01: Creating a Database

```
CREATE DATABASE Restaurant;
```

```
USE Restaurant;
```

02: Creating Tables

```
CREATE TABLE Customers (
    CustomerID INT(6) NOT NULL AUTO_INCREMENT,
    PRIMARY KEY(CustomerID)
);
```

```
CREATE TABLE Customers (
    CustomerID INT(6) NOT NULL AUTO_INCREMENT,
    FirstName VARCHAR(200) NOT NULL,
    LastName VARCHAR(200) NOT NULL,
    Email VARCHAR(200),
    Address VARCHAR(200),
    City VARCHAR(200),
    State CHAR(2),
    Phone VARCHAR(20) NOT NULL,
    Birthday DATE,
    PRIMARY KEY(CustomerID)
);
```

03: Writing SQL Queries

```
SELECT * FROM Customers;
```

```
SELECT * FROM Dishes;
```

```
SELECT Name FROM Customers;
```

```
SELECT FirstName, LastName, Email FROM Customers;
```

04: Narrowing query results

```
SELECT FirstName, LastName, State FROM Customers;
```

```
SELECT FirstName, LastName, State FROM Customers WHERE State = "CA";
```

```
SELECT FirstName, LastName, State FROM Customers WHERE State = "TX";
```

```
SELECT FirstName, LastName, State FROM Customers WHERE State = "CA" OR
State = "CO;
```

```
SELECT FirstName, LastName, State FROM Customers WHERE State LIKE "C%";
```

```
SELECT FirstName, LastName, State FROM Customers WHERE Name = "Taylor";
```

```
SELECT ID, FirstName, LastName, State FROM Customers WHERE Name =
"Taylor”;
```

```
SELECT * FROM Reservations WHERE Date > "2019-02-06" AND Date < "2019-02-
07”;
```

05: Sorting results

```
SELECT * FROM Dishes ORDER BY `Name`;
```

```
SELECT * FROM Dishes ORDER BY `Name` ASC;
```

```
SELECT * FROM Dishes ORDER BY `Name` DESC;
```

```
SELECT * FROM Dishes ORDER BY Price;
```

```
SELECT * FROM Reservations ORDER BY `Date`;
```

```
SELECT * FROM Reservations WHERE `Date` > "2019-02-06" AND `Date` < "2019-
02-07" ORDER BY `Date`;
```

06: Aggregate functions

```
SELECT COUNT(FirstName) FROM Customers;
```

```
SELECT COUNT(FirstName) FROM Customers WHERE State = "CA";
```

```
SELECT COUNT(State) FROM Customers WHERE State = "CA";
```

```
SELECT SUM(Price) FROM Dishes;
```

```
SELECT SUM(Price), AVG(Price) FROM Dishes;
```

```
SELECT SUM(Price), AVG(Price), MIN(Price), MAX(Price) FROM Dishes;
```

07: Joining tables

```
SELECT FirstName, LastName, FavoriteDish FROM Customers JOIN Dishes;
```

```
SELECT FirstName, LastName, FavoriteDish FROM Customers JOIN Dishes ON
Customers.FavoriteDish = Dishes.DishID;
```

```
SELECT FirstName, LastName, FavoriteDish, Dishes.`Name` FROM Customers
JOIN Dishes ON Customers.FavoriteDish = Dishes.DishID;
```

```
SELECT FirstName, LastName, FavoriteDish, Dishes.DishID, Dishes.`Name`
FROM Customers JOIN Dishes ON Customers.FavoriteDish = Dishes.DishID;
```

```
SELECT FirstName, LastName, Dishes.`Name` FROM Customers JOIN Dishes ON
Customers.FavoriteDish = Dishes.DishID;
```

```
SELECT * FROM Reservations;
```

```
SELECT FirstName, LastName, Reservations.Date, Reservations.PartySize FROM
Customers JOIN Reservations ON Reservations.CustomerID =
Customers.CustomerID ORDER BY Reservations.Date;
```

```
SELECT
ord.OrderID AS order_id,
COUNT(*) AS num_dishes,
GROUP_CONCAT(ord_dish.DishID) AS dish_ids
FROM Orders AS ord
JOIN OrdersDishes AS ord_dish
ON ord.OrderID = ord_dish.OrderID
GROUP BY ord.OrderID
ORDER BY ord.OrderID
```

For MySQL:

```
SELECT OrdersDishes.OrderID, Orders.OrderDate,
Customers.FirstName, Customers.LastName, Customers.Phone,
GROUP_CONCAT(Dishes.`Name` SEPARATOR ', ') AS Items,
COUNT(OrdersDishes.DishID) AS Qty, SUM(Dishes.Price) AS Total
FROM OrdersDishes
JOIN Dishes on OrdersDishes.DishID = Dishes.DishID
JOIN Orders on Orders.OrderID = OrdersDishes.OrderID
JOIN Customers on Orders.CustomerID = Customers.CustomerID
GROUP BY(Orders.OrderID);
```

For SQLite: *SQLite doesn't support the `SEPARATOR` keyword in `GROUP_CONCAT()`.*

```
SELECT OrdersDishes.OrderID, Orders.OrderDate, Customers.FirstName,
Customers.LastName, Customers.Phone, GROUP_CONCAT(Dishes.`Name`, ', ') AS
Items, COUNT(OrdersDishes.DishID) AS Qty, SUM(Dishes.Price) AS Total
FROM OrdersDishes
JOIN Dishes on OrdersDishes.DishID = Dishes.DishID
JOIN Orders on Orders.OrderID = OrdersDishes.OrderID
JOIN Customers on Orders.CustomerID = Customers.CustomerID
GROUP BY(Orders.OrderID);
```

08: Modifying data

```
INSERT INTO Customers;
```

```
INSERT INTO Customers (FirstName, LastName, Email, Phone) VALUES ("Jane",
"Smith", "jsmith2019@landonhotel.com", "415-555-1234");
```

```
SELECT * FROM Customers WHERE FirstName = "Taylor" AND LastName =
"Jenkins";
```

```
SELECT * FROM Customers WHERE CustomerID = 1;
```

```
UPDATE Customers SET Email = "tjenkins@landonhotel.com" WHERE CustomerID =
1;
```

```
SELECT * FROM Customers WHERE CustomerID = 1;
```

```
SELECT * FROM Customers WHERE FirstName = "Taylor" AND LastName =
"Jenkins";
```

```
DELETE FROM Customers WHERE CustomerID = 26;
```

```
SELECT * FROM Customers;
```