**Advanced Database System Lab**

**Batch: T5**

**Assignment No. 6**

Sumit Narake

2020BTECS00023

**To design and implement a data warehouse for a customer order processing system in a company. [ Use any Database ]**

**Introduction:**

The objective of this project is to design and implement a data warehouse for a customer order processing system in a company. The data warehouse system will extract data from existing operational databases and provide online analytical processing with roll up, drill down, slice and dice features to meet user requirements.

**Business Requirements:**

The data warehouse system is required to answer several queries related to the stores, items, customers and orders in the enterprise. These queries include finding all the stores that hold a particular item of stock, finding all the orders that can be fulfilled by a given store, finding all the stores that hold items ordered by a given customer, and finding the stock level of a particular item in all stores in a particular city.

**Functional Specification:**

The data warehouse system will provide a user interface for users to input their queries and generate OLAP reports. The input specification will include the selection of dimensions and measures for each query. The output specification will include the format and presentation of the OLAP reports.

**Data Warehousing Design:**

The data warehouse will be designed using a star schema. The fact table will be the Ordered\_item table, and the dimension tables will be the Customer, Store, Item, and Time tables. The Customer dimension table will have attributes such as customer name, city, and state. The Store dimension table will have attributes such as city, state, phone, and headquarter address. The Item dimension table will have attributes such as description, size, weight, and unit price. The Time dimension table will have attributes such as order date and time.

**Data Cube Implementation:**

The data warehouse system will load data into data cubes for fast retrieval and processing of OLAP queries. The data cubes will be implemented using a multidimensional database management system (MDBMS) such as Microsoft SQL Server Analysis Services or Oracle OLAP.

**DATABASE:-**

create database warehouseass6;

use warehouseass6;

-- Dimension Tables

**-- Create Cities table**

**CREATE TABLE Cities** (

City\_id INT PRIMARY KEY,

City\_name VARCHAR(50) NOT NULL,

State VARCHAR(50) NOT NULL

);

**-- Create Items table**

**CREATE TABLE Items** (

Item\_id INT PRIMARY KEY,

Description VARCHAR(50) NOT NULL,

Size VARCHAR(20),

Weight FLOAT,

Unit\_price FLOAT

);

-- **Create Customers table**

**CREATE TABLE Customers** (

Customer\_id INT PRIMARY KEY,

Customer\_name VARCHAR(50) NOT NULL,

City\_id INT,

First\_order\_date DATE,

FOREIGN KEY (City\_id) REFERENCES Cities(City\_id)

);

**-- Create Stores table**

**CREATE TABLE Stores** (

Store\_id INT PRIMARY KEY,

City\_id INT,

Phone VARCHAR(20) NOT NULL,

FOREIGN KEY (City\_id) REFERENCES Cities(City\_id)

);

-- Fact Tables

**-- Create Sales table**

**CREATE TABLE Sales** (

Order\_no INT PRIMARY KEY,

Order\_date DATE NOT NULL,

Customer\_id INT,

Item\_id INT,

Quantity\_ordered INT,

Ordered\_price FLOAT,

Time TIMESTAMP NOT NULL,

FOREIGN KEY (Customer\_id) REFERENCES Customers(Customer\_id),

FOREIGN KEY (Item\_id) REFERENCES Items(Item\_id)

);

**-- Create Inventory table**

**CREATE TABLE Inventory** (

Store\_id INT,

Item\_id INT,

Quantity\_held INT,

Time TIMESTAMP NOT NULL,

PRIMARY KEY (Store\_id, Item\_id, Time),

FOREIGN KEY (Store\_id) REFERENCES Stores(Store\_id),

FOREIGN KEY (Item\_id) REFERENCES Items(Item\_id)

);

-- ===================================================

-- 2

INSERT INTO Cities (City\_id, City\_name, State) VALUES

(1, 'New York City', 'New York'),

(2, 'Los Angeles', 'California'),

(3, 'Chicago', 'Illinois'),

(4, 'Houston', 'Texas'),

(5, 'Philadelphia', 'Pennsylvania');

-- 4

INSERT INTO Items (Item\_id, Description, Size, Weight, Unit\_price) VALUES

(1, 'T-shirt', 'M', 0.2, 10.99),

(2, 'Jeans', 'L', 0.6, 35.99),

(3, 'Sneakers', '9', 0.9, 79.99),

(4, 'Jacket', 'M', 1.2, 99.99),

(5, 'Hoodie', 'S', 0.4, 29.99);

-- 1

INSERT INTO Customers (Customer\_id, Customer\_name, City\_id, First\_order\_date) VALUES

(1, 'John Doe', 1, '2022-01-01'),

(2, 'Jane Smith', 2, '2022-02-05'),

(3, 'David Brown', 3, '2022-03-10'),

(4, 'Mary Johnson', 2, '2022-02-15'),

(5, 'Mark Lee', 1, '2022-01-20');

-- 3

INSERT INTO Stores (Store\_id, City\_id, Phone) VALUES

(1, 1, '212-555-1234'),

(2, 2, '310-555-5678'),

(3, 3, '312-555-4321'),

(4, 2, '323-555-8765'),

(5, 1, '917-555-5555');

-- 5

INSERT INTO Sales (Order\_no, Order\_date, Customer\_id, Item\_id, Quantity\_ordered, Ordered\_price, Time) VALUES

(1, '2022-01-01', 1, 1, 2, 21.98, '2022-04-01'),

(2, '2022-02-05', 2, 2, 1, 35.99, '2022-04-01'),

(3, '2022-03-10', 3, 3, 1, 79.99, '2022-04-01'),

(4, '2022-02-15', 4, 4, 1, 99.99, '2022-04-01'),

(5, '2022-01-20', 5, 5, 2, 59.98, '2022-04-01');

-- 6

INSERT INTO Inventory (Store\_id, Item\_id, Quantity\_held, Time) VALUES

(1, 1, 10, '2022-04-01'),

(2, 2, 20, '2022-04-01'),

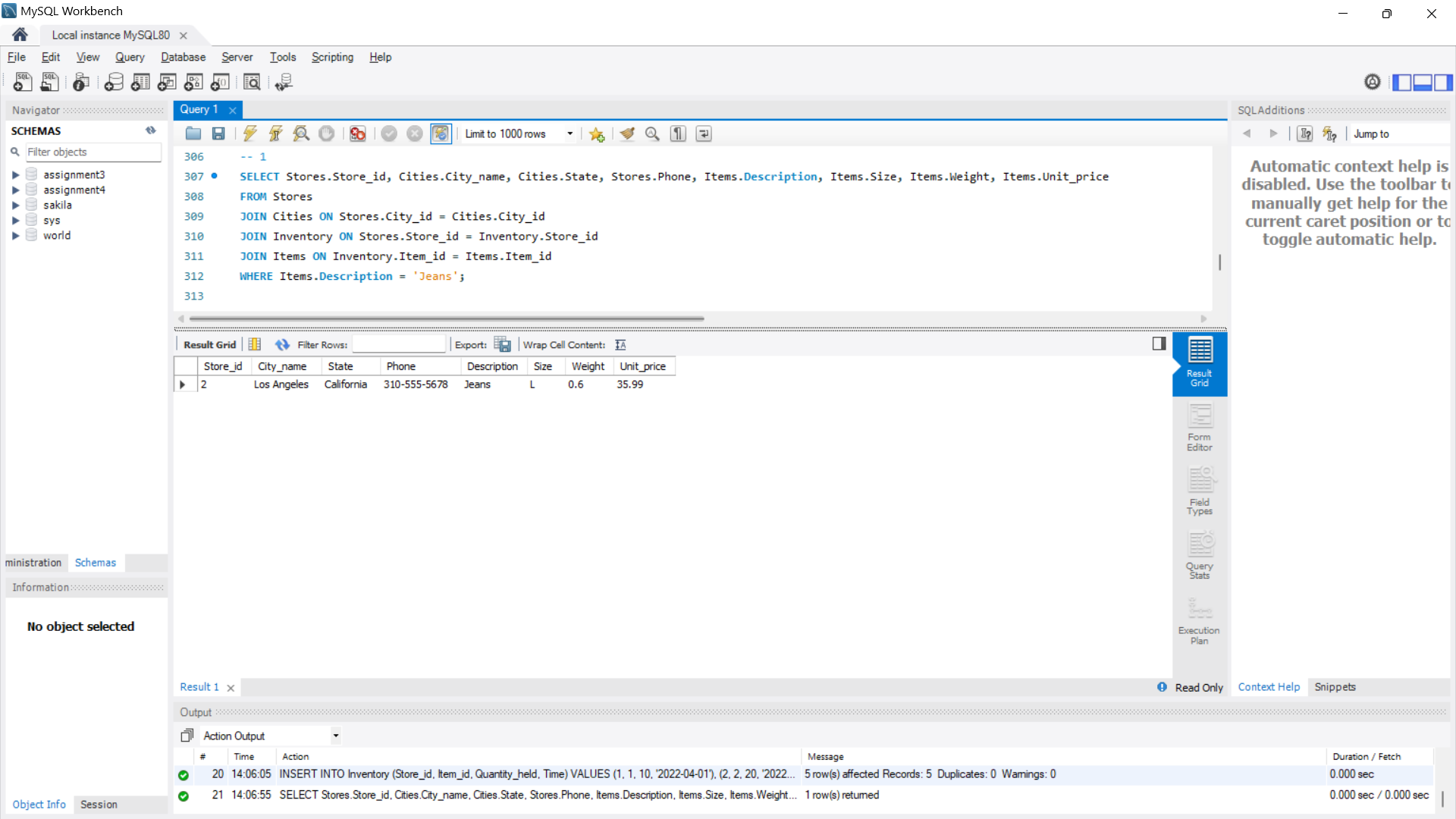
(3, 3, 15, '2022-04-01'),

(4, 4, 5, '2022-04-01'),

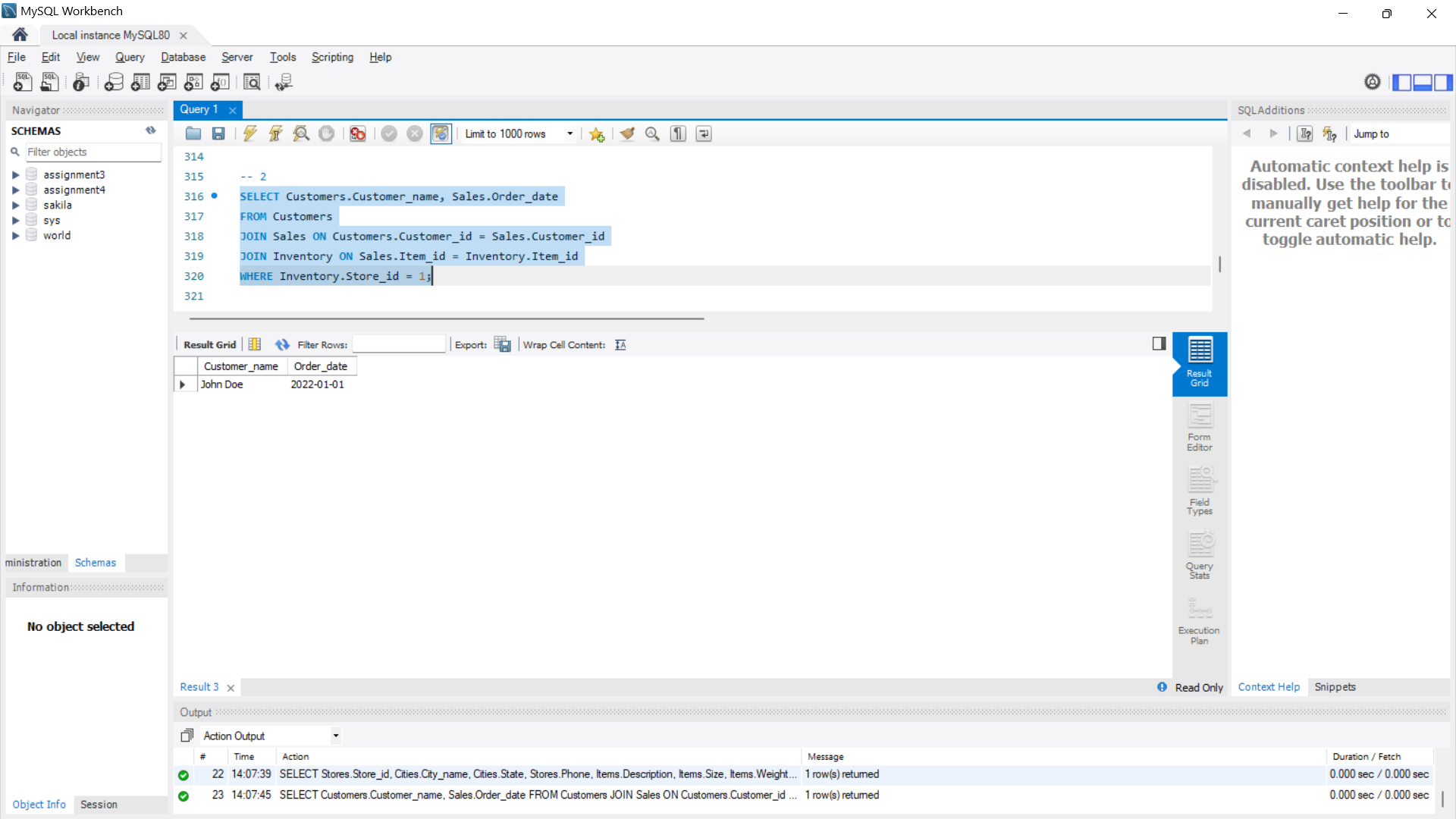
(5, 5, 8, '2022-04-01');

**Observations:**

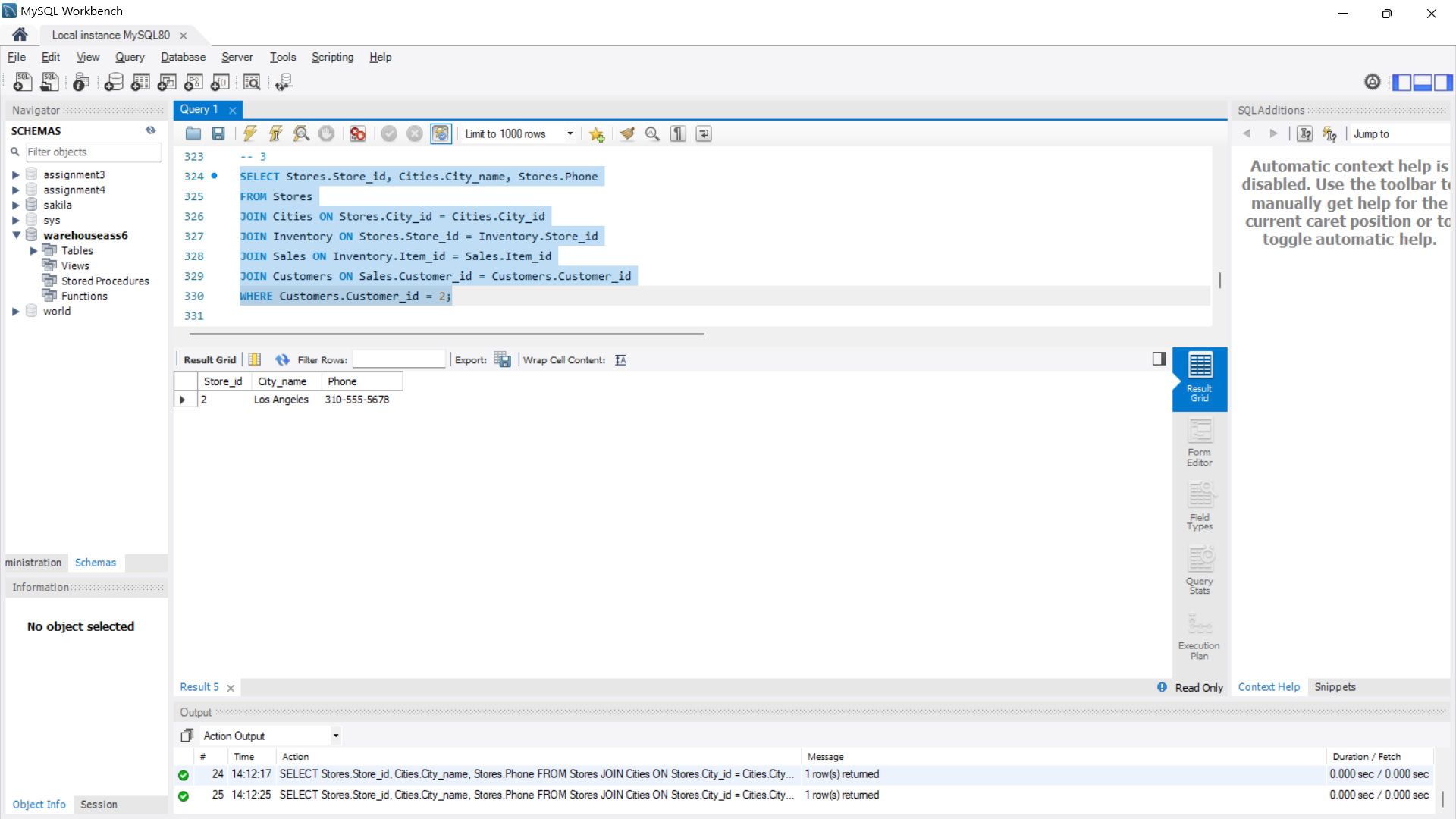
1. Find all the stores along with city, state, phone, description, size, weight and unit price that hold a particular item of stock.



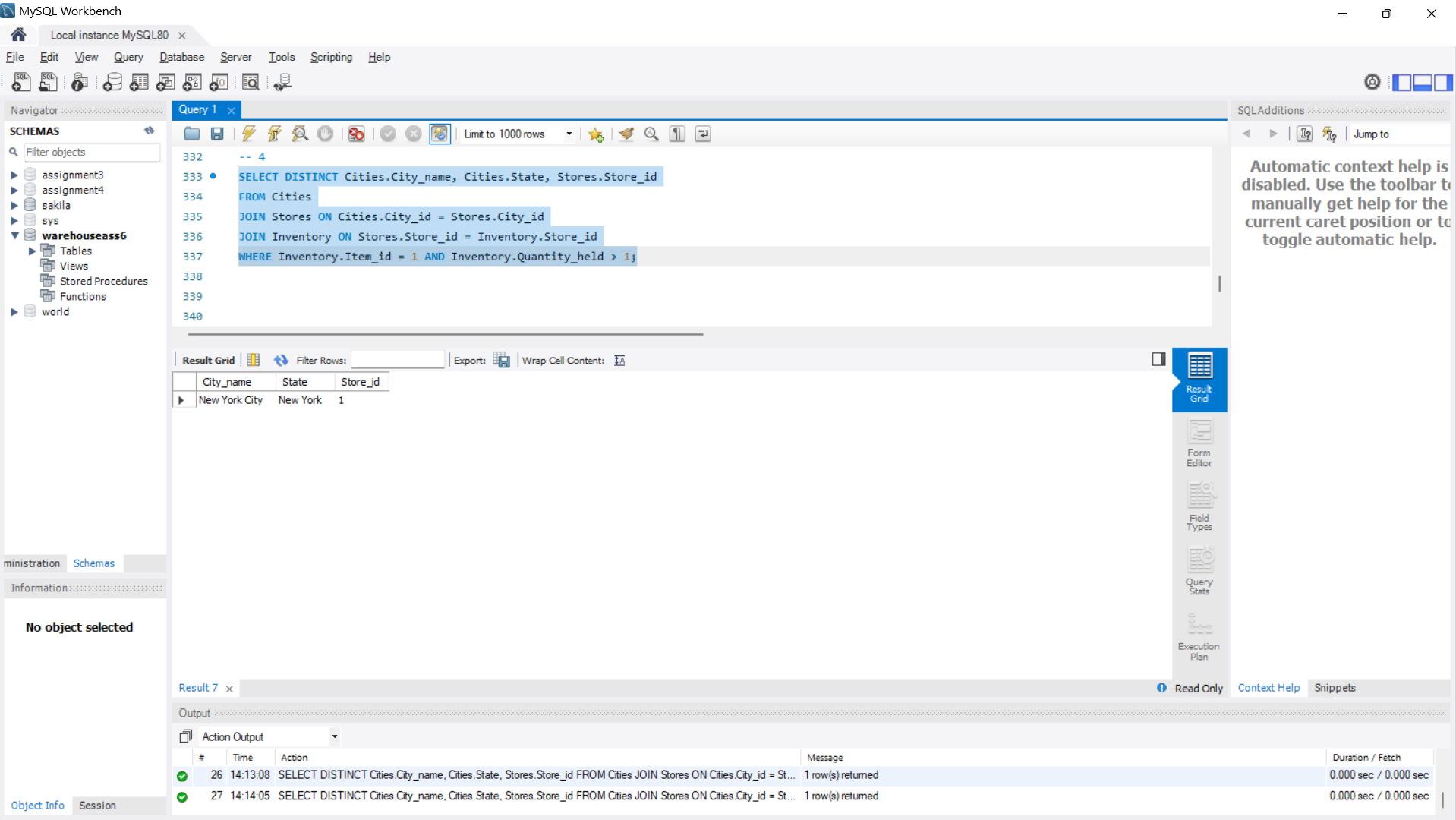
2. Find all the orders along with customer name and order date that can be fulfilled by a given store.



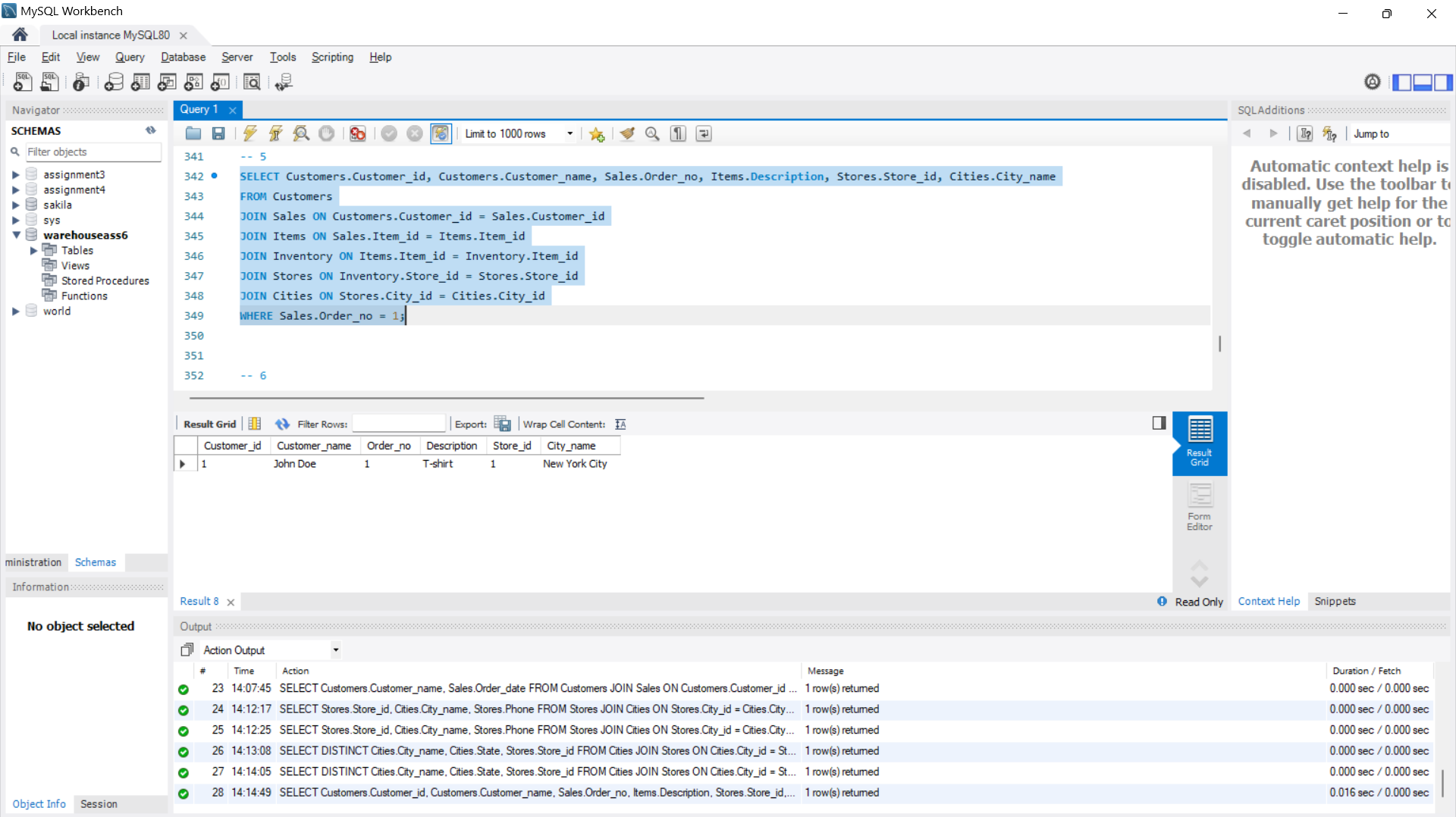
3.Find all stores along with city name and phone that hold items ordered bygiven customer.



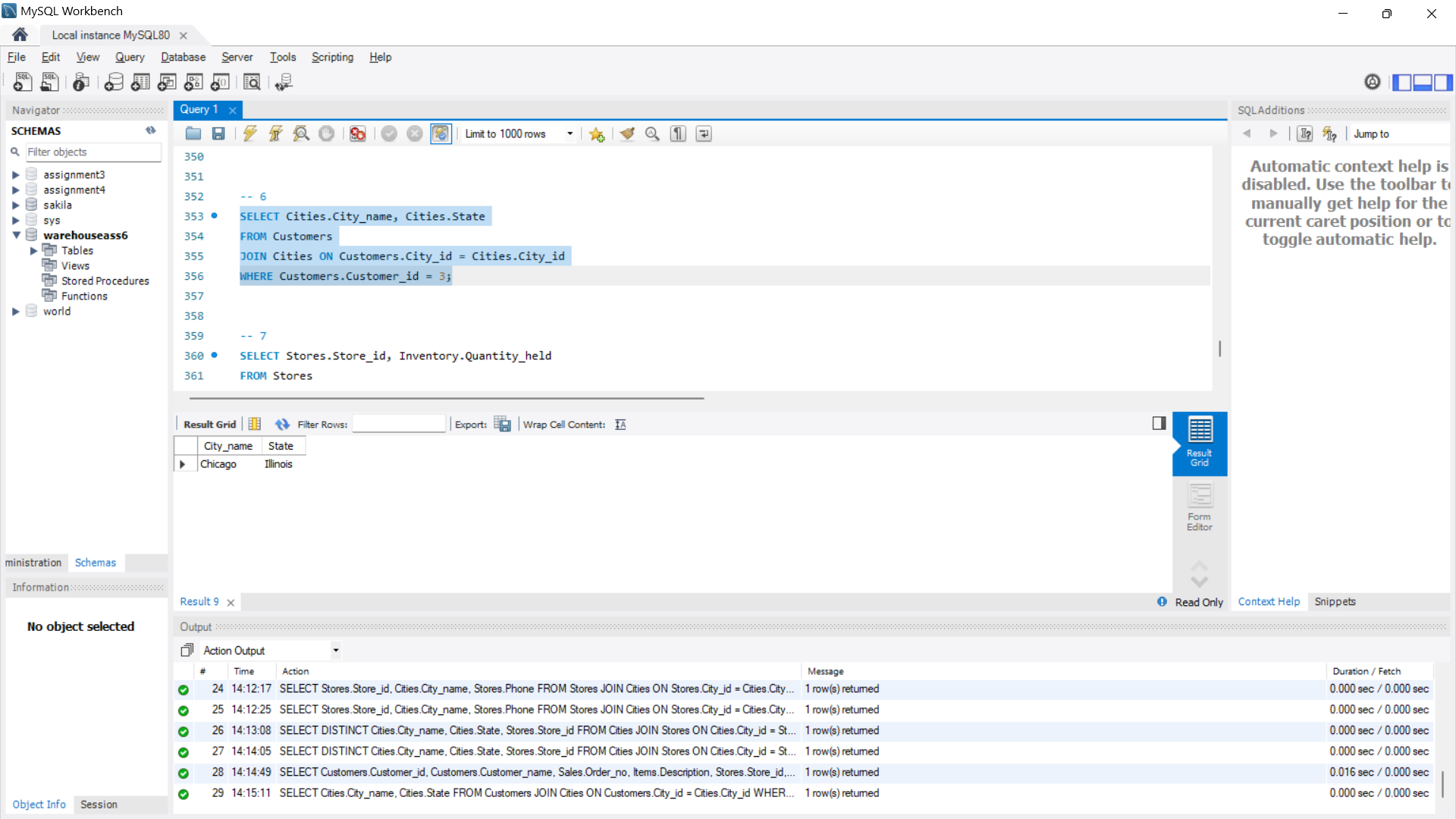
4. Find the headquarter address along with city and state of all stores that hold stocks of an item above a particular level.

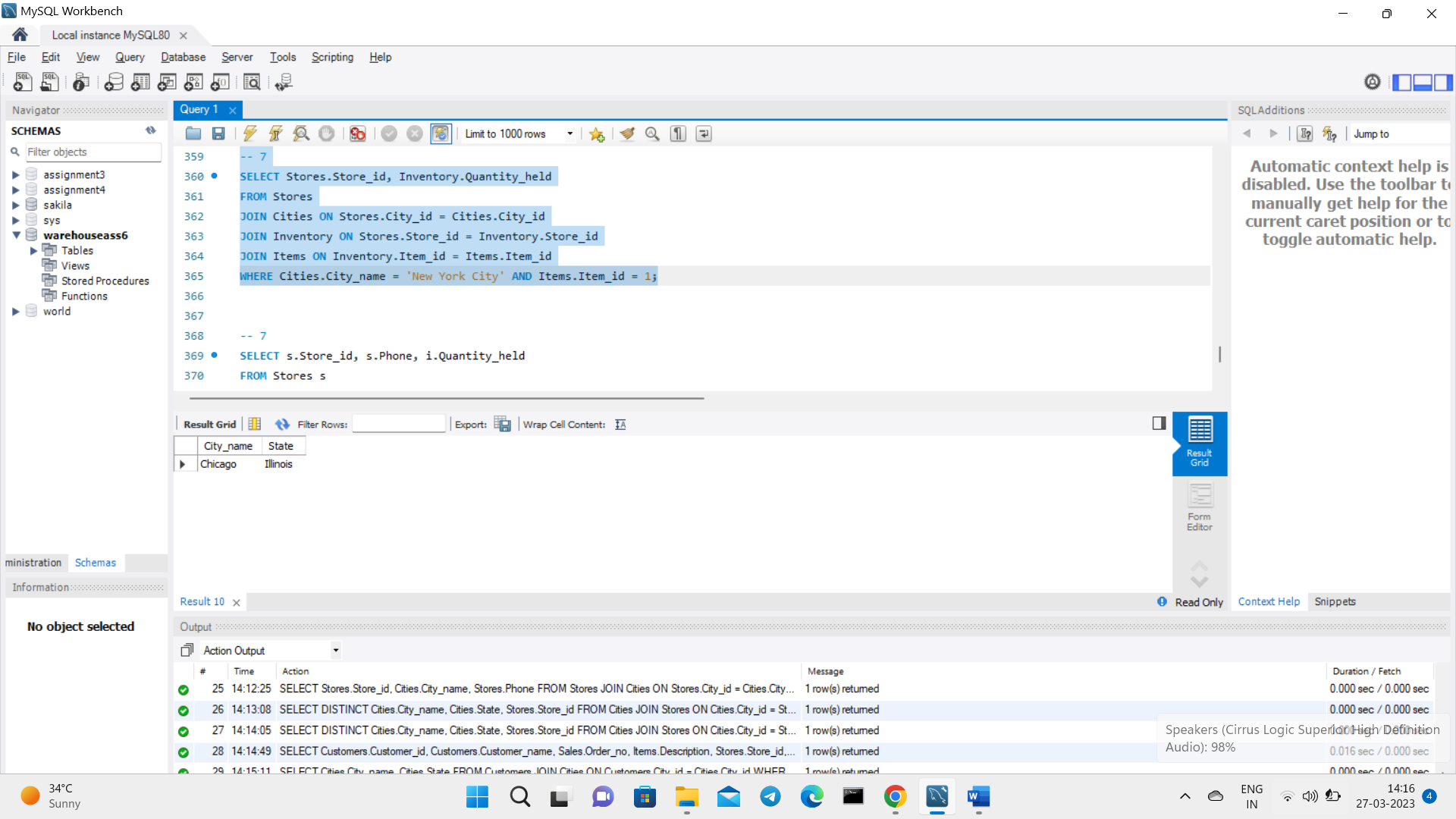


5. For each customer order, show the items ordered along with description, store id and city name and the stores that hold the items.

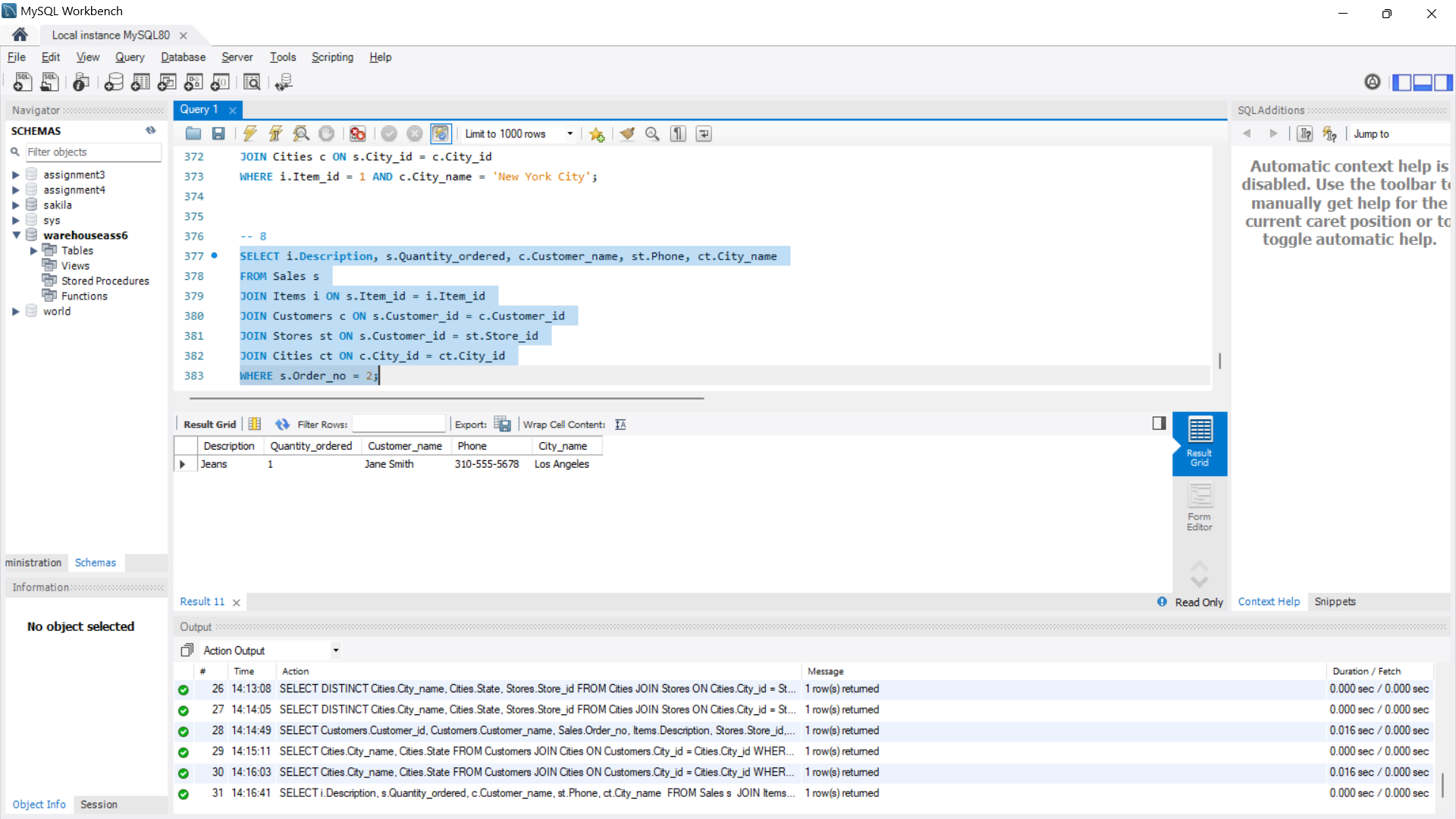


6. Find the city and the state in which a given customer lives.

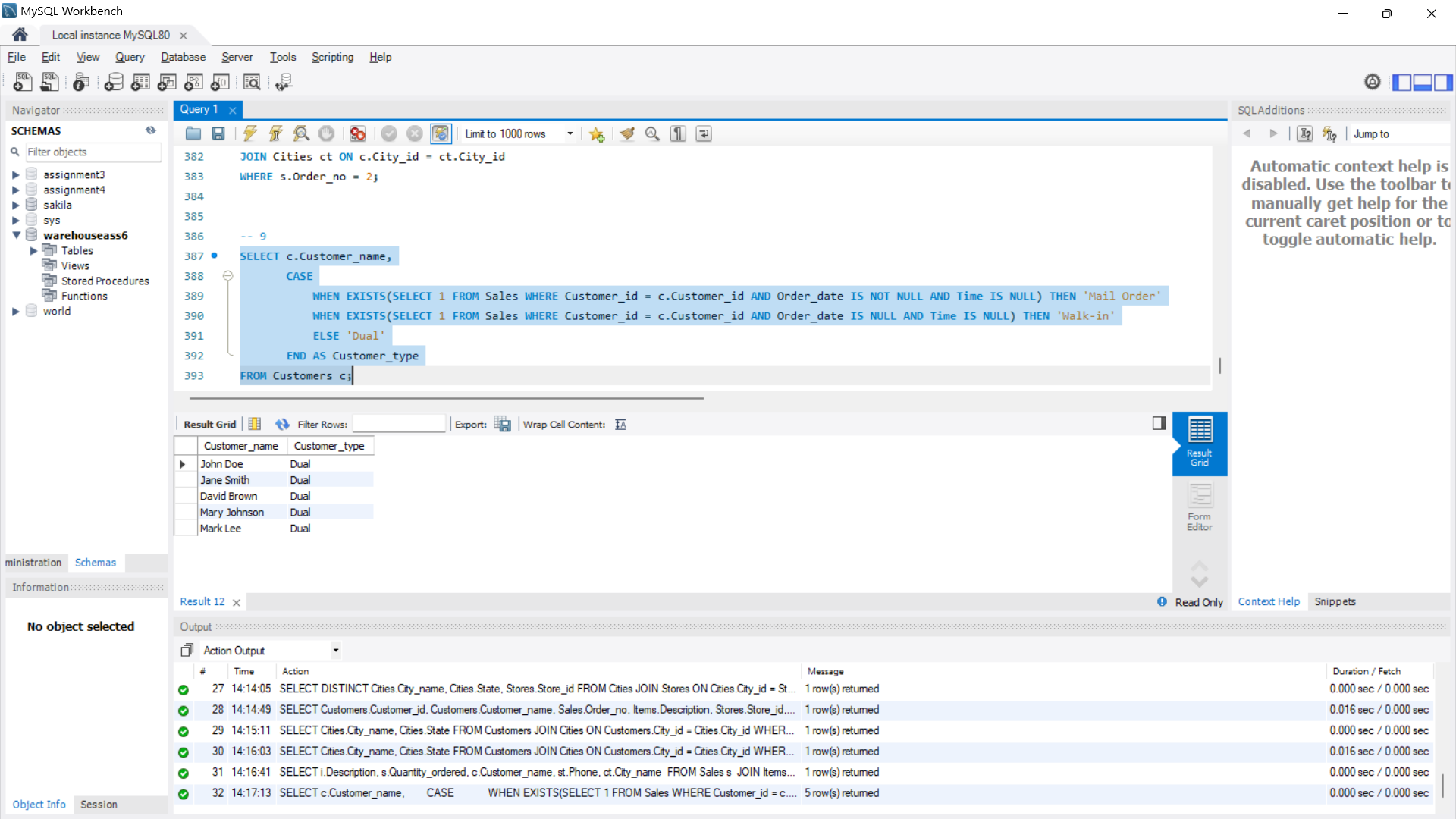


7. Find the stock level of a particular item in all stores in a particular city.  


8. Find the items, quantity ordered, customer, store and city of an order.



9. Find the walk in customers, mail order customers and dual customers (both walk-in and mail order)



**Conclusion:**

In conclusion, the data warehouse system designed and implemented for the customer order processing system in the company meets the business requirements and provides a user-friendly interface for generating OLAP reports. The use of a star schema and data cubes enables fast retrieval and processing of OLAP queries. The accuracy and completeness of the reports generated by the data warehouse system will be ensured by verifying the source relational tables' data.