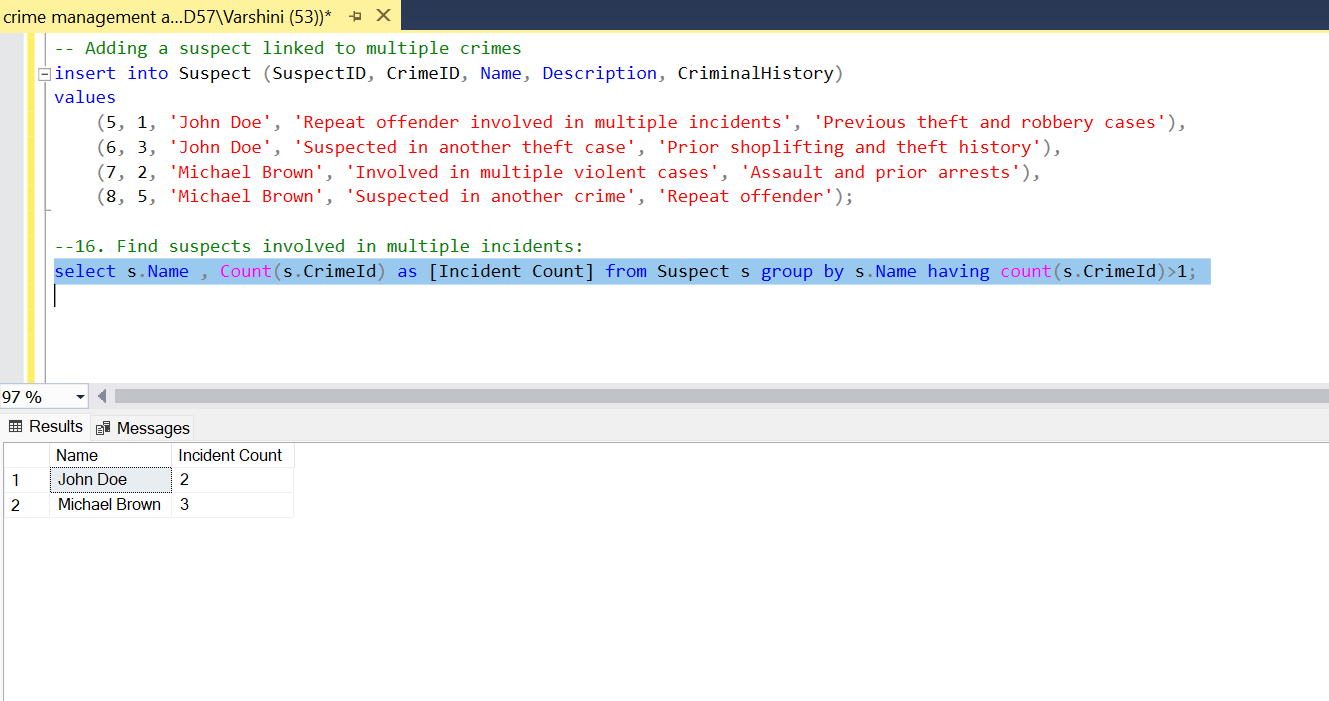
(6, 3, 'John Doe', 'Suspected in another theft case', 'Prior shoplifting and theft history'),

(7, 2, 'Michael Brown', 'Involved in multiple violent cases', 'Assault and prior arrests'),

(8, 5, 'Michael Brown', 'Suspected in another crime', 'Repeat offender');

--16. Find suspects involved in multiple incidents:

select s.Name , Count(s.CrimeId) as [Incident Count] from Suspect s group by s.Name having count(s.CrimeId)>1;



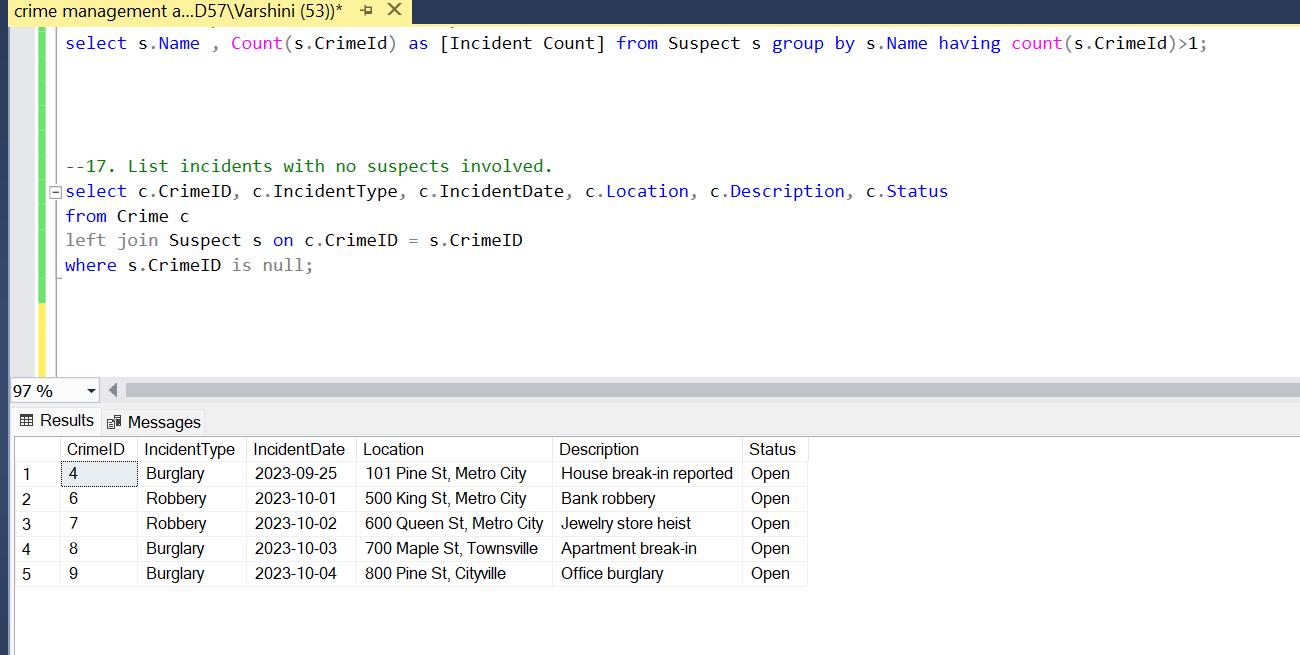
--17. List incidents with no suspects involved.

select c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status

from Crime c

left join Suspect s on c.CrimeID = s.CrimeID

where s.CrimeID is null;



-- Case 1: Valid case with one Homicide and other Robberies

insert into Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)

values (11, 'Homicide', '2023-09-20', '111 Cedar St, Townsville', 'Murder investigation', 'Open'),

(12, 'Robbery', '2023-09-21', '112 Cedar St, Townsville', 'Bank Robbery', 'Open'),

(13, 'Robbery', '2023-09-22', '113 Cedar St, Townsville', 'ATM Theft', 'Open');

-- Case 2: Invalid case (should not appear in output) because it has a 'Burglary'

insert into Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)

values (14, 'Homicide', '2023-09-25', '200 Birch St, Metro City', 'Another murder case', 'Under Investigation'),

(15, 'Burglary', '2023-09-26', '201 Birch St, Metro City', 'Office break-in', 'Open');

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

select distinct c1.CrimeID, c1.IncidentType, c1.IncidentDate, c1.Location,

CAST(c1.Description AS VARCHAR(MAX)) AS Description, c1.Status

from Crime c1

where c1.IncidentType = 'Homicide'

AND NOT EXISTS (

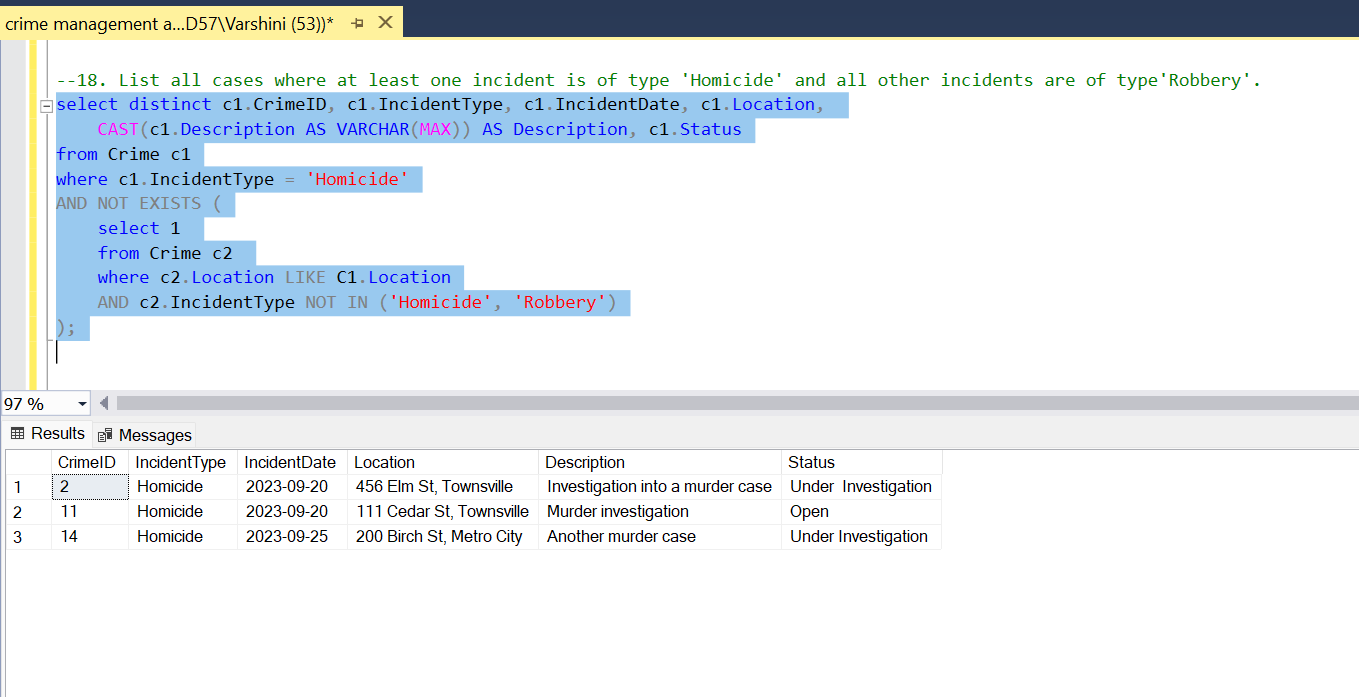
select 1

from Crime c2

where c2.Location LIKE C1.Location

AND c2.IncidentType NOT IN ('Homicide', 'Robbery')

);



--19. Retrieve a list of all incidents and the associated suspects, showing suspects for each incident,

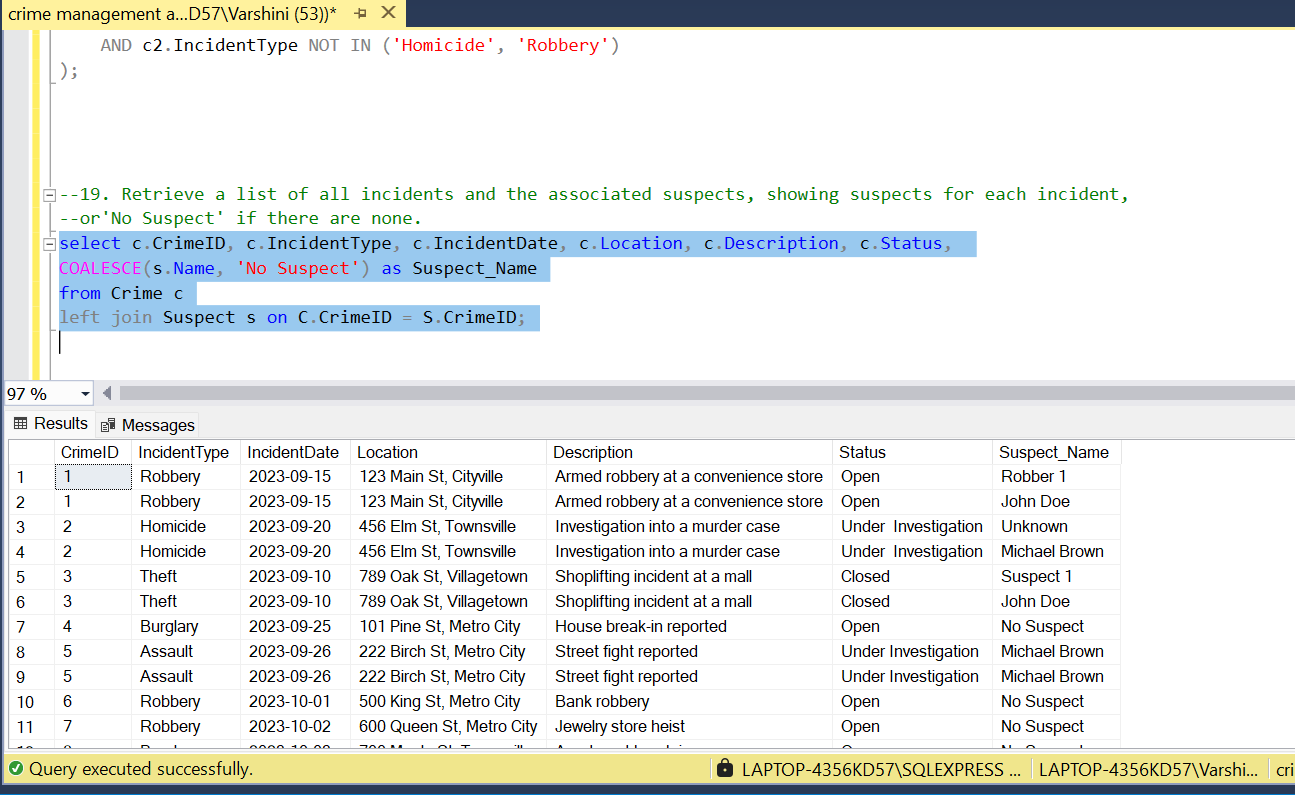
--or'No Suspect' if there are none.

select c.CrimeID, c.IncidentType, c.IncidentDate, c.Location, c.Description, c.Status,

COALESCE(s.Name, 'No Suspect') as Suspect\_Name

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.SuspectID, s.Name, s.Description, s.CriminalHistory

from Suspect s

join Crime c on s.CrimeID = c.CrimeID

where c.IncidentType in ('Robbery', 'Assault');

