Task 1. Retrieve employee device data

In this task, you need to obtain information on employee devices because your team needs to update them. The information you need is in the `machines` table in the `organization` database.

First, you need to retrieve all the information about the employee devices.

1. Run the following query to select all device information from the `machines` table:

```
SELECT *
FROM machines;
```

Note: *Using the asterisk (`*`) returns all data from the specified table. Also, table names in MySQL are case-sensitive.*

The output returns all the contents of the `machines` table:

```
clear
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.3.39-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [organization]> clear
MariaDB [organization]> SELECT *
    ->
    -> FROM machines;
+--------------+------------------+----------------+--------------+-------------+
| device_id    | operating_system | email_client   | OS_patch_date | employee_id |
+--------------+------------------+----------------+--------------+-------------+
| a184b775c707 | OS 1             | Email Client 1 | 2021-09-01    |        1156 |
| a192b174c940 | OS 2             | Email Client 1 | 2021-06-01    |        1052 |
| a305b818c708 | OS 3             | Email Client 2 | 2021-06-01    |        1182 |
| a317b635c465 | OS 1             | Email Client 2 | 2021-03-01    |        1130 |
| a320b137c219 | OS 2             | Email Client 2 | 2021-03-01    |        1000 |
| a398b471c573 | OS 3             | Email Client 2 | 2021-12-01    |           0 |
| a667b270c984 | OS 1             | Email Client 1 | 2021-03-01    |        1078 |
| a821b452c176 | OS 2             | Email Client 2 | 2021-12-01    |        1104 |
| a998b568c863 | OS 3             | Email Client 1 | 2021-12-01    |        1026 |
| b157c491d493 | OS 2             | Email Client 1 | 2021-03-01    |           0 |
| b239c825d303 | OS 1             | Email Client 1 | 2021-03-01    |        1001 |
| b264c773d977 | OS 2             | Email Client 2 | 2021-03-01    |        1157 |
| b265c937d713 | OS 2             | Email Client 1 | 2021-09-01    |        1131 |
| b433c245d868 | OS 1             | Email Client 1 | 2021-06-01    |        1079 |
| b551c837d758 | OS 3             | Email Client 1 | 2021-03-01    |        1105 |
```

Next, you want to focus on the email client running on various devices.

2. Run the following query to select only the `device_id` and `email_client` columns from the `machines` table. Replace `X` with `device_id` and `Y` with `email_client`:

The correct query to solve this step:

```
SELECT device_id, email_client

FROM machines;
```

The output should return only the selected columns of the `machines` table:

```
MariaDB [organization]> SELECT device_id, email_client
    ->
    -> FROM machines;
+--------------+----------------+
| device_id    | email_client   |
+--------------+----------------+
| a184b775c707 | Email Client 1 |
| a192b174c940 | Email Client 1 |
| a305b818c708 | Email Client 2 |
| a317b635c465 | Email Client 2 |
| a320b137c219 | Email Client 2 |
| a398b471c573 | Email Client 2 |
| a667b270c984 | Email Client 1 |
| a821b452c176 | Email Client 2 |
| a998b568c863 | Email Client 1 |
| b157c491d493 | Email Client 1 |
| b239c825d303 | Email Client 1 |
| b264c773d977 | Email Client 2 |
| b265c937d713 | Email Client 1 |
| b433c245d868 | Email Client 1 |
| b551c837d758 | Email Client 1 |
| b566c710d544 | Email Client 1 |
| b806c503d354 | Email Client 1 |
| b979c871d361 | Email Client 1 |
| c116d593e558 | Email Client 1 |
| c150d982e144 | Email Client 2 |
| c185d679e493 | Email Client 2 |
| c406d877e950 | Email Client 1 |
| c547d140e477 | Email Client 1 |
```

Now, you need information on the operating systems used on various devices and their last patch date.

3. Complete the query to return only the `device_id`, `operating_system`, and `OS_patch_date` columns from the `machines` table. Replace `X`, `Y`, and `Z` with the columns that you need to return:

The correct query to solve this step:

```
SELECT device_id, operating_system, OS_patch_date
```

```
FROM machines;
```

```
MariaDB [organization]> SELECT device_id, operating_system, OS_patch_date
    ->
    -> FROM machines;
+--------------+------------------+---------------+
| device_id    | operating_system | OS_patch_date |
+--------------+------------------+---------------+
| a184b775c707 | OS 1             | 2021-09-01    |
| a192b174c940 | OS 2             | 2021-06-01    |
| a305b818c708 | OS 3             | 2021-06-01    |
| a317b635c465 | OS 1             | 2021-03-01    |
| a320b137c219 | OS 2             | 2021-03-01    |
| a398b471c573 | OS 3             | 2021-12-01    |
| a667b270c984 | OS 1             | 2021-03-01    |
| a821b452c176 | OS 2             | 2021-12-01    |
| a998b568c863 | OS 3             | 2021-12-01    |
| b157c491d493 | OS 2             | 2021-03-01    |
| b239c825d303 | OS 1             | 2021-03-01    |
| b264c773d977 | OS 2             | 2021-03-01    |
| b265c937d713 | OS 2             | 2021-09-01    |
| b433c245d868 | OS 1             | 2021-06-01    |
```

Task 2. Investigate login activity

In this task, you need to analyze the information from the `log_in_attempts` table to determine if any unusual activity has occurred.

First, you need to investigate the locations where login attempts were made to ensure that they're in expected areas (the United States, Canada, or Mexico).

1. Write a SQL query to select the `event_id` and `country` columns from the `log_in_attempts` table.

The correct query to solve this step:

```
SELECT event_id, country

FROM log_in_attempts;
```

```
MariaDB [organization]> SELECT event_id, country
    ->
    -> FROM log_in_attempts;
+----------+---------+
| event_id | country |
+----------+---------+
|        1 | CAN     |
|        2 | CAN     |
|        3 | USA     |
|        4 | USA     |
|        5 | CANADA  |
|        6 | MEXICO  |
|        7 | CAN     |
|        8 | US      |
|        9 | MEX     |
|       10 | CANADA  |
|       11 | CANADA  |
|       12 | USA     |
|       13 | USA     |
|       14 | US      |
|       15 | USA     |
|       16 | CAN     |
|       17 | USA     |
|       18 | US      |
|       19 | US      |
|       20 | MEXICO  |
|       21 | US      |
|       22 | MEX     |
|       23 | MEXICO  |
|       24 | MEXICO  |
|       25 | US      |
|       26 | CANADA  |
```

Next, you need to check if login attempts were made outside of the organization's working hours.

2. Write a SQL query that selects the `username`, `login_date`, and `login_time` columns from the `log_in_attempts` table.

The correct query to solve this step:

```
SELECT username, login_date, login_time

FROM log_in_attempts;
```

```
MariaDB [organization]> SELECT username, login_date, login_time
    ->
    -> FROM log_in_attempts;
+----------+------------+------------+
| username | login_date | login_time |
+----------+------------+------------+
| jrafael  | 2022-05-09 | 04:56:27   |
| apatel   | 2022-05-10 | 20:27:27   |
| dkot     | 2022-05-09 | 06:47:41   |
| dkot     | 2022-05-08 | 02:00:39   |
| jrafael  | 2022-05-11 | 03:05:59   |
| arutley  | 2022-05-12 | 17:00:59   |
| eraab    | 2022-05-11 | 01:45:14   |
| bisles   | 2022-05-08 | 01:30:17   |
| yappiah  | 2022-05-11 | 13:47:29   |
| jrafael  | 2022-05-12 | 09:33:19   |
| sgilmore | 2022-05-11 | 10:16:29   |
| dkot     | 2022-05-08 | 09:11:34   |
| mrah     | 2022-05-11 | 09:29:34   |
| sbaelish | 2022-05-10 | 10:20:18   |
| lyamamot | 2022-05-09 | 17:17:26   |
| mcouliba | 2022-05-11 | 06:44:22   |
| pwashing | 2022-05-11 | 02:33:02   |
| pwashing | 2022-05-11 | 19:28:50   |
| jhill    | 2022-05-12 | 13:09:04   |
| tshah    | 2022-05-12 | 18:56:36   |
| iuduike  | 2022-05-11 | 17:50:00   |
| rjensen  | 2022-05-11 | 00:59:26   |
| yappiah  | 2022-05-10 | 18:11:53   |
| arusso   | 2022-05-09 | 06:49:39   |
```

3. Write a SQL query that selects all columns from the `log_in_attempts` table, using a single symbol after the `SELECT` keyword.

The correct query to solve this step:

```
SELECT *

FROM log_in_attempts;
```

```
MariaDB [organization]> SELECT *
    ->
    -> FROM log_in_attempts;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       1 |
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       1 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
|        5 | jrafael  | 2022-05-11 | 03:05:59   | CANADA  | 192.168.86.232  |       0 |
|        6 | arutley  | 2022-05-12 | 17:00:59   | MEXICO  | 192.168.3.24    |       0 |
|        7 | eraab    | 2022-05-11 | 01:45:14   | CAN     | 192.168.170.243 |       1 |
|        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.119.173 |       0 |
|        9 | yappiah  | 2022-05-11 | 13:47:29   | MEX     | 192.168.59.136  |       1 |
|       10 | jrafael  | 2022-05-12 | 09:33:19   | CANADA  | 192.168.228.221 |       0 |
|       11 | sgilmore | 2022-05-11 | 10:16:29   | CANADA  | 192.168.140.81  |       0 |
|       12 | dkot     | 2022-05-08 | 09:11:34   | USA     | 192.168.100.158 |       1 |
|       13 | mrah     | 2022-05-11 | 09:29:34   | USA     | 192.168.246.135 |       1 |
|       14 | sbaelish | 2022-05-10 | 10:20:18   | US      | 192.168.16.99   |       1 |
|       15 | lyamamot | 2022-05-09 | 17:17:26   | USA     | 192.168.183.51  |       0 |
|       16 | mcouliba | 2022-05-11 | 06:44:22   | CAN     | 192.168.172.189 |       1 |
|       17 | pwashing | 2022-05-11 | 02:33:02   | USA     | 192.168.81.89   |       1 |
|       18 | pwashing | 2022-05-11 | 19:28:50   | US      | 192.168.66.142  |       0 |
|       19 | jhill    | 2022-05-12 | 13:09:04   | US      | 192.168.142.245 |       1 |
|       20 | tshah    | 2022-05-12 | 18:56:36   | MEXICO  | 192.168.109.50  |       0 |
|       21 | iuduike  | 2022-05-11 | 17:50:00   | US      | 192.168.131.147 |       1 |
|       22 | rjensen  | 2022-05-11 | 00:59:26   | MEX     | 192.168.213.128 |       0 |
|       23 | yappiah  | 2022-05-10 | 18:11:53   | MEXICO  | 192.168.200.48  |       1 |
|       24 | arusso   | 2022-05-09 | 06:49:39   | MEXICO  | 192.168.171.192 |       1 |
```

Task 3. Order login attempts data

In this task, you need to use the `ORDER BY` keyword. You'll sequence the data that your query returns according to the login date and time.

First, you need to sort the information by date.

1. Run the following query, which orders `log_in_attempts` data by `login_date`:

```
SELECT *

FROM log_in_attempts
```

```
ORDER BY login_date;
```

```
MariaDB [organization]> SELECT *
    ->
    -> FROM log_in_attempts
    ->
    -> ORDER BY login_date;
+----------+----------+------------+------------+---------+----------------+---------+
| event_id | username | login_date | login_time | country | ip_address     | success |
+----------+----------+------------+------------+---------+----------------+---------+
|      145 | ivelasco | 2022-05-08 | 09:06:02   | CANADA  | 192.168.39.196 |       1 |
|      163 | tmitchel | 2022-05-08 | 09:21:16   | MEX     | 192.168.119.29 |       0 |
|       36 | asundara | 2022-05-08 | 09:00:42   | US      | 192.168.78.151 |       1 |
|      165 | jreckley | 2022-05-08 | 15:28:43   | MEXICO  | 192.168.34.193 |       0 |
|      168 | jlansky  | 2022-05-08 | 13:25:42   | USA     | 192.168.210.94 |       1 |
|      169 | alevitsk | 2022-05-08 | 08:10:43   | CANADA  | 192.168.210.228 |       0 |
|       72 | alevitsk | 2022-05-08 | 12:09:10   | CANADA  | 192.168.139.176 |       1 |
|      101 | sbaelish | 2022-05-08 | 12:01:22   | US      | 192.168.145.158 |       0 |
|      172 | mabadi   | 2022-05-08 | 08:06:50   | US      | 192.168.180.41  |       1 |
|      150 | nmason   | 2022-05-08 | 14:40:02   | CAN     | 192.168.204.124 |       0 |
|       68 | mrah     | 2022-05-08 | 17:16:13   | US      | 192.168.42.248  |       1 |
|       66 | aestrada | 2022-05-08 | 21:58:32   | MEX     | 192.168.67.223  |       1 |
|       53 | nmason   | 2022-05-08 | 11:51:38   | CAN     | 192.168.133.188 |       1 |
|      147 | yappiah  | 2022-05-08 | 06:04:34   | MEX     | 192.168.65.245  |       0 |
|      148 | daquino  | 2022-05-08 | 06:15:55   | CANADA  | 192.168.135.6   |       1 |
|       49 | asundara | 2022-05-08 | 14:00:01   | US      | 192.168.173.213 |       0 |
|       47 | dkot     | 2022-05-08 | 05:06:45   | US      | 192.168.233.24  |       1 |
|       44 | daquino  | 2022-05-08 | 07:02:35   | CANADA  | 192.168.168.144 |       0 |
|       43 | mcouliba | 2022-05-08 | 02:35:34   | CANADA  | 192.168.16.208  |       0 |
|       56 | acook    | 2022-05-08 | 04:56:30   | CAN     | 192.168.209.130 |       1 |
|       80 | cjackson | 2022-05-08 | 02:18:10   | CANADA  | 192.168.33.140  |       1 |
|      117 | bsand    | 2022-05-08 | 00:19:11   | USA     | 192.168.197.187 |       0 |
```

Now, you need to further organize the previous results by ordering them by `login_time`.

2. Modify the query from the previous step by adding the login time to the `ORDER BY` clause. You must replace `X` with the appropriate column name:

The correct query to solve this step:

```
SELECT *

FROM log_in_attempts
```

```
ORDER BY login_date, login_time;
```

```
MariaDB [organization]> SELECT *
    ->
    -> FROM log_in_attempts
    ->
    -> ORDER BY login_date, login_time;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|      117 | bsand    | 2022-05-08 | 00:19:11   | USA     | 192.168.197.187 |       0 |
|       92 | pwashing | 2022-05-08 | 00:36:12   | US      | 192.168.247.219 |       0 |
|        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.119.173 |       0 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
|       80 | cjackson | 2022-05-08 | 02:18:10   | CANADA  | 192.168.33.140  |       1 |
|       43 | mcouliba | 2022-05-08 | 02:35:34   | CANADA  | 192.168.16.208  |       0 |
|      184 | alevitsk | 2022-05-08 | 03:09:48   | CAN     | 192.168.33.70   |       0 |
|       56 | acook    | 2022-05-08 | 04:56:30   | CAN     | 192.168.209.130 |       1 |
|       47 | dkot     | 2022-05-08 | 05:06:45   | US      | 192.168.233.24  |       1 |
|      189 | nmason   | 2022-05-08 | 05:37:24   | CANADA  | 192.168.168.117 |       1 |
|      147 | yappiah  | 2022-05-08 | 06:04:34   | MEX     | 192.168.65.245  |       0 |
|      148 | daquino  | 2022-05-08 | 06:15:55   | CANADA  | 192.168.135.6   |       1 |
|      191 | cjackson | 2022-05-08 | 06:46:07   | CANADA  | 192.168.7.187   |       0 |
|       44 | daquino  | 2022-05-08 | 07:02:35   | CANADA  | 192.168.168.144 |       0 |
|      193 | lrodriqu | 2022-05-08 | 07:11:29   | US      | 192.168.125.240 |       0 |
|      172 | mabadi   | 2022-05-08 | 08:06:50   | US      | 192.168.180.41  |       1 |
|       83 | lrodriqu | 2022-05-08 | 08:10:23   | USA     | 192.168.67.69   |       1 |
|      169 | alevitsk | 2022-05-08 | 08:10:43   | CANADA  | 192.168.210.228 |       0 |
|       36 | asundara | 2022-05-08 | 09:00:42   | US      | 192.168.78.151  |       1 |
|      197 | jsoto    | 2022-05-08 | 09:05:09   | US      | 192.168.36.21   |       0 |
|      145 | ivelasco | 2022-05-08 | 09:06:02   | CANADA  | 192.168.39.196  |       1 |
|       12 | dkot     | 2022-05-08 | 09:11:34   | USA     | 192.168.100.158 |       1 |
|      163 | tmitchel | 2022-05-08 | 09:21:16   | MEX     | 192.168.119.29  |       0 |
```