- **Title:** DB Assignment 3
- **Your Name:** Ryan Farley
- **Date:** 11 Oct 24

Question 1

1. List names and sellers of products that are no longer available (quantity=0)

```
-- Question one
SELECT p.name AS product_name, m.name AS merchant_name, s.quantity_available
FROM products p
JOIN sell s ON p.pid = s.pid
JOIN merchants m ON s.mid = m.mid
WHERE s.quantity_available = 0;
```

product_name	merchant_name	quantity_available
Printer	Apple	0
Laptop	HP	0
Router	Apple	0
Ethernet Adapter	Lenovo	0
Router	HP	0
Router	Acer	0
Super Drive	HP	0
Router	Dell	0
Network Card	Acer	0

Quantity available wasn't technically required to be shown by the question but it was a good way of confirming. Selects names and sellers (and the quantity), joins them product id, and finds the targets by checking which ones have a quantity of zero.

Question 2

2. List names and descriptions of products that are not sold.

```
-- Question two
SELECT p.name, p.description
FROM products p
LEFT JOIN sell s ON p.pid = s.pid
WHERE s.pid IS NULL;
```

name	description
Super Drive	External CD/DVD/RW
Super Drive	UInternal CD/DVD/RW

Filters only the products that have no corresponding rows in the sell table (they are not sold

Question 3

1. How many customers bought SATA drives but not any routers?

```
-- Question three
SELECT COUNT(DISTINCT pl.cid) AS num_customers
FROM place pl
JOIN orders o ON pl.oid = o.oid
JOIN contain c ON o.oid = c.oid
JOIN products p ON c.pid = p.pid
WHERE p.category = 'SATA'
AND pl.cid NOT IN (
    SELECT DISTINCT pl2.cid
    FROM place pl2
    JOIN orders o2 ON pl2.oid = o2.oid
    JOIN contain c2 ON o2.oid = c2.oid
    JOIN products p2 ON c2.pid = p2.pid
    WHERE p2.category = 'Router'
```

num_customers
0

Things I hate: This query

Joins with product and order ids, looks for ‘sata’ purchases and lack of ‘router’. Thanks I hate it.

Question 4

HP has a 20% sale on all its Networking products

```sql
-- Question four
SELECT products.name AS product_name, sell.price AS original_price, (sell.price * 0.8) AS discounted_price
FROM sell
INNER JOIN products ON sell.pid = products.pid
WHERE sell.mid = (SELECT mid FROM merchants WHERE name = 'HP')
  AND products.category = 'Networking';
```

product_name	original_price	discounted_price
Network Card	1154.68	923.7440000000001
Network Card	345.01	276.008
Network Card	262.2	209.76
Ethernet Adapter	1260.45	1008.3600000000001
Router	205.56	164.448

Price is set to 80% of its normal amount. Checks for brand and category

Displays original and discount price.

Question 5

1. What did Uriel Whitney order from Acer? (make sure to at least retrieve product names and prices).

```sql
-- Question five
SELECT DISTINCT customers.fullname AS customer_name, products.name AS product_name, sell.price
FROM customers
INNER JOIN place ON customers.cid = place.cid
INNER JOIN orders ON place.oid = orders.oid
INNER JOIN contain ON orders.oid = contain.oid
INNER JOIN products ON contain.pid = products.pid
INNER JOIN sell ON products.pid = sell.pid
INNER JOIN merchants ON sell.mid = merchants.mid
WHERE customers.fullname = 'Uriel Whitney' AND merchants.name = 'Acer';
```

customer_name	product_name	price
Uriel Whitney	Network Card	130.43
Uriel Whitney	Super Drive	356.13
Uriel Whitney	Printer	310.83
Uriel Whitney	Printer	1345.37
Uriel Whitney	Hard Drive	836.99

Table is much bigger but couldnt fit

Matches the customer id, finds all the orders between her and acer and what they contain

Question 6

1. List the annual total sales for each company (sort the results along the company and the year attributes).

```
-- question six
SELECT YEAR(order_date) AS year, merchants.name AS company, SUM(sell.price * sell.quantity_available) AS total_revenue
FROM place
JOIN orders ON place.oid = orders.oid
JOIN contain ON place.oid = contain.oid
JOIN sell ON contain.pid = sell.pid
JOIN merchants ON sell.mid = merchants.mid
GROUP BY YEAR(order_date), merchants.name
ORDER BY YEAR(order_date) DESC;
```

I want to go to bed. Connect product ids order ids merchant ids, just every id at this point. This should work but my files have to be f'ed again because I keep getting null for years.

year	company	total_revenue
NULL	Acer	6072519.339999975
NULL	Apple	6643368.529999991
NULL	Dell	9825264.260000024
NULL	HP	5744609.259999982
NULL	Lenovo	8019973.509999987

Question 7

1. Which company had the highest annual revenue and in what year?

```
67  -- question seven
68  SELECT YEAR(order_date) AS year, merchants.name AS company, SUM(sell.price * sell.quantity_available) AS total_revenue
69  FROM place
70  JOIN orders ON place.oid = orders.oid
71  JOIN contain ON place.oid = contain.oid
72  JOIN sell ON contain.pid = sell.pid
73  JOIN merchants ON sell.mid = merchants.mid
74  GROUP BY YEAR(order_date), merchants.name
75  ORDER BY total_revenue DESC
76  LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

year	company	total_revenue
NULL	Dell	9825264.260000024

Dell is 100% right but my years are bugged

Question 8

1. On average, what was the cheapest shipping method used ever?

```sql
-- question eight
SELECT AVG(lowest_shipping_cost) AS average_lowest_shipping_cost
FROM (
    SELECT MIN(orders.shipping_cost) AS lowest_shipping_cost
    FROM orders
    GROUP BY orders.oid
) AS lowest_shipping_methods;
```

average_lowest_shipping_cost
7.603340000000003

Question 9

1. What is the best sold ($) category for each company?
2.

```sql
-- question nine
WITH totalSales AS (
    SELECT merchants.name AS company, products.category, SUM(sell.price * sell.quantity_available) AS total_sales
    FROM sell
    JOIN products ON sell.pid = products.pid
    JOIN merchants ON sell.mid = merchants.mid
    GROUP BY merchants.name, products.category
)
SELECT totalSales.company, totalSales.category, totalSales.total_sales
FROM totalSales
JOIN (
    SELECT company, MAX(total_sales) AS max_sales
    FROM totalSales
    GROUP BY company
) max_sales_per_company ON totalSales.company = max_sales_per_company.company AND totalSales.total_sales = max_sales_per_company.max_sales;
```

esult Grid | Filter Rows: | Export: | Wrap Cell Content:

company	category	total_sales
Lenovo	Peripheral	83479.82999999999
Dell	Peripheral	100753.95999999999
Apple	Peripheral	63974.73999999998
Acer	Peripheral	78136.53
HP	Peripheral	51133.469999999994

Peripherals clear

Question 10

1. For each company find out which customers have spent the most and the least amounts.

```sql
-- question 10
WITH customerleastmost AS (
    SELECT merchants.name AS company,
           customers.fullname AS customer_name,
           SUM(sell.price * sell.quantity_available) AS total_spent
    FROM place
    JOIN orders ON place.oid = orders.oid
    JOIN contain ON place.oid = contain.oid
    JOIN sell ON contain.pid = sell.pid
    JOIN merchants ON sell.mid = merchants.mid
    JOIN customers ON place.cid = customers.cid
    GROUP BY merchants.name, customers.fullname
)
SELECT customer1.company, customer1.customer_name, customer1.total_spent
FROM customerleastmost customer1
JOIN (
    SELECT company,
           MAX(total_spent) AS max_spent,
           MIN(total_spent) AS min_spent
    FROM customerleastmost
    GROUP BY company
) customer2
ON customer1.company = customer2.company AND (customer1.total_spent = customer2.max_spent OR customer1.total_spent = customer2.min_spent)
ORDER BY company, total_spent DESC;
```

2.

company	customer_name	total_spent
Acer	Dean Heath	443713.32000000007
Acer	Inez Long	190191.55999999994
Apple	Clementine Travis	497858.47999999975
Apple	Wynne Mckinney	193504.62999999998
Dell	Clementine Travis	741615.8399999999

company	customer_name	total_spent
Dell	Inez Long	259552.37000000008
HP	Clementine Travis	412323.26000000007
HP	Wynne Mckinney	168651.54
Lenovo	Haviva Stewart	536047.37
Lenovo	Inez Long	243477.23