**Solution: Subquery Mania**

1. Provide the **name** of the **sales\_rep** in each **region** with the largest amount of **total\_amt\_usd** sales.  
     
   First, I wanted to find the **total\_amt\_usd** totals associated with each **sales rep**, and I also wanted the region in which they were located. The query below provided this information.
2. **SELECT** s.**name** rep\_name, r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt
3. **FROM** sales\_reps s
4. **JOIN** accounts a
5. **ON** a.sales\_rep\_id = s.**id**
6. **JOIN** orders o
7. **ON** o.account\_id = a.**id**
8. **JOIN** region r
9. **ON** r.**id** = s.region\_id
10. **GROUP** **BY** 1,2
11. **ORDER** **BY** 3 **DESC**;

Next, I pulled the max for each region, and then we can use this to pull those rows in our final result.

**SELECT** region\_name, **MAX**(total\_amt) total\_amt

**FROM**(**SELECT** s.**name** rep\_name, r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt

**FROM** sales\_reps s

**JOIN** accounts a

**ON** a.sales\_rep\_id = s.**id**

**JOIN** orders o

**ON** o.account\_id = a.**id**

**JOIN** region r

**ON** r.**id** = s.region\_id

**GROUP** **BY** 1, 2) t1

**GROUP** **BY** 1;

Essentially, this is a **JOIN** of these two tables, where the region and amount match.

**SELECT** t3.rep\_name, t3.region\_name, t3.total\_amt

**FROM**(**SELECT** region\_name, **MAX**(total\_amt) total\_amt

**FROM**(**SELECT** s.**name** rep\_name, r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt

**FROM** sales\_reps s

**JOIN** accounts a

**ON** a.sales\_rep\_id = s.**id**

**JOIN** orders o

**ON** o.account\_id = a.**id**

**JOIN** region r

**ON** r.**id** = s.region\_id

**GROUP** **BY** 1, 2) t1

**GROUP** **BY** 1) t2

**JOIN** (**SELECT** s.**name** rep\_name, r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt

**FROM** sales\_reps s

**JOIN** accounts a

**ON** a.sales\_rep\_id = s.**id**

**JOIN** orders o

**ON** o.account\_id = a.**id**

**JOIN** region r

**ON** r.**id** = s.region\_id

**GROUP** **BY** 1,2

**ORDER** **BY** 3 **DESC**) t3

**ON** t3.region\_name = t2.region\_name **AND** t3.total\_amt = t2.total\_amt;

1. For the region with the largest sales **total\_amt\_usd**, how many **total** orders were placed?  
     
   The first query I wrote was to pull the **total\_amt\_usd** for each **region**.
2. **SELECT** r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt
3. **FROM** sales\_reps s
4. **JOIN** accounts a
5. **ON** a.sales\_rep\_id = s.**id**
6. **JOIN** orders o
7. **ON** o.account\_id = a.**id**
8. **JOIN** region r
9. **ON** r.**id** = s.region\_id
10. **GROUP** **BY** r.**name**;

Then we just want the region with the max amount from this table. There are two ways I considered getting this amount. One was to pull the max using a subquery. Another way is to order descending and just pull the top value.

**SELECT** **MAX**(total\_amt)

**FROM** (**SELECT** r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt

**FROM** sales\_reps s

**JOIN** accounts a

**ON** a.sales\_rep\_id = s.**id**

**JOIN** orders o

**ON** o.account\_id = a.**id**

**JOIN** region r

**ON** r.**id** = s.region\_id

**GROUP** **BY** r.**name**) sub;

Finally, we want to pull the total orders for the region with this amount:

**SELECT** r.**name**, **COUNT**(o.total) total\_orders

**FROM** sales\_reps s

**JOIN** accounts a

**ON** a.sales\_rep\_id = s.**id**

**JOIN** orders o

**ON** o.account\_id = a.**id**

**JOIN** region r

**ON** r.**id** = s.region\_id

**GROUP** **BY** r.**name**

**HAVING** **SUM**(o.total\_amt\_usd) = (

**SELECT** **MAX**(total\_amt)

**FROM** (**SELECT** r.**name** region\_name, **SUM**(o.total\_amt\_usd) total\_amt

**FROM** sales\_reps s

**JOIN** accounts a

**ON** a.sales\_rep\_id = s.**id**

**JOIN** orders o

**ON** o.account\_id = a.**id**

**JOIN** region r

**ON** r.**id** = s.region\_id

**GROUP** **BY** r.**name**) sub);

This provides the **Northeast** with **2357** orders.

1. **How many accounts** had more **total** purchases than the account **name** which has bought the most **standard\_qty** paper throughout their lifetime as a customer?  
     
   First, we want to find the account that had the most **standard\_qty** paper. The query here pulls that account, as well as the total amount:
2. **SELECT** a.**name** account\_name, **SUM**(o.standard\_qty) total\_std, **SUM**(o.total) total
3. **FROM** accounts a
4. **JOIN** orders o
5. **ON** o.account\_id = a.**id**
6. **GROUP** **BY** 1
7. **ORDER** **BY** 2 **DESC**
8. **LIMIT** 1;

Now, I want to use this to pull all the accounts with more total sales:

**SELECT** a.**name**

**FROM** orders o

**JOIN** accounts a

**ON** a.**id** = o.account\_id

**GROUP** **BY** 1

**HAVING** **SUM**(o.total) > (**SELECT** total

**FROM** (**SELECT** a.**name** act\_name, **SUM**(o.standard\_qty) tot\_std, **SUM**(o.total) total

**FROM** accounts a

**JOIN** orders o

**ON** o.account\_id = a.**id**

**GROUP** **BY** 1

**ORDER** **BY** 2 **DESC**

**LIMIT** 1) sub);

This is now a list of all the accounts with more total orders. We can get the count with just another simple subquery.

**SELECT** **COUNT**(\*)

**FROM** (**SELECT** a.**name**

**FROM** orders o

**JOIN** accounts a

**ON** a.**id** = o.account\_id

**GROUP** **BY** 1

**HAVING** **SUM**(o.total) > (**SELECT** total

**FROM** (**SELECT** a.**name** act\_name, **SUM**(o.standard\_qty) tot\_std, **SUM**(o.total) total

**FROM** accounts a

**JOIN** orders o

**ON** o.account\_id = a.**id**

**GROUP** **BY** 1

**ORDER** **BY** 2 **DESC**

**LIMIT** 1) inner\_tab)

) counter\_tab;

1. For the customer that spent the most (in total over their lifetime as a customer) **total\_amt\_usd**, how many **web\_events** did they have for each channel?  
     
   Here, we first want to pull the customer with the most spent in lifetime value.
2. **SELECT** a.**id**, a.**name**, **SUM**(o.total\_amt\_usd) tot\_spent
3. **FROM** orders o
4. **JOIN** accounts a
5. **ON** a.**id** = o.account\_id
6. **GROUP** **BY** a.**id**, a.**name**
7. **ORDER** **BY** 3 **DESC**
8. **LIMIT** 1;

Now, we want to look at the number of events on each channel this company had, which we can match with just the **id**.

**SELECT** a.**name**, w.channel, **COUNT**(\*)

**FROM** accounts a

**JOIN** web\_events w

**ON** a.**id** = w.account\_id **AND** a.**id** = (**SELECT** **id**

**FROM** (**SELECT** a.**id**, a.**name**, **SUM**(o.total\_amt\_usd) tot\_spent

**FROM** orders o

**JOIN** accounts a

**ON** a.**id** = o.account\_id

**GROUP** **BY** a.**id**, a.**name**

**ORDER** **BY** 3 **DESC**

**LIMIT** 1) inner\_table)

**GROUP** **BY** 1, 2

**ORDER** **BY** 3 **DESC**;

I added an **ORDER BY** for no real reason, and the account name to assure I was only pulling from one account.

1. What is the lifetime average amount spent in terms of **total\_amt\_usd** for the top 10 total spending **accounts**?  
     
   First, we just want to find the top 10 accounts in terms of highest **total\_amt\_usd**.
2. **SELECT** a.**id**, a.**name**, **SUM**(o.total\_amt\_usd) tot\_spent
3. **FROM** orders o
4. **JOIN** accounts a
5. **ON** a.**id** = o.account\_id
6. **GROUP** **BY** a.**id**, a.**name**
7. **ORDER** **BY** 3 **DESC**
8. **LIMIT** 10;

Now, we just want the average of these 10 amounts.

**SELECT** **AVG**(tot\_spent)

**FROM** (**SELECT** a.**id**, a.**name**, **SUM**(o.total\_amt\_usd) tot\_spent

**FROM** orders o

**JOIN** accounts a

**ON** a.**id** = o.account\_id

**GROUP** **BY** a.**id**, a.**name**

**ORDER** **BY** 3 **DESC**

**LIMIT** 10) temp;

1. What is the lifetime average amount spent in terms of **total\_amt\_usd**, including only the companies that spent more per order, on average, than the average of all orders.  
     
   First, we want to pull the average of all accounts in terms of **total\_amt\_usd**:
2. **SELECT** **AVG**(o.total\_amt\_usd) avg\_all
3. **FROM** orders o

Then, we want to only pull the accounts with more than this average amount.

**SELECT** o.account\_id, **AVG**(o.total\_amt\_usd)

**FROM** orders o

**GROUP** **BY** 1

**HAVING** **AVG**(o.total\_amt\_usd) > (**SELECT** **AVG**(o.total\_amt\_usd) avg\_all

**FROM** orders o);

Finally, we just want the average of these values.

**SELECT** **AVG**(avg\_amt)

**FROM** (**SELECT** o.account\_id, **AVG**(o.total\_amt\_usd) avg\_amt

**FROM** orders o

**GROUP** **BY** 1

**HAVING** **AVG**(o.total\_amt\_usd) > (**SELECT** **AVG**(o.total\_amt\_usd) avg\_all

**FROM** orders o)) temp\_table;

**Wow! That was intense. Nice job if you got these!**