Interactive - 15 - select with group data of data

SQL has the ability to group data. When we use the count() function we are grouping by data. You can only group by data that you select - you can not group by hidden columns. However you can use a nested select to pick just some of the columns after you have grouped. The last part of this homework will show this. First we have to group the data.

```
SELECT
        fed_area,
        min(gdp_growth) as min_growth,
        avg(gdp_growth) as avg_growth,
        max(gdp_growth) as max_growth,
        string_agg(state, ',' ORDER BY state) as  state_list
    FROM us_state
    GROUP BY fed_area
    ORDER BY fed_area
;
```

You can group by more than one column. All the non-grouped columns have to be in some form aggregated. In this example we have min, max, avg and string_agg.

To group columns the database has to sort them. So an order by is usually free. If you group by a column that there is a b-tree index on the group by is much faster. The database will use the index instead of re-sorting the data. Other index types do not provide sorted data (gin, gist, hash etc).

Let's say we just want the avg_growth and the list of states. We have to have fed_area to do the grouping - but we can nest our select.

```
SELECT t1.avg_growth, t1.state_list
    FROM (
        SELECT
                t2.fed_area,
                avg(t2.gdp_growth) as avg_growth,
                string_agg(t2.state, ',' ORDER BY state) as  state_list
            FROM us_state as t2
            GROUP BY t2.fed_area
    ) as t1
;
```

Take Away

1. group by
2. order by
3. min/avg/max

4. nested selects in FROM
5. Aliasing tables int the “FROM” with “as”

Book

- group by - page 120 to 121
- min, max - page 119
- avg - 64, 195
- order by - page 15, 16
- alias table name - page 86
- nested query in From - page 195 (avg used also)

Tags: “min”,“max”,“avg”,“group by”,“order by”,“nested query”,“sub query”

Validate: SQL-Select,“select ‘PASS’ as x”