Show the associated query(ies) for each question below. For each example, use aliases for any table names (for any table joins) and for field names to rename them if/as necessary. For example, use “Product Number” and not ProductNumber. When asked, be sure to include the # of records found.

1. Show the query to add yourself as a new customer. Use your name and either your real address info or mythical.

**INSERT INTO CUSTOMERS (customerfirstname, customerlastname,**

**customerstreetaddress, customercity,**

**customerstate, customerzipcode,**

**customerareacode, customerphonenumber)**

**VALUES ('John', 'Doe', '123 Main St.', 'San Diego', 'CA', '91911', 8584, '555-5555');**

1. Show two different queries to show all of the customers in the customers table **except you**. Note: **One** of the queries must use the IN operator.\*

**SELECT \* FROM CUSTOMERS WHERE customerID != 1030;**

**SELECT \* FROM CUSTOMERS WHERE customerID <> 1030;**

**SELECT \* FROM CUSTOMERS WHERE customerID NOT IN (1028);**

1. Show the query to select each customer's first name, last name, order number, product number, and quoted price.

**SELECT c.customerfirstname, c.customerlastname,**

**o.orderid, p.productid, od.orderdetailquotedprice**

**FROM customers c, orders o, products p, orderdetails od**

**WHERE c.customerid = o.customerid**

**AND o.orderid = od.orderid**

**AND p.productid = od.productid**

1. Show the query to select all customers whose first name begins with a vowel (a, e, i, o or u) and whose last name ends with a vowel (a, e, i, o or u). Sort results by last name, then first name, both in ascending order.

**SELECT customerfirstname, customerlastname**

**FROM customers**

**WHERE (((customerfirstname LIKE 'a%') OR**

**(customerfirstname LIKE 'e%') OR**

**(customerfirstname LIKE 'i%') OR**

**(customerfirstname LIKE 'o%') OR**

**(customerfirstname LIKE 'u%')) AND**

**((customerlastname LIKE '%a') OR**

**(customerlastname LIKE '%e') OR**

**(customerlastname LIKE '%i') OR**

**(customerlastname LIKE '%o') OR**

**(customerlastname LIKE '%u')));**

1. Show the query to select all customers whose first name begins with a (a, e, i, o or u) or whose last name ends with a vowel (a, e, i, o or u). Sort results by last name, then first name, both in ascending order.

**SELECT customerfirstname, customerlastname**

**FROM customers**

**WHERE (((customerfirstname LIKE 'a%') OR**

**(customerfirstname LIKE 'e%') OR**

**(customerfirstname LIKE 'i%') OR**

**(customerfirstname LIKE 'o%') OR**

**(customerfirstname LIKE 'u%')) OR**

**((customerlastname LIKE '%a') OR**

**(customerlastname LIKE '%e') OR**

**(customerlastname LIKE '%i') OR**

**(customerlastname LIKE '%o') OR**

**(customerlastname LIKE '%u')));**

1. Show the query to select all category descriptions, product names, and retail prices. Sort results by category description in ascending order, then retail price descending order.

**SELECT c.categorydescription,**

**p.productname,**

**p.productprice**

**FROM categories c,**

**products p**

**WHERE c.categoryid = p.categoryid**

**ORDER BY c.categorydescription, p.productprice DESC;**

1. Show the query to select all category descriptions, product names, and retail prices, but only for a category of “Bikes” or “Wheels”. Sort results by category description in ascending order, then retail price in descending order.

**SELECT c.categorydescription,**

**p.productname,**

**p.productprice**

**FROM categories c,**

**products p**

**WHERE c.categoryid = p.categoryid**

**AND c.categorydescription IN ('bikes** **', 'wheels** **')**

**ORDER BY c.categorydescription,**

**p.productprice DESC;**

1. Show the query to select all category descriptions, product names, and retail prices, but only for the categories that are not “Bikes” or “Wheels”. Sort results by category description in ascending order, then retail price in descending order.

**SELECT c.categorydescription,**

**p.productname,**

**p.productprice**

**FROM categories c,**

**products p**

**WHERE c.categoryid = p.categoryid**

**AND c.categorydescription NOT IN ('bikes ', 'wheels ')**

**ORDER BY c.categorydescription,**

**p.productprice DESC;**

1. Show the query to select all category descriptions, product names, retail price, wholesale price, and vendor name. Sort results by category description in ascending order, then vendor name in descending order, then retail price in descending order, then wholesale price in descending order.

**SELECT c.categorydescription,**

**p.productname,**

**p.productprice,**

**pv.productvendorwholesaleprice,**

**v.vendorname**

**FROM categories c,**

**products p,**

**productvendors pv,**

**vendors v**

**WHERE c.categoryid = p.categoryid**

**AND pv.vendorid = v.vendorid**

**ORDER BY c.categorydescription,**

**v.vendorname desc,**

**p.productprice desc,**

**pv.productvendorwholesaleprice desc;**

1. Show the query to select all category descriptions, product names, retail price, wholesale price, and vendor name. Sort results by category description in ascending order, then vendor name in descending order, then retail price in descending order, then wholesale price in descending order, but only for vendor “Lone Star Bike Supply” or category “Bikes”. How many records were returned?

**SELECT c.categorydescription,**

**p.productname,**

**p.productprice,**

**pv.productvendorwholesaleprice,**

**v.vendorname**

**FROM categories c,**

**products p,**

**productvendors pv,**

**vendors v**

**WHERE c.categoryid = p.categoryid**

**AND (v.vendorname = 'Lone Star Bike Supply** **' OR**

**c.categorydescription = 'Bikes** **')**

**AND pv.vendorid = v.vendorid**

**ORDER BY c.categorydescription,**

**v.vendorname desc,**

**p.productprice desc,**

**pv.productvendorwholesaleprice desc;**

1. Show the query to select all category descriptions, product names, retail price, wholesale price, and vendor name. Sort results by category description in ascending order, then vendor name in descending order, then retail price in descending order, then wholesale price in descending order, but only for vendor “Lone Star Bike Supply” and category “Bikes”. How many records were returned?

**SELECT c.categorydescription,**

**p.productname,**

**p.productprice,**

**pv.productvendorwholesaleprice,**

**v.vendorname**

**FROM categories c,**

**Products p,**

**productvendors pv,**

**vendors v**

**WHERE c.categoryid = p.categoryid**

**AND (v.vendorname = 'Lone Star Bike Supply** **' AND**

**c.categorydescription = 'Bikes** **')**

**AND pv.vendorid = v.vendorid**

**ORDER BY c.categorydescription,**

**v.vendorname desc,**

**p.productprice desc,**

**pv.productvendorwholesaleprice desc;**

1. Show the query to select all employee first name, employee last name, order number, product number, and quantity ordered. Sort results by quantity ordered in descending order.

**SELECT e.employeefirstname, e.employeelastname,**

**o.orderid, p.productid,**

**od.orderdetailquantityordered**

**FROM employees e, orders o, products p, orderdetails od**

**WHERE e.employeeid = o.employeeid**

**AND o.orderid = od.orderid**

**AND p.productid = od.productid**

**ORDER BY od.orderdetailquantityordered DESC;**

1. Show the query to select all employee first name, employee last name, order number, quoted price, and quantity ordered where the quantity ordered is greater than 4. Sort results by quantity ordered in descending order and then by quoted price in descending order. How many records are there?

**SELECT e.employeefirstname, e.employeelastname,**

**o.orderid, od.orderdetailquotedprice,**

**od.orderdetailquantityordered**

**FROM employees e, orders o, products p, orderdetails od**

**WHERE e.employeeid = o.employeeid**

**AND o.orderid = od.orderid**

**AND p.productid = od.productid**

**AND od.orderdetailquantityordered > 4**

**ORDER BY od.orderdetailquantityordered DESC,**

**od.orderdetailquotedprice DESC;**

1. Show two different queries to select all employee first name, employee last name, order number, quoted price, and quantity ordered where the quantity ordered is not equal to 4. Sort results by quantity ordered in descending order and then by quoted price in descending order.

**SELECT e.employeefirstname, e.employeelastname,**

**o.orderid, od.orderdetailquotedprice,**

**od.orderdetailquantityordered**

**FROM employees e, orders o, products p, orderdetails od**

**WHERE e.employeeid = o.employeeid**

**AND o.orderid = od.orderid**

**AND p.productid = od.productid**

**AND od.orderdetailquantityordered != 4**

**ORDER BY od.orderdetailquantityordered DESC,**

**od.orderdetailquotedprice DESC;**

**SELECT e.employeefirstname, e.employeelastname,**

**o.orderid, od.orderdetailquotedprice,**

**od.orderdetailquantityordered**

**FROM employees e, orders o, products p, orderdetails od**

**WHERE e.employeeid = o.employeeid**

**AND o.orderid = od.orderid**

**AND p.productid = od.productid**

**AND od.orderdetailquantityordered <> 4**

**ORDER BY od.orderdetailquantityordered DESC,**

**od.orderdetailquotedprice DESC;**

1. Show the query to select customer first name, customer last name, employee ID, order number, product number, and quoted price. Sort results by employee ID in ascending order.

**SELECT c.customerfirstname, c.customerlastname,**

**e.employeeid, o.orderid,**

**p.productid, od.orderdetailquotedprice**

**FROM customers c, employees e, orders o,**

**products p, orderdetails od**

**WHERE c.customerid = o.customerid**

**AND e.employeeid = o.employeeid**

**AND o.orderid = od.orderid**

**AND p.productid = od.productid**

**ORDER BY e.employeeid;**

1. Do a Union query to combine all customers (do not include the customer ID) with all employees (do not include the employee ID).

**SELECT c.customerfirstname, c.customerlastname,**

**c.customerstreetaddress,**

**c.customercity, c.customerstate, c.customerzipcode,**

**c.customerareacode, c.customerphonenumber**

**FROM customers c**

**UNION SELECT**

**e.employeefirstname, e.employeelastname,**

**e.employeestreetaddress, e.employeecity,**

**e.employeestate, e.employeezipcode,**

**e.employeeareacode, e.employeephonenumber**

**FROM employees e;**