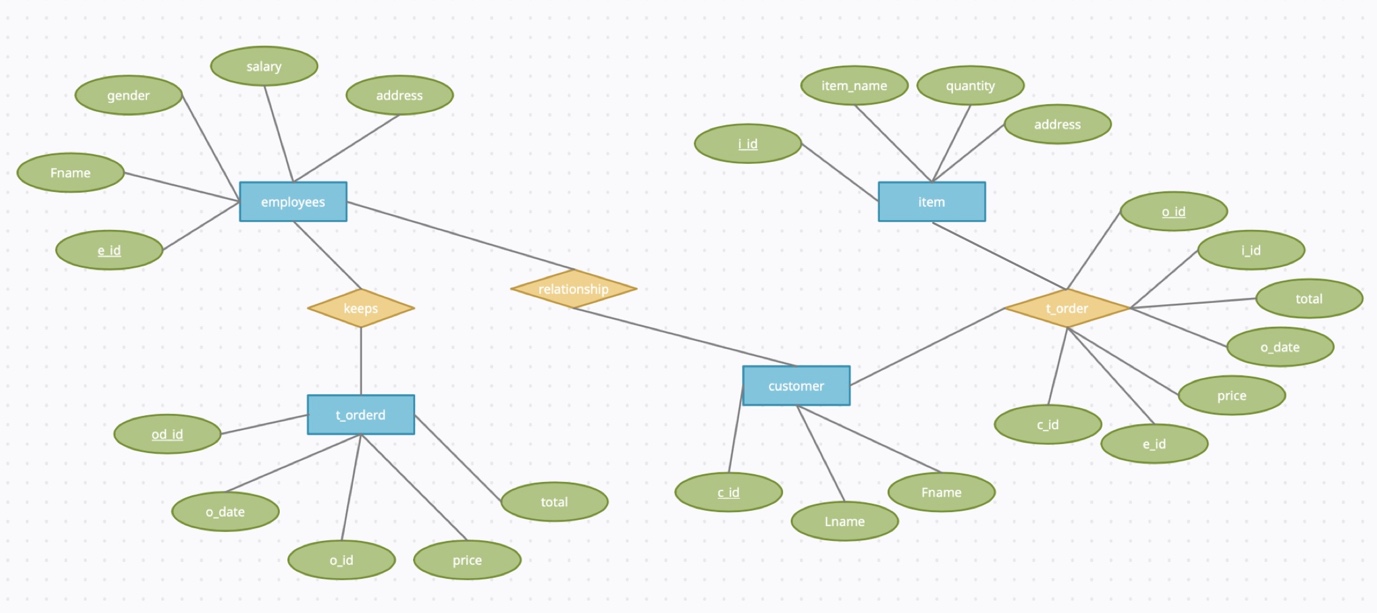
Part 1

1. Suggest a situation where you can use a database to manage and record daily transactions.

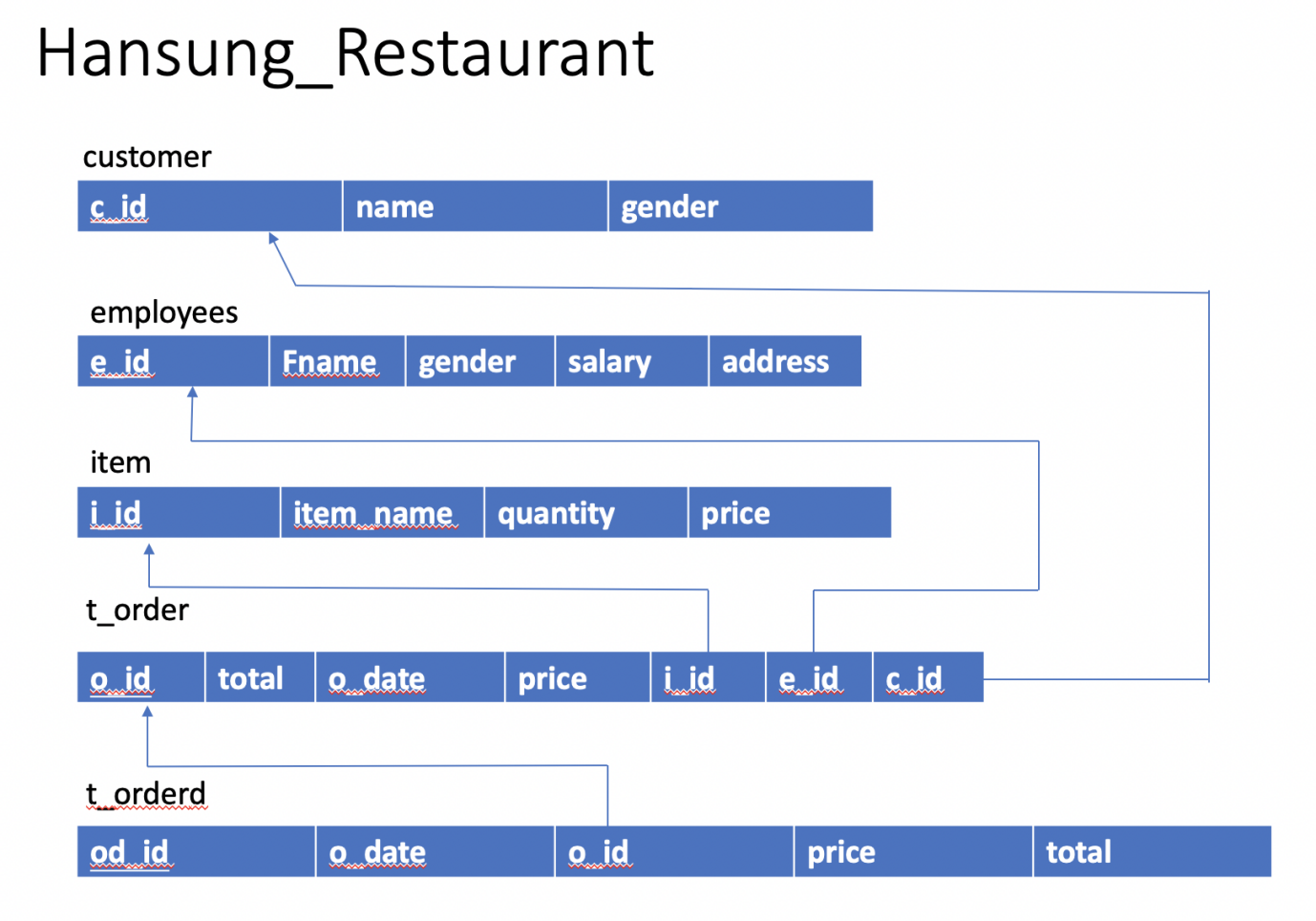
A restaurant consists of number of customers, employees and orders. Each customer offers several orders. A number of order details make up each order. Each order contains order ID, order date, the items price and the total price. The employees in the restaurant is an important part of preparing the item of the restaurant, the employees make customer orders according to order details.

However, the employees may need to make changes to item details based on customer needs. This can help the restaurant to get a higher ratings and increase the exposure of the restaurant on the delivery app, which can lead to more orders. Each order detail is very similar to the constituent elements of order, but is different. Orders can be generated on a weekly, monthly or yearly basis for each order detail. At the same time, the order details can only show the order details of each customer, if the restaurant need to count the monthly orders or annual orders, we need the orders as the total data support. The data for each order detail can help calculate the most and least sold items for each month, as well as the difference in sales for different time periods.

1. ER diagram

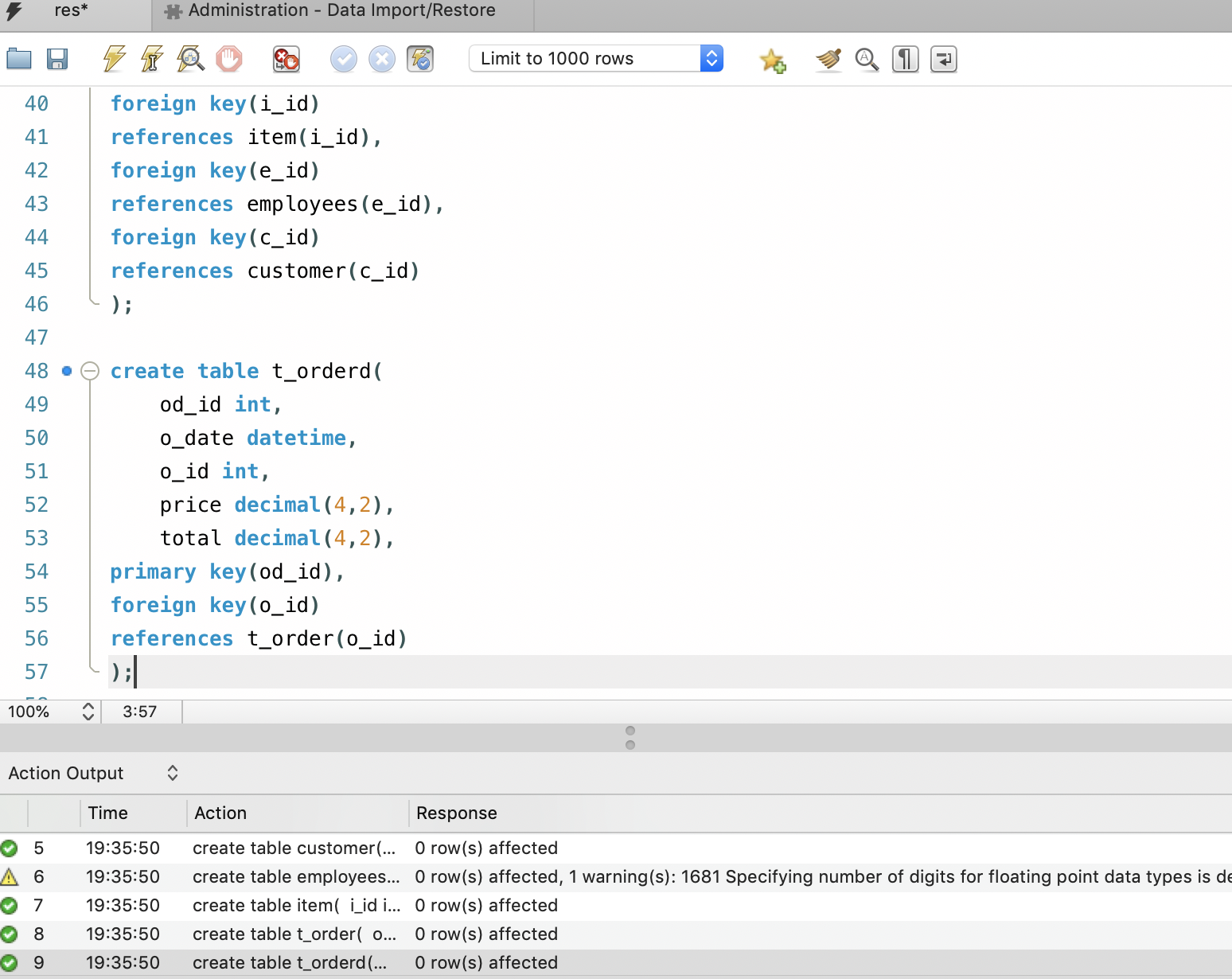


1. Conceptual design into a relational model

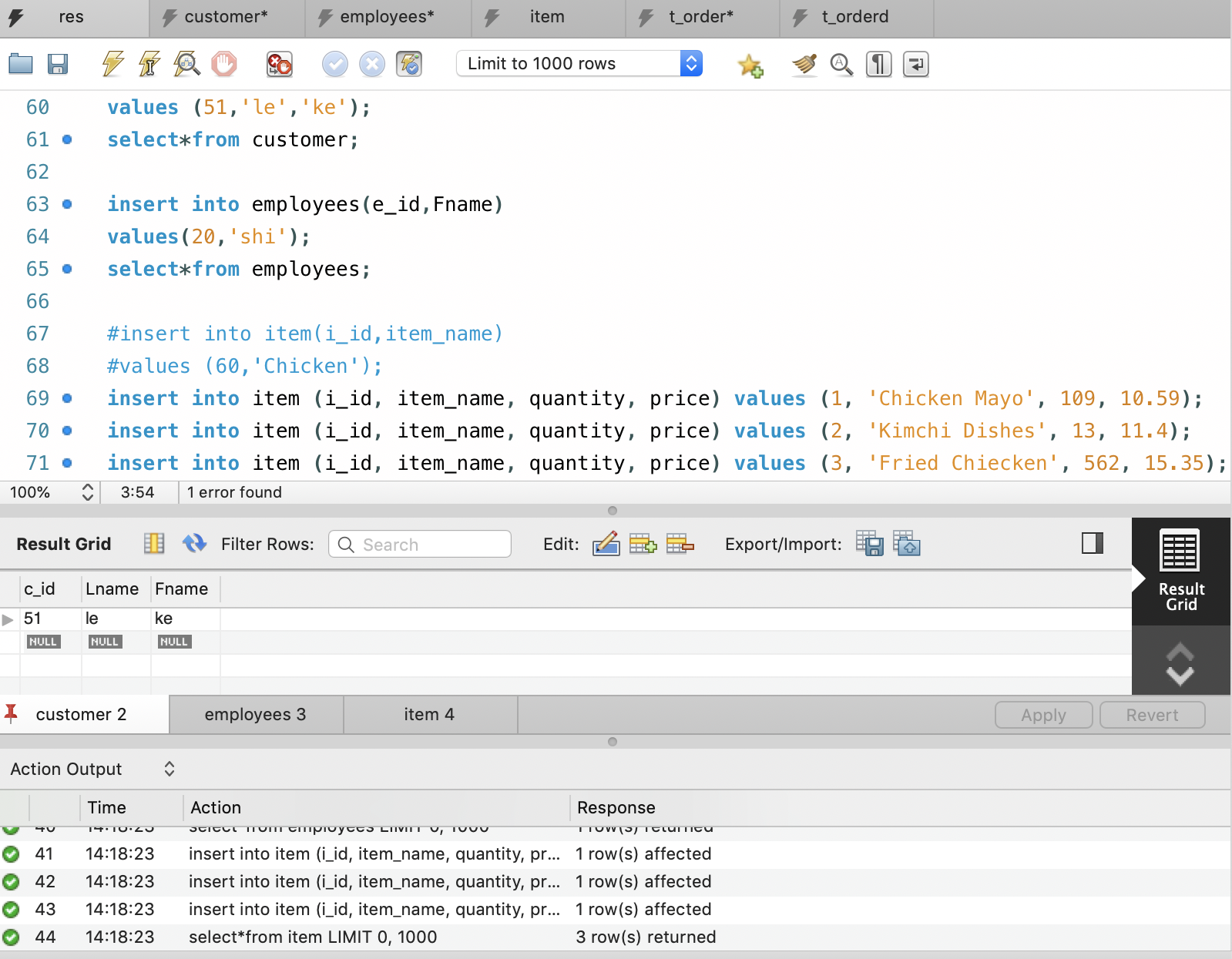


Part 2

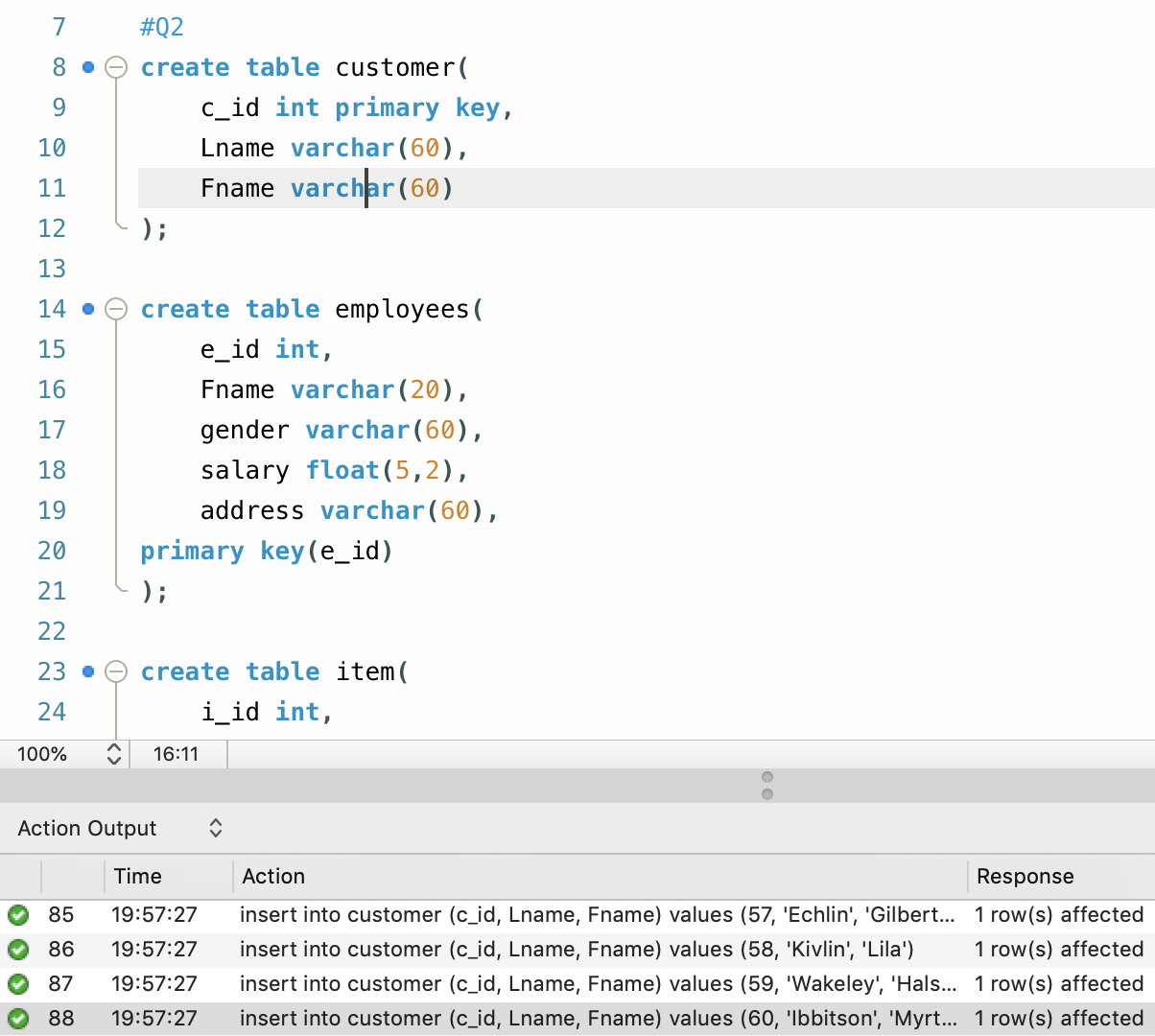
1. Create the corresponding database using DDL and
2. Create all the necessary tables identified above using DDL



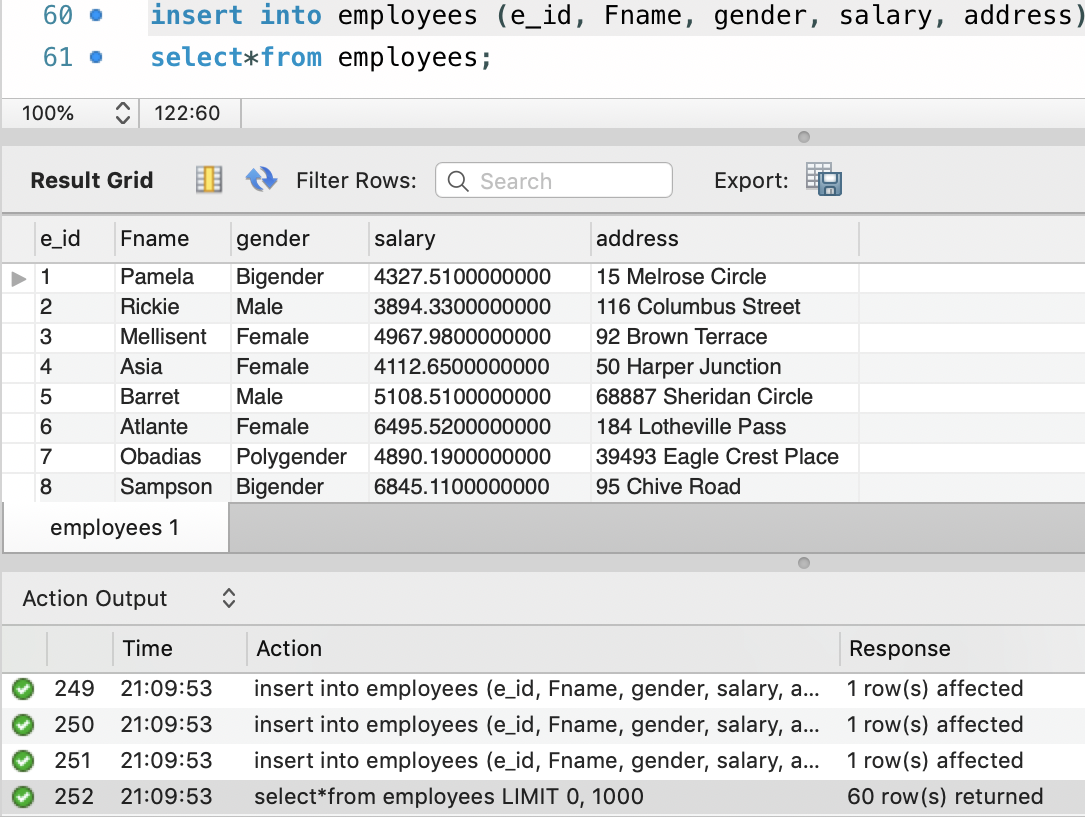
1. Populate at least three of your tables with some data using DML



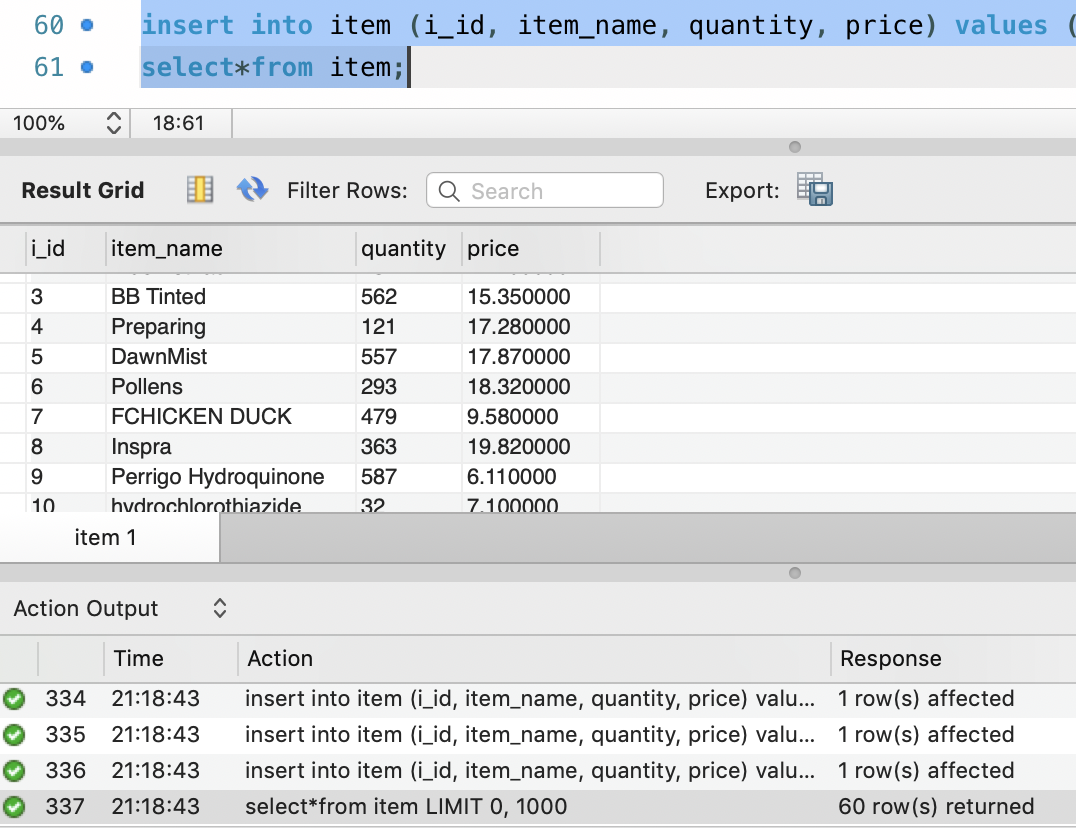
4.

1).

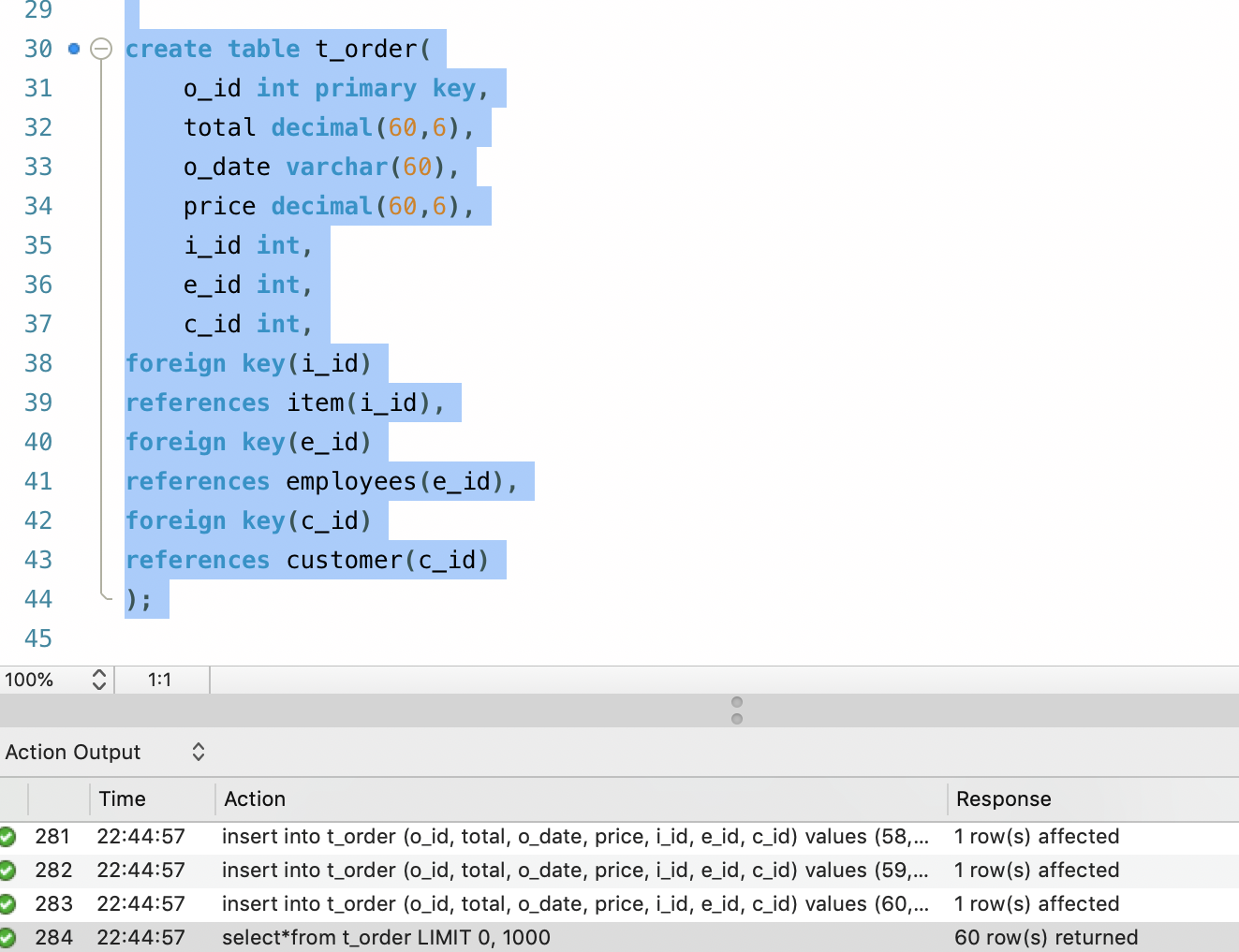
2).



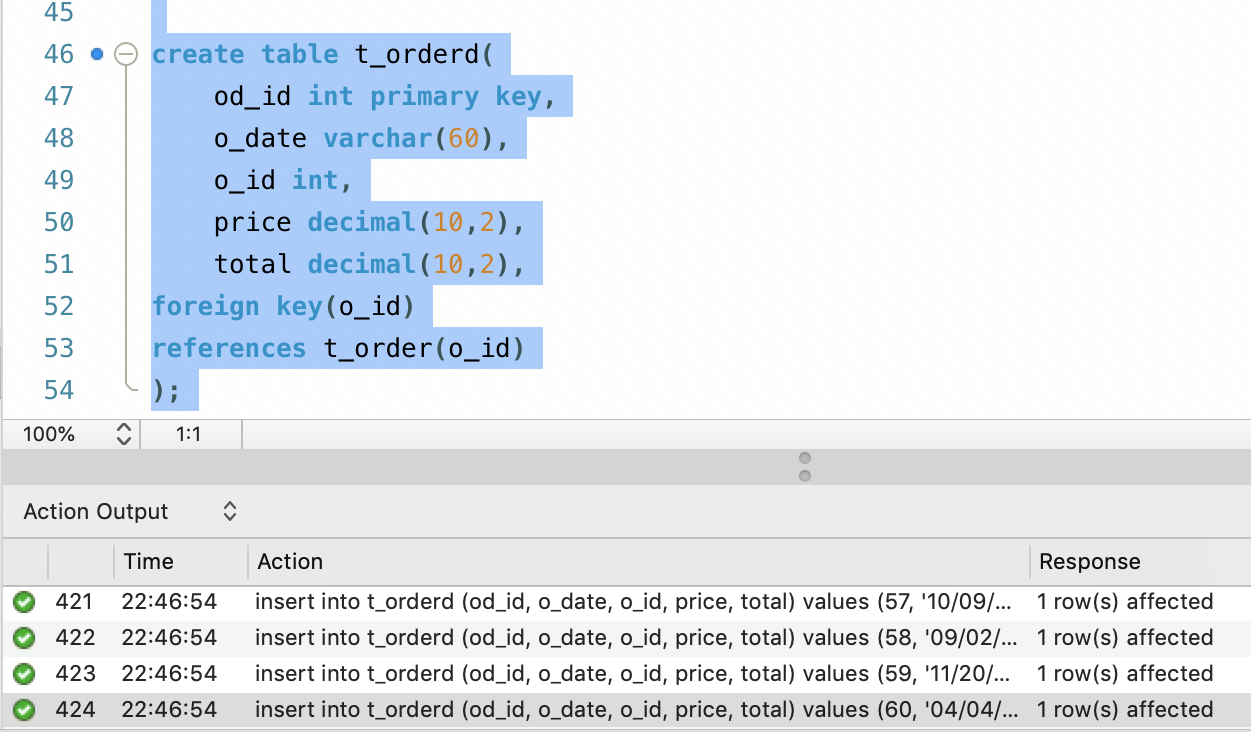
3).



4).

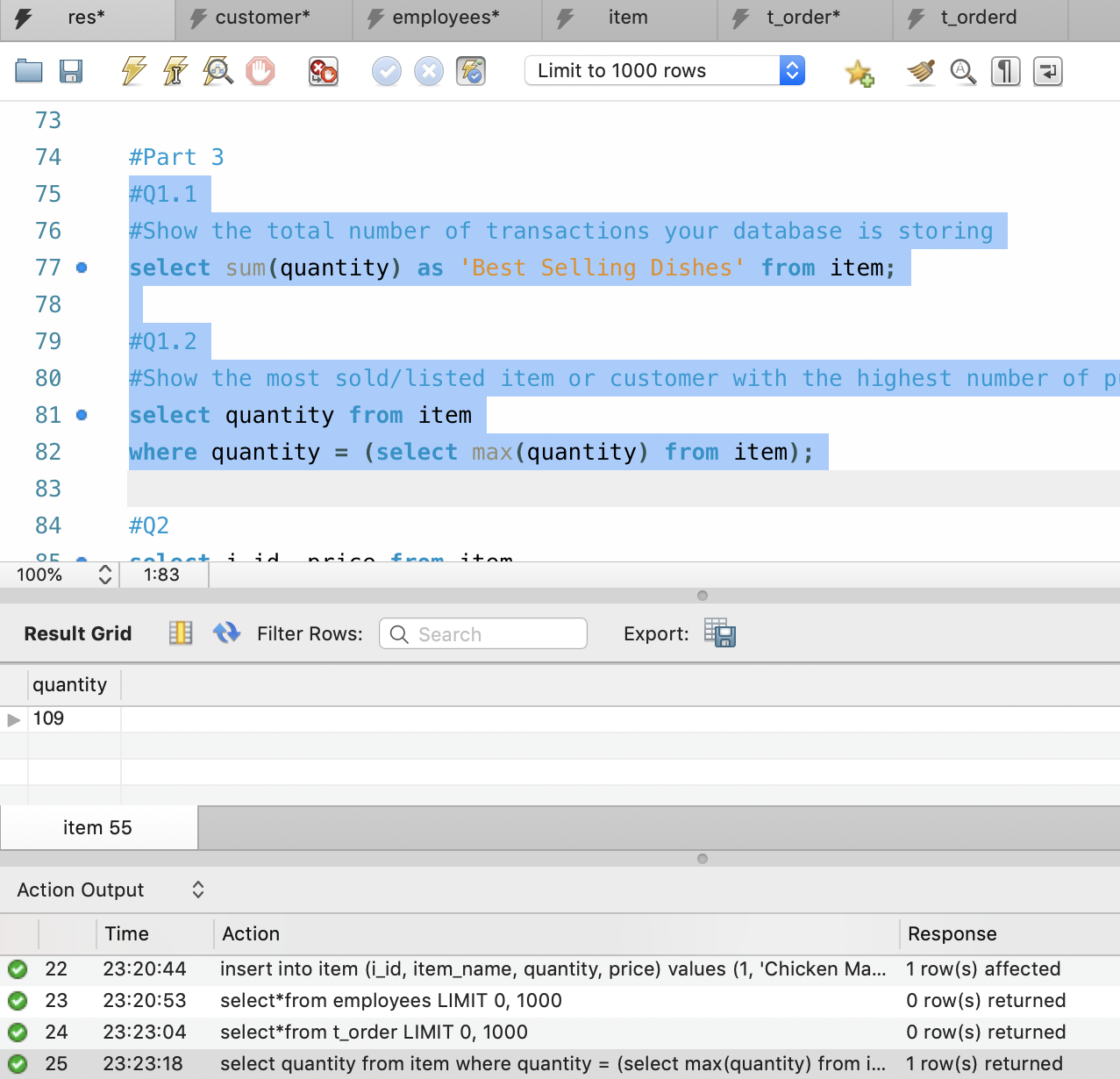


5).

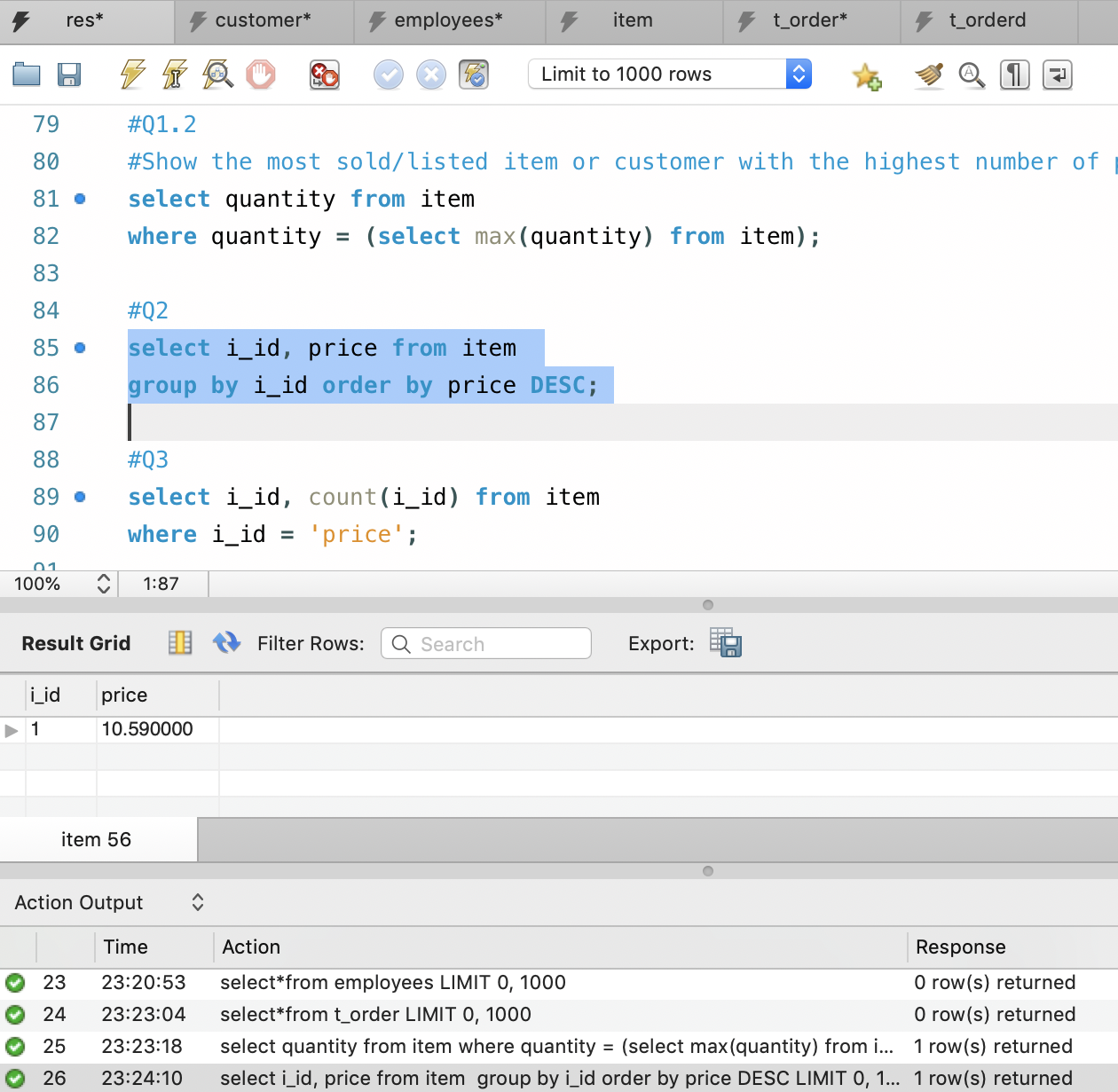


Part 3

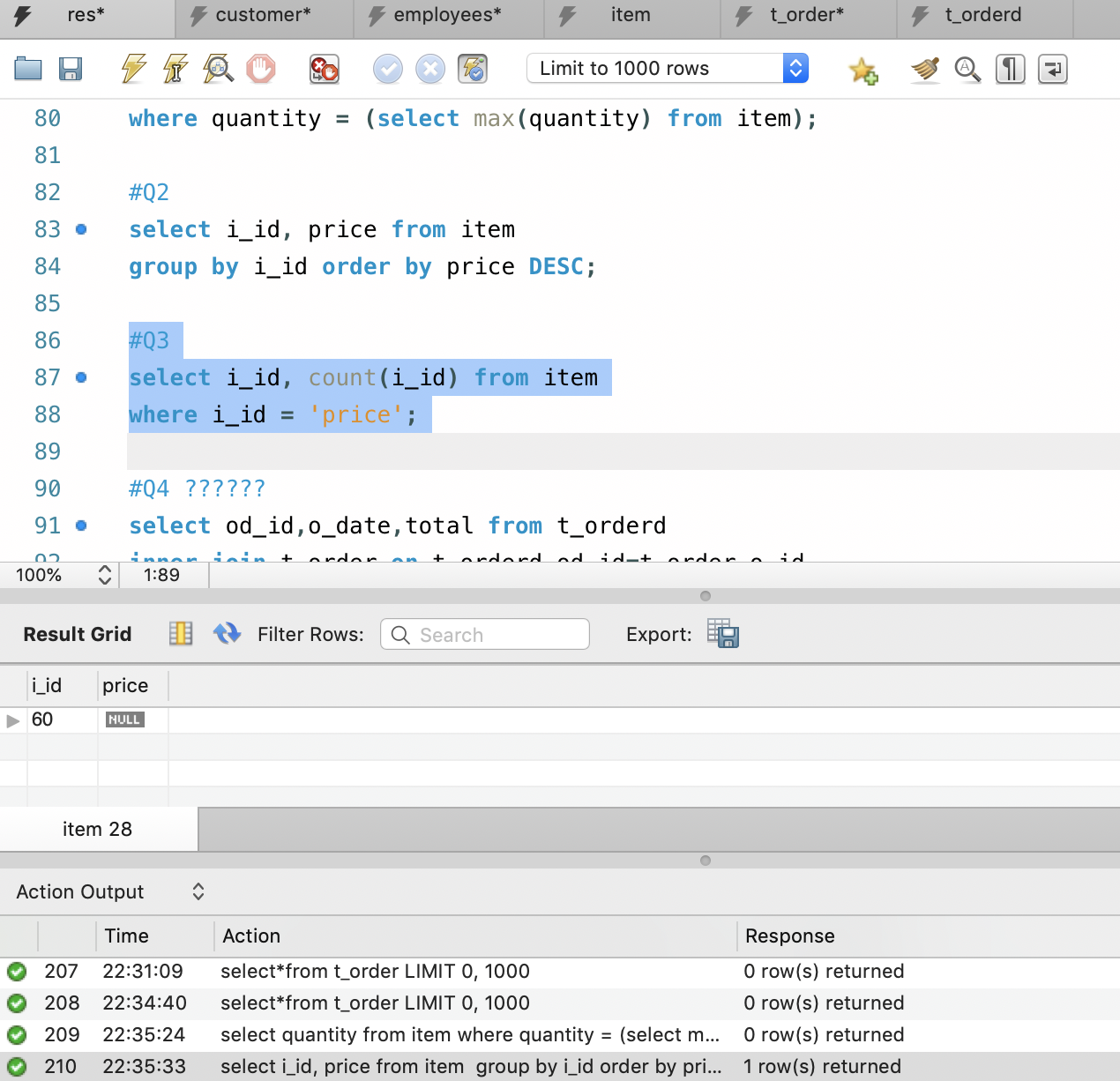
1. Show the total number of transactions your database is storing and, depending on your database, the most sold/listed item or customer with the highest number of purchases.



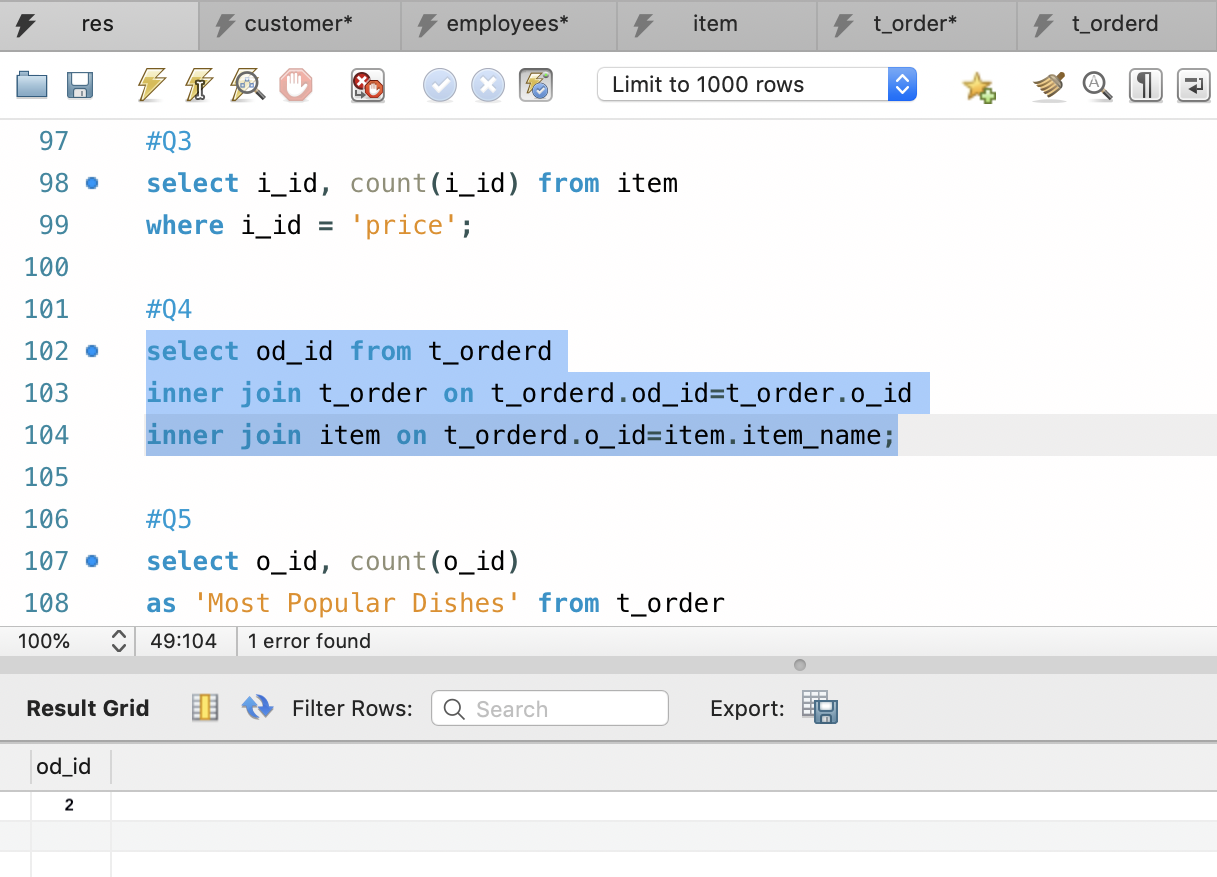
1. Write a query statement that includes “Order by” and “Group by”.



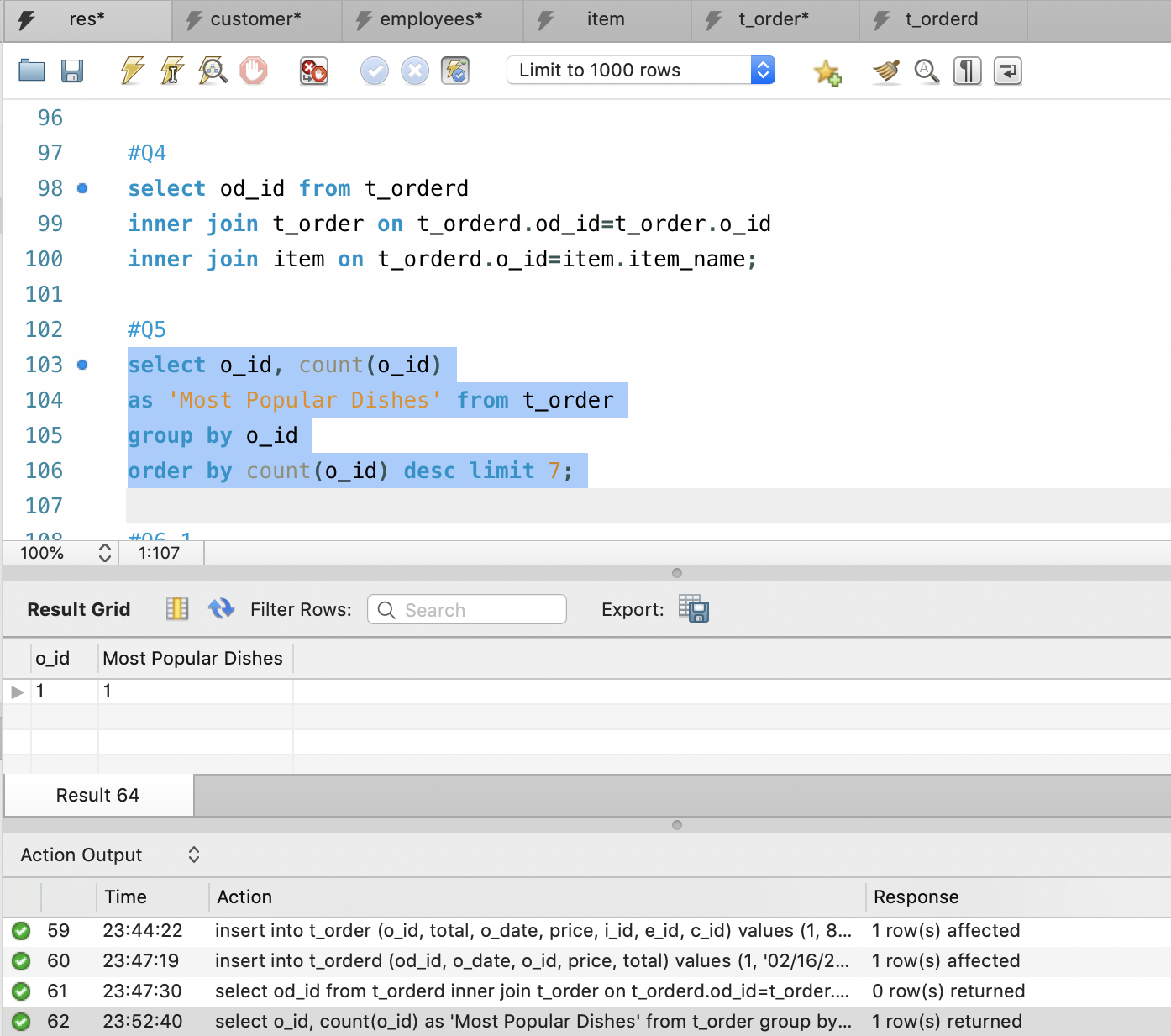
1. Write a query statement that uses pattern matching



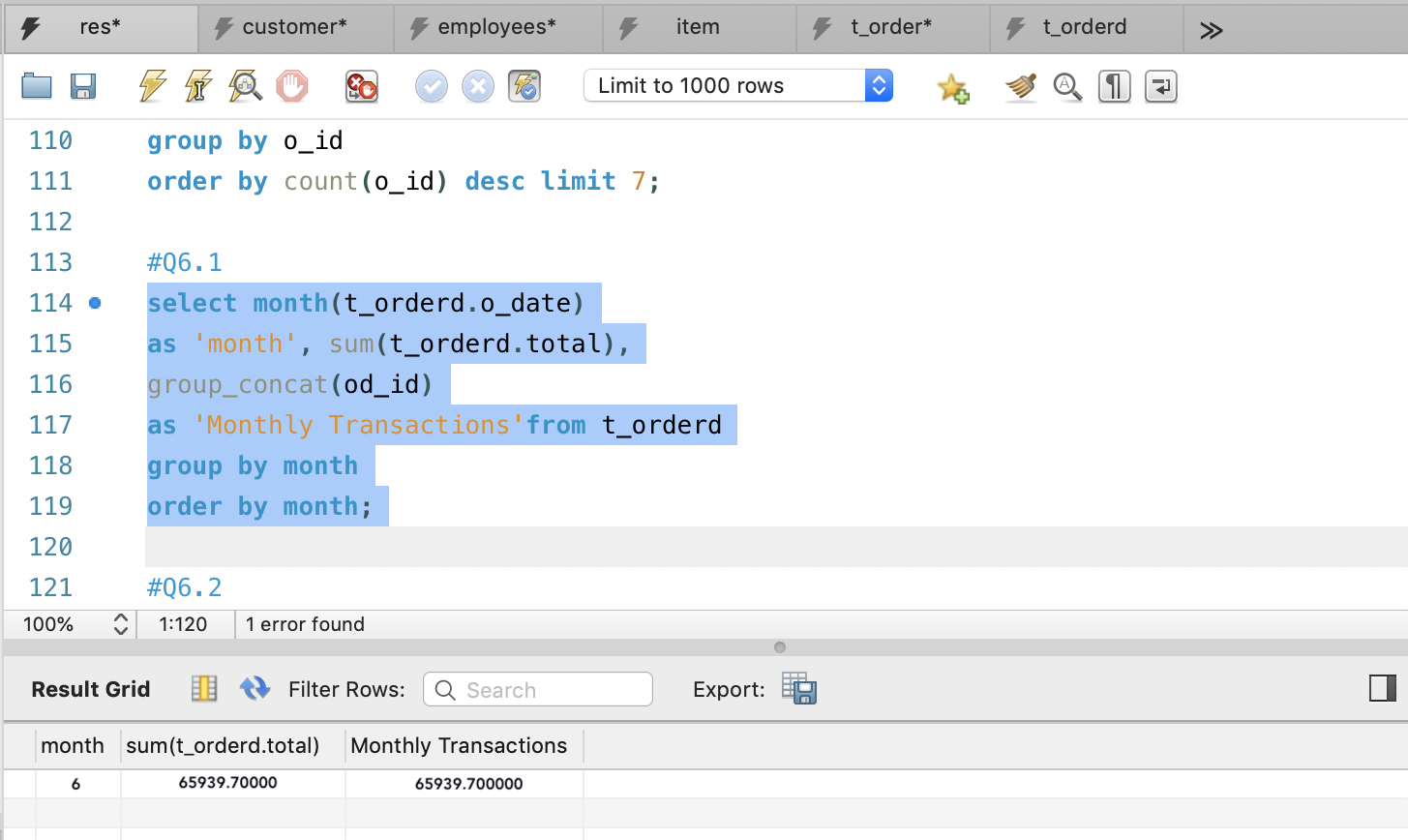
1. Show information from three tables based on criteria of your choice



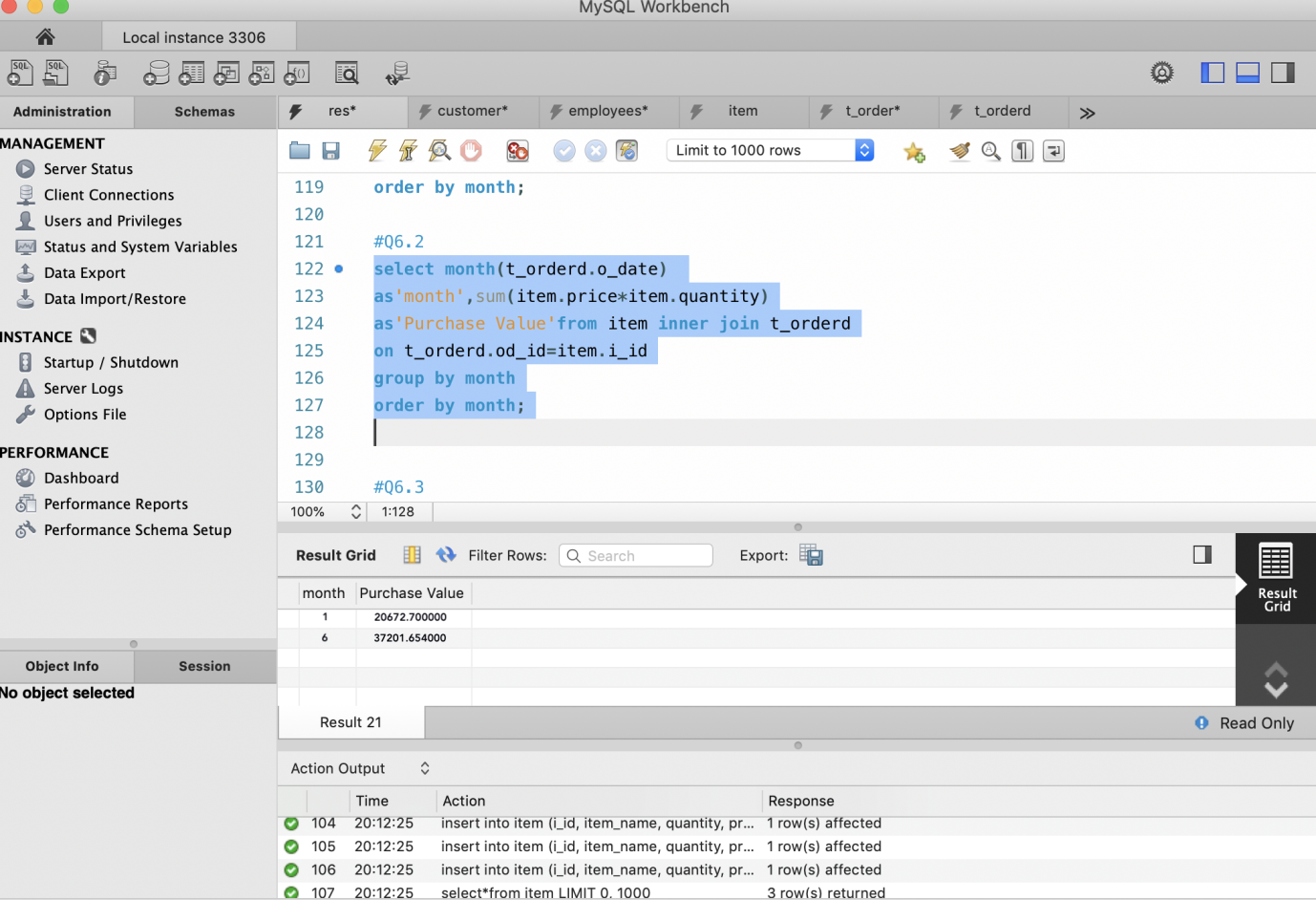
1. Create a view that includes information from the most frequent seven transactions



1. 1). Shows the total number of transactions with corresponding details every month



2). Shows customer purchase value per month



3). Shows name of product and number sold each month

