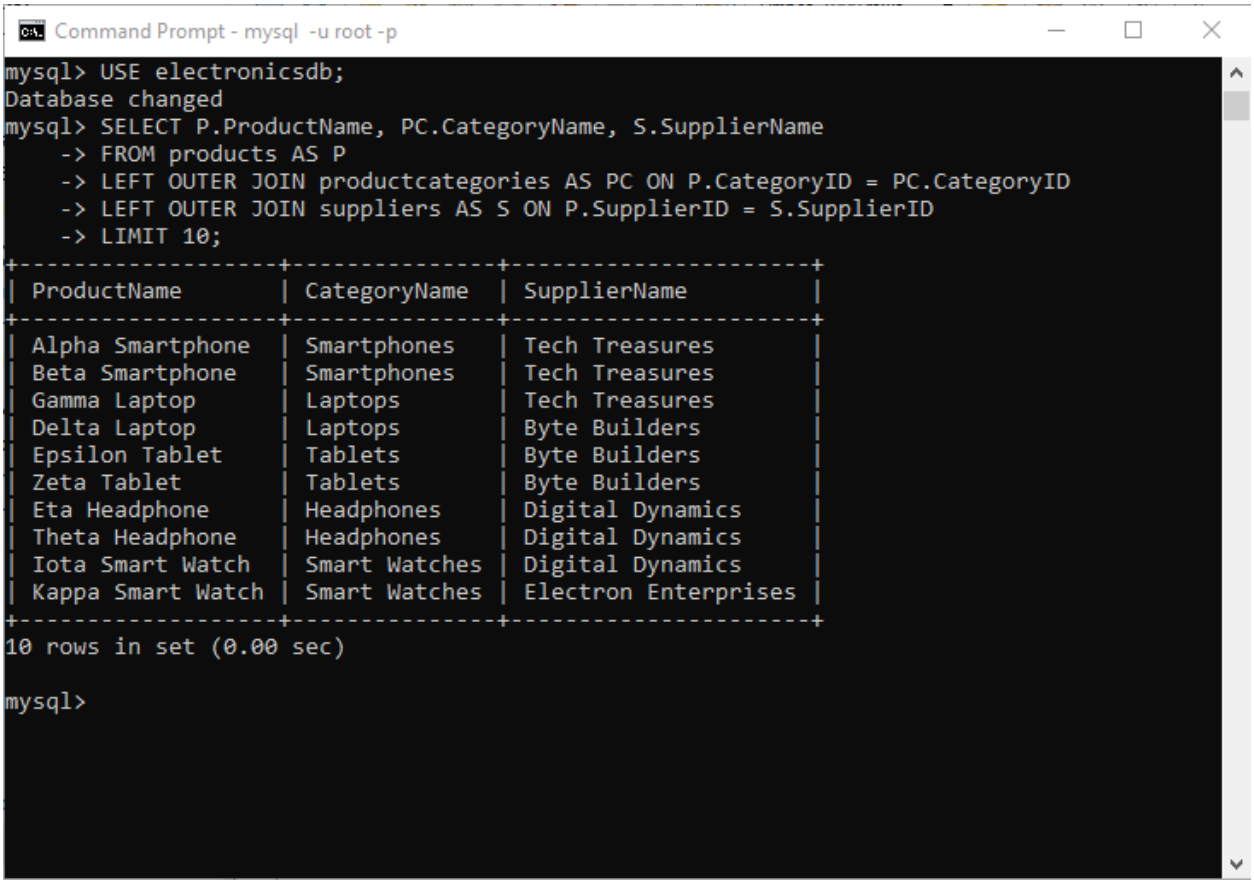
**Part II: [Total Points: 70]**
By Stephanie Boissonneault
**Notes**

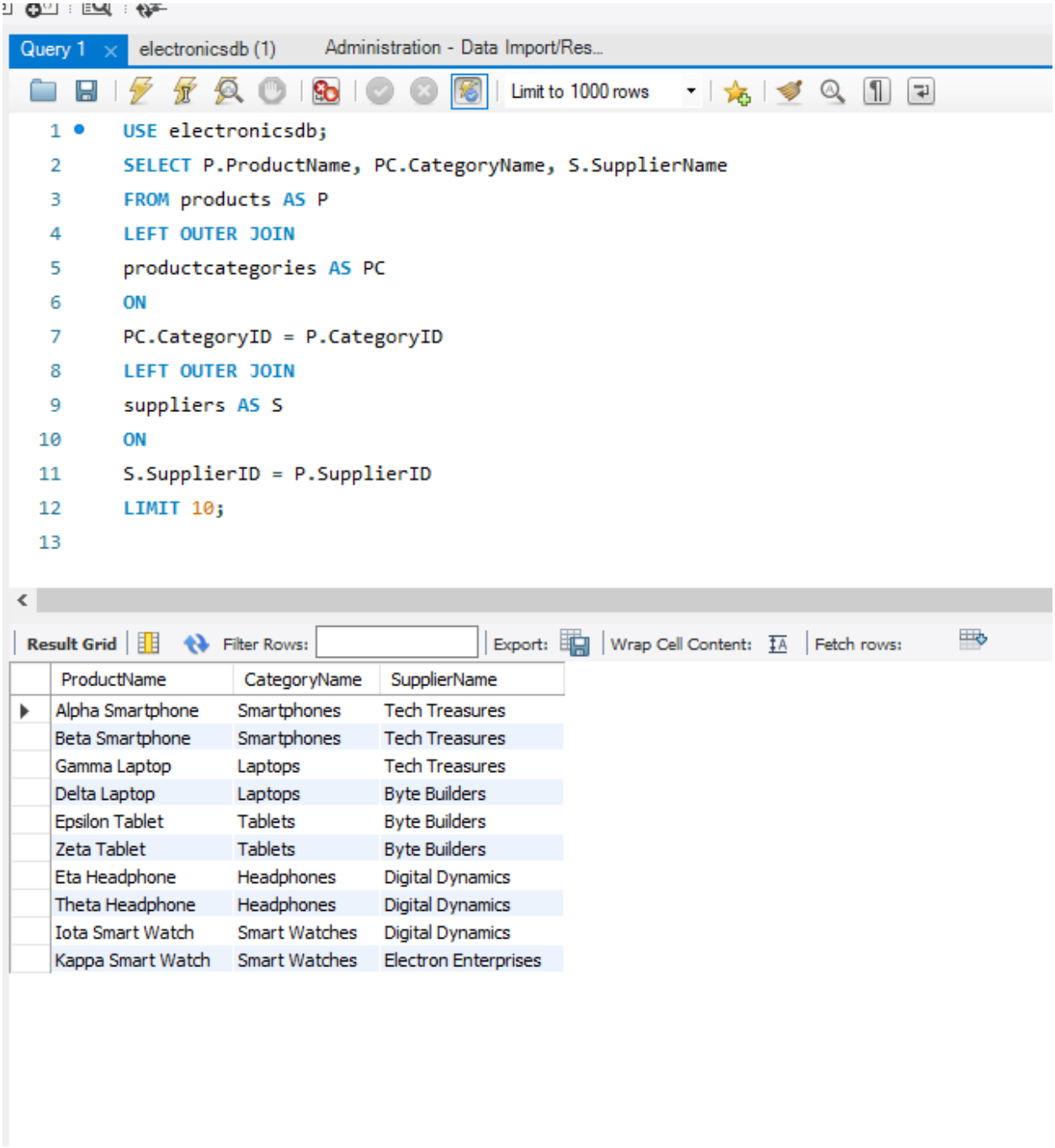
• For each of the following 7 questions: Write your answer as a MySQL query statement, execute it, and take a screenshot of the results. If the query result has more than 10 records, limit the output to the top 10 before taking the screenshot. Then, paste the answer and the respective screenshot into a Word or Google document file with the question number.

• **Submission:** Submit your answer sheet for this part either in PDF or DOCX file format.

**1. [8 pts.] Retrieve all products with their corresponding category names and supplier names.**

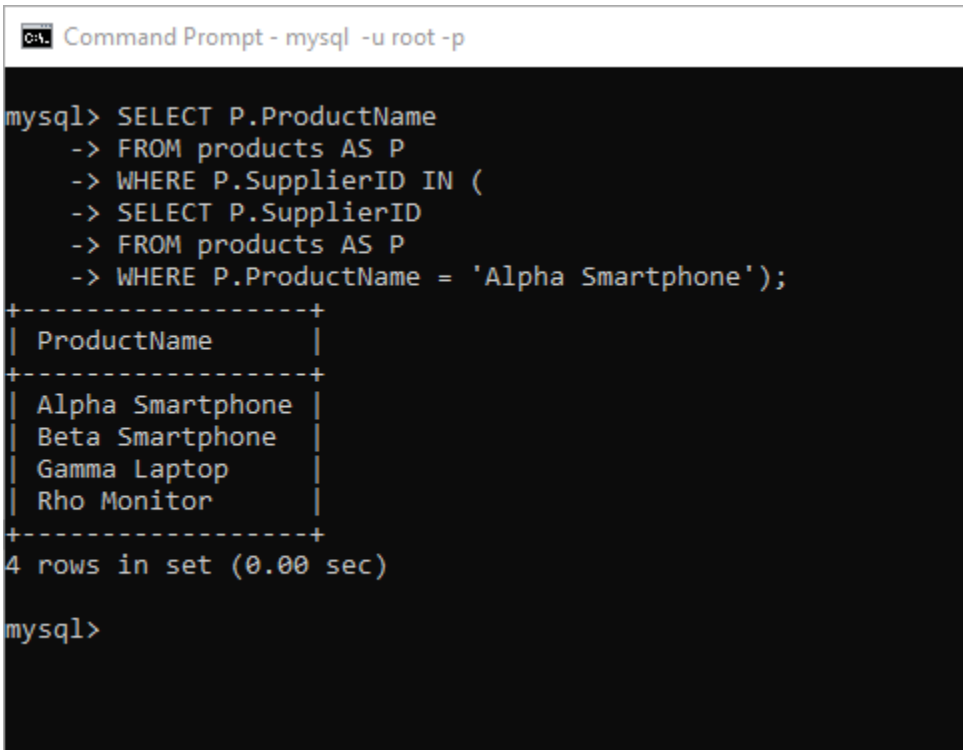
```
USE electronicsdb;
SELECT P.ProductName, PC.CategoryName, S.SupplierName
FROM products AS P
LEFT OUTER JOIN productcategories AS PC ON P.CategoryID = PC.CategoryID
LEFT OUTER JOIN suppliers AS S ON P.SupplierID = S.SupplierID
LIMIT 10;
```



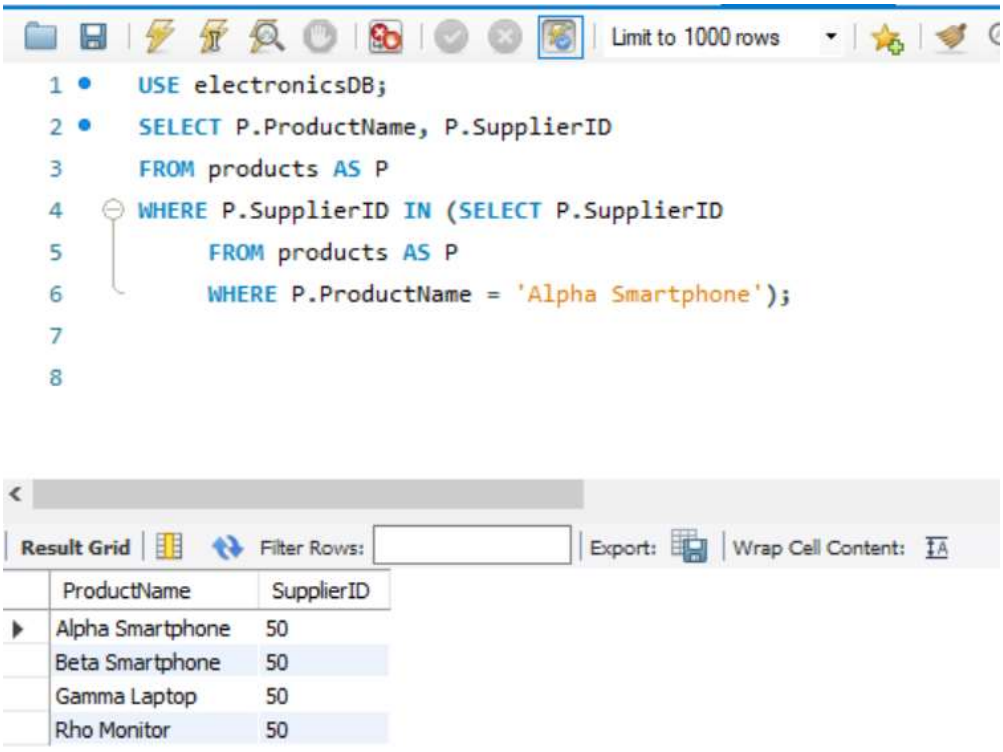



**2. [8 pts.] Retrieve the names of all products with the same supplier as 'Alpha Smartphone'.**

```
SELECT P.ProductName
FROM products AS P
WHERE P.SupplierID IN (SELECT P.SupplierID
                       FROM products AS P
                       WHERE P.ProductName = 'Alpha Smartphone');
```

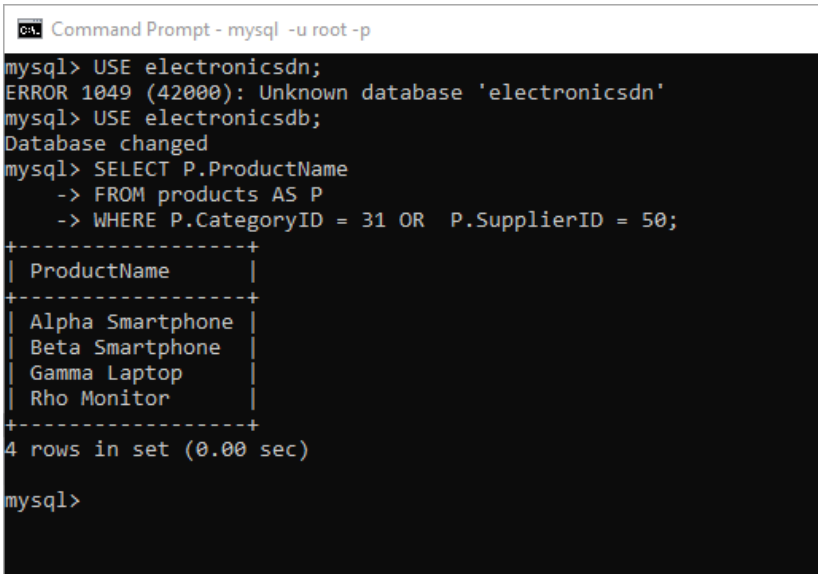



In the following screenshot, I have also selected to display the supplier ID to confirm that they are all the same supplier as Alpha Smartphone:

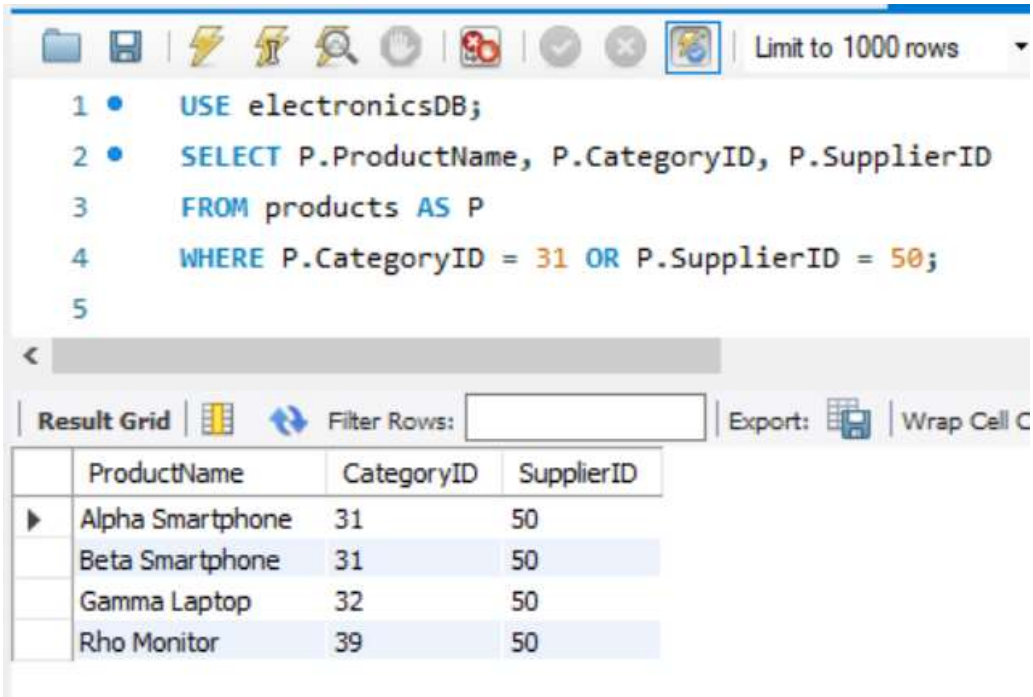


**3. [10 pts.] Retrieve all products that are either in the 'Smartphones' category (CategoryID 31) or supplied by SupplierID 50.**

SELECT P.ProductName, P.CategoryID, P.SupplierID
FROM products AS P
WHERE P.CategoryID = 31 OR  P.SupplierID = 50;

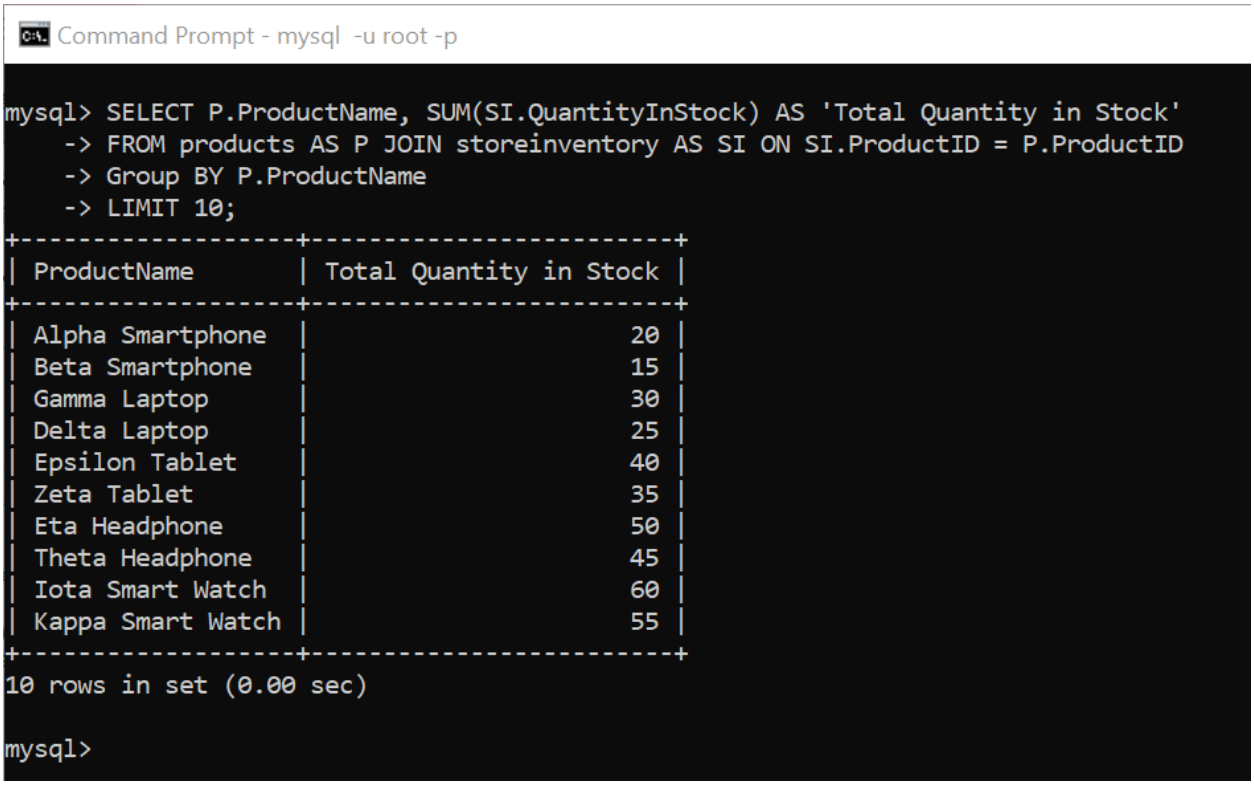



Here I have additionally selected to display the CategoryID and the SupplierID to confirm that retrievals are either (CategoryID 31) or SupplierID 50:



**4. [10 pts.] Find the total quantity in stock for each product.**

SELECT P.ProductName, SUM(SI.QuantityInStock) AS 'Total Quantity in Stock'
FROM products AS P JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
Group BY P.ProductName
LIMIT 10;



```
1  USE electronicsDB;
2  SELECT P.ProductName, SUM(SI.QuantityInStock) AS 'Total Quantity in Stock'
3  FROM products AS P JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
4  Group BY P.ProductName
5  LIMIT 10;
6
```

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

| ProductName | Total Quantity in Stock |
|---|---|
| Alpha Smartphone | 20 |
| Beta Smartphone | 15 |
| Gamma Laptop | 30 |
| Delta Laptop | 25 |
| Epsilon Tablet | 40 |
| Zeta Tablet | 35 |
| Eta Headphone | 50 |
| Theta Headphone | 45 |
| Iota Smart Watch | 60 |
| Kappa Smart Watch | 55 |

**5. [10 pts.] Calculate the average stock quantity for each product category and list the categories with an average stock above 70.**

SELECT PC.CategoryName, AVG(SI.QuantityInStock) AS 'Average Stock Quantity'
FROM productcategories AS PC JOIN products AS P ON P.CategoryID = PC.CategoryID
JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
Group BY PC.CategoryName
HAVING AVG(SI.QuantityInStock) > 70;

```
Command Prompt - mysql -u root -p
mysql>
mysql> SELECT PC.CategoryName, AVG(SI.QuantityInStock) AS 'Average Stock Quantity'
    -> FROM productcategories AS PC JOIN products AS P ON P.CategoryID = PC.CategoryID
    -> JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
    -> Group BY PC.CategoryName
    -> HAVING AVG(SI.QuantityInStock) > 70;
+--------------+------------------------+
| CategoryName | Average Stock Quantity |
+--------------+------------------------+
| Cameras      |               77.5000  |
| Printers     |               87.5000  |
| Monitors     |               97.5000  |
| Speakers     |              107.5000  |
+--------------+------------------------+
4 rows in set (0.00 sec)

mysql>
```

```
Limit to 1000 rows
1  USE electronicsDB;
2  SELECT PC.CategoryName, AVG(SI.QuantityInStock) AS 'Average Stock Quantity'
3  FROM productcategories AS PC JOIN products AS P ON P.CategoryID = PC.CategoryID
4  JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
5  Group BY PC.CategoryName
6  HAVING AVG(SI.QuantityInStock) > 70;
7
```

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

| CategoryName | Average Stock Quantity |
|---|---|
| Cameras | 77.5000 |
| Printers | 87.5000 |
| Monitors | 97.5000 |
| Speakers | 107.5000 |

**6. [12 pts.] Which products in the ’Cameras’ category are supplied by at least one supplier?**

SELECT P.ProductName, P.ProductID, P.SupplierID, COUNT(P.SupplierID) AS SupplierCount
FROM products AS P LEFT OUTER JOIN productcategories AS PC
ON PC.CategoryID = P.CategoryID
WHERE PC.CategoryName = 'Cameras'
GROUP BY P.ProductName, P.ProductID, P.SupplierID
HAVING COUNT(P.SupplierID) > 0;

Command Prompt - mysql -u root -p

```
mysql> USE electronicsDB;
Database changed
mysql> SELECT P.ProductName, P.ProductID, P.SupplierID, COUNT(P.SupplierID) AS SupplierCount
    -> FROM products AS P LEFT OUTER JOIN productcategories AS PC
    -> ON PC.CategoryID = P.CategoryID
    -> WHERE PC.CategoryName = 'Cameras'
    -> GROUP BY P.ProductName, P.ProductID, P.SupplierID
    -> HAVING COUNT(P.SupplierID) > 0;
+-------------+-----------+------------+---------------+
| ProductName | ProductID | SupplierID | SupplierCount |
+-------------+-----------+------------+---------------+
| Nu Camera   |        53 |         33 |             1 |
| Xi Camera   |        54 |         33 |             1 |
+-------------+-----------+------------+---------------+
2 rows in set (0.00 sec)

mysql>
```

```sql
USE electronicsDB;
SELECT P.ProductName, PC.CategoryName, P.SupplierID, COUNT(P.SupplierID) AS SupplierCount
FROM products AS P LEFT OUTER JOIN productcategories AS PC
ON PC.CategoryID = P.CategoryID
WHERE PC.CategoryName = 'Cameras'
GROUP BY P.ProductName, PC.CategoryName, P.SupplierID
HAVING COUNT(P.SupplierID) > 0;
```

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

| ProductName | CategoryName | SupplierID | SupplierCount |
|---|---|---|---|
| Nu Camera | Cameras | 33 | 1 |
| Xi Camera | Cameras | 33 | 1 |

**Could also check that they are supplied by at least one supplier by checking that the SupplierID is NOT NULL (implying that there is at least one supplier):**

SELECT P.ProductName, P.ProductID, PC.CategoryName, P.SupplierID
FROM products AS P
LEFT JOIN productcategories AS PC ON PC.CategoryID = P.CategoryID
WHERE PC.CategoryName = 'Cameras' AND P.SupplierID IS NOT NULL;

Command Prompt - mysql -u root -p

```
mysql> SELECT P.ProductName, P.ProductID, PC.CategoryName, P.SupplierID
    -> FROM products AS P
    -> LEFT JOIN productcategories AS PC ON PC.CategoryID = P.CategoryID
    -> WHERE PC.CategoryName = 'Cameras' AND P.SupplierID IS NOT NULL;
+-------------+-----------+--------------+------------+
| ProductName | ProductID | CategoryName | SupplierID |
+-------------+-----------+--------------+------------+
| Nu Camera   |        53 | Cameras      |         33 |
| Xi Camera   |        54 | Cameras      |         33 |
+-------------+-----------+--------------+------------+
2 rows in set (0.00 sec)

mysql>
```

```
24 •  SELECT P.ProductName, P.ProductID, PC.CategoryName, P.SupplierID
25    FROM products AS P
26    LEFT JOIN productcategories AS PC ON PC.CategoryID = P.CategoryID
27    WHERE PC.CategoryName = 'Cameras' AND P.SupplierID IS NOT NULL
28
29
```

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

| ProductName | ProductID | CategoryName | SupplierID |
|---|---|---|---|
| Nu Camera | 53 | Cameras | 33 |
| Xi Camera | 54 | Cameras | 33 |

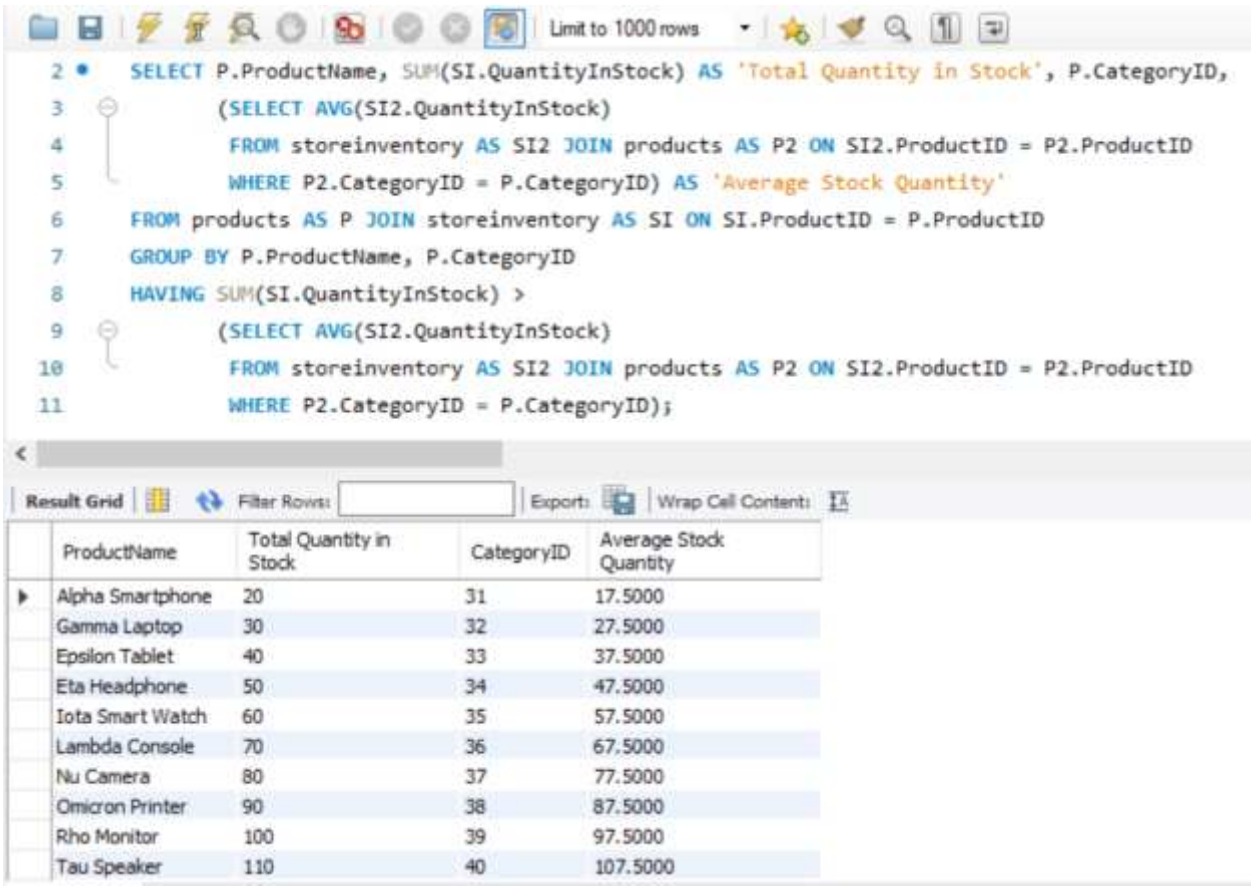
**7. [12 pts.] Find products with a total quantity in stock higher than the average stock quantity of all products in their respective category.**

```
SELECT P.ProductName, SUM(SI.QuantityInStock) AS 'Total Quantity in Stock', P.CategoryID,
            (SELECT AVG(SI2.QuantityInStock)
             FROM storeinventory AS SI2 JOIN products AS P2
             ON SI2.ProductID = P2.ProductID
             WHERE P2.CategoryID = P.CategoryID) AS 'Average Stock Quantity'
FROM products AS P JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
GROUP BY P.ProductName, P.CategoryID
HAVING SUM(SI.QuantityInStock) >
            (SELECT AVG(SI2.QuantityInStock)
             FROM storeinventory AS SI2 JOIN products AS P2
             ON SI2.ProductID = P2.ProductID
             WHERE P2.CategoryID = P.CategoryID);
```

Command Prompt - mysql -u root -p

```
mysql> SELECT P.ProductName, SUM(SI.QuantityInStock) AS 'Total Quantity in Stock', P.CategoryID,
    ->        (SELECT AVG(SI2.QuantityInStock)
    ->         FROM storeinventory AS SI2 JOIN products AS P2
    ->         ON SI2.ProductID = P2.ProductID
    ->         WHERE P2.CategoryID = P.CategoryID) AS 'Average Stock Quantity'
    -> FROM products AS P JOIN storeinventory AS SI ON SI.ProductID = P.ProductID
    -> GROUP BY P.ProductName, P.CategoryID
    -> HAVING SUM(SI.QuantityInStock) >
    ->        (SELECT AVG(SI2.QuantityInStock)
    ->         FROM storeinventory AS SI2 JOIN products AS P2
    ->         ON SI2.ProductID = P2.ProductID
    ->         WHERE P2.CategoryID = P.CategoryID);
+-----------------+-------------------------+------------+-----------------------+
| ProductName     | Total Quantity in Stock | CategoryID | Average Stock Quantity |
+-----------------+-------------------------+------------+-----------------------+
| Alpha Smartphone |                      20 |         31 |               17.5000 |
| Gamma Laptop    |                      30 |         32 |               27.5000 |
| Epsilon Tablet  |                      40 |         33 |               37.5000 |
| Eta Headphone   |                      50 |         34 |               47.5000 |
| Iota Smart Watch |                     60 |         35 |               57.5000 |
| Lambda Console  |                      70 |         36 |               67.5000 |
| Nu Camera       |                      80 |         37 |               77.5000 |
| Omicron Printer |                      90 |         38 |               87.5000 |
| Rho Monitor     |                     100 |         39 |               97.5000 |
| Tau Speaker     |                     110 |         40 |              107.5000 |
+-----------------+-------------------------+------------+-----------------------+
10 rows in set (0.00 sec)

mysql>
```




| ProductName | Total Quantity in Stock | CategoryID | Average Stock Quantity |
|---|---|---|---|
| Alpha Smartphone | 20 | 31 | 17.5000 |
| Gamma Laptop | 30 | 32 | 27.5000 |
| Epsilon Tablet | 40 | 33 | 37.5000 |
| Eta Headphone | 50 | 34 | 47.5000 |
| Iota Smart Watch | 60 | 35 | 57.5000 |
| Lambda Console | 70 | 36 | 67.5000 |
| Nu Camera | 80 | 37 | 77.5000 |
| Omicron Printer | 90 | 38 | 87.5000 |
| Rho Monitor | 100 | 39 | 97.5000 |
| Tau Speaker | 110 | 40 | 107.5000 |