

Assignment #1

CS 348 - Fall 2022

Due : 11:59pm p.m., Tue, Sep 29 2022

Appeal Deadline: One week after being returned

(Total weight 10%)

Submission Instruction

This assignment consists of 2 parts. Part I will be submitted through Crowdmark. See the website for more detailed instructions. In particular, do not forget to submit one file per question to make the lives of TAs easier. Part II is a programming question and requires submission of files through Marmoset. See the Piazza post for detailed information about how to test your programs and submit your final files.

You don't have to type the questions if you are using Latex. Just specifying the question number is good enough. You don't need to keep the same font style. Here is a draft/empty latex template you may download and use: <https://www.overleaf.com/read/dnrknxzcncq>. Handwritten solutions and scanned to pdf are also acceptable.

Part I

[50 marks, 5% of Final Grade in Total]

This part consists of 3 questions on relational model, relation algebra, and basic SQL.

Question 1.

[8 marks in total] Warm up question:

- (a) (4 marks) Describe a database system used by a company or a organization and what the system is used for (2-3 sentences). Please cite your reference.

- (b) (4 marks) Use 3-4 sentences to explain what is physical data independence and why it is important. Please cite your reference if there is any.

Question 2.

[20 marks in total] Consider the following database schema:

Product (maker, model, type)

PC (model, speed, ram, hd, price)

Laptop (model, speed, ram, hd, screen, price)

Printer (model, color, type, price)

The **Product** relation gives the manufacturer (or maker), model number, and type (PC, laptop, or printer) of various products. The **PC** relation gives for each model number that is a PC the speed (of the processor, in gigahertz), the amount of RAM (in megabytes), the size of the hard disk (in gigabytes), and the price. The **Laptop** relation is similar, except that the screen size (in inches) is also included. The **Printer** relation records for each printer model whether the printer produces color output (true, if so), the process type (laser or ink-jet, typically), and the price. The primary keys designated for these four relations are underlined in the schema above. In addition, the models for all the pc products in the **Product** relation must be contained in the **PC** relation. Similarly, the models for all the laptop products in the **Product** must be contained in the **Laptop** relation, and the models for all the printer products in the **Product** must be contained in the **Printer** relation. The following is a sample instance:

Product			PC				
maker	model	type	model	speed	ram	hd	price
<i>A</i>	001	<i>pc</i>	001	2.66	1024	250	2114
<i>B</i>	003	<i>printer</i>	002	2.10	512	250	995
<i>D</i>	001	<i>laptop</i>	003	1.40	512	80	478
<i>D</i>	001	<i>printer</i>	004	2.80	1024	250	649
...

Laptop						Printer			
model	speed	ram	hd	screen	price	model	color	type	price
001	2.00	2048	240	20.1	3673	001	<i>true</i>	<i>inkjet</i>	99
007	1.73	1024	80	17.0	949	003	<i>false</i>	<i>laser</i>	209
003	1.80	512	60	15.4	549	004	<i>true</i>	<i>laser</i>	391
006	2.00	512	60	13.3	1150	006	<i>true</i>	<i>dry</i>	129
...

Based on the given schema and instance, for each of the following questions, please **circle** (or **state**) “**T**” (for “true”) or “**F**” (for “false”) and **explain your choice** using no more than 4 sentences.

(a) (4 marks) {**maker**, **type**} is a candidate key for the **Product** relation. Ans: “**T**” or “**F**”

Hint: The database on page 3 is just one possible instance. Your decision should not simply base on this instance.

(b) (4 marks) {**model**, **speed**} is a superkey for the **Laptop** relation? Ans: “**T**” or “**F**”

(c) (4 marks) We cannot insert a new row with value (“A, 001, pc”) to the **Product** table instance shown on the previous page?

Ans: “**T**” or “**F**”

(d) (4 marks) If the **PC** table only has 4 models listed in the instance on the previous page (001, 002, 003, 004), we can still insert a new row with value ("A, 005, pc") to the Product table. Ans: "**T**" or "**F**"

(e) (4 marks) The **Printer** relation has 8 superkeys. Ans: "**T**" or "**F**"

Question 3.

[22 marks in total] Using the database schema and sample instance from Question 1, which shows the `Product`, `PC`, `Laptop`, `Printer` relations, write the following queries in relational algebra.

Notes: (i) Only use the relational algebra operators listed on the lecture slides (including selection, projection, cross product, union, difference, renaming, join, natural join, intersection).
(ii) The answers can be expressed either in trees or lines of algebraic expressions.

- (a) (3 marks) Which manufacturers make laptops that have a speed of at least 2.00;
- (b) (4 marks) Find the model number, type, and price of all products (of any type) made by manufacturer B;
- (c) (4 marks) Find those pairs of PC models that have both the same speed and RAM (A pair should be listed only once; e.g. `list(i,j)` but not `(j,i)`);

(d) (5 marks) Find the manufacturers that make the cheapest printers.

(e) (6 marks) Find printer models that are sold by *exactly* two manufacturers.

Part II.

[21 marks, 5% of Final Grade in Total]

Students will work with a state-of-the-art commercial DMBS. A number of systems can be used. The School supports IBM DB2. Follow the instructions below to get started:

Get Started

DB2 is available on

ubuntu2004-002.student.cs.uwaterloo.ca

ubuntu2004-004.student.cs.uwaterloo.ca

Every CS348 student will have an account on these servers.

First, login in by typing the following command in a terminal followed by your password:

```
softbase.cs$ ssh <userid>@linux.cs.uwaterloo.ca
```

This will connect you to one of the two Ubuntu servers where you can access DB2. If you want to choose which of the two servers to connect to you can now ssh directly to either ubuntu2004-002.student.cs.uwaterloo.ca or ubuntu2004-004.student.cs.uwaterloo.ca. If you start editing a database on one of these servers you will need to reconnect to that same server or you will not see your changes. This is not a concern if you populate the database using data stored in files.

Second, [set up your environment correctly](#):

for sh, bash, et al:

```
$ source ~cs348/public/db2profile
```

Then, you can start DB2 without any DB2 password:

```
$ db2
(c) Copyright IBM Corporation 1993,2007
Command Line Processor for DB2 Client 11.1.1.1
[...]
For more detailed help, refer to the Online Reference Manual.
db2 =>
```

Last, connect to the database managed by DB2 for this class by typing in DB2

```
db2 => connect to cs348
```

You can now enter SQL statements directly at the db2 => prompt. For example,

```
db2 => list tables
```

You can read more examples from DB2Tutorial.pdf uploaded on Learn (under Assignment 1) or find online reference for IBM DB2 at https://www.ibm.com/support/knowledgecenter/SSEPGG_10.1.0/com.ibm.db2.luw.kc.doc/welcome.html.

Run Test Database

Download Database.zip from Learn under Assignment 1. Transfer the unzipped files onto one of the servers which has DB2, e.g. you may use

```
$ scp -r Database/ <username>@ubuntu2004-002.student.cs.uwaterloo.ca:~/
```

After you login, remember to set up your environment correctly (refer to the instruction above).

In Database/testdb, there are several .sql scripts (connectCS348.sql, droptables.sql, createtables.sql, populatetables.sql).

You can run .sql scripts in the following way: for example,

```
$ db2 -stvf connectCS348.sql
```

Remember to **connect to CS348** (run the script above) before running other scripts.

You can find the schema in createtables.sql and this script creates tables.

In Database/testdb/, there are also several text files (student.txt, faculty.txt, class.txt, enrolled.txt) which are used by populatetables.sql to populate the tables. droptables.sql drops all the tables.

Queries

In this assignment, you are required to submit 6 SQL files. You can find starter files in Database/testdb/queries/: 1.sql, 2.sql, ..., 6.sql.

Each file contains an incomplete SQL statement. Complete these SQL statements based on the problems below. Note that these query statements will be marked automatically, so you need to test if the queries work in db2 by running `$ db2 -stvf 1.sql`, etc before submission.

Unless otherwise specified each file should contain only a single query.

You should NOT create views to answer any of the SQL questions.

You should NOT update the database except in question 6 where you are specifically asked to delete tuples.

Submission

You need to submit your assignment via <https://marmoset.student.cs.uwaterloo.ca/>.

IMPORTANT: Your submission must be a single zipped folder which includes only the 6 SQL files 1.sql, 2.sql, ..., 6.sql. This means that when you unzip the folder you should see the 6 SQL files. The files must be named exactly 1.sql, 2.sql, ..., 6.sql.

Each question has only 1 public test and many private tests. You will only see the results of the public test (whether or not you passed or failed the public test). Passing the public test does NOT guarantee that your submission will pass all of the private tests. To receive full marks your SQL must pass all of the tests. Therefore, it is important that you test your SQL yourself.

To pass a test the tuples returned by your query must match the expected output. The number of columns returned by your query must match what the question asks for however, the column names and column order do not matter so feel free to use aliases if you like. The row/tuple order matters iff the question asks for a specific order.

You can re-submit until the deadline and your best submission will be used for the final grade. Do not be alarmed if marmoset is slow, marmoset can often take a rather long time to run and grade your submission.

All of the SQL files that you submit must work in DB2 and they must NOT produce any errors/warning. Queries that produce errors/warnings will automatically fail the test cases.

1.sql

[4 marks in total] Print the average age of students enrolled in classes taught by faculty members that have taught at least 2 courses.

2.sql

[3 marks in total] Print the list of names of faculty members (with no repetitions) that do not teach any classes. Sort the list of names in ascending order. Note that it is possible for two faculty members with different fid have the same fname. For example, two different faculty members share the same name 'Xi He', then 'Xi He' should appear twice in the output. On the other hand, the name of the same person should not appear more than once in the output.

3.sql

[3 marks in total] Print the list of names of students who take the class 'Operating System Design', and the class 'Database Systems'. The output list should be sorted by the student name in ascending order. Note that it is possible for two students with different snum have the same sname. For example, two different students share the same name 'Lisa Walker', then 'Lisa Walker' should appear twice in the output. On the other hand, the name of the same person should not appear more than once in the output.

4.sql

[4 marks in total] Print the list of class names (with no repetitions) from the class relation, together with the name of the corresponding lecturer and class sizes. The output list should be sorted according to the class size in descending order. (Note that some classes in class relation may be not enrolled by any one. These classes should appear in the output as well, with a count of 0).

5.sql

[4 marks in total] Print the number of students who are enrolled in exactly one class of size < 5 . (The students considered in the output can take classes of size ≥ 5 , but cannot take two or more than two classes of size < 5).

6.sql

[3 marks in total] Delete classes that are enrolled by fewer than 2 students. (You may use more than 1 queries.)