

6/3/2024



### **Project Response Team**

**Team Members :Mackenzie Falla, Elizaveta Sorokina**

## 10 Queries

1. Select users who have reported incidents:

```
SELECT DISTINCT u.UserID, u.Name  
FROM user u  
JOIN user_incident ui ON u.UserID = ui.UserID;
```

The screenshot shows the MySQL Workbench interface with the query results displayed in the Result Grid. The results show 10 rows of user information, each consisting of a UserID and Name. The names listed are John Doe, Jane Smith, Alice Johnson, Bob Brown, Charlie Black, David White, Eva Green, Frank Red, Grace Blue, and Hannah Yellow.

| UserID | Name          |
|--------|---------------|
| 1      | John Doe      |
| 2      | Jane Smith    |
| 3      | Alice Johnson |
| 4      | Bob Brown     |
| 5      | Charlie Black |
| 6      | David White   |
| 7      | Eva Green     |
| 8      | Frank Red     |
| 9      | Grace Blue    |
| 10     | Hannah Yellow |

The screenshot shows the MySQL Workbench interface with the results of the second query displayed in the Result Grid. The results show 10 rows of user information, identical to the first query's results. The names listed are John Doe, Jane Smith, Alice Johnson, Bob Brown, Charlie Black, David White, Eva Green, Frank Red, Grace Blue, and Hannah Yellow.

| UserID | Name          |
|--------|---------------|
| 1      | John Doe      |
| 2      | Jane Smith    |
| 3      | Alice Johnson |
| 4      | Bob Brown     |
| 5      | Charlie Black |
| 6      | David White   |
| 7      | Eva Green     |
| 8      | Frank Red     |
| 9      | Grace Blue    |
| 10     | Hannah Yellow |

2. Select all incidents and their associated user:

```
SELECT ei.IncidentID, ei.Type AS IncidentType, u.Name AS UserName
FROM emergency_incident ei
JOIN user_incident ui ON ei.IncidentID = ui.IncidentID
JOIN user u ON ui.UserID = u.UserID;
```

| IncidentID | IncidentType | UserName      |
|------------|--------------|---------------|
| 1          | Fire         | John Doe      |
| 2          | Accident     | Jane Smith    |
| 3          | Flood        | Alice Johnson |
| 4          | Robbery      | Bob Brown     |
| 5          | Medical      | Charlie Black |
| 6          | Earthquake   | Diana White   |
| 7          | Fire         | Eva Green     |
| 8          | Accident     | Frank Red     |
| 9          | Flood        | Grace Blue    |
| 10         | Medical      | Hannah Yellow |

| IncidentID | IncidentType | UserName      |
|------------|--------------|---------------|
| 1          | Fire         | John Doe      |
| 2          | Accident     | Jane Smith    |
| 3          | Flood        | Alice Johnson |
| 4          | Robbery      | Bob Brown     |
| 5          | Medical      | Charlie Black |
| 6          | Earthquake   | Diana White   |
| 7          | Fire         | Eva Green     |
| 8          | Accident     | Frank Red     |
| 9          | Flood        | Grace Blue    |
| 10         | Medical      | Hannah Yellow |

3. Select the number of incidents each response team has handled:

```
SELECT ResponseTeamID, COUNT(IncidentID) AS NumberOfIncidents
FROM responseteam_incident
GROUP BY ResponseTeamID;
```

The screenshot shows two separate instances of MySQL Workbench running on the same machine. Both instances have the same session details and are executing the same SQL query.

**Session 1 (Top):**

- Navigator:** Local instance MySQL80, showing Schemas (classicmodels), Tables (responseteam\_incident), Views, and Stored Procedures.
- SQL Editor:** Query 911db# containing the SQL code from step 3.
- Result Grid:** Shows the results of the query:
 

| ResponseTeamID | NumberOfIncidents |
|----------------|-------------------|
| 1              | 1                 |
| 2              | 1                 |
| 3              | 1                 |
| 4              | 1                 |
| 5              | 1                 |
| 6              | 1                 |
| 7              | 1                 |
| 8              | 1                 |
| 9              | 1                 |
| 10             | 1                 |
- Output:** Action Output showing log entries for the executed statements.

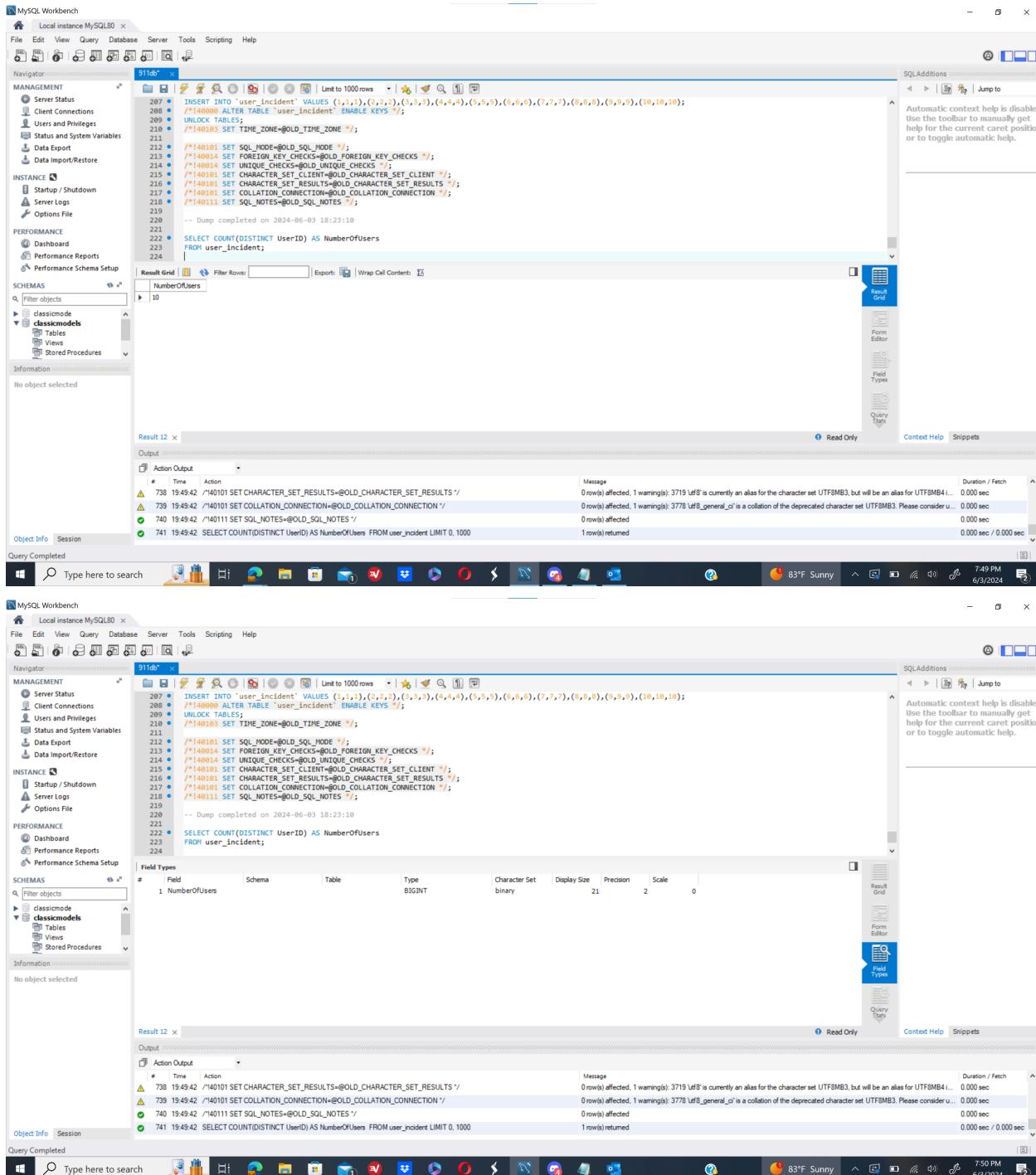
**Session 2 (Bottom):**

- Navigator:** Local instance MySQL80, showing the same schema structure.
- SQL Editor:** Query 911db# containing the same SQL code.
- Result Grid:** Shows the results of the query:
 

| ResponseTeamID | NumberOfIncidents |
|----------------|-------------------|
| 1              | 1                 |
| 2              | 1                 |
| 3              | 1                 |
| 4              | 1                 |
| 5              | 1                 |
| 6              | 1                 |
| 7              | 1                 |
| 8              | 1                 |
| 9              | 1                 |
| 10             | 1                 |
- Output:** Action Output showing log entries for the executed statements.

4. Select the number of users who have reported incidents:

```
SELECT COUNT(DISTINCT UserID) AS NumberOfUsers
FROM user_incident;
```



The screenshot shows the MySQL Workbench interface with two panes. The left pane displays the SQL editor with the following code:

```
207 • INSERT INTO `user_incident` VALUES (1,1),(2,2),(3,3),(4,4),(5,5),(6,6),(7,7),(8,8),(9,9),(10,10);
208 • /*140100 ALTER TABLE `user_incident` ENABLE KEYS */;
209 • 209
210 • /*140103 SET TIME_ZONE=@OLD_TIME_ZONE */;
211
212 • /*140101 SET SQL_MODE=@OLD_SQL_MODE */;
213 • /*140101 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
214 • /*140101 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
215 • /*140101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
216 • /*140101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
217 • /*140101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
218 • /*140111 SET SQL_NOTES=@OLD_SQL_NOTES */;
219
220 -- Dump completed on 2024-06-03 18:23:10
221
222 • SELECT COUNT(DISTINCT UserID) AS NumberOfUsers
223 FROM user_incident;
224
```

The right pane shows the results of the query in a grid format:

| NumberOfUsers |
|---------------|
| 10            |

Below the results, the "Action Output" section shows the following log entries:

- 738 19:49:42 /140101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS' 0 row(s) affected, 1 warning(s); 3719 'utf8' is currently an alias for the character set UTF8MB3, but will be an alias for UTF8MB4... 0.000 sec
- 739 19:49:42 /140101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION' 0 row(s) affected, 1 warning(s); 3778 'utf8\_general\_ci' is a collation of the deprecated character set UTF8MB3. Please consider u... 0.000 sec
- 740 19:49:42 /140111 SET SQL\_NOTES=@OLD\_SQL\_NOTES' 0 rows affected 0.000 sec
- 741 19:49:42 SELECT COUNT(DISTINCT UserID) AS NumberOfUsers FROM user\_incident LIMIT 0, 1000 1 row(s) returned 0.000 sec / 0.000 sec

The status bar at the bottom indicates "83°F Sunny" and the date "6/3/2024".

5. Select all users and their emergency contact:

```
SELECT u.UserID, u.Name AS UserName, ec.Name AS ContactName, ec.PhoneNumber, ec.Relationship
```

FROM user u

```
LEFT JOIN emergency_contact ec ON u.EmergencyContactID = ec.EmergencyContactID;
```

The screenshot shows the MySQL Workbench interface with two windows. The top window displays the results of the SQL query, showing 10 rows of data. The bottom window shows the 'Field Types' panel, which lists the columns and their characteristics.

**Result Grid Data:**

| User ID | User Name     | Contact Name | Phone Number | Relationship |
|---------|---------------|--------------|--------------|--------------|
| 1       | John Doe      | Alice        | 1234567890   | Friend       |
| 2       | Jane Smith    | Bob          | 2345678901   | Family       |
| 3       | Alice Johnson | Charlie      | 3456789012   | Colleague    |
| 4       | Bob Johnson   | David        | 4567890123   | Friend       |
| 5       | Charlie Bad   | Eva          | 5678901234   | Family       |
| 6       | David White   | Frank        | 6789012345   | Colleague    |
| 7       | Eva Green     | Grace        | 7890123456   | Friend       |
| 8       | Frank Red     | Hannah       | 8901234567   | Family       |
| 9       | Grace Blue    | Ian          | 9012345678   | Colleague    |
| 10      | Hannah Yellow | Jack         | 0123456789   | Friend       |

**Field Types Data:**

| # | Field        | Schema        | Table             | Type    | Character Set | Display Size | Precision | Scale |
|---|--------------|---------------|-------------------|---------|---------------|--------------|-----------|-------|
| 1 | UserID       | classicmodels | user              | INT     | binary        | 11           | 2         | 0     |
| 2 | UserName     | classicmodels | user              | VARCHAR | utf8          | 45           | 13        | 0     |
| 3 | ContactName  | classicmodels | emergency_contact | utf8    | 45            | 7            | 0         |       |
| 4 | PhoneNumber  | classicmodels | emergency_contact | VARCHAR | utf8          | 10           | 10        | 0     |
| 5 | Relationship | classicmodels | emergency_contact | VARCHAR | utf8          | 45           | 9         | 0     |

6. Select all incidents and their associated response teams:

```
SELECT ei.IncidentID, ei.Type AS IncidentType, rt.Location AS ResponseTeamLocation
FROM emergency_incident ei
JOIN responseteam_incident rti ON ei.IncidentID = rti.IncidentID
JOIN response_team rt ON rti.ResponseTeamID = rt.ResponseTeamID;
```

| IncidentID | IncidentType | ResponseTeamLocation |
|------------|--------------|----------------------|
| 1          | Fire         | Station 1            |
| 2          | Accident     | Station 2            |
| 3          | Flood        | Station 3            |
| 4          | Robbery      | Station 4            |
| 5          | Medical      | Station 5            |
| 6          | Earthquake   | Station 6            |
| 7          | Fire         | Station 7            |
| 8          | Accident     | Station 8            |
| 9          | Flood        | Station 9            |
| 10         | Medical      | Station 10           |

| IncidentID | IncidentType | ResponseTeamLocation |
|------------|--------------|----------------------|
| 1          | Fire         | Station 1            |
| 2          | Accident     | Station 2            |
| 3          | Flood        | Station 3            |
| 4          | Robbery      | Station 4            |
| 5          | Medical      | Station 5            |
| 6          | Earthquake   | Station 6            |
| 7          | Fire         | Station 7            |
| 8          | Accident     | Station 8            |
| 9          | Flood        | Station 9            |
| 10         | Medical      | Station 10           |

7. Select the most recent incident report for each incident:

```
SELECT ei.IncidentID, ir.Timestamp, ir.Description
FROM emergency_incident ei
JOIN incident_report ir ON ei.IncidentID = ir.IncidentID
WHERE ir.Timestamp = (SELECT MAX(Timestamp) FROM incident_report WHERE IncidentID = ei.IncidentID);
```

| IncidentID | Timestamp              | Description   |
|------------|------------------------|---|
| 1          | 2024-06-02 19:45:55.00 | Initial report of building on fire                        |
| 2          | 2024-06-02 19:45:55.00 | Car accident details [Initial report of building on fire] |
| 3          | 2024-06-02 19:45:55.00 | Flood situation monitored                                 |
| 4          | 2024-06-02 19:45:55.00 | Robbery details recorded                                  |
| 5          | 2024-06-02 19:45:55.00 | Medical emergency attended                                |
| 6          | 2024-06-02 19:45:55.00 | Earthquake impact assessed                                |
| 7          | 2024-06-02 19:45:55.00 | Forest fire status updated                                |
| 8          | 2024-06-02 19:45:55.00 | Accident scene managed                                    |
| 9          | 2024-06-02 19:45:55.00 | Flooding managed  |
| 10         | 2024-06-02 19:45:55.00 | Injury treated and logged                                 |

| IncidentID | Timestamp              | Description   |
|------------|------------------------|---|
| 1          | 2024-06-02 19:45:55.00 | Initial report of building on fire                        |
| 2          | 2024-06-02 19:45:55.00 | Car accident details [Initial report of building on fire] |
| 3          | 2024-06-02 19:45:55.00 | Flood situation monitored                                 |
| 4          | 2024-06-02 19:45:55.00 | Robbery details recorded                                  |
| 5          | 2024-06-02 19:45:55.00 | Medical emergency attended                                |
| 6          | 2024-06-02 19:45:55.00 | Earthquake impact assessed                                |
| 7          | 2024-06-02 19:45:55.00 | Forest fire status updated                                |
| 8          | 2024-06-02 19:45:55.00 | Accident scene managed                                    |
| 9          | 2024-06-02 19:45:55.00 | Flooding managed  |
| 10         | 2024-06-02 19:45:55.00 | Injury treated and logged                                 |

8. Select incidents and the number of reports associated with each incident:

```
SELECT ei.IncidentID, ei.Type, COUNT(ir.ReportID) AS ReportCount
FROM emergency_incident ei
JOIN incident_report ir ON ei.IncidentID = ir.IncidentID
GROUP BY ei.IncidentID, ei.Type;
```

| IncidentID | Type       | ReportCount |
|------------|------------|-------------|
| 1          | Fire       | 1           |
| 2          | Accident   | 1           |
| 3          | Flood      | 1           |
| 4          | Robbery    | 1           |
| 5          | Medical    | 1           |
| 6          | Earthquake | 1           |
| 7          | Fire       | 1           |
| 8          | Accident   | 1           |
| 9          | Flood      | 1           |
| 10         | Medical    | 1           |

Result 17 | Read Only | Context Help | Snippets

Action Output

| #    | Time     | Action  | Message  | Duration / Fetch      |
|------|----------|---|--|-----------------------|
| 1183 | 20:01:56 | /40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS /   | 0 rows(a) affected, 1 warning(s): 3719 'utf8' is currently an alias for the character set UTF8MB3, but will be an alias for UTF8MB4... | 0.000 sec             |
| 1184 | 20:01:56 | /40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION /   | 0 rows(a) affected, 1 warning(s): 3778 'utf8_general_ci' is a collation of the deprecated character set UTF8MB3. Please consider u...  | 0.000 sec             |
| 1185 | 20:01:56 | /40111 SET SQL_NOTES=@OLD_SQL_NOTES /   | 0 rows(a) affected   | 0.000 sec             |
| 1186 | 20:01:56 | SELECT ei.IncidentID, ei.Type, COUNT(ir.ReportID) AS ReportCount FROM emergency_incident ei JOIN incident_report ir ON e... | 10 row(s) returned   | 0.015 sec / 0.000 sec |

| # | Field       | Schema        | Table              | Type    | Character Set | Display Size | Precision | Scale |
|---|-------------|---------------|--------------------|---------|---------------|--------------|-----------|-------|
| 1 | IncidentID  | classicmodels | emergency_incident | INT     | binary        | 11           | 2         | 0     |
| 2 | Type        | classicmodels | emergency_incident | VARCHAR | utf8          | 100          | 10        | 0     |
| 3 | ReportCount |               |                    | BIGINT  | binary        | 21           | 1         | 0     |

Result 17 | Read Only | Context Help | Snippets

Action Output

| #    | Time     | Action  | Message  | Duration / Fetch      |
|------|----------|---|--|-----------------------|
| 1183 | 20:01:56 | /40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS /   | 0 rows(a) affected, 1 warning(s): 3719 'utf8' is currently an alias for the character set UTF8MB3, but will be an alias for UTF8MB4... | 0.000 sec             |
| 1184 | 20:01:56 | /40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION /   | 0 rows(a) affected, 1 warning(s): 3778 'utf8_general_ci' is a collation of the deprecated character set UTF8MB3. Please consider u...  | 0.000 sec             |
| 1185 | 20:01:56 | /40111 SET SQL_NOTES=@OLD_SQL_NOTES /   | 0 rows(a) affected   | 0.000 sec             |
| 1186 | 20:01:56 | SELECT ei.IncidentID, ei.Type, COUNT(ir.ReportID) AS ReportCount FROM emergency_incident ei JOIN incident_report ir ON e... | 10 row(s) returned   | 0.015 sec / 0.000 sec |

9. Select incidents and their total number of response teams:

```
SELECT ei.IncidentID, ei.Type, COUNT(rti.ResponseTeamID) ASResponseTeamCount
FROM emergency_incident ei
JOIN responseteam_incident rti ON ei.IncidentID = rti.IncidentID
GROUP BY ei.IncidentID, ei.Type;
```

| IncidentID | Type       | ASResponseTeamCount |
|------------|------------|---------------------|
| 1          | Fire       | 1                   |
| 2          | Accident   | 1                   |
| 3          | Flood      | 1                   |
| 4          | Robbery    | 1                   |
| 5          | Medical    | 1                   |
| 6          | Earthquake | 1                   |
| 7          | Fire       | 1                   |
| 8          | Accident   | 1                   |
| 9          | Flood      | 1                   |
| 10         | Medical    | 1                   |

| # | Field               | Schema        | Table              | Type    | Character Set | Display Size | Precision | Scale |
|---|---------------------|---------------|--------------------|---------|---------------|--------------|-----------|-------|
| 1 | IncidentID          | classicmodels | emergency_incident | INT     | binary        | 11           | 2         | 0     |
| 2 | Type                | classicmodels | emergency_incident | VARCHAR | utf8          | 100          | 10        | 0     |
| 3 | ASResponseTeamCount |               |                    | BIGINT  | binary        | 21           | 1         | 0     |

10. Select the count of incidents handled by each response team:

```
SELECT rt.ResponseTeamID, rt.Location, COUNT(rti.IncidentID) AS IncidentCount
FROM response_team rt
JOIN responseteam_incident rti ON rt.ResponseTeamID = rti.ResponseTeamID
GROUP BY rt.ResponseTeamID, rt.Location;
```

The screenshot shows the MySQL Workbench interface with a query editor and results grid.

**Query Editor:**

```
208  /*!40080 ALTER TABLE `user` incident` ENABLE KEYS */;
209  /*!40080 UNLOCK TABLES */;
210  /*!40183 SET TIME_ZONE=@OLD_TIME_ZONE */;
211
212  /*!40181 SET SQL_MODE=@OLD_SQL_MODE */;
213  /*!40141 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
214  /*!40129 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
215  /*!40181 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
216  /*!40181 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
217  /*!40181 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
218  /*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2024-06-03 18:23:10

222  SELECT rt.ResponseTeamID, rt.Location, COUNT(rt.IncidentID) AS IncidentCount FROM response_team rt
223  JOIN responseteam_incident rti ON rt.ResponseTeamID = rti.ResponseTeamID
224  GROUP BY rt.ResponseTeamID, rt.Location;
225
```

**Result Grid:**

| ResponseTeamID | Location   | IncidentCount |
|----------------|------------|---------------|
| 1              | Station 1  | 1             |
| 2              | Station 2  | 1             |
| 3              | Station 3  | 1             |
| 4              | Station 4  | 1             |
| 5              | Station 5  | 1             |
| 6              | Station 6  | 1             |
| 7              | Station 7  | 1             |
| 8              | Station 8  | 1             |
| 9              | Station 9  | 1             |
| 10             | Station 10 | 1             |

**Output:**

| Action | Time     | Action  | Message   | Duration / Fetch      |
|--------|----------|---|---|-----------------------|
| 1094   | 19:59:35 | /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;   | 0 row(s) affected, 1 warning(s); 3719 utf8' is currently an alias for the character set UTF8MB3, but will be an alias for UTF8MB4 in... | 0.000 sec             |
| 1095   | 19:59:35 | /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;   | 0 row(s) affected, 1 warning(s); 3778 utf8_general_ci is a collation of the deprecated character set UTF8MB3. Please consider us...     | 0.000 sec             |
| 1096   | 19:59:35 | /*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;   | 0 row(s) affected   | 0.000 sec             |
| 1097   | 19:59:35 | SELECT rt.ResponseTeamID, rt.Location, COUNT(rt.IncidentID) AS IncidentCount FROM response_team rt JOIN responseteam_i... | 10 row(s) returned  | 0.000 sec / 0.000 sec |

**Object Info:** Session

**Query Completed**

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

The screenshot shows the MySQL Workbench interface with the following details:

- File Menu:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** MANAGEMENT, INSTANCE, PERFORMANCE.
- Central Area:** A large text area displays a MySQL dump log from 2024-06-03 18:23:18. The log includes various SQL statements such as `ALTER TABLE`, `SET TIME\_ZONE`, and `SELECT` queries.
- Table:** A table titled "Field Types" lists columns for ResponseTeamID, Location, and IncidentCount with their respective data types and character sets.
- Result Grid:** A grid view of the same data as the table.
- Form Editor:** A form editor interface.
- Context Help:** A help panel on the right side.
- Output Tab:** Shows the results of the executed SQL statements, including messages about character set changes and warnings about deprecated character sets.
- Object Info:** Shows the selected object is a session.
- Session Tab:** Shows the session status.
- Bottom Bar:** Includes a search bar, system icons, and a status bar indicating 83°F Sunny at 8:00 PM on 6/3/2024.

## Each Table:

`SELECT * FROM user LIMIT 10;`

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the schema `classicmodels` and its tables.
- SQL File 3:** Contains the SQL code for the query: `SELECT * FROM user LIMIT 10;`
- Result Grid:** Displays the results of the query, showing 10 rows of data from the `user` table.
- Action Output:** Shows the execution log with the following entries:
  - 274 18:03:46 /140101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS /
  - 275 18:03:46 /140101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION /
  - 276 18:03:46 /140111 SET SQL\_NOTES=@OLD\_SQL\_NOTES /
  - 277 18:03:46 SELECT \* FROM user LIMIT 10;
- System Tray:** Shows the Windows taskbar with various icons and system status.

`SELECT * FROM emergency_contact LIMIT 10;`

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the schema `classicmodels` and its tables.
- SQL File 3:** Contains the SQL code for the query: `SELECT * FROM user LIMIT 10;`
- Result Grid:** Displays the results of the query, showing 10 rows of data from the `user` table.
- Action Output:** Shows the execution log with the following entries:
  - 274 18:03:46 /140101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS /
  - 275 18:03:46 /140101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION /
  - 276 18:03:46 /140111 SET SQL\_NOTES=@OLD\_SQL\_NOTES /
  - 277 18:03:46 SELECT \* FROM user LIMIT 10;
- System Tray:** Shows the Windows taskbar with various icons and system status.

```
SELECT * FROM user_incident LIMIT 10;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the schema `classicmodels` and table `customers`.
- SQL Editor:** Contains the SQL command `SELECT \* FROM user\_incident LIMIT 10;` and its execution log.
- Result Grid:** Displays the 10 rows of data from the `user\_incident` table.
- Action Output:** Shows the execution log with 452, 453, 454, and 456 entries.
- System Bar:** Includes the Windows taskbar with various pinned icons like File Explorer, Edge, and Mail.

```
SELECT * FROM emergency_incident LIMIT 10;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the schema `classicmodels` and table `customers`.
- SQL Editor:** Contains the SQL command `SELECT \* FROM emergency\_incident LIMIT 10;` and its execution log.
- Result Grid:** Displays the 10 rows of data from the `emergency\_incident` table.
- Action Output:** Shows the execution log with 541, 542, 543, and 548 entries.
- System Bar:** Includes the Windows taskbar with various pinned icons like File Explorer, Edge, and Mail.

`SELECT * FROM responseteam_incident LIMIT 10;`

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database structure for the `classicmodels` schema, including tables like `customers`.
- SQL Editor:** Contains the query `SELECT * FROM responseteam_incident LIMIT 10;`. The results show 10 rows of data from the `responseteam_incident` table.
- Result Grid:** Displays the data in columns: `ID`, `ResponseTeamID`, and `IncidentID`. The data is as follows:

| ID | ResponseTeamID | IncidentID |
|----|----------------|------------|
| 1  | 1              | 1          |
| 2  | 2              | 2          |
| 3  | 3              | 3          |
| 4  | 4              | 4          |
| 5  | 5              | 5          |
| 6  | 6              | 6          |
| 7  | 7              | 7          |
| 8  | 8              | 8          |
| 9  | 9              | 9          |
| 10 | 10             | 10         |
- Action Output:** Shows the execution log with entries for each row insertion and the final query execution.
- System Bar:** Includes the Windows taskbar with various pinned icons.

`SELECT * FROM incident_report LIMIT 10;`

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database structure for the `classicmodels` schema, including tables like `customers`.
- SQL Editor:** Contains the query `SELECT * FROM incident_report LIMIT 10;`. The results show 10 rows of data from the `incident_report` table.
- Result Grid:** Displays the data in columns: `ReportID`, `IncidentID`, `Timestamp`, and `Description`. The data is as follows:

| ReportID | IncidentID | Timestamp              | Description                            |
|----------|------------|------------------------|--|
| 1        | 1          | 2024-06-02 23:45:55.00 | Call received about a building on fire |
| 2        | 2          | 2024-06-02 23:45:55.00 | Car accident details logged            |
| 3        | 3          | 2024-06-02 23:45:55.00 | Flood status monitored                 |
| 4        | 4          | 2024-06-02 23:45:55.00 | Robbery details recorded               |
| 5        | 5          | 2024-06-02 23:45:55.00 | Medical emergency details recorded     |
| 6        | 6          | 2024-06-02 23:45:55.00 | Earthquake impact assessed             |
| 7        | 7          | 2024-06-02 23:45:55.00 | Forest fire status updated             |
| 8        | 8          | 2024-06-02 23:45:55.00 | Accident scene managed                 |
| 9        | 9          | 2024-06-02 23:45:55.00 | Flooding managed                       |
| 10       | 10         | 2024-06-02 23:45:55.00 | Injury treated and logged              |
- Action Output:** Shows the execution log with entries for each row insertion and the final query execution.
- System Bar:** Includes the Windows taskbar with various pinned icons.

SELECT \* FROM response\_team LIMIT 10;

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** MANAGEMENT, INSTANCE, PERFORMANCE.
- SQL Editor:** SQL File 3 (911db\*), containing the query: "SELECT \* FROM response\_team LIMIT 10;".
- Results Grid:** Shows the "response\_team" table with 10 rows of data:

| ResponseTeamID | Location   |
|----------------|------------|
| 1              | Station 1  |
| 2              | Station 2  |
| 3              | Station 3  |
| 4              | Station 4  |
| 5              | Station 5  |
| 6              | Station 6  |
| 7              | Station 7  |
| 8              | Station 8  |
| 9              | Station 9  |
| 10             | Station 10 |

- Output Tab:** Action Output, showing the execution log:

| Action       | Time   | Message   | Duration / Fetch      |
|--------------|--|---|-----------------------|
| 808 18:10:06 | /40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS '/ | 0 row(s) affected, 1 warning(s): 3719 'utf8' is currently an alias for the character set UTF8MB3, but will be an alias for UTF8MB4 in a future rel... | 0.000 sec             |
| 809 18:10:06 | /40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION '/   | 0 row(s) affected, 1 warning(s): 3778 'utf8_general_ci' is a collation of the deprecated character set UTF8MB3. Please consider using UTF8M...        | 0.000 sec             |
| 810 18:10:06 | /40111 SET SQL_NOTES=@OLD_SQL_NOTES '/                         | 0 row(s) affected   | 0.000 sec             |
| 811 18:10:06 | SELECT * FROM response_team LIMIT 10                           | 10 row(s) returned  | 0.000 sec / 0.000 sec |

- System Bar:** Shows the taskbar with various application icons and the system clock indicating 6:10 PM on 6/4/2024.

## Code:

```
-- MySQL dump 10.13 Distrib 8.0.36, for Win64 (x86_64)
--
-- Host: localhost Database: 911_db
-----
-- Server version      8.4.0

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!50503 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `emergency_contact`
--

DROP TABLE IF EXISTS `emergency_contact`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `emergency_contact` (
  `EmergencyContactID` int NOT NULL,
  `PhoneNumber` varchar(10) DEFAULT NULL,
  `Relationship` varchar(45) DEFAULT NULL,
  `Location` varchar(45) DEFAULT NULL,
  `Name` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`EmergencyContactID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `emergency_contact`
--

LOCK TABLES `emergency_contact` WRITE;
```

```

/*!40000 ALTER TABLE `emergency_contact` DISABLE KEYS */;
INSERT INTO `emergency_contact` VALUES (1,'1234567890','Friend','City
A','Alice'),(2,'2345678901','Family','City B','Bob'),(3,'3456789012','Colleague','City
C','Charlie'),(4,'4567890123','Friend','City D','David'),(5,'5678901234','Family','City
E','Eva'),(6,'6789012345','Colleague','City F','Frank'),(7,'7890123456','Friend','City
G','Grace'),(8,'8901234567','Family','City H','Hannah'),(9,'9012345678','Colleague','City
I','Ian'),(10,'0123456789','Friend','City J','Jack');

/*!40000 ALTER TABLE `emergency_contact` ENABLE KEYS */;
UNLOCK TABLES;

-- 
-- Table structure for table `emergency_incident`
-- 

DROP TABLE IF EXISTS `emergency_incident`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;

CREATE TABLE `emergency_incident` (
  `IncidentID` int NOT NULL,
  `Type` varchar(100) DEFAULT NULL,
  `Description` varchar(200) DEFAULT NULL,
  `Location` varchar(100) DEFAULT NULL,
  `Timestamp` timestamp(2) NULL DEFAULT NULL,
  PRIMARY KEY (`IncidentID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

-- 
-- Dumping data for table `emergency_incident`
-- 

LOCK TABLES `emergency_incident` WRITE;
/*!40000 ALTER TABLE `emergency_incident` DISABLE KEYS */;
INSERT INTO `emergency_incident` VALUES (1,'Fire','Building on fire','City A','2024-06-02
23:45:55.00'),(2,'Accident','Car accident on highway','City B','2024-06-02
23:45:55.00'),(3,'Flood','Flood in residential area','City C','2024-06-02
23:45:55.00'),(4,'Robbery','Bank robbery in progress','City D','2024-06-02
23:45:55.00'),(5,'Medical','Heart attack','City E','2024-06-02 23:45:55.00'),(6,'Earthquake','Minor
earthquake reported','City F','2024-06-02 23:45:55.00'),(7,'Fire','Forest fire','City G','2024-06-02
23:45:55.00'),(8,'Accident','Multiple vehicle collision','City H','2024-06-02
23:45:55.00'),(9,'Flood','River overflowing','City I','2024-06-02
23:45:55.00'),(10,'Medical','Serious injury','City J','2024-06-02 23:45:55.00');

/*!40000 ALTER TABLE `emergency_incident` ENABLE KEYS */;
UNLOCK TABLES;

```

```

--  

-- Table structure for table `incident_report`  

--  

DROP TABLE IF EXISTS `incident_report`;  

/*!40101 SET @saved_cs_client    = @@character_set_client */;  

/*!50503 SET character_set_client = utf8mb4 */;  

CREATE TABLE `incident_report` (  

  `ReportID` int NOT NULL,  

  `IncidentID` int DEFAULT NULL,  

  `Timestamp` timestamp(2) NULL DEFAULT NULL,  

  `Description` varchar(200) DEFAULT NULL,  

  PRIMARY KEY (`ReportID`),  

  KEY `EI_EmergencyIncident_idx` (`IncidentID`),  

  CONSTRAINT `EI_EmergencyIncident` FOREIGN KEY (`IncidentID`) REFERENCES  

  `emergency_incident` (`IncidentID`)  

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  

/*!40101 SET character_set_client = @saved_cs_client */;  

--  

-- Dumping data for table `incident_report`  

--  

LOCK TABLES `incident_report` WRITE;  

/*!40000 ALTER TABLE `incident_report` DISABLE KEYS */;  

INSERT INTO `incident_report` VALUES (1,1,'2024-06-02 23:45:55.00','Initial report of  

building on fire'),(2,2,'2024-06-02 23:45:55.00','Car accident details logged'),(3,3,'2024-06-02  

23:45:55.00','Flood situation monitored'),(4,4,'2024-06-02 23:45:55.00','Robbery details  

recorded'),(5,5,'2024-06-02 23:45:55.00','Medical emergency attended'),(6,6,'2024-06-02  

23:45:55.00','Earthquake impact assessed'),(7,7,'2024-06-02 23:45:55.00','Forest fire status  

updated'),(8,8,'2024-06-02 23:45:55.00','Accident scene managed'),(9,9,'2024-06-02  

23:45:55.00','Flooding managed'),(10,10,'2024-06-02 23:45:55.00','Injury treated and logged');  

/*!40000 ALTER TABLE `incident_report` ENABLE KEYS */;  

UNLOCK TABLES;  

--  

-- Table structure for table `response_team`  

--  

DROP TABLE IF EXISTS `response_team`;  

/*!40101 SET @saved_cs_client    = @@character_set_client */;  

/*!50503 SET character_set_client = utf8mb4 */;  

CREATE TABLE `response_team` (

```

```

`ResponseTeamID` int NOT NULL,
`Location` varchar(200) DEFAULT NULL,
PRIMARY KEY (`ResponseTeamID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

-- 
-- Dumping data for table `response_team`
-- 

LOCK TABLES `response_team` WRITE;
/*!40000 ALTER TABLE `response_team` DISABLE KEYS */;
INSERT INTO `response_team` VALUES (1,'Station 1'),(2,'Station 2'),(3,'Station 3'),(4,'Station 4'),(5,'Station 5'),(6,'Station 6'),(7,'Station 7'),(8,'Station 8'),(9,'Station 9'),(10,'Station 10');
/*!40000 ALTER TABLE `response_team` ENABLE KEYS */;
UNLOCK TABLES;

-- 
-- Table structure for table `responseteam_incident`
-- 

DROP TABLE IF EXISTS `responseteam_incident`;
/*!40101 SET @saved_cs_client  = @@character_set_client */;
/*!40101 SET character_set_client = utf8mb4 */;
CREATE TABLE `responseteam_incident` (
`ID` int NOT NULL,
`ResponseTeamID` int DEFAULT NULL,
`IncidentID` int DEFAULT NULL,
PRIMARY KEY (`ID`),
KEY `RI_IncidentID_idx` (`IncidentID`),
KEY `RI_ResponseTeamID_idx` (`ResponseTeamID`),
CONSTRAINT `RI_IncidentID` FOREIGN KEY (`IncidentID`) REFERENCES `emergency_incident` (`IncidentID`),
CONSTRAINT `RI_ResponseTeamID` FOREIGN KEY (`ResponseTeamID`) REFERENCES `response_team` (`ResponseTeamID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

-- 
-- Dumping data for table `responseteam_incident`
-- 

LOCK TABLES `responseteam_incident` WRITE;
/*!40000 ALTER TABLE `responseteam_incident` DISABLE KEYS */;

```

```

INSERT INTO `responseteam_incident` VALUES
(1,1,1),(2,2,2),(3,3,3),(4,4,4),(5,5,5),(6,6,6),(7,7,7),(8,8,8),(9,9,9),(10,10,10);
/*!40000 ALTER TABLE `responseteam_incident` ENABLE KEYS */;
UNLOCK TABLES;

-- 
-- Table structure for table `user`
-- 

DROP TABLE IF EXISTS `user`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8mb4 */;
CREATE TABLE `user` (
  `UserID` int NOT NULL,
  `Name` varchar(45) DEFAULT NULL,
  `ContactInformation` varchar(45) DEFAULT NULL,
  `Location` varchar(45) DEFAULT NULL,
  `EmergencyContactID` int DEFAULT NULL,
  PRIMARY KEY (`UserID`),
  KEY `EC_EmergencyContactID_idx` (`EmergencyContactID`),
  CONSTRAINT `EC_EmergencyContactID` FOREIGN KEY (`EmergencyContactID`)
    REFERENCES `emergency_contact` (`EmergencyContactID`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

-- 
-- Dumping data for table `user`
-- 

LOCK TABLES `user` WRITE;
/*!40000 ALTER TABLE `user` DISABLE KEYS */;
INSERT INTO `user` VALUES (1,'John Doe','john@example.com','City A',1),(2,'Jane Smith','jane@example.com','City B',2),(3,'Alice Johnson','alice@example.com','City C',3),(4,'Bob Brown','bob@example.com','City D',4),(5,'Charlie Black','charlie@example.com','City E',5),(6,'David White','david@example.com','City F',6),(7,'Eva Green','eva@example.com','City G',7),(8,'Frank Red','frank@example.com','City H',8),(9,'Grace Blue','grace@example.com','City I',9),(10,'Hannah Yellow','hannah@example.com','City J',10);
/*!40000 ALTER TABLE `user` ENABLE KEYS */;
UNLOCK TABLES;

-- 
-- Table structure for table `user_incident`
-- 

```

```

DROP TABLE IF EXISTS `user_incident`;
/*!40101 SET @saved_cs_client    = @@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `user_incident` (
  `ID` int NOT NULL,
  `UserID` int DEFAULT NULL,
  `IncidentID` int DEFAULT NULL,
  PRIMARY KEY (`ID`),
  KEY `UI_UserID_idx` (`UserID`),
  KEY `UI_IncidentID_idx` (`IncidentID`),
  CONSTRAINT `UI_IncidentID` FOREIGN KEY (`IncidentID`) REFERENCES
  `emergency_incident` (`IncidentID`) ON DELETE CASCADE,
  CONSTRAINT `UI_UserID` FOREIGN KEY (`UserID`) REFERENCES `user` (`UserID`) ON
  DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

-- 
-- Dumping data for table `user_incident`
-- 

LOCK TABLES `user_incident` WRITE;
/*!40000 ALTER TABLE `user_incident` DISABLE KEYS */;
INSERT INTO `user_incident` VALUES
(1,1,1),(2,2,2),(3,3,3),(4,4,4),(5,5,5),(6,6,6),(7,7,7),(8,8,8),(9,9,9),(10,10,10);
/*!40000 ALTER TABLE `user_incident` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2024-06-03 18:23:10

SELECT DISTINCT u.UserID, u.Name
FROM user u
JOIN user_incident ui ON u.UserID = ui.UserID;

SELECT ei.IncidentID, ei.Type AS IncidentType, u.Name AS UserName

```

```
FROM emergency_incident ei
JOIN user_incident ui ON ei.IncidentID = ui.IncidentID
JOIN user u ON ui.UserID = u.UserID;

SELECT ResponseTeamID, COUNT(IncidentID) AS NumberOfIncidents
FROM responseteam_incident
GROUP BY ResponseTeamID;

SELECT COUNT(DISTINCT UserID) AS NumberOfUsers
FROM user_incident;

SELECT u.UserID, u.Name AS UserName, ec.Name AS ContactName, ec.PhoneNumber,
ec.Relationship
FROM user u
LEFT JOIN emergency_contact ec ON u.EmergencyContactID = ec.EmergencyContactID;

SELECT ei.IncidentID, ei.Type AS IncidentType, rt.Location AS ResponseTeamLocation FROM
emergency_incident ei
JOIN responseteam_incident rti ON ei.IncidentID = rti.IncidentID
JOIN response_team rt ON rti.ResponseTeamID = rt.ResponseTeamID;

SELECT ei.IncidentID, ir.Timestamp, ir.Description
FROM emergency_incident ei
JOIN incident_report ir ON ei.IncidentID = ir.IncidentID
WHERE ir.Timestamp = (SELECT MAX(Timestamp) FROM incident_report WHERE
IncidentID = ei.IncidentID);

SELECT ei.IncidentID, ei.Type, COUNT(ir.ReportID) AS ReportCount
FROM emergency_incident ei
JOIN incident_report ir ON ei.IncidentID = ir.IncidentID
GROUP BY ei.IncidentID, ei.Type;

SELECT ei.IncidentID, ei.Type, COUNT(rti.ResponseTeamID) AS ResponseTeamCount
FROM emergency_incident ei
JOIN responseteam_incident rti ON ei.IncidentID = rti.IncidentID
GROUP BY ei.IncidentID, ei.Type;

SELECT rt.ResponseTeamID, rt.Location, COUNT(rti.IncidentID) AS IncidentCount FROM
response_team rt
JOIN responseteam_incident rti ON rt.ResponseTeamID = rti.ResponseTeamID
GROUP BY rt.ResponseTeamID, rt.Location;

SELECT * FROM user LIMIT 10;
```

```
SELECT * FROM emergency_contact LIMIT 10;
```

```
SELECT * FROM user_incident LIMIT 10;
```

```
SELECT * FROM emergency_incident LIMIT 10;
```

```
SELECT * FROM responseteam_incident LIMIT 10;
```

```
SELECT * FROM incident_report LIMIT 10;
```

```
SELECT * FROM response_team LIMIT 10;
```