

# **Record Management System**

## **Milestone: Implementation in MySQL**

Group 3

Gudipalli Sri Sai Prabhath Reddy

Yashwant Dontam

857-398-7947 (Prabhath)

857-230-5413 (Yashwant)

[gudipalli.s@northeastern.edu](mailto:gudipalli.s@northeastern.edu)

[dontam.y@northeastern.edu](mailto:dontam.y@northeastern.edu)

**Percentage of Effort Contributed by Student1: 50%**

**Percentage of Effort Contributed by Student2: 50%**

**Signature of Student 1:** Gudipalli Sri Sai Prabhath Reddy

**Signature of Student 2:** Yashwant Dontam

**Submission Date:** 11/12/2023

# SQL queries showing the Tables and the Columns

1.

CREATE TABLE AccessRights (

AccessID INT PRIMARY KEY,

AccessName TEXT,

AccessLevel TEXT,

UserID INT NOT NULL,

FOREIGN KEY (UserID) REFERENCES User(UserID)

);

The screenshot shows a database management interface. On the left, a 'Navigator' pane displays a tree view of the database schema. Under the 'rms1' schema, the 'Tables' folder is expanded, showing a list of tables including 'accessrights', 'administrative documents', 'assignedto', 'cases', 'evidences', 'incidents', 'involved', 'officers', 'shiftschedule', 'suspects', 'user', 'vehicles', and 'victims'. The 'accessrights' table is selected. Below the Navigator, the 'Table: accessrights' is detailed with its columns: 'AccessID' (int PK), 'AccessName' (text), 'AccessLevel' (text), and 'UserID' (int). The main window displays the 'accessrights' table with a query 'SELECT \* FROM rms1.accessrights;' and a 'Result Grid' showing the following data:

AccessID	AccessName	AccessLevel	UserID
5000	ManageCases	Full	4500
5001	ManageUsers	Read-Only	4501
5002	ManageUsers	NoAccess	4502
5003	ViewCases	Read-Only	4503
5004	ViewReports	NoAccess	4504
5005	ManageUsers	Limited	4505
5006	ManageCases	Read-Only	4506
5007	ManageUsers	Read-Only	4507

Below the Result Grid, an 'Output' pane shows a log of actions performed, including two 'SELECT \* FROM suspects WHERE CriminalRecord = 'Yes' LIMIT 0, 1000' queries.

2.

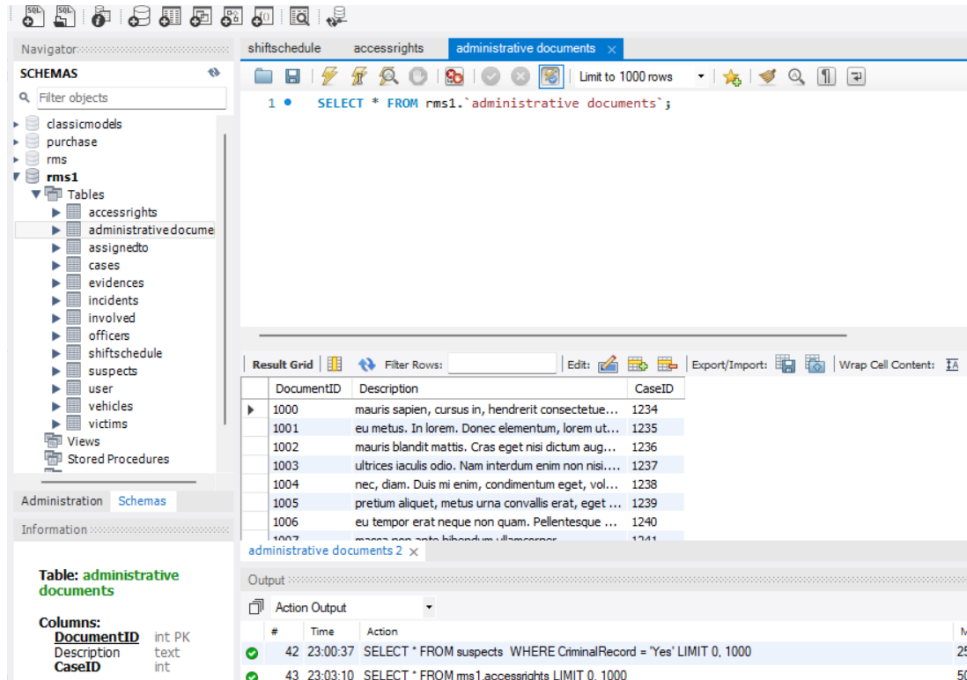
CREATE TABLE AdministrativeDocuments (

DocumentID INT PRIMARY KEY,

Description TEXT,

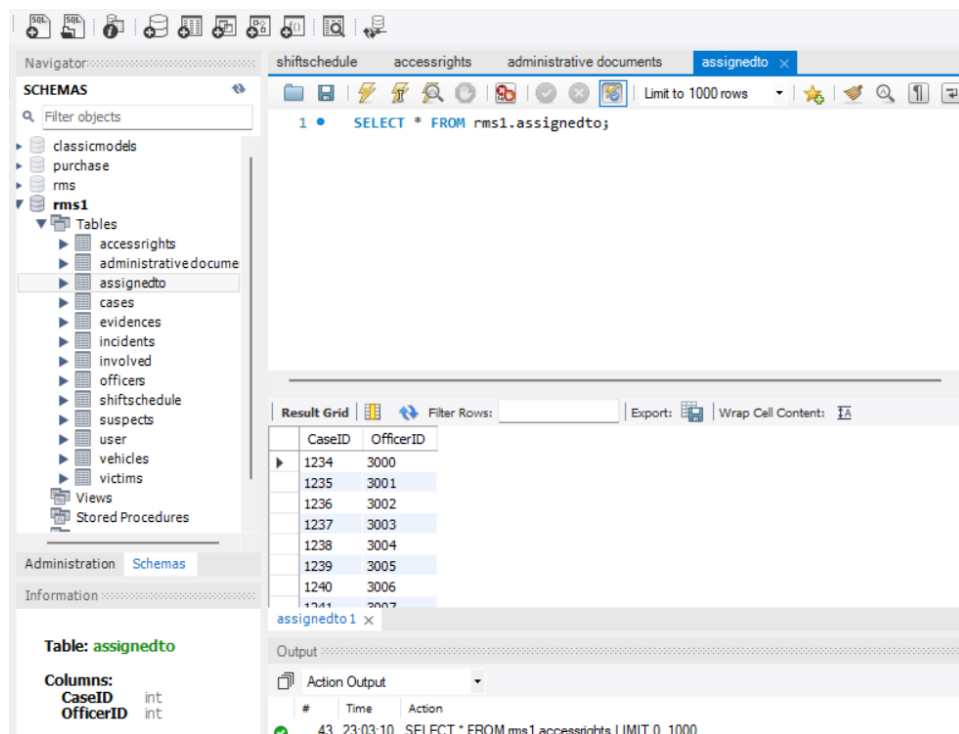
CaseID INT NOT NULL,  
 FOREIGN KEY (CaseID) REFERENCES Cases(CaseID)

);



3.

CREATE TABLE AssignedTo (  
 CaseID INT NOT NULL,  
 OfficerID INT NOT NULL,  
 PRIMARY KEY (CaseID, OfficerID),  
 FOREIGN KEY (CaseID) REFERENCES Cases(CaseID),  
 FOREIGN KEY (OfficerID) REFERENCES Officers(OfficerID)  
 );



4.

CREATE TABLE cases (

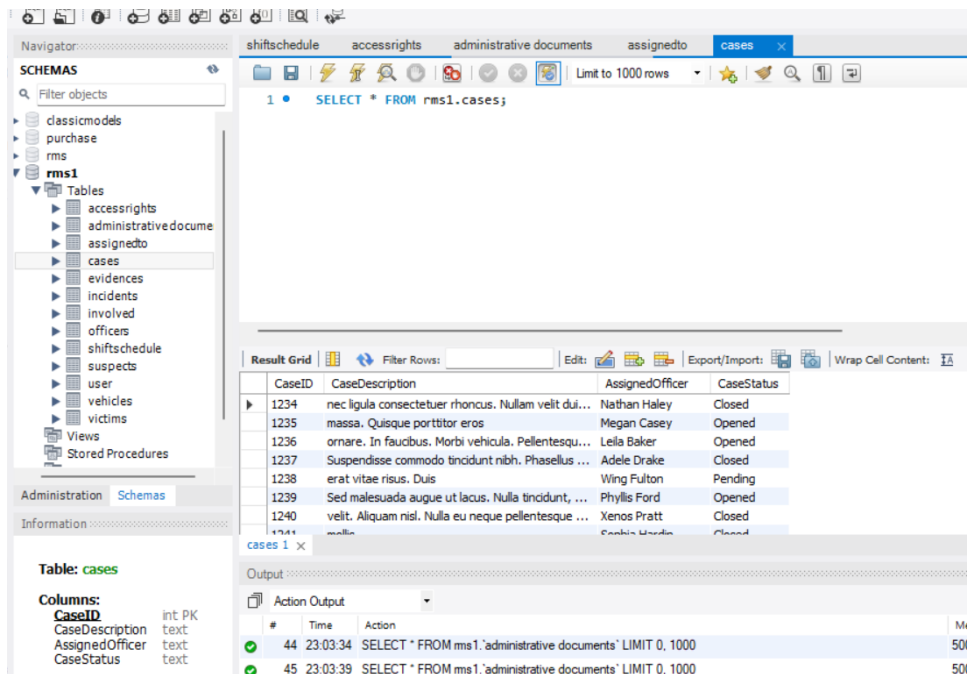
CaseID INT PRIMARY KEY,

CaseDescription TEXT,

AssignedOfficers TEXT,

CaseStatus TEXT

);



5.

CREATE TABLE evidences (

EvidenceID INT PRIMARY KEY,

EvidenceType TEXT,

ChainOfCustody TEXT,

EvidenceDescription TEXT,

StorageLocation TEXT,

CaseID INT NOT NULL,

FOREIGN KEY (CaseID) REFERENCES Case(CaseID)

);

The screenshot displays a database management interface. On the left, a 'SCHEMAS' pane shows a tree view of the database structure, including tables like 'evidences', 'incidents', 'involved', 'officers', 'shiftschedule', 'suspects', 'user', 'vehicles', and 'victims'. The 'evidences' table is selected, and its columns are listed in the 'Table: evidences' pane: EvidenceID (int, PK), EvidenceType (te), ChainOfCustody (te), EvidenceDescription (te), StorageLocation (te), and CaseID (int).

The main area shows a query editor with the following SQL statement:

```
SELECT * FROM rms1.evidences;
```

Below the query editor, a 'Result Grid' displays the query results. The grid has columns: EvidenceID, EvidenceType, ChainOfCustody, EvidenceDescription, StorageLocation, and CaseID. The results are as follows:

EvidenceID	EvidenceType	ChainOfCustody	EvidenceDescription	StorageLocation	CaseID
2000	Digital	TransportOfficer	ipsum cursus vestibulum. Mauris magna. Duis di...	ForensicLab	1234
2001	Digital	EvidenceClerk	consequat auctor, nunc nulla vulputate dul,	Off-SiteFacility	1235
2002	Tools	LawEnforcementOfficer	sit amet ultricies sem magna nec quam.	Off-SiteFacility	1236
2003	Photographic	ForensicAnalyst	eu erat semper rutrum. Fusce dolor quam, elem...	DigitalStorage	1237
2004	Vehicle	Other	tristique aliquet. Phasellus fermentum convallis	DigitalStorage	1238
2005	Digital	LawEnforcementOfficer	dictum. Proin eget odio. Aliquam vulputate ullam...	ColdStorage	1239
2006	Tools	TransportOfficer	molestie. Sed id	DigitalStorage	1240
2007	Document	Other	...	DigitalStorage	1241

At the bottom, an 'Output' pane shows the execution log with the following entries:

#	Time	Action	Message
45	23:03:39	SELECT * FROM rms1.'administrative documents' LIMIT 0, 1000	500 row(s) retu
46	23:05:15	SELECT * FROM rms1.accessrights LIMIT 0, 1000	500 row(s) retu
47	23:05:45	SELECT * FROM rms1.'administrative documents' LIMIT 0, 1000	500 row(s) retu
48	23:06:18	SELECT * FROM rms1.assignedto LIMIT 0, 1000	500 row(s) retu

6.

CREATE TABLE incidents (

IncidentID INT PRIMARY KEY,

IncidentLocation TEXT,

IncidentStatus TEXT,

IncidentDate DATE,

IncidentTime TIME,

CaseID INT NOT NULL,

FOREIGN KEY (CaseID) REFERENCES Cases(CaseID)

);

The screenshot shows a database management interface with a 'SCHEMAS' pane on the left and a 'Result Grid' on the right. The 'incidents' table is selected in the schemas pane, and its structure is displayed in the 'Table: incidents' section. The 'Result Grid' shows the output of a query: `SELECT * FROM rms1.incidents;`

**Table: incidents**

**Columns:**

- IncidentID** int PK
- IncidentLocation** tex
- IncidentStatus** tex
- IncidentDate** tex
- IncidentTime** tex
- CaseID** int

**Result Grid:**

IncidentID	IncidentLocation	IncidentStatus	IncidentDate	IncidentTime	CaseID
2500	Iowa	Closed	Jan 24, 2023	3:29 AM	1234
2501	Idaho	Resolved	Feb 8, 2023	8:40 AM	1235
2502	Kentucky	UnderInvestigation	Nov 22, 2022	8:16 AM	1236
2503	Iowa	Closed	Dec 13, 2023	2:41 PM	1237
2504	Wyoming	Resolved	Mar 18, 2024	10:30 PM	1238
2505	Wyoming	Resolved	Aug 16, 2023	2:14 AM	1239
2506	Pennsylvania	Closed	Oct 12, 2024	1:29 AM	1240

7.

CREATE TABLE Involved (

CaseID INT NOT NULL,

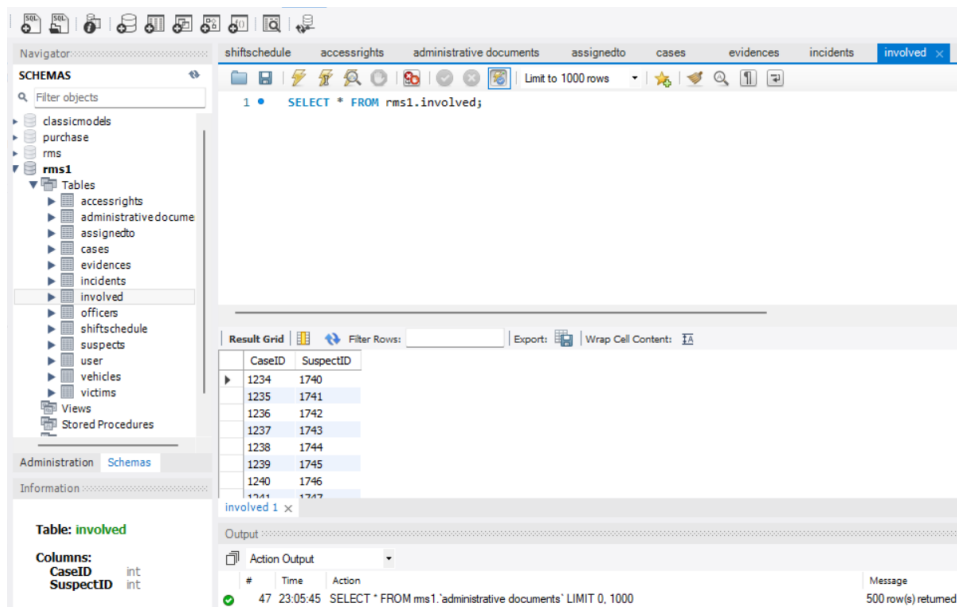
SuspectID INT NOT NULL,

PRIMARY KEY (CaseID, SuspectID),

FOREIGN KEY (CaseID) REFERENCES Cases(CaseID),

FOREIGN KEY (SuspectID) REFERENCES Suspect(SuspectID)

);



8.

CREATE TABLE Officers (

OfficerID INT PRIMARY KEY,

OfficerName TEXT,

OfficerContact TEXT,

OfficerRank TEXT,

BadgeNumber TEXT,

VehicleID INT NOT NULL,

ScheduleID INT NOT NULL,

UserID INT NOT NULL,

FOREIGN KEY (VehicleID) REFERENCES Vehicle(VehicleID),

FOREIGN KEY (ScheduleID) REFERENCES Schedule(ScheduleID),

FOREIGN KEY (UserID) REFERENCES User(UserID)

);



Navigator: shiftschedule, accessrights, administrative documents, assignedto, cases, evidences, incidents, involved, officers

SCHEMAS

Filter objects

classicmodels, purchase, rms, rms1

Tables: accessrights, administrative documents, assignedto, cases, evidences, incidents, involved, officers, shiftschedule, suspects, user, vehicles, victims

Views: Stored Procedures

Administration: Schemas

Information: Table: officers

Columns: OfficerID (int, PK), OfficerName (text), OfficerContact (text), OfficerRank (text), BadgeNumber (text), VehicleID (int), ScheduleID (int), UserID (int)

1 • SELECT \* FROM rms1.officers;

Result Grid

OfficerID	OfficerName	OfficerContact	OfficerRank	BadgeNumber	VehicleID	ScheduleID	UserID
3000	Gisela Farley	(275) 462-3737	Lieutenant	GPC54FWK10W	3500	4000	4500
3001	Hiroko Burt	(207) 265-9587	Sergeant	MQW68ZUJ9FU	3501	4001	4501
3002	Gregory Townsend	(370) 638-4515	Inspector	DUB66PDO2BT	3502	4002	4502
3003	Sonya Hayes	(585) 262-9722	Chief	ODY22XBY9MZ	3503	4003	4503
3004	Dominic Greer	(813) 738-8474	Inspector	KFI22PMQ7GX	3504	4004	4504
3005	Francesca Keller	(472) 985-6378	Captain	OEP90JFW2PF	3505	4005	4505
3006	Tyler Singleton	(636) 172-9602	Lieutenant	AYF34CUB3QI	3506	4006	4506

Output

#	Time	Action	Message
48	23:06:18	SELECT * FROM rms1.assignedto LIMIT 0, 1000	500 row(s) returned
49	23:07:55	SELECT * FROM rms1.cases LIMIT 0, 1000	500 row(s) returned
50	23:08:22	SELECT * FROM rms1.evidences LIMIT 0, 1000	500 row(s) returned
51	23:08:47	SELECT * FROM rms1.incidents LIMIT 0, 1000	500 row(s) returned
52	23:09:25	SELECT * FROM rms1.involved LIMIT 0, 1000	500 row(s) returned

9.

CREATE TABLE ShiftSchedule (  
 ScheduleID INT PRIMARY KEY,  
 ShiftDates DATE,  
 ShiftTime TIME  
 );

Navigator: shiftschedule, accessrights, administrative documents, assignedto, cases, evidences, incidents, involved, officers, shiftschedule

SCHEMAS

Filter objects

classicmodels, purchase, rms, rms1

Tables: accessrights, administrative documents, assignedto, cases, evidences, incidents, involved, officers, shiftschedule, suspects, user, vehicles, victims

Views: Stored Procedures

Administration: Schemas

Information: Table: shiftschedule

Columns: ScheduleID (int, PK), ShiftDates (text), ShiftTime (text)

1 • SELECT \* FROM rms1.shiftschedule;

Result Grid

ScheduleID	ShiftDates	ShiftTime
4000	Sat, Nov 25th, 2023	2:13 PM
4001	Sat, Apr 22nd, 2023	1:29 AM
4002	Sat, Sep 16th, 2023	3:33 AM
4003	Wed, Jul 3rd, 2024	11:15 AM
4004	Tue, Jan 10th, 2023	8:45 PM
4005	Tue, Feb 28th, 2023	1:02 AM
4006	Thu, Jan 11th, 2024	6:39 AM

Output

#	Time	Action	Message
49	23:07:55	SELECT * FROM rms1.cases LIMIT 0, 1000	500 row(s) returned

10.

```

CREATE TABLE suspects (

    SuspectID INT PRIMARY KEY,

    SuspectName TEXT,

    SuspectDOB DATE,

    SuspectAddress TEXT,

    CriminalRecord TEXT

);

```

The screenshot shows a database management interface with a sidebar on the left containing a 'SCHEMAS' tree. The 'suspects' table is selected. The main area displays the table's structure and data.

**Table: suspects**

**Columns:**

Column	Type	PK
SuspectID	int	PK
SuspectName	text	
SuspectDOB	text	
SuspectAddress	text	
CriminalRecord	text	

**Result Grid:**

SuspectID	SuspectName	SuspectDOB	SuspectAddress	CriminalRecord
1740	Or Short	May 20, 1988	P.O. Box 239, 9577 Alqum St.	Yes
1741	Ira McDowell	Dec 12, 1968	Ap #506-5373 Lectus, St.	Yes
1742	Iola Pratt	Apr 7, 1997	4282 Turpe St.	Yes
1743	Brittany Olsen	Jan 25, 1996	6117 Tempor Avenue	Yes
1744	Ora Reese	Jun 21, 1968	197-7769 Sem St.	Yes
1745	Katelyn Wang	Feb 1, 1990	P.O. Box 457, 323 Molestie Road	No
1746	Risa Kramer	Sep 8, 1982	Ap #518-5011 Brandt St.	No

**Output:**

#	Time	Action	Message
50	23:08:22	SELECT * FROM ms1.evidences LIMIT 0, 1000	500 row(s) returned
51	23:08:47	SELECT * FROM ms1.incidents LIMIT 0, 1000	500 row(s) returned

11.

```

CREATE TABLE Users (

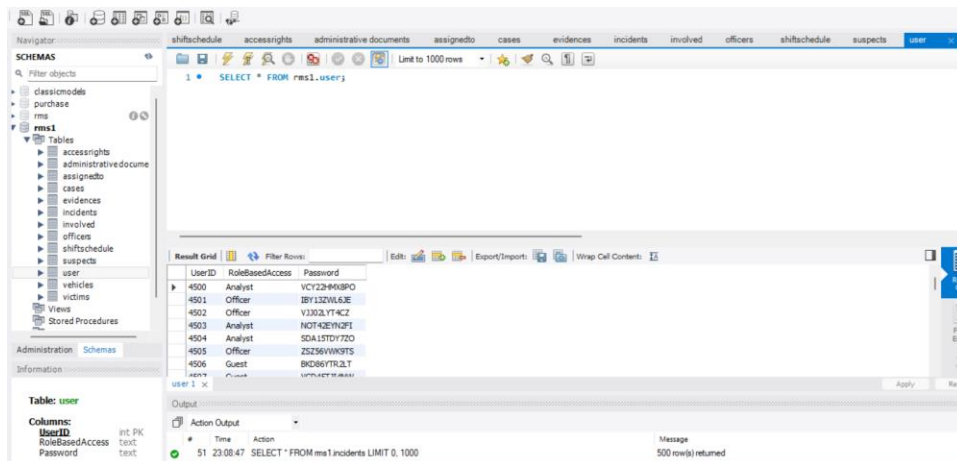
    UserID INT PRIMARY KEY,

    RoleBasedAccess TEXT,

    Password TEXT

);

```



12.

CREATE TABLE Vehicles (

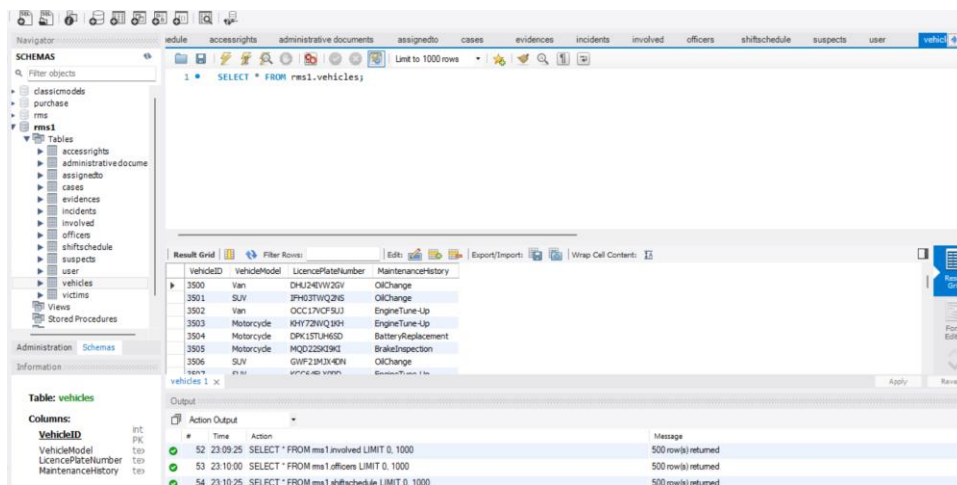
VehicleID INT PRIMARY KEY,

VehicleModel TEXT,

LicencePlateNumber TEXT,

MaintenanceHistory TEXT

);



13.

CREATE TABLE Victims (

VictimID INT PRIMARY KEY,

VictimName TEXT,  
 VictimContact TEXT,  
 VictimStatement TEXT,  
 CaseID INT NOT NULL,  
 FOREIGN KEY (CaseID) REFERENCES Cases(CaseID)  
 );

Table: **victims**

Columns:

- VictimID: INT PK
- VictimName: text
- VictimContact: text
- VictimStatement: text
- CaseID: INT

Result Grid:

VictimID	VictimName	VictimContact	VictimStatement	CaseID
1500	Henry Butler	(212) 450-4658	metus. Vivamus euismod urna. Nullam lobortis g...	1234
1501	Axel Finley	(503) 545-1784	magna nec quam. Curabitur vel lectus. Cum	1235
1502	Hannah Lawrence	(967) 997-7588	commodo auctor velit. Aliquam nisi. Nulla eu neq...	1236
1503	Zachary McDonald	(219) 131-8126	et enim. Curabitur massa. Vestibulum accumsan...	1237
1504	Amal Luna	(418) 621-4850	a sollicitudin orci sem eget massa. Suspendisse...	1238
1505	Jasper Cantu	(114) 306-1965	ligula consectetur rhoncus. Nullam velit dui, se...	1239
1506	Driscoll McKee	(502) 700-1515	egret metus. In nec orci. Donec nibh. Quisque n...	1240

Output:

#	Time	Action	Message
53	23:10:00	SELECT * FROM rms1.officers LIMIT 0, 1000	500 row(s) returned
54	23:10:25	SELECT * FROM rms1.shiftschedule LIMIT 0, 1000	500 row(s) returned

# Sample Queries

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Query 1 x shiftschedule

Limit to 1000 rows

SCHEMAS

Filter objects

- classicmodels
- purchase
- rms
- rms1
  - Tables
    - accessrights
    - administrative docume
    - assignedto
    - cases
    - evidences
    - incidents
    - involved
    - officers
    - shiftschedule
    - suspects
    - user
    - vehicles
    - victims
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

Table: accessrights

Columns:

- AccessID int PK
- AccessName text
- AccessLevel text
- UserID int

Object Info Session

Query 1 x shiftschedule

```
1 -- Retrieving details about cases, including CaseID, CaseDescription, AssignedOfficer, CaseStatus, and OfficerName
2 SELECT cases.CaseID, cases.CaseDescription, cases.AssignedOfficer, cases.CaseStatus, officers.OfficerName
3 FROM cases
4 LEFT JOIN AssignedTo ON cases.CaseID = AssignedTo.CaseID
5 LEFT JOIN officers ON assignedto.OfficerID = officers.OfficerID;
```

Result Grid

CaseID	CaseDescription	AssignedOfficer	CaseStatus	OfficerName
1234	nec ligula consectetur rhoncus. Nullam velit du...	Nathan Haley	Closed	Gisela Farley
1235	massa. Quisque porttitor eros	Megan Casey	Opened	Hiroko Burt
1236	ornare. In faucibus. Morbi vehicula. Pellentesqu...	Leila Baker	Opened	Gregory Townsend
1237	Suspendisse commodo tincidunt nibh. Phasellus ...	Adele Drake	Closed	Sonya Hayes
1238	erat vitae risus. Duis	Wing Fulton	Pending	Dominic Greer
1239	Sed malesuada augue ut lacus. Nulla tincidunt, ...	Phyllis Ford	Opened	Francesca Keller
1240	velit. Aliquam nisl. Nulla eu neque pellentesque ...	Xenos Pratt	Closed	Tyler Singleton
1241	mollis	Sophia Hardin	Closed	Samantha Burton

Output

Action Output

#	Time	Action	Message	Duration / Fetch
34	22:45:12	Apply changes to accessrights	Changes applied	
35	22:45:23	SELECT AccessName, AccessLevel FROM accessrights WHERE UserID = 4513 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
36	22:45:52	SELECT AccessName, AccessLevel FROM accessrights WHERE UserID = '4513', '4578'	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL se...	0.000 sec
37	22:46:10	SELECT AccessName, AccessLevel FROM accessrights WHERE UserID = '4513' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
38	22:47:02	SELECT AccessName, AccessLevel FROM accessrights WHERE UserID IN (4513, 4890, 4657) LIMIT 0, 1...	3 row(s) returned	0.000 sec / 0.000 sec
39	22:49:46	SELECT cases.CaseID, cases.CaseDescription, cases.AssignedOfficer, cases.CaseStatus, officers.OfficerNam...	500 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Read Only Context Help Snippets

ENG US

10:50 PM 11/12/2023

Navigation

SCHEMAS

Filter objects

classimodels

purchase

rms

**rms1**

Tables

accessrights

administrative docume

assignedto

cases

evidences

incidents

involved

officers

shiftschedule

suspects

user

vehicles

victims

Views

Stored Procedures

Administration

Schemas

Information

Table: accessrights

Columns:

AccessID int PK

AccessName text

AccessLevel text

UserID int

Object Info

Session

shiftschedule

Limit to 1000 rows

1 -- Retrieving information about incidents, specifically their location, date, and time, for incidents that occurred after January 1, 2022

2 • SELECT IncidentLocation, IncidentDate, IncidentTime

3 FROM incidents

4 WHERE IncidentDate > '2022-01-01'

5 LIMIT 0, 1000

6

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

IncidentLocation IncidentDate IncidentTime

Iowa Jan 24, 2023 3:29 AM

Idaho Feb 8, 2023 8:40 AM

Kentucky Nov 22, 2022 8:16 AM

Iowa Dec 13, 2023 2:41 PM

Wyoming Mar 18, 2024 10:30 PM

Wyoming Aug 16, 2023 2:14 AM

Pennsylvania Oct 12, 2024 1:29 AM

Massachusetts Oct 9, 2022 5:55 AM

incidents 2 x

Read Only

Output

Action Output

# Time Action Message

35 22:45:23 SELECT AccessName, AccessLevel FROM accessrights WHERE UserID = 4513 LIMIT 0, 1000 1 row(s) returned

36 22:45:52 SELECT AccessName, AccessLevel FROM accessrights WHERE UserID = '4513', '4678' Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''4678'' at line 1

37 22:46:10 SELECT AccessName, AccessLevel FROM accessrights WHERE UserID = '4513' LIMIT 0, 1000 1 row(s) returned

38 22:47:02 SELECT AccessName, AccessLevel FROM accessrights WHERE UserID IN ('4513', '4890', '4657') LIMIT 0, 1000 3 row(s) returned

39 22:49:46 SELECT cases CaseID, cases CaseDescription, cases AssignedOfficer, cases CaseStatus, officers OfficerName FROM cases WHERE CaseID = 4513 500 row(s) returned

40 22:56:55 SELECT IncidentLocation, IncidentDate, IncidentTime FROM incidents WHERE IncidentDate > '2022-01-01' 500 row(s) returned







SQL Server Enterprise Manager interface showing a SQL query and its results.

**Navigator:**

- SCHEMAS
  - Filter objects
  - classicmodels
  - purchase
  - rms
  - rms1**
    - Tables
      - accessrights
      - administrative docume
      - assignedto
      - cases
      - evidences
      - incidents
      - involved
      - officers
      - shiftschedule
      - suspects
      - user
      - vehicles
      - victims
    - Views
    - Stored Procedures

**Administration Schemas**

**Information**

**Table: victims**

**shiftschedule** **cases**

Limit to 1000 rows

```

1  -- Listing all the suspects who are born in November
2  • SELECT * FROM suspects
3    WHERE MONTH(STR_TO_DATE(SuspectDOB, '%M %d, %Y')) = 11;
4

```

**Result Grid** | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

SuspectID	SuspectName	SuspectDOB	SuspectAddress	CriminalRecord
1751	Sara Delaney	Nov 18, 1982	P.O. Box 813, 6054 Facilis Street	Yes
1777	Yuri Schmidt	Nov 7, 1992	Ap #949-9881 Ultricies Street	Yes
1787	Caleb Nelson	Nov 12, 1978	8931 Maecenas St.	No
1792	Basia Cantrell	Nov 4, 1998	P.O. Box 628, 4775 Dui. St.	No
1811	Tad Ferguson	Nov 6, 1970	505-3946 Sem St.	No
1863	Gail Padilla	Nov 20, 1977	Ap #698-3853 Lectus St.	No
1891	Cedric Burch	Nov 20, 1994	451-3019 Orci Avenue	No
1907	Oliver Ruiz	Nov 16, 1988	3500 Libero St.	Yes

suspects 8 x

**Output**