

Notes for Format:

1. Please follow the instructions mentioned in Canvas “Project Phase 2”. Important parts are captured below:

All inline SQL queries should be written to retrieve the exact set of results from the database. "SELECT *" is bad practice since you are relying on application logic to parse out the result set and not writing targeted queries, which is what we want you to learn. You should also avoid utilizing temp tables, as in most instances a subquery or table expression would also suffice and require less overhead. For any reports that you create, write a single query that outputs the report whenever possible. Do not run separate queries and then join the results in code – you should let the database do all the heavy lifting of data. We may also validate your abstract code SQL to see that it runs successfully against your SQL schema. If you decide to use functions or features specific to certain SQL dialect that we may not be familiar with, please be sure to let us know via a footnote in your abstract code. **You are not required to create indices in this phase, and please do not create views or stored procedures yet (you may use these in Phase 3).**

Be careful when copying/pasting/typing SQL in different editors, as they may hinder instead of help you, such as "smart quotes (' or ') instead of an apostrophe ('). Depending on the SQL dialect, you may need to escape certain characters in order for your queries to run.

Make sure to have a semicolon (;) at the end of every query.

2. Please follow the instructions in the sample submission “team001_p2_ac+SQL.pdf”.
3. One can update the “team058_p2_ac+SQL_draft.docx” via the way mentioned on next page by adding TextBox like the sample below.

```
SELECT first_name, last_name, gender, birthdate, current_city, home_town
FROM `User`
INNER JOIN RegularUser ON `User`.email=RegularUser.email
WHERE `User`.email='$UserID';
```

3.1 Any existing comments in the docx file can be ignored. Any new changes suggested by TA can be handled by the person taking charge of that part.

3.2 The best way to add TextBox might be copying this sample box (in section “View Product by Category Report”) directly since it should contain all the formatting.

3.3 Keep the keyword UPPERCASE, e.g. SELECT

3.4 Change to a new line for e.g. SELECT, FROM, any JOIN, WHERE, GROUP BY, HAVING, ORDER BY, as shown in the above sample

3.5 The naming convention doesn't need to be the same as that in the TextBox, please use the same as that in the “team058_p2_schema_v2.sql” and AC we have

3.6 Keep the TextBox at the same level (indentation) of the AC you want to show

3.7 Some important formatting in 3.2 includes:

Font Style and Size in TextBox are the same as that in the corresponding AC

Spacing: Before - 0 pt, After - 0 pt

Line Spacing: Multiple At 1.25

Shape Border Weight: 1 pt

Wrap Text: Top and Bottom

How to change “team058_p2_ac+SQL_draft.docx” in local and push in Git without overwriting others’ work?

1. Before making any changes, git pull
2. Copy the docx and rename it to any other name (e.g. xxx_copy.docx) you want
3. Make any changes to “xxx_copy.docx”
4. After finishing the change to “xxx_copy.docx”, git pull
5. Check if the modified time for “team058_p2_ac+SQL_draft.docx” is just now:
 - 5.1 If not, then no one has updated the file since your last pull, it's safe to delete “team058_p2_ac+SQL_draft.docx” and rename “xxx_copy.docx” to “team058_p2_ac+SQL_draft.docx”, then git add, commit, pull, and push
 - 5.2 If so, then copy your changes from “xxx_copy.docx” to “team058_p2_ac+SQL_draft.docx”, save “team058_p2_ac+SQL_draft.docx”, delete “xxx_copy.docx” or move it to some other folder, then git add, commit, pull, and push
6. Check and make sure “team058_p2_ac+SQL_draft.docx” in the Git reflects your change

Sharable link for this Google Doc:

<https://docs.google.com/document/d/1Mn9nQ7FJK1F1R9UXLgWanUh0LV00DGcEO-9H6cH5Mhc/edit?usp=sharing>