

Aggregate Function

```
mysql> SELECT
->   EmailAddress,
->   LEFT(EmailAddress, INSTR(EmailAddress, "@")-1) As UserName
-> FROM Customers
-> LIMIT 10;
+-----+-----+
| EmailAddress | UserName |
+-----+-----+
| jon24@learnsector.com | jon24 |
| eugene10@learnsector.com | eugene10 |
| ruben35@learnsector.com | ruben35 |
| christy12@learnsector.com | christy12 |
| elizabeth5@learnsector.com | elizabeth5 |
| julio1@learnsector.com | julio1 |
| marco14@learnsector.com | marco14 |
| rob4@learnsector.com | rob4 |
| shannon38@learnsector.com | shannon38 |
| jacquelyn20@learnsector.com | jacquelyn20 |
+-----+-----+
10 rows in set (0.01 sec)
```

1. Calling of an Aggregate Functions.
2. Use of Group By to find each instance of the value.
3. Group BY + Order BY with Index.
4. Use of HAVING to apply filter.

Whenever you call aggregated column.
All the column next to aggregated columns
needs to be Grouped.

COUNT()

```
2062 rows in set (0.01 sec)

mysql> SELECT
->   COUNT(customerKey)
-> FROM Customers;
+-----+
| COUNT(customerKey) |
+-----+
|          2062 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT
->   COUNT(*)
-> FROM Customers;
+-----+
| COUNT(*) |
+-----+
|      2062 |
+-----+
1 row in set (0.01 sec)
```

```

mysql> SELECT
->   DISTINCT AnnualIncome
->   FROM Customers;
+-----+
| AnnualIncome |
+-----+
|      90000   |
|      60000   |
|      NULL    |
|      80000   |
|      70000   |
|     100000   |
|      30000   |
|      20000   |
|      40000   |
|      10000   |
|     160000   |
|     170000   |
|     130000   |
|     110000   |
|     120000   |
|     150000   |
|      50000   |
+-----+
17 rows in set (0.00 sec)

mysql> SELECT
->   COUNT( DISTINCT AnnualIncome)
->   FROM Customers;
+-----+
| COUNT( DISTINCT AnnualIncome) |
+-----+
|                  16           |
+-----+
1 row in set (0.01 sec)

```

```

SELECT
  DISTINCTROW(AnnualIncome)
FROM Customers;

SELECT
  DISTINCT AnnualIncome
FROM Customers;

SELECT
  COUNT(DISTINCT AnnualIncome)
FROM Customers;

```

DISTINCT Annual Income - 17 records
NULL - wouldn't be counted
So, your final Count is 16.

SUM [Aggregated Function] with Group BY

```

mysql> DESC territories;
+-----+-----+-----+-----+-----+-----+
| Field        | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SalesTerritoryKey | int    | YES  |     | NULL    |       |
| Region        | text   | YES  |     | NULL    |       |
| Country        | text   | YES  |     | NULL    |       |
| Continent      | text   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

mysql> SELECT * FROM territories;
+-----+-----+-----+-----+
| SalesTerritoryKey | Region        | Country        | Continent      |
+-----+-----+-----+-----+
|          1         | Northwest     | United States  | North America  |
|          2         | Northeast     | United States  | North America  |
|          3         | Central        | United States  | North America  |
|          4         | Southwest     | United States  | North America  |
|          5         | Southeast     | United States  | North America  |
|          6         | Canada        | Canada         | North America  |
|          7         | France        | France         | Europe         |
|          8         | Germany       | Germany        | Europe         |
|          9         | Australia     | Australia      | Pacific        |
|         10         | United Kingdom | United Kingdom | Europe         |
+-----+-----+-----+-----+
10 rows in set (0.01 sec)

```

```
mysql> SELECT * FROM returns LIMIT 10;
+-----+-----+-----+-----+
| ReturnDate | TerritoryKey | ProductKey | ReturnQuantity |
+-----+-----+-----+-----+
| 1/18/2015 | 9           | 312        | 1             |
| 1/18/2015 | 10          | 310        | 1             |
| 1/21/2015 | 8           | 346        | 1             |
| 1/22/2015 | 4           | 311        | 1             |
| 2/2/2015   | 6           | 312        | 1             |
| 2/15/2015  | 1           | 312        | 1             |
| 2/19/2015  | 9           | 311        | 1             |
| 2/24/2015  | 8           | 314        | 1             |
| 3/8/2015   | 8           | 350        | 1             |
| 3/13/2015  | 9           | 350        | 1             |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql> DESC returns;
+-----+-----+-----+-----+-----+-----+
| Field      | Type   | Null  | Key   | Default | Extra  |
+-----+-----+-----+-----+-----+-----+
| ReturnDate | text    | YES   |       | NULL    |        |
| TerritoryKey | int    | YES   |       | NULL    |        |
| ProductKey  | int    | YES   |       | NULL    |        |
| ReturnQuantity | int   | YES   |       | NULL    |        |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> SELECT
->   TerritoryKey,
->   SUM(ReturnQuantity) AS TotalReturn
->   FROM Returns;
ERROR 1140 (42000): In aggregated query without GROUP BY, expression #1 of SELECT list contains nonaggregated column 'sales_db>Returns
.TerritoryKey'; this is incompatible with sql_mode=only_full_group_by
m.l>
```

```
mysql> SELECT
->   SUM(ReturnQuantity) AS TotalReturn
->   FROM Returns;
+-----+
| TotalReturn |
+-----+
|      1828   |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT
->   COUNT(TerritoryKey) as TotalCount
->   FROM Returns;
+-----+
| TotalCount |
+-----+
|      1809   |
+-----+
1 row in set (0.00 sec)
```

TerritoryKey Total Return[SUM]

---	1828

upto 1809 records	

Any Column next to aggregated columns
needs to be fixed with GROUP BY.

```
mysql> SELECT
->   TerritoryKey,
->     SUM(ReturnQuantity) AS TotalReturn
->   FROM Returns
->   GROUP BY TerritoryKey;
+-----+-----+
| TerritoryKey | TotalReturn |
+-----+-----+
|         9    |      404 |
|        10    |      204 |
|         8    |      163 |
|         4    |      362 |
|         6    |      238 |
|         1    |      270 |
|         7    |      186 |
|         5    |       1 |
+-----+-----+
8 rows in set (0.01 sec)
```

-- SUM with GROUP BY
-- Which territory has high return orders.
SELECT
 TerritoryKey,
 SUM(ReturnQuantity) AS TotalReturn
FROM Returns
GROUP BY TerritoryKey;

-- On individual day how many returns appeared.

```
SELECT
  ReturnDate,
  SUM(ReturnQuantity) AS TotalReturn
FROM Returns
GROUP BY ReturnDate;
```

```
+-----+-----+
| ReturnDate | TotalReturn |
+-----+-----+
| 1/18/2015  |      2 |
| 1/21/2015  |      1 |
| 1/22/2015  |      1 |
| 2/2/2015   |      1 |
| 2/15/2015  |      1 |
| 2/19/2015  |      1 |
| 2/24/2015  |      1 |
| 3/8/2015   |      1 |
| 3/13/2015  |      1 |
| 3/14/2015  |      1 |
+-----+-----+
10 rows in set (0.00 sec)
```

```

mysql> SELECT
->   ReturnDate,
->     SUM(ReturnQuantity) AS TotalReturn
-> FROM Returns
-> GROUP BY ReturnDate
-> ORDER BY TotalReturn DESC
-> LIMIT 5;
+-----+-----+
| ReturnDate | TotalReturn |
+-----+-----+
| 6/7/2017    |      12 |
| 5/30/2017   |      11 |
| 3/1/2017    |      11 |
| 1/11/2017   |      11 |
| 2/24/2017   |      11 |
+-----+-----+
5 rows in set (0.01 sec)

```

```

mysql> SELECT
->   ReturnDate,
->     SUM(ReturnQuantity) AS TotalReturn
-> FROM Returns
-> GROUP BY 1
-> ORDER BY 2 DESC
-> LIMIT 5;
+-----+-----+
| ReturnDate | TotalReturn |
+-----+-----+
| 6/7/2017    |      12 |
| 5/30/2017   |      11 |
| 3/1/2017    |      11 |
| 1/11/2017   |      11 |
| 2/24/2017   |      11 |
+-----+-----+
5 rows in set (0.01 sec)

```

```

mysql> SELECT
->   ReturnDate,
->     SUM(ReturnQuantity) AS TotalReturn
-> FROM Returns
-> GROUP BY ReturnDate
-> ORDER BY TotalReturn DESC
-> LIMIT 10;
+-----+-----+
| ReturnDate | TotalReturn |
+-----+-----+
| 6/7/2017    |      12 |
| 5/30/2017   |      11 |
| 3/1/2017    |      11 |
| 1/11/2017   |      11 |
| 2/24/2017   |      11 |
| 6/28/2017   |      11 |
| 11/7/2016   |      10 |
| 8/30/2016   |      10 |
| 8/24/2016   |      10 |
| 9/26/2016   |      10 |
+-----+-----+
10 rows in set (0.01 sec)

```

```

mysql> SELECT ProductName, ProductPrice, ProductCost FROM Products LIMIT 10;
+-----+-----+-----+
| ProductName          | ProductPrice | ProductCost |
+-----+-----+-----+
| Sport-100 Helmet, Red | 34.99       | 13.0863    |
| Sport-100 Helmet, Black | 33.6442    | 12.0278    |
| Mountain Bike Socks, M | 9.5         | 3.3963     |
| Mountain Bike Socks, L | 9.5         | 3.3963     |
| Sport-100 Helmet, Blue | 33.6442    | 12.0278    |
| AWC Logo Cap          | 8.6442     | 5.7052     |
| Long-Sleeve Logo Jersey, S | 48.0673   | 31.7244    |
| Long-Sleeve Logo Jersey, M | 48.0673   | 31.7244    |
| Long-Sleeve Logo Jersey, L | 48.0673   | 31.7244    |
| Long-Sleeve Logo Jersey, XL | 48.0673   | 31.7244    |
+-----+-----+-----+
10 rows in set (0.02 sec)

```

Total Profit =
ProductPrice - ProductCost

Based on Unique ProductSubcategory, Can we find the Total Profit.

```
mysql> DESC `product-subcategories`;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ProductSubcategoryKey | int | YES |   | NULL |   |
| SubcategoryName | text | YES | P.K | NULL |   |
| ProductCategoryKey | int | YES |   | NULL |   |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> DESC products;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ProductKey | int | YES | F.K | NULL |   |
| ProductSubcategoryKey | int | YES |   | NULL |   |
| ProductSKU | text | YES |   | NULL |   |
| ProductName | text | YES |   | NULL |   |
| ModelName | text | YES |   | NULL |   |
| ProductDescription | text | YES |   | NULL |   |
| ProductColor | text | YES |   | NULL |   |
| ProductSize | text | YES |   | NULL |   |
| ProductStyle | text | YES |   | NULL |   |
| ProductCost | double | YES |   | NULL |   |
| ProductPrice | double | YES |   | NULL |   |
+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)
```

SELECT

```
ProductSubcategoryKey,
SUM(ProductPrice) AS TotalProductPrice,
SUM(ProductCost) AS TotalProductCost
FROM Products
GROUP BY ProductSubcategoryKey;
```

ProductSubcategoryKey	TotalProductPrice	TotalProductCost
31	102.27839999999999	37.1419
23	36.98000000000004	13.51719999999999
19	8.6442	5.7052
21	408.22920000000005	293.18679999999995
14	20825.8651	12036.2716
12	18035.296100000003	9482.0868
2	65774.4484	40130.4307
1	52384.29339999996	28998.836099999982
10	553.2	245.6209
11	261.22	115.9817
4	549.8964000000001	244.15410000000003
17	3093.01	1373.2964
22	449.93	177.5125
35	125	51.5625
34	25	10.3125
36	44.98	18.5543
33	93.97	38.76269999999995
24	224.9699999999997	92.80019999999999

Round(Value , 2)

```

mysql> SELECT
->   ProductSubcategoryKey,
->   ROUND(SUM(ProductPrice),2) AS TotalProductPrice,
->   ROUND(SUM(ProductCost),2) AS TotalProductCost
-> FROM Products
-> GROUP BY ProductSubcategoryKey;

```

ProductSubcategoryKey	TotalProductPrice	TotalProductCost
31	102.28	37.14
23	36.98	13.52
19	8.64	5.71
21	408.23	293.19
14	20825.87	12036.27
12	18035.3	9482.09
2	65774.45	40130.43
1	52384.29	28998.84
10	553.2	245.62
11	261.22	115.98
4	549.9	244.15
17	3093.01	1373.3
22	449.93	177.51
35	125	51.56
34	25	10.31
36	44.98	18.55
33	93.97	38.76
24	224.97	92.8
18	269.97	111.36
20	184.61	76.15
25	190.5	71.25
28	23.97	8.96

Total Profit

P.P - P.C

```

mysql> SELECT
->   ProductSubcategoryKey,
->   ROUND(SUM(ProductPrice),2) AS TotalProductPrice,
->   ROUND(SUM(ProductCost),2) AS TotalProductCost,
->   ROUND(SUM(ProductPrice) - SUM(ProductCost),2) AS TotalProfit
-> FROM Products
-> GROUP BY ProductSubcategoryKey
-> ORDER BY TotalProfit DESC
-> LIMIT 10;

```

ProductSubcategoryKey	TotalProductPrice	TotalProductCost	TotalProfit
2	65774.45	40130.43	25644.02
1	52384.29	28998.84	23385.46
3	31355.46	19490.55	11864.91
14	20825.87	12036.27	8789.59
12	18035.3	9482.09	8553.21
16	11365.48	6812.47	4553.01
17	3093.01	1373.3	1719.71
8	836.97	371.61	465.36
10	553.2	245.62	307.58
4	549.9	244.15	305.74

10 rows in set (0.00 sec)

```

mysql> DESC `Sales-2015`;

```

Field	Type	Null	Key	Default	Extra
OrderDate	text	YES		NULL	
StockDate	text	YES		NULL	
OrderNumber	text	YES		NULL	
ProductKey	int	YES		NULL	
CustomerKey	int	YES		NULL	
TerritoryKey	int	YES		NULL	
OrderLineItem	int	YES		NULL	
OrderQuantity	int	YES		NULL	

8 rows in set (0.01 sec)

SUM - Aggregation

F.K - Group BY

```

mysql> SELECT ProductName FROM Products WHERE productKey = 312;
+-----+
| ProductName      |
+-----+
| Road-150 Red, 48 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT
    --> ProductKey,
    --> SUM(OrderQuantity) AS total_quantity_sold
    --> FROM `sales-2015`
    --> GROUP BY 1
    --> ORDER BY 2 DESC
    --> LIMIT 10;
+-----+-----+
| ProductKey | total_quantity_sold |
+-----+-----+
|      312   |          179 |
|      310   |          169 |
|      313   |          168 |
|      314   |          157 |
|      311   |          139 |
|      377   |          108 |
|      371   |          106 |
|      369   |           90 |
|      375   |           83 |
|      370   |           80 |
+-----+-----+
10 rows in set (0.01 sec)

```

```

mysql> SELECT
    --> SUM(OrderQuantity) AS totalOrderQty
    --> FROM `sales-2015`;
+-----+
| totalOrderQty |
+-----+
|        2630 |
+-----+
1 row in set (0.01 sec)

```

AVG - Function

```

mysql> SELECT
    --> ProductSubcategoryKey,
    --> ROUND(AVG(ProductPrice),2) AS AvgProductPrice,
    --> ROUND(AVG(ProductCost),2) AS AvgProductCost,
    --> ROUND(AVG(ProductPrice) - AVG(ProductCost),2) AS AverageProfit
    --> FROM Products
    --> GROUP BY 1
    --> ORDER BY 4 DESC;
+-----+-----+-----+-----+
| ProductSubcategoryKey | AvgProductPrice | AvgProductCost | AverageProfit |
+-----+-----+-----+-----+
|          1            |     1637.01   |      906.21   |       730.8   |
|          2            |     1529.64   |      933.27   |       596.37  |
|          3            |     1425.25   |      885.93   |       539.31  |
|         12            |      644.12   |      338.65   |       305.47  |
|         14            |      671.8    |      388.27   |       283.54  |
|         16            |      631.42   |      378.47   |       252.95  |
|          8            |      278.99   |      123.87   |       155.12  |
|         17            |      220.93   |       98.09   |       122.84  |
|         10            |      184.4    |      81.87   |       102.53  |
|         27            |       159     |      59.47   |       99.53  |
|         26            |       120     |      44.88   |       75.12  |
|         35            |       125     |      51.56   |       73.44  |
|          6            |      106.5    |      47.29   |       59.21  |
|          9            |      106.48   |      47.27   |       59.2    |
|         18            |      89.99    |      37.12   |       52.87  |
|          5            |      92.24    |      40.95   |       51.29  |
|         11            |      87.07    |      38.66   |       48.41  |
|         24            |      74.99    |      30.93   |       44.06  |
+-----+-----+-----+-----+

```

```

SELECT
    ProductSubcategoryKey,
    ROUND(AVG(ProductPrice),2) AS AvgProductPrice,
    ROUND(AVG(ProductCost),2) AS AvgProductCost,
    ROUND(AVG(ProductPrice) - AVG(ProductCost),2) AS AverageProfit
FROM Products
GROUP BY 1
ORDER BY 4 DESC;

```

```

mysql> SELECT DISTINCT Occupation FROM Customers;
+-----+
| Occupation |
+-----+
| Professional |
| Management |
| Skilled Manual |
| Clerical |
| Manual |
+-----+
5 rows in set (0.01 sec)

```

```

mysql> SELECT
    --> Occupation,
    -->     ROUND(AVG(AnnualIncome),0) AS AvgIncome
    --> FROM Customers
    --> GROUP BY 1
    --> ORDER BY 2 DESC;
+-----+-----+
| Occupation | AvgIncome |
+-----+-----+
| Management | 98567 |
| Professional | 74830 |
| Skilled Manual | 51620 |
| Clerical | 31862 |
| Manual | 16093 |
+-----+-----+
5 rows in set (0.01 sec)

```

```

mysql> SELECT
    --> Occupation,
    -->     ROUND(AVG(AnnualIncome),0) AS AvgIncome,
    -->     COUNT(*) AS TotalPerson
    --> FROM Customers
    --> GROUP BY 1
    --> ORDER BY 2 DESC;
+-----+-----+-----+
| Occupation | AvgIncome | TotalPerson |
+-----+-----+-----+
| Management | 98567 | 330 |
| Professional | 74830 | 561 |
| Skilled Manual | 51620 | 540 |
| Clerical | 31862 | 350 |
| Manual | 16093 | 281 |
+-----+-----+-----+
5 rows in set (0.01 sec)

```

```

mysql> SELECT
    ->   EducationLevel,
    ->   ROUND(AVG(AnnualIncome),0) AS AvgIncome,
    ->   COUNT(*) AS TotalPerson
    -> FROM Customers
    -> GROUP BY 1
    -> ORDER BY 2 DESC;
+-----+-----+-----+
| EducationLevel | AvgIncome | TotalPerson |
+-----+-----+-----+
| Graduate Degree | 67727 | 418 |
| Bachelors | 62707 | 595 |
| Partial College | 53707 | 585 |
| High School | 47243 | 342 |
| Partial High School | 39836 | 122 |
+-----+-----+-----+
5 rows in set (0.01 sec)

```

```

mysql> SELECT
    ->   Occupation,
    ->   EducationLevel,
    ->   ROUND(AVG(AnnualIncome),0) AS AvgIncome,
    ->   COUNT(*) AS TotalPerson
    -> FROM Customers
    -> GROUP BY 1,2
    -> ORDER BY 3 DESC;
+-----+-----+-----+-----+
| Occupation | EducationLevel | AvgIncome | TotalPerson |
+-----+-----+-----+-----+
| Management | Partial High School | 143333 | 3 |
| Management | Partial College | 131111 | 9 |
| Professional | Partial High School | 114545 | 11 |
| Management | High School | 107188 | 32 |
| Management | Graduate Degree | 99593 | 123 |
| Management | Bachelors | 93416 | 163 |
| Professional | Partial College | 86645 | 156 |
| Professional | High School | 76709 | 79 |
| Skilled Manual | Partial High School | 70000 | 23 |
| Professional | Graduate Degree | 67479 | 119 |
| Professional | Bachelors | 66923 | 196 |
| Skilled Manual | Partial College | 58734 | 159 |
| Skilled Manual | Graduate Degree | 56415 | 106 |
| Skilled Manual | Bachelors | 47612 | 135 |
| Skilled Manual | High School | 38534 | 117 |
| Clerical | Partial College | 35422 | 167 |
| Clerical | Graduate Degree | 31270 | 63 |
| Clerical | High School | 30000 | 6 |
| Clerical | Bachelors | 27955 | 88 |
| Clerical | Partial High School | 24231 | 26 |
| Manual | High School | 18241 | 108 |
| Manual | Partial High School | 15763 | 59 |
| Manual | Partial College | 15000 | 94 |
| Manual | Graduate Degree | 11429 | 7 |
| Manual | Bachelors | 10000 | 13 |
+-----+-----+-----+-----+
25 rows in set (0.01 sec)

```

```
-- FIND the average AnnualIncome of different Occupation.
```

```
SELECT
```

```
    Occupation,  
    ROUND(AVG(AnnualIncome),0) AS AvgIncome,  
    COUNT(*) AS TotalPerson
```

```
FROM Customers
```

```
GROUP BY 1
```

```
ORDER BY 2 DESC;
```

```
-- FIND the average AnnualIncome of different EducationLevel.
```

```
SELECT
```

```
    EducationLevel,  
    ROUND(AVG(AnnualIncome),0) AS AvgIncome,  
    COUNT(*) AS TotalPerson
```

```
FROM Customers
```

```
GROUP BY 1
```

```
ORDER BY 2 DESC;
```

```
-- FIND the average AnnualIncome of different Occupation & EducationLevel.
```

```
SELECT
```

```
    Occupation,  
    EducationLevel,  
    ROUND(AVG(AnnualIncome),0) AS AvgIncome,  
    COUNT(*) AS TotalPerson
```

```
FROM Customers
```

```
GROUP BY 1,2
```

```
ORDER BY 3 DESC;
```