

COMP-421/764 Database Systems, Winter 2021

Written Assignment 1: SQL

Due Date Feb 18, 05:00pm

This is an individual assignment. You are required to work on your own to create the solution.

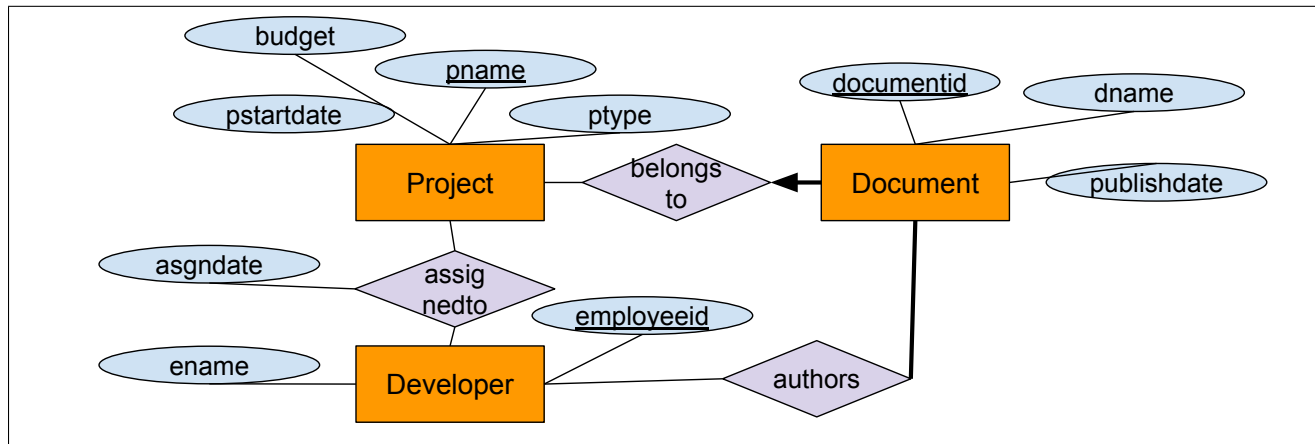
This assignment will be graded by an automated system. It is very important to read the instructions and follow them exactly. Not following those instructions can result in your assignment receiving 0 points. Turn in one tar file in mycourses.

Ex. 1 — SQL (50 Points)

Although the individual questions add up to a total of 55 points, the maximum points that you will get for this assignment is capped at 50 points. Use the extra 5 points as a buffer.

The following (part) of ER model and the corresponding relational model are designed for a project management system. There are various projects along with some associated metadata like their start dates, budgets, project type (internal/external), etc. The system also keeps track of the developers who are assigned to specific projects (one person could be in multiple projects). The system also keeps track of documents associated with a project, and the person(s) responsible for authoring them. It is not necessary that a person who is authoring a document is also assigned to work on the project (and vice versa).

A sample schema and few records have been provided as `setup.sh`. Please add more records into this as you may need to test various scenarios. The automated system will use a slightly different data set. So ensure your logic works even for different data sets. **Remember! Just because your query returned the correct results does not mean it is correct. Your query could produce incorrect results with slightly different data.** It is important to review your query and also to make sure that your data set has some “corner” cases (it is not about the number of records you use, but the variety of records that address different scenarios). The table definitions used by the automated system will be the same as the ones in the sample scripts given to you. Use your individual database accounts to work on the assignment.



```

project(pname, pstartdate, ptype, budget)
developer(employeeid, ename)
document(documentid, dname, publishdate, pname)
pname references project
devassignments(employeeid, pname, asgndate)
employeeid references developer, pname references project
documentauthors(employeeid, documentid)
employeeid references developer, documentid references document
    
```

Important !!

All the sql solutions will be evaluated by an automated system, which compares the output data produced by executing your query on our dataset with the expected output result for the correct query. So it is important that you include the correct column names, in the correct order, perform any ordering on output tuples as asked etc.

Double check your SQL for **typos**, for example if you spelt 'Kodak' instead of 'Kodiak', a query might not return the correct records and you will not get any points.

While the column and table names are not case sensitive, **the data itself can be case sensitive**. So do not write 'KODIAK', where it was required to write 'Kodiak' as this may produce no results or wrong results.

For more details read the attached sql formatting guide. If you have questions about this, post it in Piazza. **Remember you will either get 0 or all points for a given SQL question !!**

For this assignment you **MUST NOT** create views or tables in your solution. You can however use derived tables as we saw in class in your SQL. You **may also use the SQL WITH clause** if you wish to (not covered in lectures and not essential to writing solutions). **Do not use any explicit SQL CAST operations** to perform data conversions of your own. All your answers should be comprised of only **a SELECT query**. **Output ONLY the attributes in the question, following the exact order mentioned in the question.** **Adding attributes not mentioned can result in a 0 score !**

Unless explicitly allowed in the question, your output query **should not produce duplicate rows in your output resultset**. Use the technique taught in class to eliminate duplicate records from the output when a query can result in such a scenario. Keep in mind that some SQL queries do not produce any duplicate records while others may.

Where an output ordering is asked for, remember to order the output records. The technique for this was also shown in class. **Unless stated otherwise, the default ordering should be in the ascending order.**

1. (2 Pts) List the name and start date of all projects of the type **internal**. The output should be sorted by the project name.
2. (2 Pts) Find the information of all the authors of the document with document id 22. Output the employee id and name, order the output based on the employee id.
3. (3 Pts) Give the names and employee id of the developers who are assigned to the project **Kodiak** but has not authored any documents associated with the project. Order the output by the employee id.
4. (2 Pts) Give the project name, assignment date and project type of the projects assigned to the developer with the employee id 82102. Order the output by project names.
5. (2 Pts) Give a list of projects that started last summer (2020 May through August). Output project name and start date, order the output by start date and then by the project name.
6. (3 Pts) Give the name and employee id of the developers who are on at least one **internal** project but is not on any **external** project (these are the values of the project type column). Order the output by the employee id.
7. (4 Pts) Give the project name and employee id of the developers who have authored a document for a project, but is not assigned to that project. Order the output by the project and then by the employee id.
8. (2 Pts) List the names of all the projects for which the employee with the employee id 93401 has authored a document. Order the output by the project name.
9. (2 Pts) Find the number of projects that started this year. Name the output column **numprojects**.
10. (3 Pts) Find the number of projects for which the employee with the employee id 93401 has authored a document. Call the output column **numprojects**.
11. (3 Pts) List the name of all the projects that have more than 2 developers assigned to it. Order by the project names.
12. (4 Pts) List the name of all the projects that have 2 or less developers assigned to it. Order by the project names.
13. (3 Pts) List the names of the internal projects and the number of documents associated with them. Name the latter column **numdocs**. Order the output with the projects having the highest number of documents being first. You can then order by the project names if two of them have the same number of documents. Remember to take into account the projects with no documents in the output of the query.

14. (3 Pts) Find the name and start date of the oldest project (based on start date). If there are multiple projects with the same start date, order the output by the project names.
15. (4 Pts) Find the name and start date of the youngest project (based on start date) as well as the number of developers assigned to the project. If there are multiple projects with the same start date, order the output by the project names. Call the number column `numdevs`. Remember there could be a case where you may have no developers assigned yet, which you should consider as 0.
16. (4 Pts) Find the names and employee ids of developers who are either assigned to the project `Kodiak` or has authored a document for it. Order the output by their employee ids.
17. (4 Pts) Find the names of (all types of) projects whose budget is more than the average internal project budget. Also include the budget of such projects in the output. Order the output such that the highest budget projects are at the top of the list. If there are two projects with same budget, then order them by the project name. You can assume that there are some internal projects.
18. (5 Pts) Find the name of the project whose budget to the number of developers assigned ratio is the highest as well as that ratio (call this column `devcost`). If there are multiple projects with the same ratio, order the output by the project name. Do not include projects with no developers assigned for this calculation.

Questions ?

Please use Piazza for any clarifications you need. Do not email the instructor or TAs as this leads to a lot of duplicate questions and responses (not an efficient system). Please check the pinned post “A1 general clarifications” before you post a new question. It might have been already addressed there, in which case we will not address it again.

There will be specific TA office hours for the assignment that will be announced closer to the due date.