

/\*\*

\* @author: Andressa Machado - 040923007

\* CST8215 - DATABASE

\* Section: 302

\* Professor: Kumari Gurusamy.

\* Lab 9 - Query Inventory Database

\* last updated April 2nd, 2018.

\*/

/\*question 1: What is the average sell\_price for each product. Round to 2 decimal places.

If null value is returned, enter a 0. Make sure to enter both decimal places.\*/

-- Answer: 0, 66.70, 23.41, 90.71, 54.04, 63.61, 81.36, 143.13, 41.57, 453.99

SELECT

Products.name,  
AVG(Order\_lines.sell\_price)

FROM

Products LEFT OUTER JOIN Order\_lines  
ON  
Products.id = Order\_lines.product\_id

GROUP BY

Products.name

ORDER BY

Products.name;

--Question 2: How many order lines are there?

--22490

SELECT COUNT (id) FROM order\_lines;

--Question 3: How many orders do not have an order\_total?

--1097

SELECT order\_total FROM orders

SELECT COUNT(\*) FROM orders WHERE order\_total IS NULL

-- Question 4: How many orders have an order\_total that is less than the sum of their line\_total(s)?

-- 383

SELECT \* FROM order\_lines

SELECT

Orders.id,  
Orders.order\_total,  
SUM(Order\_lines.line\_total)

```
FROM
    Orders JOIN Order_lines
    ON
        Orders.id = Order_lines.order_id
GROUP BY
    Orders.id
HAVING
    Orders.order_total < SUM(Order_lines.line_total)
ORDER BY
    Orders.id;
```

-- Question 5: How many orders were placed in September 2013?  
--749

```
SELECT COUNT (*) FROM orders WHERE order_date BETWEEN '2013-09-01' AND '2013-09-30'
```

--another valid answer:  
--776

```
SELECT
    count(id)
FROM
    Orders
WHERE
    EXTRACT(YEAR FROM order_date) = 2013
    AND
    EXTRACT(MONTH FROM order_date) = 09;
```

--Question 6:What is the smallest order total that is greater than 0 (zero)? Include all digits including leading zeros if applicable.  
--0.52

```
SELECT order_total FROM orders WHERE order_total > 0 ORDER BY order_total
--or:
SELECT
    MIN(order_total)
FROM
    Orders
WHERE
    order_total > 0;
```

--Question 7:How many orders have no order lines?  
--1055

```
SELECT
    count(Orders.id)
```

```

FROM
    Orders LEFT OUTER JOIN Order_lines
    ON
    Orders.id = Order_lines.order_id
WHERE
    Order_lines.id IS NULL;

```

--Question 8:How many customers have an orphaned customer type?  
-- 442

```

SELECT * FROM customer_types

SELECT
    count(Customers.customer_type_id)
FROM
    Customer_types RIGHT OUTER JOIN Customers
    ON
    Customer_types.id = Customers.customer_type_id
WHERE
    Customers.customer_type_id NOT IN (1, 2, 3, 4);

```

--Question 9: Which product had the largest total sell\_price?  
-- Widget 1.0

```

SELECT DISTINCT
    products.id AS productID, sell_price, quantity, products.name
FROM
    order_lines LEFT JOIN products
    ON products.id = order_lines.product_id
WHERE sell_price * quantity = (
    SELECT
        max(sell_price * quantity)
    FROM
        order_lines LEFT JOIN products
    ON
        products.id = order_lines.product_id );

```

--or:

```

SELECT
    Products.name,
    MAX(Order_lines.sell_price) AS "Max"
FROM
    Products LEFT OUTER JOIN Order_lines
    ON
    Products.id = Order_lines.product_id
GROUP BY

```

```
        Products.name
ORDER BY
        "Max" DESC;
```

--Question 10: How many orders have a cvv of 668?  
--17

```
SELECT COUNT (order_date) FROM orders WHERE cvv LIKE '668';
```

--Question 11: What was the total line\_total for each product that was sold in the largest order? (largest order\_total).

--Enter 0 if the product was not in the largest order. Make sure to enter both decimal places.

```
SELECT
        *
FROM
        orders
WHERE
        order_total IS NOT NULL
ORDER BY
        order_total DESC
LIMIT 1
```

```
SELECT * FROM order_lines WHERE order_id = 926
```

```
SELECT
        Products.name,
        SUM(Order_lines.line_total)
FROM
        Products JOIN Order_lines
        ON
        Products.id = Order_lines.product_id
WHERE
        Order_lines.order_id = 926
GROUP BY
        Products.name
ORDER BY
        Products.name;
```

--Question 12: Which product was not sold?  
--AlienGlows Notebook

```
SELECT
        Products.name,
        count(Order_lines.id)
FROM
        Products LEFT OUTER JOIN Order_lines
```

```
        ON
        Products.id = Order_lines.product_id
GROUP BY
    Products.name
ORDER BY
    Products.name;
```

--Question 13: Who placed the largest order?  
--Christopher Contreras

```
SELECT
    name, orders.*
FROM
    customers JOIN orders
    ON customers.id = orders.customer_id
WHERE
    order_total IS NOT NULL
ORDER BY
    order_total DESC
LIMIT 1
```

--Question 14: What is the gross margin for Widget 1.0.  
--The gross margin is calculated by dividing the total line\_total by the total quantity sold.  $\text{total \$\$} / \text{total quantity} = \text{gross margin}$

```
SELECT SUM(line_total)/SUM(quantity) FROM order_lines WHERE product_id = 1
--or:
SELECT
    Products.name,
    SUM(Order_lines.line_total)/SUM(Order_lines.quantity) AS "Gross Margin"
FROM
    Products JOIN Order_lines
    ON
    Products.id = Order_lines.product_id
WHERE
    Products.name = 'Widget 1.0'
GROUP BY
    Products.name;
```

--Question 15: How many orders did Gage Burke place?  
--2

```
SELECT
    order_date, customers.name
FROM
    orders JOIN customers
    ON orders.customer_id = customers.id
WHERE
```

```
        name
    ILIKE
        'Gage%'
```

--Question 16: How many products were sold for less than their price? Sell price is in the orders lines whereas the regular price in the products listing.

--Return the count of the order lines and not the sum of the quantity.

-- 4355

```
SELECT COUNT
    (ol.id)
FROM
    order_lines AS ol
    JOIN
    products AS p
    ON
    ol.product_id = p.id
WHERE
    ol.sell_price < p.price;
```

```
SELECT * FROM order_lines AS ol JOIN products AS p ON ol.product_id = p.id WHERE ol.sell_price <
p.price
```

--Question 17: What was the largest order total for Gage Burke?

--3142.96

```
SELECT
    order_total, order_date, customers.name
FROM
    orders JOIN customers
    ON
    orders.customer_id = customers.id
WHERE name ILIKE 'Gage%'
ORDER BY order_total
```

--Question 18: How many customer orders contain dohicky 2.5?

--2466

```
SELECT id, name FROM products WHERE name ILIKE 'dohicky 2.5'
```

```
SELECT COUNT(id) FROM order_lines WHERE product_id = 2
```

--or:

```
SELECT
    count(Orders.id)
FROM
    Orders JOIN Order_lines
    ON
    Orders.id = Order_lines.order_id
    JOIN Products
```

```
        ON
        Order_lines.product_id = Products.id
WHERE
        Products.name = 'Dohicky 2.5';
```

--Question 19: In what city was the largest order placed?  
--Valcourt

```
SELECT
        Customers.city,
        Orders.id
FROM
        Customers JOIN Orders
        ON
        Customers.id = Orders.customer_id
WHERE
        Orders.order_total = (SELECT MAX(order_total) FROM Orders);
```

--Question 20: How many customers have an address2?  
--327

```
SELECT COUNT(*) FROM customers WHERE address2 IS NOT NULL
```

--Question 21: How many customers are in the prairie provinces? The prairie provinces consist of Alberta (AB), Saskatchewan (SK) and Manitoba (MB).  
--367

```
SELECT COUNT(*) FROM customers WHERE province IN ( 'AB' , 'SK', 'MB')
```

--Question 22: What is the smallest order total (greater than zero) for Orders that have the Dohicky Master Edition 2.5 in it?  
--7.20

```
SELECT id, name FROM products WHERE name ILIKE 'Dohicky Master Edition 2.5'
```

```
SELECT
        MIN(NULLIF(order_total, 0)), order_lines.product_id
FROM
        orders JOIN order_lines
        ON orders.id = order_lines.order_id
WHERE
        product_id = 4
GROUP BY
        order_lines.product_id
```

--or:

```
SELECT
```

```

        MIN(Orders.order_total)
FROM
    Orders JOIN Order_lines
    ON
    Orders.id = Order_lines.order_id
    JOIN Products
    ON
    Order_lines.product_id = Products.id
WHERE
    Orders.order_total > 0
    AND
    Products.name = 'Dohicky Master Edition 2.5';

```

--Question 23: How many orders where placed for customers in the city of Ajax?  
--56

```

SELECT COUNT
    (customers.city)
FROM
    orders JOIN customers
    ON orders.customer_id = customers.id
WHERE
    city ILIKE 'Ajax'
GROUP BY
    customers.city;

```

--Question 24: How many customers do not have an order?  
--18

```

SELECT * FROM orders;

SELECT
    count(Customers.id)
FROM
    Customers LEFT OUTER JOIN Orders
    ON
    Customers.id = Orders.customer_id
WHERE
    Orders.customer_id IS NULL;

```

--Question 25: How many customers have a name that starts with 'Ginger'?  
--2

```

SELECT COUNT(*) FROM customers WHERE name ILIKE 'Ginger%'

```

--Question 26: How many of each of the following products were sold in the largest order? (largest order\_total)

--



```
SELECT
    *
FROM
    orders
WHERE order_total IS NOT NULL
ORDER BY order_total DESC
LIMIT 1
```

--or:

```
SELECT
    id
FROM
    Orders
WHERE
    order_total = (SELECT MAX(order_total) FROM Orders);
```

---

```
SELECT
    p.name, ol.quantity
FROM
    products AS p
    JOIN
    order_lines AS ol
    ON ol.product_id = p.id
WHERE
    ol.order_id = 926;
```

-or:

```
SELECT
    Products.name,
    SUM(Order_lines.quantity)
FROM
    Products JOIN Order_lines
    ON
    Products.id = Order_lines.product_id
WHERE
    Order_lines.order_id = 926
GROUP BY
    Products.name
ORDER BY
    Products.name;
```