# **Crime Management Coding Challenge SQL**

# **Creating database**

# Query

```
create database crime;
show databases;
```

# Output



# **Creating Tables**

#### **Crime Table**

## Query

```
create table Crime (
    CrimeID int primary key,
    IncidentType varchar(255),
    IncidentDate date,
    Location varchar(255),
    Description text,
    Status varchar(20)
);
```

# desc crime;



	Field	Туре	Null	Key	Default	Extra
•	CrimeID	int	NO	PRI	NULL	
	IncidentType	varchar(255)	YES		NULL	
	IncidentDate	date	YES		NULL	
	Location	varchar(255)	YES		NULL	
	Description	text	YES		NULL	
	Status	varchar(20)	YES		NULL	

#### **Victim Table**

# Query

```
• Create table Victim (
        VictimID int primary key,
        CrimeID int,
        Name varchar(255),
        ContactInfo varchar(255),
        Injuries varchar(255),
        Age int,
        foreign key (CrimeID) references Crime(CrimeID)
);
• desc Victim;
```

0	16	11:58:52	create table Victim (	VictimID int primary key,	Cr	0 row(s) affected
0	17	11:59:09	desc Victim			6 row(s) returned

	Field	Type	Null	Key	Default	Extra
١	VictimID	int	NO	PRI	NULL	
	CrimeID	int	YES	MUL	NULL	
	Name	varchar(255)	YES		NULL	
	ContactInfo	varchar(255)	YES		NULL	
	Injuries	varchar(255)	YES		NULL	
	Age	int	YES		NULL	

## **Suspect Table**

## Query

```
create table Suspect (
    SuspectID int primary key,
    CrimeID int,
    Name varchar(255),
    Description text,
    CriminalHistory text,
    Age int,
    foreign key (CrimeID) references Crime(CrimeID)
);

desc Suspect;
```

## Output

```
    22 12:05:45 create table Suspect ( SuspectID int primary key, CrimeID int, Name varchar(255), Description text, ... 0 row(s) affected
    23 12:05:49 desc Suspect
    6 row(s) returned
```

	Field	Туре	Null	Key	Default	Extra
•	SuspectID	int	NO	PRI	NULL	
	CrimeID	int	YES	MUL	NULL	
	Name	varchar(255)	YES		NULL	
	Description	text	YES		NULL	
	CriminalHistory	text	YES		NULL	
	Age	int	YES		NULL	

# Inserting the values into the tables

#### Crime table

```
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'),
(4, 'Robbery', '2023-09-02', '22 Lakeview Ave, Rivertown', 'Street mugging case', 'Open'),
(5, 'Assault', '2023-09-05', '33 Hill St, Cityville', 'Bar fight incident', 'Closed'),
(6, 'Homicide', '2023-09-07', '12 Beach Rd, Seaville', 'Double homicide case', 'Open'),
(7, 'Burglary', '2023-09-03', '88 Ocean Dr, Coastville', 'Home burglary', 'Closed'),
(8, 'Assault', '2023-09-04', '77 Forest Rd, Greenfield', 'Park assault', 'Open'),
(9, 'Robbery', '2023-09-08', '101 River St, Newcity', 'Bank robbery', 'Open'),
(10, 'Vandalism', '2023-09-22', '99 Street Ln, Oldtown', 'School wall graffiti', 'Closed');
select * from Crime;
```

```
INSERT INTO Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)
VALUES (11, 'Burglary', '2023-09-09', '200 Willow St, Windtown', 'Break-in reported, no suspect identified', 'Open');
```

- 19 12:04:12 insert into Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status) values (1, "Robbery", "2023.... 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
- 20 12:04:21 select \*from Crime LIMIT 0, 1000 10 row(s) returned
- 26 12:12:08 INSERT INTO Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status) VALUES (11, 'Burglary... 1 row(s) affected
- 27 12:12:12 select \*from Crime LIMIT 0, 1000 11 row(s) returned

	CrimeID	IncidentType	IncidentDate	Location	Description	Status
•	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
	4	Robbery	2023-09-02	22 Lakeview Ave, Rivertown	Street mugging case	Open
	5	Assault	2023-09-05	33 Hill St, Cityville	Bar fight incident	Closed
	6	Homicide	2023-09-07	12 Beach Rd, Seaville	Double homicide case	Open
	7	Burglary	2023-09-03	88 Ocean Dr, Coastville	Home burglary	Closed
	8	Assault	2023-09-04	77 Forest Rd, Greenfield	Park assault	Open
	9	Robbery	2023-09-08	101 River St, Newcity	Bank robbery	Open
	10	Vandalism	2023-09-22	99 Street Ln, Oldtown	School wall graffiti	Closed
_	11 NULL	Burglary NULL	2023-09-09 NULL	200 Willow St, Windtown	Break-in reported, no suspect identified	Open NULL

#### Victim table

## Query

```
insert into Victim (VictimID, CrimeID, Name, ContactInfo, Injuries, Age) values
(1, 1, 'John Doe', 'johndoe@example.com', 'Minor injuries', 32),
(2, 2, 'Jane Smith', 'janesmith@example.com', 'Deceased', 45),
(3, 3, 'Alice Johnson', 'alicej@example.com', 'None', 28),
(4, 4, 'Emily Doe', 'emilydoe@example.com', 'Bruised arm', 30),
(5, 5, 'Carlos Vega', 'carlosv@example.com', 'Fractured rib', 35),
(6, 6, 'Sarah Connor', 'sconnor@example.com', 'Critical condition', 27),
(7, 7, 'Michael Ray', 'mray@example.com', 'Broken window injury', 40),
(8, 8, 'Laura Doe', 'lauradoe@example.com', 'Cut on hand', 33),
(9, 9, 'James Lee', 'jlee@example.com', 'Gunshot wound', 36),
(10, 10, 'Tony Stark', 'tstark@example.com', 'Paint fumes', 50),
(11, 11, 'Diana Prince', 'diana@example.com', 'No physical injuries', 38);
select * from Victim;
```

```
    32 12:18:10 insert into Victim (VictimID, CrimeID, Name, ContactI... 11 row(s) affected Records: 11 Duplicates: 0 Wam... 0.
    33 12:18:23 select *from Victim LIMIT 0, 1000 11 row(s) returned 0.
```

	VictimID	CrimeID	Name	ContactInfo	Injuries	Age
•	1	1	John Doe	johndoe@example.com	Minor injuries	32
	2	2	Jane Smith	janesmith@example.com	Deceased	45
	3	3	Alice Johnson	alicej@example.com	None	28
	4	4	Emily Doe	emilydoe@example.com	Bruised arm	30
	5	5	Carlos Vega	carlosv@example.com	Fractured rib	35
	6	6	Sarah Connor	sconnor@example.com	Critical condition	27
	7	7	Michael Ray	mray@example.com	Broken window injury	40
	8	8	Laura Doe	lauradoe@example.com	Cut on hand	33
	9	9	James Lee	jlee@example.com	Gunshot wound	36
	10	10	Tony Stark	tstark@example.com	Paint fumes	50
	11	11	Diana Prince	diana@example.com	No physical injuries	38
	PALIFI	BILLI	BILLI	BILL I	BILLI	MILIT

### Suspect table

## Query

```
INSERT INTO Suspect (SuspectID, CrimeID, Name, Description, CriminalHistory, Age) VALUES
(1, 1, 'Robber 1', 'Armed and masked robber', 'Previous robbery convictions', 35),
(2, 2, 'Unknown', 'Investigation ongoing', NULL, NULL),
(3, 3, 'Suspect 1', 'Shoplifting suspect', 'Prior shoplifting arrests',25),
(4, 4, 'John Doe', 'Repeat offender', 'Assault and robbery', 40),
(5, 5, 'Emily Ray', 'Involved in altercation', 'First-time offender', 30),
(6, 6, 'Suspect X', 'Witness identified', 'Murder suspect', 42),
(7, 7, 'Unknown', 'Suspect fled scene', NULL, NULL),
(8, 8, 'John Wayne', 'Seen running', 'Assault in 2019', 31),
(9, 9, 'Robber 1', 'Same as case #1', 'Repeat robbery suspect', 35),
(10, 10, 'Tony Stark', 'Graffiti suspect', 'Minor vandalism', 50),
(11, 5, 'John Doe', 'Also seen near crime scene', 'Repeat offender', 40);
SELECT * FROM suspect;
```

```
    34 12:20:44 INSERT INTO Suspect (SuspectID, CrimeID, Name... 11 row(s) affected Records: 11 Duplicates: 0 Wam...
    35 12:21:10 SELECT * FROM suspect LIMIT 0, 1000 11 row(s) returned
```

	SuspectID	CrimeID	Name	Description	CriminalHistory	Age
<b>&gt;</b>	1	1	Robber 1	Armed and masked robber	Previous robbery convictions	35
	2	2	Unknown	Investigation ongoing	NULL	NULL
	3	3	Suspect 1	Shoplifting suspect	Prior shoplifting arrests	25
	4	4	John Doe	Repeat offender	Assault and robbery	40
	5	5	Emily Ray	Involved in altercation	First-time offender	30
	6	6	Suspect X	Witness identified	Murder suspect	42
	7	7	Unknown	Suspect fled scene	NULL	NULL
	8	8	John Wayne	Seen running	Assault in 2019	31
	9	9	Robber 1	Same as case #1	Repeat robbery suspect	35
	10	10	Tony Stark	Graffiti suspect	Minor vandalism	50
	11	5	John Doe	Also seen near crime scene	Repeat offender	40
	NULL	NULL	NULL	NULL	NULL	NULL

## **Queries**

## 1. Select all open incidents.

## Query

```
/*1. Select all open incidents*/
select * from crime
where status = 'Open';
```

## Output

37 12:25:14 select \*from crime where status = 'Open' LIMIT 0, 1000

6 row(s) returned

CrimeID	IncidentType	IncidentDate	Location	Description	Status
1	Robbery	2023-09-15	123 Main St, Cityville	Armed robbery at a convenience store	Open
4	Robbery	2023-09-02	22 Lakeview Ave, Rivertown	Street mugging case	Open
6	Homicide	2023-09-07	12 Beach Rd, Seaville	Double homicide case	Open
8	Assault	2023-09-04	77 Forest Rd, Greenfield	Park assault	Open
9	Robbery	2023-09-08	101 River St, Newcity	Bank robbery	Open
11	Burglary	2023-09-09	200 Willow St, Windtown	Break-in reported, no suspect identified	Open

#### 2. Find the total number of incidents.

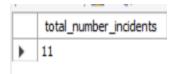
# Query

```
select count(*) as total_number_incidents from crime;
```

# Output

39 12:31:16 select count(\*) as total\_number\_incidents from crime LIMIT 0, 1000

1 row(s) returned



# 3. List all unique incident types.

## Query

```
select distinct incidenttype from crime;
```

### Output

40 12:34:03 select distinct incidenttype from crime LIMIT 0, 1000

6 row(s) returned

	incidenttype
•	Robbery
	Homicide
	Theft
	Assault
	Burglary
	Vandalism

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

# Query

```
select * from crime
where IncidentDate between '2023-09-01' and '2023-09-10';
```

## Output



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

# Query

```
select name, age from victim
union
select name, age from suspect where age is not null
order by age desc;
```

## Output

17 row(s) returned

	name	age
•	Tony Stark	50
	Jane Smith	45
	Suspect X	42
	Michael Ray	40
	John Doe	40
	Diana Prince	38
	James Lee	36
	Carlos Vega	35
	Robber 1	35
	Laura Doe	33
	John Doe	32
	John Wayne	31
	Emily Doe	30
	Emily Ray	30
	Alice Johnson	28
	Sarah Connor	27
	Suspect 1	25

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

# Query

```
select avg(age) as average_age

from (
    select age from victim
    union all
    select age from suspect where age is not null

) as all_persons;
```

# Output

45 12:51:53 select avg(age) as average\_age from ( select age from victim union all select age from suspect where ag... 1 row(s) returned

	average_age
•	36.1000

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

```
select incidenttype, count(*) from crime
where status = 'Open'
group by incidenttype;
```

49 12:57:37 select incidenttype, count(") from crime where status = "Open" group by incidenttype LIMIT 0, 1000

4 row(s) returned

	incidenttype	count(*)
١	Robbery	3
	Homicide	1
	Assault	1
	Burglary	1

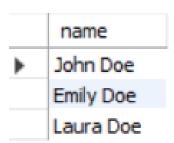
8. Find persons with names containing 'Doe'.

## Query

```
where name like '%Doe%'
union
select name from suspect
where name like '%Doe%';
```

# **Output**

53 13:02:33 select name from victim where name like "%Doe%" union select name from suspect where name like "%Doe%" 3 row(s) returned



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

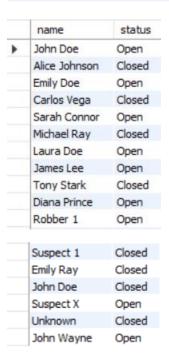
## Query

```
select v.name, c.status
from victim v

join crime c on v.crimeid = c.crimeid
where c.status in ('Open', 'Closed')
union
select s.name, c.status
from suspect s
join crime c on s.crimeid = c.crimeid
where c.status in ('Open', 'Closed');
```

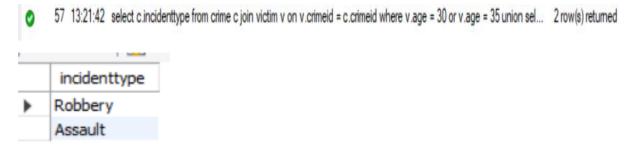
## Output

54 13:14:53 select v.name, c.status from victim v join crime c on v.crimeid = c.crimeid where c.status in ('Open', 'Closed') unio... 17 row(s) returned



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

```
select c.incidenttype from crime c
join victim v on v.crimeid = c.crimeid
where v.age = 30 or v.age = 35
union
select c.incidenttype from crime c
join suspect s on s.crimeid = c.crimeid
where s.age = 30 or s.age = 35
```



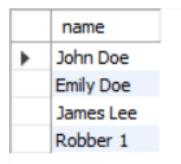
## 11. Find persons involved in incidents of the same type as 'Robbery'.

## Query

```
select name from victim
where crimeid in (
    select crimeid from crime
    where incidenttype = 'Robbery'
)
union
select name from suspect
where crimeid in (
    select crimeid from crime
    where incidenttype = 'Robbery'
);
```

# Output

59 13:29:21 select name from victim where crimeid in ( select crimeid from crime where incidenttype = 'Robbery') union s... 4 row(s) returned



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

# Query

Updating the row before finding the open case more than one

```
update crime set status = 'Open' where crimeid = 2;
select incidenttype, COUNT(*) as open_case_count from crime
where status = 'Open'
group by incidenttype
having COUNT(*) > 1;
```

## Output

```
65 13:34:41 update crime set status = 'Open' where crimeid = 2

66 13:35:59 select incidenttype, COUNT(') as open_case_count from crime where status = 'Open' group by incidenttype havi... 2 row(s) returned

incidenttype open_case_count

Robbery 3

Homicide 2
```

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

```
select distinct c.* FROM crime c
join suspect s on c.crimeid = s.crimeid
where s.name in (
    select v.name from victim v
);
```

67 13:38:26 select distinct c.\* FROM crime c join suspect s on c.crimeid = s.crimeid where s.name in ( select v.name from v... 3 row(s) returned

	CrimeID	IncidentType	IncidentDate	Location	Description	Status
•	4	Robbery	2023-09-02	22 Lakeview Ave, Rivertown	Street mugging case	Open
	5	Assault	2023-09-05	33 Hill St, Cityville	Bar fight incident	Closed
	10	Vandalism	2023-09-22	99 Street Ln, Oldtown	School wall graffiti	Closed

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

# Query

```
select c.CrimeID, c.IncidentType,
  v.Name as VictimName,
  s.Name as SuspectName
from crime c
left join victim v on c.CrimeID = v.CrimeID
left join suspect s on c.CrimeID = s.CrimeID;
```

# Output

o 70 13:43:48 select c.CrimeID, c.IncidentType, v.Name as VictimName, s.Name as SuspectName from crime c left join victi... 12 row(s) returned

	CrimeID	IncidentType	VictimName	SuspectName	
١	1	Robbery	John Doe	Robber 1	
	2	Homicide	Jane Smith	Unknown	
	3	Theft	Alice Johnson	Suspect 1	
	4	Robbery	Emily Doe	John Doe	
	5	Assault	Carlos Vega	Emily Ray	
	5	Assault	Carlos Vega	John Doe	
	6	Homicide	Sarah Connor	Suspect X	
	7	Burglary	Michael Ray	Unknown	
	8	Assault	Laura Doe	John Wayne	
	9	Robbery	James Lee	Robber 1	
	10	Vandalism	Tony Stark	Tony Stark	
	11	Burglary	Diana Prince	NULL	

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

```
select * from crime c
where c.CrimeID in (
    select s.CrimeID from suspect s
    where s.Age > (select MIN(v.Age) from victim v
    )
);
```

71 13:47:03 select \*from crime c where c.CrimeID in ( select s.CrimeID from suspect s where s.Age > (select MIN(v.Age)... 7 row(s) returned

	CrimeID	IncidentType	IncidentDate	Location	Description	Status
•	1	Robbery	2023-09-15	123 Main St, Cityville	Armed robbery at a convenience store	Open
	4	Robbery	2023-09-02	22 Lakeview Ave, Rivertown	Street mugging case	Open
	5	Assault	2023-09-05	33 Hill St, Cityville	Bar fight incident	Closed
	6	Homicide	2023-09-07	12 Beach Rd, Seaville	Double homicide case	Open
	8	Assault	2023-09-04	77 Forest Rd, Greenfield	Park assault	Open
	9	Robbery	2023-09-08	101 River St, Newcity	Bank robbery	Open
	10	Vandalism	2023-09-22	99 Street Ln, Oldtown	School wall graffiti	Closed
-	NULL	NULL	HULL	NULL	HULL	NULL

# 16. Find suspects involved in multiple incidents

# Query

```
select name
from suspect
group by name
having count(distinct crimeid) > 1;
```

## **Output**

72 13:48:29 select name from suspect group by name having count(distinct crimeid) > 1 LIMIT 0, 1000 3 row(s) returned



# 17. List incidents with no suspects involved.

# Query

Inserting one more value before running the above question's query

```
insert into crime (crimeid, incidenttype, incidentdate, location, description, status)
values (12, 'Vandalism', '2023-09-25', '202 Pine St, Greentown', 'Graffiti on public property', 'Open');
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'.

## Query

# Output

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

# Query

```
select
  c.crimeid,
  c.incidenttype,
  c.incidentdate,
  c.location,
  c.status,
  ifnull(s.name, 'No Suspect') as suspect_name
from crime c
left join suspect s on c.crimeid = s.crimeid;
```

crimeid	incidenttype	incidentdate	location	status	suspect_name
1	Robbery	2023-09-15	123 Main St, Cityville	Open	Robber 1
2	Homicide	2023-09-20	456 Elm St, Townsville	Open	Unknown
3	Theft	2023-09-10	789 Oak St, Villagetown	Closed	Suspect 1
4	Robbery	2023-09-02	22 Lakeview Ave, Rivertown	Open	John Doe
5	Assault	2023-09-05	33 Hill St, Cityville	Closed	Emily Ray
5	Assault	2023-09-05	33 Hill St, Cityville	Closed	John Doe
6	Homicide	2023-09-07	12 Beach Rd, Seaville	Open	Suspect X
7	Burglary	2023-09-03	88 Ocean Dr, Coastville	Closed	Unknown
8	Assault	2023-09-04	77 Forest Rd, Greenfield	Open	John Wayne
9	Robbery	2023-09-08	101 River St, Newcity	Open	Robber 1
10	Vandalism	2023-09-22	99 Street Ln, Oldtown	Closed	Tony Stark
11	Burglary	2023-09-09	200 Willow St, Windtown	Open	No Suspect
12	Vandalism	2023-09-25	202 Pine St, Greentown	Open	No Suspect
14	Robbery	2023-09-30	501 Pine St. Springfield	Onen	No Suspect

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

# Query

```
select distinct s.name
from suspect s
join crime c on s.crimeid = c.crimeid
where c.incidenttype in ('Robbery', 'Assault');
```

# Output

2 13:56:44 select distinct s.name from suspect s join crime c on s.crimeid = c.crimeid where c.incidenttype in ('Robbery', 'Ass... 4 row(s) returned

