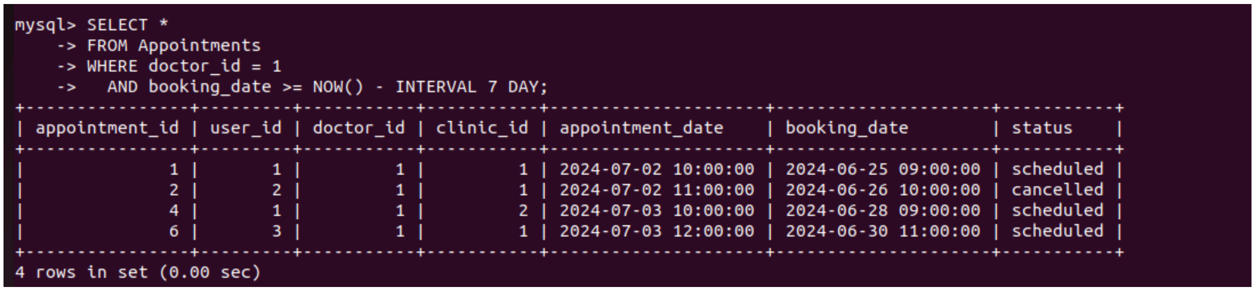
1. All appointments booked in the last 7 days for a doctor.

```
SELECT *
FROM Appointments
WHERE doctor_id = 1
AND booking_date >= NOW() - INTERVAL 7 DAY;
```

Explanation:

We want to retrieve all appointments that were booked in the last 7 days for a specific doctor. **doctor_id = 1:** This condition filters the results to only include appointments for the doctor with ID = 1.**booking_date >= NOW() - INTERVAL 7 DAY:** This condition ensures that only appointments booked within the last 7 days are included in the result. **NOW()** returns the current date and time, and **INTERVAL 7 DAY** subtracts 7 days from the current date and time.



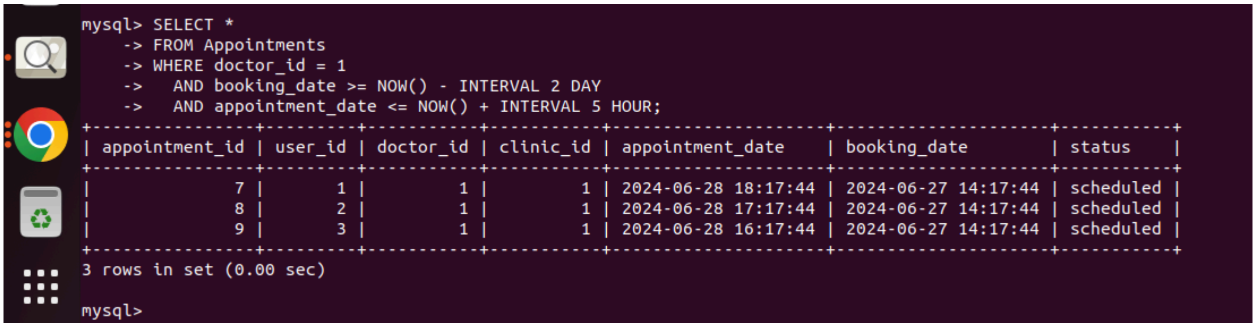
```
mysql> SELECT *
    -> FROM Appointments
    -> WHERE doctor_id = 1
    ->   AND booking_date >= NOW() - INTERVAL 7 DAY;
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
| appointment_id | user_id | doctor_id | clinic_id | appointment_date    | booking_date        | status    |
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
|              1 |       1 |         1 |         1 | 2024-07-02 10:00:00 | 2024-06-25 09:00:00 | scheduled |
|              2 |       2 |         1 |         1 | 2024-07-02 11:00:00 | 2024-06-26 10:00:00 | cancelled |
|              4 |       1 |         1 |         2 | 2024-07-03 10:00:00 | 2024-06-28 09:00:00 | scheduled |
|              6 |       3 |         1 |         1 | 2024-07-03 12:00:00 | 2024-06-30 11:00:00 | scheduled |
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
4 rows in set (0.00 sec)
```

2. All appointments booked in the last 2 days and scheduled within the next 5 hours for a doctor

```
SELECT *
FROM Appointments
WHERE doctor_id = 1
  AND booking_date >= NOW() - INTERVAL 2 DAY
  AND appointment_date <= NOW() + INTERVAL 5 HOUR;
```

Explanation:

We Want to retrieve all appointments that were booked in the last 2 days and are scheduled within the next 5 hours for a specific doctor.I am considering **doctor_id = 1** as the specific id for a doctor this condition filters the results to only include appointments for the doctor with ID = 1.**booking_date >= NOW() - INTERVAL 2 DAY:** This condition ensures that only appointments booked within the last 2 days are included.**appointment_date <= NOW() + INTERVAL 5 HOUR:** This condition ensures that only appointments scheduled to occur within the next 5 hours are included.



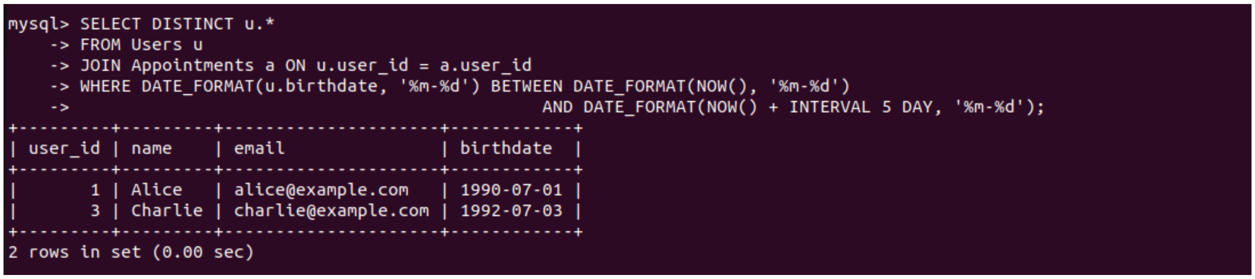
```
mysql> SELECT *
    -> FROM Appointments
    -> WHERE doctor_id = 1
    ->   AND booking_date >= NOW() - INTERVAL 2 DAY
    ->   AND appointment_date <= NOW() + INTERVAL 5 HOUR;
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
| appointment_id | user_id | doctor_id | clinic_id | appointment_date    | booking_date        | status    |
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
|              7 |       1 |         1 |         1 | 2024-06-28 18:17:44 | 2024-06-27 14:17:44 | scheduled |
|              8 |       2 |         1 |         1 | 2024-06-28 17:17:44 | 2024-06-27 14:17:44 | scheduled |
|              9 |       3 |         1 |         1 | 2024-06-28 16:17:44 | 2024-06-27 14:17:44 | scheduled |
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
3 rows in set (0.00 sec)

mysql>
```

3. Users who have at least 1 appointment and have their birthday coming in the next 5 days

```
SELECT DISTINCT u.*
FROM Users u
JOIN Appointments a ON u.user_id = a.user_id
WHERE DATE_FORMAT(u.birthdate, '%m-%d') BETWEEN DATE_FORMAT(NOW(), '%m-%d')
                              AND DATE_FORMAT(NOW() + INTERVAL 5 DAY, '%m-%d');
```

Explanation:

Retrieving all the users who have at least one appointment and whose birthdays are within the next 5 days.**JOIN Appointments a ON u.user_id = a.user_id:** This join ensures that only users with at least one appointment are considered.**DATE_FORMAT(u.birthdate, '%m-%d'):** This formats the user's birthdate to only consider the month and day, ignoring the year.**BETWEEN DATE_FORMAT(NOW(), '%m-%d') AND DATE_FORMAT(NOW() + INTERVAL 5 DAY, '%m-%d'):** This condition ensures that the user's birthdate falls within the next 5 days from the current date.



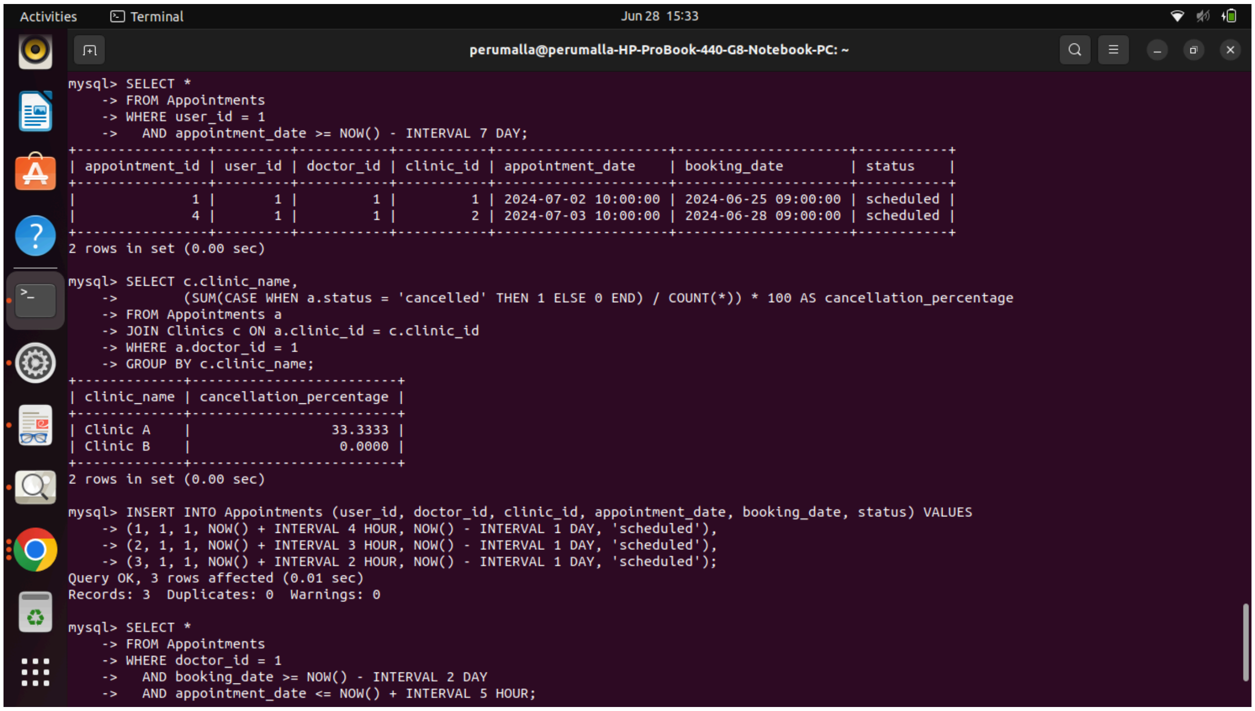
```
mysql> SELECT DISTINCT u.*
    -> FROM Users u
    -> JOIN Appointments a ON u.user_id = a.user_id
    -> WHERE DATE_FORMAT(u.birthdate, '%m-%d') BETWEEN DATE_FORMAT(NOW(), '%m-%d')
    ->                                              AND DATE_FORMAT(NOW() + INTERVAL 5 DAY, '%m-%d');
+---------+---------+---------------------+------------+
| user_id | name    | email               | birthdate  |
+---------+---------+---------------------+------------+
|       1 | Alice   | alice@example.com   | 1990-07-01 |
|       3 | Charlie | charlie@example.com | 1992-07-03 |
+---------+---------+---------------------+------------+
2 rows in set (0.00 sec)
```

4. Appointments for a particular patient in the last 7 days

```
SELECT *
FROM Appointments
WHERE user_id = 1
  AND appointment_date >= NOW() - INTERVAL 7 DAY;
```

Explanation:

Retrieving all the appointments for a specific user that were scheduled in the last 7 days.**user_id = 1:** This condition filters the results to only include appointments for the user with ID 1.**appointment_date >= NOW() - INTERVAL 7 DAY:** This condition ensures that only appointments scheduled within the last 7 days are included.



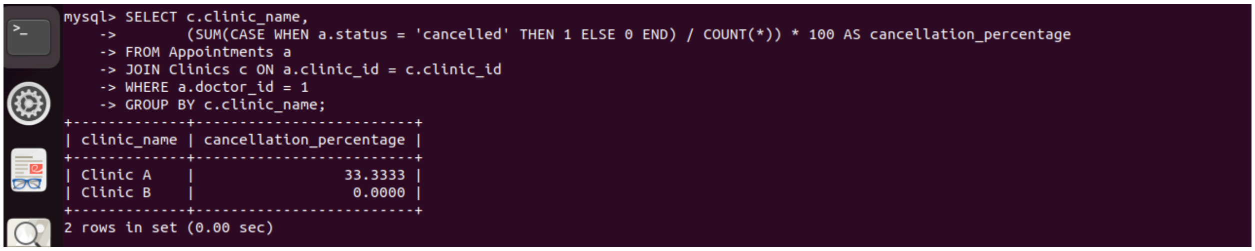
```
mysql> SELECT *
    -> FROM Appointments
    -> WHERE user_id = 1
    ->   AND appointment_date >= NOW() - INTERVAL 7 DAY;
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
| appointment_id | user_id | doctor_id | clinic_id | appointment_date    | booking_date        | status    |
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
|              1 |       1 |         1 |         1 | 2024-07-02 10:00:00 | 2024-06-25 09:00:00 | scheduled |
|              4 |       1 |         1 |         2 | 2024-07-03 10:00:00 | 2024-06-28 09:00:00 | scheduled |
+----------------+---------+-----------+-----------+---------------------+---------------------+-----------+
2 rows in set (0.00 sec)

mysql> SELECT c.clinic_name,
    ->        (SUM(CASE WHEN a.status = 'cancelled' THEN 1 ELSE 0 END) / COUNT(*)) * 100 AS cancellation_percentage
    -> FROM Appointments a
    -> JOIN Clinics c ON a.clinic_id = c.clinic_id
    -> WHERE a.doctor_id = 1
    -> GROUP BY c.clinic_name;
+-------------+-------------------------+
| clinic_name | cancellation_percentage |
+-------------+-------------------------+
| Clinic A    |                 33.3333 |
| Clinic B    |                  0.0000 |
+-------------+-------------------------+
2 rows in set (0.00 sec)

mysql> INSERT INTO Appointments (user_id, doctor_id, clinic_id, appointment_date, booking_date, status) VALUES
    -> (1, 1, 1, NOW() + INTERVAL 4 HOUR, NOW() - INTERVAL 1 DAY, 'scheduled'),
    -> (2, 1, 1, NOW() + INTERVAL 3 HOUR, NOW() - INTERVAL 1 DAY, 'scheduled'),
    -> (3, 1, 1, NOW() + INTERVAL 2 HOUR, NOW() - INTERVAL 1 DAY, 'scheduled');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> SELECT *
    -> FROM Appointments
    -> WHERE doctor_id = 1
    ->   AND booking_date >= NOW() - INTERVAL 2 DAY
    ->   AND appointment_date <= NOW() + INTERVAL 5 HOUR;
```

5. Appointment cancellation percentage for a doctor by clinic

```
SELECT c.clinic_name,
       (SUM(CASE WHEN a.status = 'cancelled' THEN 1 ELSE 0 END) /
COUNT(*)) * 100 AS cancellation_percentage
FROM Appointments a
JOIN Clinics c ON a.clinic_id = c.clinic_id
WHERE a.doctor_id = 1
GROUP BY c.clinic_name;
```

Explanation:

Calculating the percentage of cancelled appointments for a specific doctor, grouped by clinic.**JOIN Clinics c ON a.clinic_id = c.clinic_id:** This join ensures that each appointment is associated with the correct clinic.**WHERE a.doctor_id = 1:** This condition filters the results to only include appointments for the doctor with ID 1.**GROUP BY c.clinic_name:** This groups the results by clinic name, so the cancellation percentage is calculated separately for each clinic.**SUM(CASE WHEN a.status = 'cancelled' THEN 1 ELSE 0 END) / COUNT(*) * 100:** This calculates the cancellation percentage. The **CASE** Statement counts the number of cancelled appointments, and **COUNT(*)** gives the total number of appointments. Dividing the number of cancellations by the total number of appointments and multiplying by 100 gives the cancellation percentage.



```
mysql> SELECT c.clinic_name,
    ->        (SUM(CASE WHEN a.status = 'cancelled' THEN 1 ELSE 0 END) / COUNT(*)) * 100 AS cancellation_percentage
    -> FROM Appointments a
    -> JOIN Clinics c ON a.clinic_id = c.clinic_id
    -> WHERE a.doctor_id = 1
    -> GROUP BY c.clinic_name;
+-------------+-------------------------+
| clinic_name | cancellation_percentage |
+-------------+-------------------------+
| Clinic A    |                 33.3333 |
| Clinic B    |                  0.0000 |
+-------------+-------------------------+
2 rows in set (0.00 sec)
```