Homework 1A: Foundation Concepts (I)

COMS W4111: Introduction to Databases
Sections 002
Spring 2025

(v 1; 2025-JAN-28)

Notes:

- HW 1 is due on Sunday, 08-FEB at 11:59 PM.
- There are two parts to HW 1 – part A and part B.
- This document defines HW 1A.
- HW 1B will require material from the 31-JAN lecture and will be published before the 31-JAN lecture.
- Both the programming and non-programming tracks complete HW 1A.

Submission and Overview

Grading and Scope

Total points for homework assignments and exams determine final grade. The final point total is between 0 and 100. HW 1 is worth 5 points. HW 1A is worth 2.5 points and HW 1B is worth 2.5 points.

The scope of material for this HW1 is:

- The material in lecture 1.
- The material in lecture 2.
- The slides associated with the recommended textbook for
    - Chapter 1.
    - Chapter 2.
    - Chapter 3, slides 3.1 to 3.36.
    - Chapter 6, slides 6.1 to 6.24.

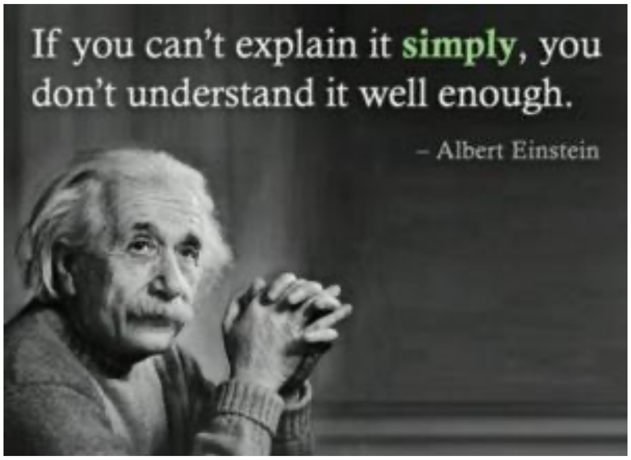
Submission

Due date: 2025-Feb-09, 11:59 PM EDT on GradeScope.

You submit on GradeScope. You upload a PDF of this document with your answers entered. You must assign pages to the questions in the outline in GradeScope.

There is a post/mega-thread on Ed Discussions that we will use to resolve questions and issues with respect to homework 1.

Brevity


If you can't explain it simply, you don't understand it well enough.
– Albert Einstein

Keep your answers focused, brief and succinct. The answers to written questions only require 3 or 4 sentences/bullet points. If you ramble and bloviate hoping to get something correct, we will deduct more points.

Questions

General Knowledge Questions

1. Is data in a spreadsheet unstructured, semi-structured or structured. Briefly explain your answer?
   A spreadsheet is typically considered semi-structured because it lacks integrity constraints and data-typing/schema enforcement, which are the hallmarks of fully structured data.

2. Professor Ferguson can export SSOL data for his classes to a spreadsheet. Despite the data being spreadsheet-like, Columbia uses an application and database to manage the data. An alternative would be to make shared spreadsheets available to everyone and let students, faculty and staff collectively edit the spreadsheets. This would be chaos. List 4 functions/capabilities of a database management system that makes using a database and application superior to sharing spreadsheets.
    a. Concurrency control – meaning that multiple users can work at once without worrying that they will double overwrite or have inconsistent reads before the changes are able to push the server.
    b. Data integrity/validation – makes sure that the values that students enter into a specific column and/or row are of a consistent datatype.
    c. Access control – With typical spreadsheets, there is no way of preventing access to certain subsets of the data.; however, with Databases, we can grant specific users access to some parts of our DB but not others.
    d. Backup – Without a system for properly centralizing versions of the document, we can't create proper backups in case of any issues.
3. What are the three levels of abstraction for data that a DBMS provides? What are two disadvantages of having a user or developer directly use the lowest level and directly access the data without going through the higher abstractions?
    a. Views (External) – Deals with what users see
    b. Logical (Conceptual) - Describes what data is stored and the relationships between data.
    c. Physical (Internal) – This deals with how the data is stored (file structures, indexing, etc.)

    These come from the ANSI/SPARC three-schema architecture, although the book mentions a slightly different architecture in Chapter 1 (p. 23).

4. Is a full-stack web application a two-tier or three-tier application? Is a Jupyter notebook a two-tier or three-tier application? Explain your answer. We saw an example of a two-tier application/product in lecture 1. What was the product?
    A Jupyter notebook is three-tier because it has a presentation or UI layer (the web-based notebook UI), an application or logic layer (the Jupyter kernel executing code), and a data layer(DB or file system that the notebook queries). It is different from other IDEs because it casts a local host to run the notebook on a webpage. The two-tier application we saw was the MySQL Workbench, which only features presentation and data later.

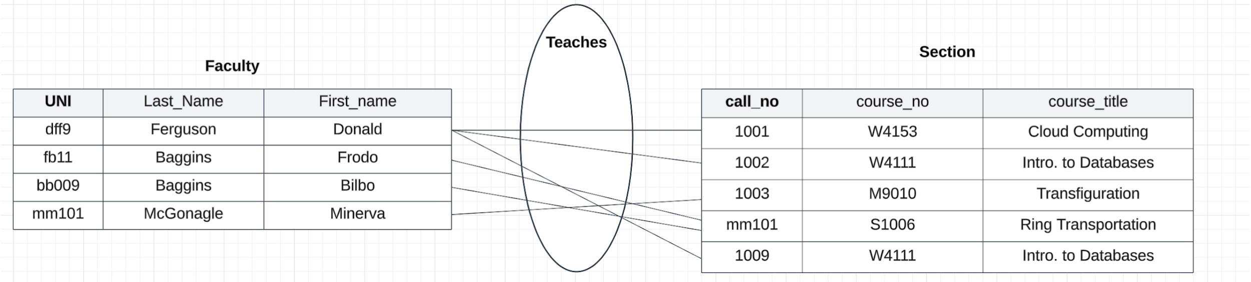
5. What are the four types/categories of database users? Which type of user is most likely to use DDL?
    From Chapter 1, pg. 24:
    a. Database Administrators (DBAs) - Responsible for designing the database schema, managing storage structures, security, etc.

b. Naïve Users – Have little to no knowledge of how to write code, query, etc., and use pre-built applications or interfaces to interact with the database.
c. Sophisticated Users – Have more advanced knowledge of how to query and create some prompts. Use query languages like SQL directly to interact with the database.
d. Application Users – Using databases by connecting via API or libraries.

Of the four categories, Database Administrators (DBAs) are most likely to use DDL, because they are responsible for defining and managing the structure of the database.

Entity Relationship Model

6. Consider the entity set with relationships below. Assume that the bold attribute/column is the primary key. Write the relational model schema definitions for representing the entity sets *Faculty* and *Section.*
Faculty (UNI, Last_name, First_name)
Section (call_no, course_no, course_title)



Faculty

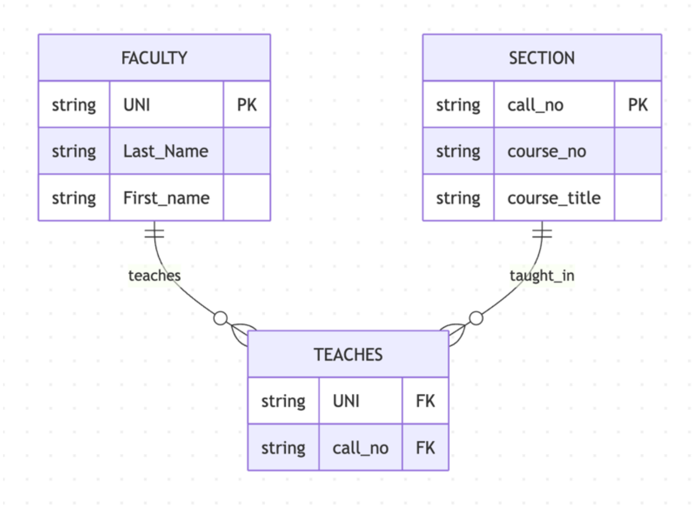
UNI	Last_Name	First_name
dff9	Ferguson	Donald
fb11	Baggins	Frodo
bb009	Baggins	Bilbo
mm101	McGonagle	Minerva

Section

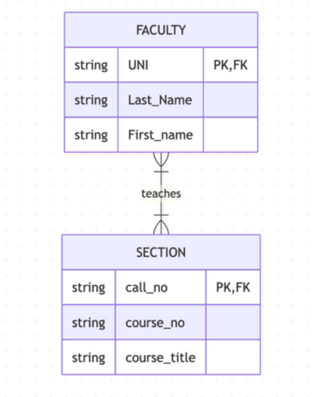
call_no	course_no	course_title
1001	W4153	Cloud Computing
1002	W4111	Intro. to Databases
1003	M9010	Transfiguration
mm101	S1006	Ring Transportation
1009	W4111	Intro. to Databases

7. Using the approach for documenting relationship sets in Lecture 1's slides, write down the relationship set *Teaches* for the diagram in question 6.
    a. (dff9, 1001)
    b. (dff9, 1002)
    c. (dff9, 1009)
    d. (fb11, mm101)
    e. (bb009, mm101)
    f. (mm101, 1003)

8. For the diagram in question 6, draw the *conceptual model* Crow's Foot Diagram using Lucidchart.



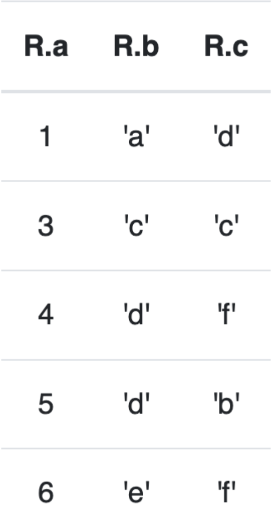
We have to do use an intermediate one:many “associative entity” because SQL doesn’t allow for many-to-many relations. If we wanted to treat “teaches” simply as a relationship, then we’d get the following



Relational Algebra

9. Using the diagram from question 6, what is the result of the relational algebra expression $\pi_{\text{course\_no, course\_title}}$ (Section)?
   This simply leads to dropping the section name from the projection.

10. Using the relation below, what is the result of the expression $\pi_{b,c}$ ($\sigma_{a>3}$ (R))



R.a	R.b	R.c
1	'a'	'd'
3	'c'	'c'
4	'd'	'f'
5	'd'	'b'
6	'e'	'f'

We would just get a smaller subsection of what is shown above:

R.b	**R.c**
‘c’	‘c’
‘d’	‘f’
‘d’	‘b’
‘e’	‘f’

The selection of a>3 would remove the first row where a=1, and then the final projection drops the column R.a altogether, so we are left with the columns R.b and R.c for the bottom four rows.

SQL

11. What is the SQL equivalent to the relational algebra expression $\pi_{\text{name, dept\_name}}(\sigma_{\text{dept\_name='Comp. Sci.'}}(\text{instructor}))$.

```
SELECT Name, Department
FROM Instructor
WHERE dept_name = ‘Comp. Sci.’
```

12. Translate the following relational model definition into SQL. You may assume that the data type of all columns is TEXT.

    section(courseNo, sectionNo, semester, year, courseName, enrollment)

DDL for this is:

```
CREATE TABLE section((
   courseNo TEXT,
   sectionNo TEXT,
   semester TEXT,
   year TEXT,
   courseName TEXT,
   enrollment TEXT,
   PRIMARY KEY (courseNo, sectionNo, semester, year)
);
```

The last line we add to form the composite primary key since no individual attribute has it. On a similar note, it is typically better to use VARCHAR() and assign the maximum length of characters for both space and data consistency reasons; however, TEXT was used to stay in line with the question.

13. Using the diagram from question 6, translate the relational algebra statement $\pi_{course\_no, course\_title}$ (Section) into SQL. What is the result of executing the SQL statement?

```
SELECT course_no, course_title
FROM Section;
```

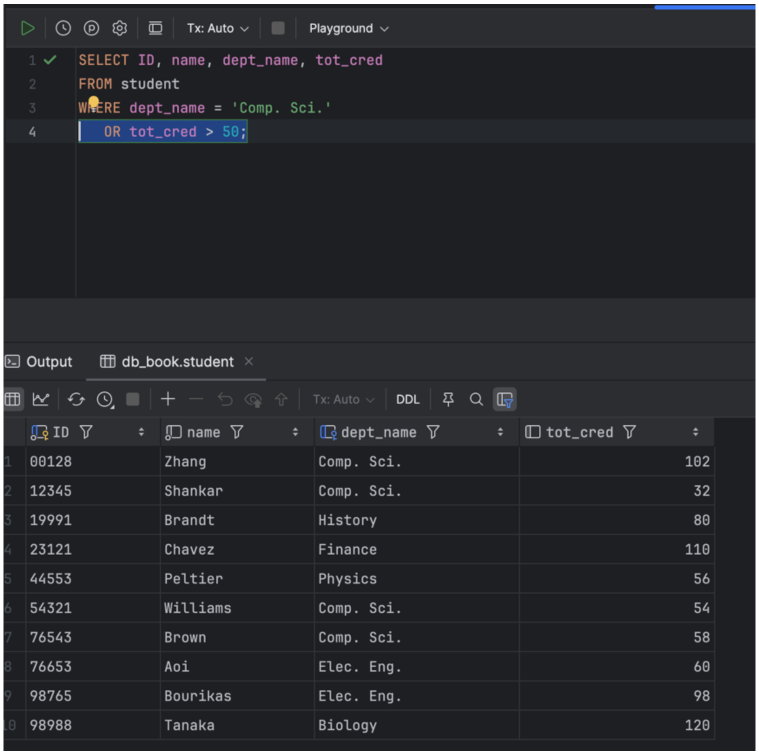
course_no	course_title
W4153	Cloud Computing
W4111	Intro. to Databases
M9010	Transfiguration
S1006	Ring Transportation
W4111	Intro. to Databases

Because we did not use DISTINCT and the default of SQL is to leave “repeats” (because ultimately the full triplet is unique but the first column, which is the PK, is not being projected).

14. Using DataGrip and the data from the recommended text book that you loaded in homework 0, write a SQL statement that selects the ID, name, dept_name and tot_cred of students. Your result should include only students that are in the ‘Comp. Sci.’ department or who have tot_cred > 50. Place the text of your query and a screen capture of the execution below.

```
SELECT ID, name, dept_name, tot_cred
FROM student
```

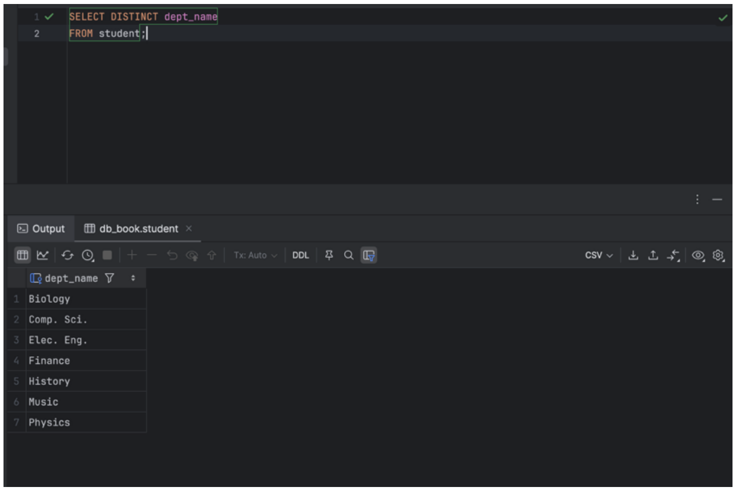
WHERE total_cred > 50 OR dept_name=’Comp. Sci.’;



```
SELECT ID, name, dept_name, tot_cred
FROM student
WHERE dept_name = 'Comp. Sci.'
    OR tot_cred > 50;
```

ID	name	dept_name	tot_cred
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
54321	Williams	Comp. Sci.	54
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

15. Using DataGrip and the data from the recommended text book that you loaded in homework 0, write a SQL statement that returns a table containing only dept_name. The table should not have duplicated. Place the text of your query and a screen capture of the execution below.
SELECT DISTINCT dept_name
FROM student;



```
SELECT DISTINCT dept_name
FROM student;
```

dept_name
Biology
Comp. Sci.
Elec. Eng.
Finance
History
Music
Physics