

Aggregate Operators

There are several aggregate operators available for use - COUNT(), AVG(), MAX(), MIN(). These aggregate operators cannot be used with non-aggregate operators without a GROUP BY clause. Below find an example:

```
mysql> SELECT S.SName, MAX(S.NumCredits)
      -> FROM Students S;
ERROR 1140 (42000): Mixing of GROUP columns (MIN(),MAX(),COUNT(),...) with no GROUP
columns is illegal if there is no GROUP BY clause
```

```
mysql> SELECT MAX(S.NumCredits) FROM Students S;
+-----+
| MAX(S.NumCredits) |
+-----+
|                72 |
+-----+
1 row in set (0.00 sec)
```

You can, however use them with the GROUP BY as long as the GROUP BY clause contains all the fields in the SELECT:

```
mysql> SELECT E.ClassID, COUNT(*) AS NumEnrolled
      -> FROM Enrolled E
      -> GROUP BY ClassID;
+-----+-----+
| ClassID | NumEnrolled |
+-----+-----+
| CS314   |            2 |
| CS430   |            1 |
| CS575   |            2 |
| CT320   |            1 |
+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> select * from Enrolled;
+----+-----+-----+
| SID | ClassID | Grade |
+----+-----+-----+
|  1  | CS314   | NULL  |
|  1  | CS575   | NULL  |
|  2  | CS430   | NULL  |
|  2  | CS575   | NULL  |
|  3  | CS314   | NULL  |
|  3  | CT320   | NULL  |
+----+-----+-----+
6 rows in set (0.00 sec)
```

More on this as we get to the GROUP BY functionality. The primary reason you cannot mix aggregate and non-aggregate is they don't make sense together. Non-aggregate produces a row-by-row answer, the aggregate is a single number. Pairing the two produces two different contexts.

What if we want to answer the following : Give me the student with the most number of credits? Why wouldn't we use:

```
SELECT S.SName, MAX(S.NumCredits) FROM Students S;
```

The first part of the SELECT asks for a row by row list of Students names, the second asks for a single number. This is not the same as asking for the student with the maximum number of credits. You might think that moving it to the WHERE clause will work such as :

```
mysql> SELECT S.SName, S.NumCredits
-> FROM Students S
-> WHERE S.NumCredits = MAX (S.NumCredits);
ERROR 1305 (42000): FUNCTION cs430.MAX does not exist
```

Nope. The reason has to do with the order in which the SELECT statement is evaluated and when a number (such as MAX) is available. More on this when we talk about the GROUP BY functionality.

The correct way is:

```
mysql> SELECT S.SName, S.NumCredits
-> FROM Students S
-> WHERE S.NumCredits =
-> (SELECT MAX(S1.NumCredits) FROM Students S1);
```

```
+-----+-----+
| SName          | NumCredits |
+-----+-----+
| Alice Wonderland |          72 |
+-----+-----+
1 row in set (0.00 sec)
```

In this manner, MAX() is created in a separate SELECT statement (with it's own order of operations) before it can be used in a comparison.