-- SHOW tables;

-- SHOW columns FROM customers;

-- Select all the records from the "Customers" table.

SELECT \* FROM customers;

-- Get distinct countries from the Customers table. In other words, each country should only appear once in the results. (You’ll probably have to search online for this one.)

SELECT DISTINCT Country FROM customers;

-- Get all the records from the table Customers where the Customer’s ID starts with “BL”.

SELECT \* FROM Customers WHERE CustomerID LIKE 'BL%';

-- Get the first 100 records of the orders table.

SELECT \* FROM orders LIMIT 100;

-- Get all customers that live in the postal codes 1010, 3012, 12209, and 05023.

SELECT \* FROM customers WHERE PostalCode = 1010 OR PostalCode = 3012 OR PostalCode = 12209 OR PostalCode = 05023;

-- Get all orders where the ShipRegion is not equal to NULL.

SELECT \* FROM orders WHERE ShipRegion IS NOT NULL;

-- Get all customers ordered by the country, then by the city.

SELECT \* FROM customers ORDER BY country, city;

-- Add a new customer to the customers table. You can use whatever values/

DELETE FROM customers WHERE CustomerID ='COCOA';

INSERT INTO customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)

values ( 'COCOA', 'Jingala', 'Miraj', 'Owner', '55 richmond rd', 'Edison', 'LAS', 12345, 'Mars', '454-4544444', '777-7777777');

SELECT \* FROM customers WHERE City = 'Edison';

-- Update all ShipRegion to the value ‘EuroZone’ in the Orders table, where the ShipCountry is equal to France. (Note: this requires safe updates to be turned off. Search online for more info.)

SET SQL\_SAFE\_UPDATES=0;

UPDATE orders SET ShipRegion ='EuroZone' WHERE ShipCountry ='France';

SET SQL\_SAFE\_UPDATES=1;

-- Delete all orders from `order details` that have quantity of 1.

-- SHOW Tables;

SELECT \* FROM `order details`;

SET SQL\_SAFE\_UPDATES=0;

DELETE FROM `order details` WHERE quantity = 1;

SET SQL\_SAFE\_UPDATES=1;

-- Calculate the average, max, and min of the quantity at the `order details` table.

SELECT avg(quantity) FROM `order details`;

SELECT max(quantity) FROM `order details`;

SELECT min(quantity) FROM `order details`;

-- Calculate the average, max, and min of the quantity at the `order details` table, grouped by the orderid. This will show the average, max, and min for each order.

SELECT avg(quantity) FROM `order details` ;

SELECT max(quantity) FROM `order details`;

SELECT min(quantity) FROM `order details`;

-- Find the CustomerID that placed order 10290 (orders table)

SELECT \* FROM orders;

SELECT CustomerID FROM orders WHERE OrderID = 10290;

-- Do an inner join, left join, right join on orders and customers tables. (These are three separate queries, one for each type of join.)

-- Use a join to get the ship city and ship country of all the orders which are associated with an employee who is in London.

-- Use a join to get the ship name of all orders that include a discontinued product. (See orders, order details, and products. 1 means discontinued.)

-- Get employees’ firstname for all employees who report to no one.

SELECT \* FROM employees;

SELECT FirstName FROM employees WHERE ReportsTo IS NULL ;

-- Get employees’ firstname for all employees who report to Andrew.--

SELECT \* FROM employees;

SELECT e1.FirstName FROM employees AS e1 INNER JOIN employees AS e2 ON e1.ReportsTo = e2.EmployeeID;