**IS 441 Exam 2 “Review Class” (Sort of), November 17, 2016**

*Dr. Yüe “Jeff” Zhang*

Part “0” – From Fall 2016 Quiz 2

Common error #1: “City is not in SELECT, so it cannot be in GROUP BY”

“Verdict” – Wrong: there is no such a rule/law.

Event A 🡪 Event B does NOT mean “Event B 🡪 Event A”

|  |  |  |
| --- | --- | --- |
| **Right** | **Wrong** | **Remark** |
| (When there is GROUP BY) The fields in SELECT-clause MUST be either of the following:  1) A field that is GROUP BY; OR  2) A field that is in an aggregate function in SELECT-clause | The field in GROUP BY  “must be”  a field in SELECT-clause | The correct one says  - “You can’t be in SELECT unless you’re GROUP BY or you’re in aggregate function”;  It does NOT say the opposite:  - “You can’t be in GROUP BY unless you’re in SELECT” |
| Event A 🡪 Event B:  You’re in GROUP BY  so you can be in SELECT | Event B 🡪 Event A: You are in SELECT so you can be GROUP BY | A🡪B is right,  does NOT mean  B🡪 A is right!! |

Common error #2: WHERE COUNT(…) >= … 🡺 many left it go without identifying that being wrong

WHERE is the condition for ROWS (values of individual fields/attributes), while

COUNT(xxx) is an aggregate function, whose roles in Boolean operation as a condition is with GROUPs.

So, WHERE COUNT/AVG/SUM >/ = / < is wrong.

Common error #3: SELECT COUNT(ID), AVG(Sales), SquareFootage …

We learned on Day 1 of SQL that “ row values and set values cannot be in one SELECT-clause”; OK, here SquareFootage is a row value, listed with COUNT/AVG in the SELECT-clause. Many failed to identify this.

Common error #4: SELECT COUNT(ID), AVG(Sales), SquareFootage FROM Restaurants

Many WRONGLY identify this as an “error”.

No: it is CORRECT, That is NO RULE that FROM must be in its own line.

In fact, the whole SELECT statement can be in a LONG line, ended with the “;”.

The convention of writing each clause in a separate line is ONLY for readability:

SELECT AVG(Sales) FROM Restaurants WHERE Sales >100000 GROUP BY City ORDER BY AVG(Sales)

Is perfectly legitimate.

SO much we learned from the mistakes of Quiz 2.

Part I: SQL Codes

1. List employee name, project, date hired; sort the results first by project, then by date hire, earliest first

SELECT emp\_lname, emp\_hiredate, proj\_num FROM employee

ORDER BY proj\_num, emp\_hiredate

Note: Watch the primary sort and secondary sort

1. List job codes, and the maximum bonus percentage in each job code. Name the column of max bonus Max Bonus.

Note the 2-word column header in brackets

SELECT job\_code, max(emp\_pct) AS [Max Bonus] FROM employee GROUP BY job\_code;

1. List projects whose average bonus percentage is higher than the average percentage of all records; with the overall avg displayed.

SELECT proj\_num, AVG(emp\_pct) AS AvgProjPct, (SELECT AVG(emp\_pct) FROM employee) AS OverallAVG FROM employee

GROUP BY proj\_num

HAVING AVG(emp\_pct)> (SELECT AVG(emp\_pct) FROM employee);

Note 1: Subquery

Note 2: column alias

Note 3: Although we defined an alias “OverallAVG” for the average over the whole company, when the value is to be used in part of SQL logic (HAVING in this question), we can only (and MUST) use the expression or the subquery (in this question):

HAVING AVG(emp\_pct)> (SELECT AVG(emp\_pct) FROM employee)

Rather than

Used for display; not in logic or calculation

HAVING AVG(emp\_pct)>  OverallAVG

Part II: Normalization

1. Solved the example in  [Normalization steps and demo questions](http://www.csun.edu/~yz73352/441/441-ppt/Normalization-steps-and-demo-V2.doc) (download the document from the class web to study).
2. Please study the tutorial “**Normalization rules/tips - NEW VERSION!! (Version 6)**”.
3. Use the following diagrams to test your understanding of basic concepts re normalization – closely examine each of the following diagrams, and tell which one is right and which one is wrong. Give reasons of your judgment (we have shown all these in class. The following serves to revive your memory). [[ Please note: in the following examples, PK is NOT necessarily the 1st field ]]

🡺In each diagram, let’s call them A, B, C from top to bottom; and 1, 2, 3, 4 from left to right🡸

|  |  |
| --- | --- |
|  | What can we say about the relationship between A and B?  Who is…  FK is in… |
|  | Relationship of A and B?  Of B and C?  PKs and FKs (identify with the number of fields (A1, B2, C3 etc) |
|  | Relationship of A and B?  Of B and C?  PKs and FKs (identify with the number of fields (A1, B2, C3 etc) |
|  | Relationship of A and B? |
|  | Relationship of A and B? |

Compare: