# Exercise: Create SQL queries to check available bookings based on user input

## **Scenario**

Little Lemon’s data model must include a Bookings table so that they can store data for table bookings. They also need a stored procedure that checks available bookings based on user input, and a MySQL transaction that can be used to cancel bookings. Use your knowledge of MySQL to help them out.

## **Prerequisites**

You should have created the Little Lemon database in an earlier module. This database should contain a basic Bookings table linked to a Customers table as illustrated below. Your tables can differ slightly from those provided in the example once the required relationship exists between the two.

You also require access to MySQL Workbench SQL editor to write the required queries.

## **Task 1**

Little Lemon wants to populate the Bookings table of their database with some records of data. Your first task is to replicate the list of records in the following table by adding them to the Little Lemon booking table.

You can use simple INSERT statements to complete this task.

| **BookingID** | **BookingDate** | **TableNumber** | **CustomerID** |
| --- | --- | --- | --- |
| 1 | 2022-10-10 | 5 | 1 |
| 2 | 2022-11-12 | 3 | 3 |
| 3 | 2022-10-11 | 2 | 2 |
| 4 | 2022-10-13 | 2 | 1 |

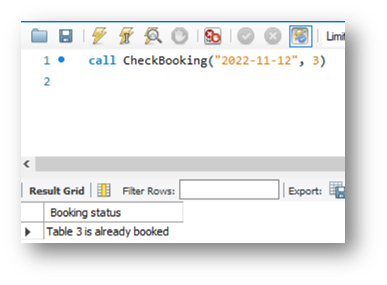
Your output must resemble the following screenshot:

## **Task 2**

For your second task, Little Lemon need you to create a stored procedure called CheckBooking to check whether a table in the restaurant is already booked. Creating this procedure helps to minimize the effort involved in repeatedly coding the same SQL statements.

The procedure should have two input parameters in the form of booking date and table number. You can also create a variable in the procedure to check the status of each table.

The output of your procedure should be similar to the following screenshot:



## **Task 3**

For your third and final task, Little Lemon need to verify a booking, and decline any reservations for tables that are already booked under another name.

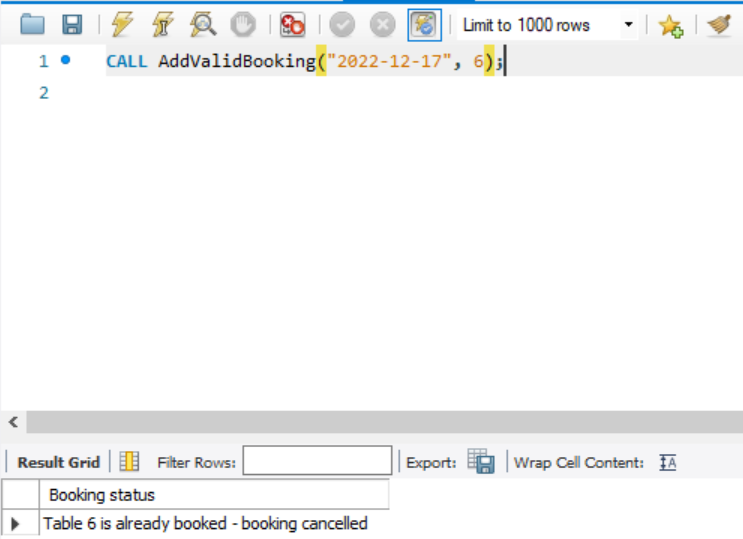
Since integrity is not optional, Little Lemon need to ensure that every booking attempt includes these verification and decline steps. However, implementing these steps requires a stored procedure and a transaction.

To implement these steps, you need to create a new procedure called AddValidBooking. This procedure must use a transaction statement to perform a rollback if a customer reserves a table that’s already booked under another name.

Use the following guidelines to complete this task:

* The procedure should include two input parameters in the form of booking date and table number.
* It also requires at least one variable and should begin with a START TRANSACTION statement.
* Your INSERT statement must add a new booking record using the input parameter's values.
* Use an IF ELSE statement to check if a table is already booked on the given date.
* If the table is already booked, then rollback the transaction. If the table is available, then commit the transaction.

The screenshot below is an example of a rollback (cancelled booking), which was enacted because table number 5 is already booked on the specified date.



## **Conclusion**

Little Lemon’s customers can now check available bookings based on user input. And you can create SQL queries that let users check records based on their input.