Exam DB1102 H2022

Home exam 13 December 2022 – 9am-3pm (6 hours)

Counts 100% of the grade in the course. Assessment: Pass / Fail.

All aids allowed.**Not allowed to collaborate with others.**

When you hand in the exam, you must collect your answers in one pdf document. The document contains answers to all the tasks.

Assignment 1 – modeling (30%)

Some friends of yours want to pick up the fight with Cutters and other hairdressers. They want to start a hairdressing salon where the customer can book an appointment online. The hairdressing salon is intended for those who want to decide on a hairstyle before arriving at the salon. They approach you as you have learned how to model data. This is how they describe their idea:

Our hairdressing salon is largely about hairstyles. There are an incredible number of cool hairstyles out there that the vast majority of people miss out on. We want to create a website where the customer can browse through different hairstyles and choose the one they want - before they come for a haircut. Therefore, we need to store information about different hairstyles we offer, what the hairstyle is called and how much it costs. And of course we also need a description of the hairstyle and a picture.

Now it is not the case that all our hairdressers can do all types of hairstyles. We therefore need an overview of which hairdressers can perform which hairstyles. The idea is that when the customer has found their dream hairstyle, they can choose from hairdressers who can perform that hairstyle. We also need to have information about our various hairdressers: What their name is, and which user they have on Instagram.

Once customers have chosen a hairdresser and hairstyle, they can book an appointment. The agreement must naturally have a time, in addition to information about which hairdresser will perform which hairstyle. And it's nice to know which customer made the deal. It is an advantage that our customers can create a user account. No, by the way; they MUST create a user account before booking an appointment. We need to know the name, address, e-mail address and telephone number of the user. Can you fix a database model for this?

(FYI: There is a data type that can store images.)

Task: Draw a model of your proposed solution. You can choose whether you want to use crow's feet or UML notation. If you choose crow's feet, you do not need to distinguish between identifying and non-identifying conditions. (UML notation does not distinguish between this anyway.) Your model must contain:

- The entities and their attributes.
- Primary keys and foreign keys.
- The relationships between the entities.
- The multiplicity (participation and cardinality) of the relationships.
- If necessary, link entities.

If you think something is unclear, make your own assumptions. In that case, remember to account for these.

We recommend using a program, such as Lucidchart, to draw the model. You can also choose to draw by hand and paste the image of your drawing, but it may then be more difficult for the sensor to decipher it.

Assignment 2 – SQL (50%)

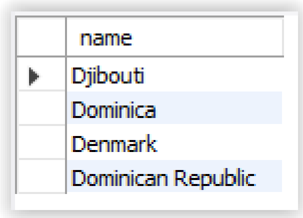
In the exam paper, you will find an SQL script (arcade.sql) that created a database (arcade) and fills tables with data. Run the SQL script against your own MySQL server. You then solve the tasks as described below. The answer to each subtask is twofold:

1. SQL that solves the task.
2. A screenshot showing your result when the SQL is run.

Example:

If I had a task from the world database: "Extract names of all countries beginning with the letter D", then an answer could look like this:

SELECT name from country WHERE name LIKE 'D%';



name
Djibouti
Dominica
Denmark
Dominican Republic

But you must use it**the arcade database**which is attached to the exam paper. Remember that it is better to make an attempt to answer a task than to answer blankly. Remember that this assignment counts for 50%, which amounts to approximately 3 hours. If you are very stuck on a task, it may be a good idea to move on to the next one.

a) Create a query that retrieves information about all players (PlayerId, Date and Code). Sort the result based on Code.

b) Create a query that retrieves all information about games launched after January 1, 2020.

c) Create a query that retrieves information about the various game categories, and how many games are in the database within each category.

d) Create a query that retrieves information (PlayerId, Date and Code) about the player who has played the most times (has the most results), and how many times the player has played. Therefore: PlayerId, Date, Code, Number of Results.

e) Create a query that retrieves information about all games (Title and Category) and the average Score achieved in the game.

f)**(Difficult)**Create a query that retrieves information about all games (Title and Category), which player (Code) has the highest Score in the game, when the record (highest Score) was set, and what the record is.

g) Choose a game, and create a view for the game that shows all results for the game i **descending order based on Score**.

- The view should be called "Scoreboard_<Game name>".
- The view must contain: The code of the player, the time of the result and Score.

h) Enter information about the new game: Ping Pong.

- The game is in the category "Sports", was launched on September 8, 2022 and needs a nice description that you can come up with.
- Also enter information about who developed the game: Jens Petrus was a programmer. Frida Frosk was the designer. A new developer named Billy Betong was the screenwriter. Billy can be contacted at billy.betong@boeljeband.no.

i) Which developers have been involved in developing games with the most results in total? Create a query that retrieves:

- Name (of game developer)
- Ant_results (for the number of results in games he has helped develop)

The result must be sorted in descending order of the most results.

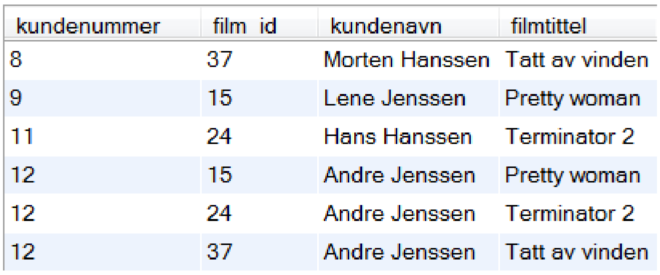
You should only include developers who have more than 30 results in total on the games theirs.

j) A change must be made in the game categories. The Action category is too extensive. There is a desire to replace the Action category with 'Shooter', 'Platformer' and 'Fighter'.

- Make a change so that the possible categories for Games are:
'Adventure', 'Shooter', 'Platformer', 'Fighter', 'RPG', 'Simulation', 'Sports', 'Puzzle', 'Other'.
- Make sure that existing Action games end up in the right category. You can decide for yourself whether they are 'Shooter', 'Platformer' or 'Fighter'.

Task 3 – Normalization (20%)

The DVD rental company "Against normal" has a table with data that describes which films have been lent to which customers. An excerpt of the table is shown below.

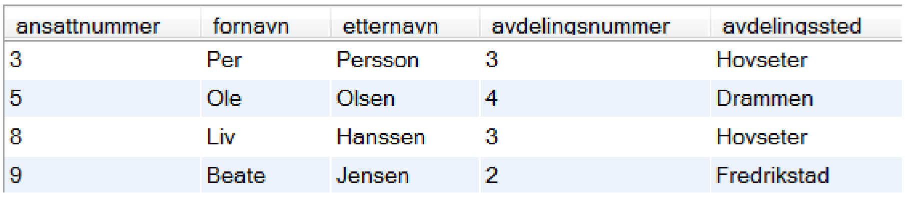


kundenummer	film_id	kundenavn	filmtittel
8	37	Morten Hanssen	Tatt av vinden
9	15	Lene Jenssen	Pretty woman
11	24	Hans Hanssen	Terminator 2
12	15	Andre Jenssen	Pretty woman
12	24	Andre Jenssen	Terminator 2
12	37	Andre Jenssen	Tatt av vinden

The table has a composite primary key (customer number, film_id). A customer can be identified by his customer number, and has the same name as the customer name. A film can be identified by its film_id, and has a title equal to the film title.

a) Show two cases of redundancy in the table.
b) Explain why the table is not in 2NF.
c) Normalize the table so that we obtain 2NF.

"Toward normal" also has a table that holds information about the employees. An excerpt of the table is shown below.



ansattnummer	fornavn	etternavn	avdelingsnummer	avdelingssted
3	Per	Persson	3	Hovseter
5	Ole	Olsen	4	Drammen
8	Liv	Hanssen	3	Hovseter
9	Beate	Jensen	2	Fredrikstad

An employee can be identified by his employee number, which is the primary key in the table. The table holds information about the employee's first and last name. It is also recorded which department the employee works in. The department has a unique department number and the location of the department tells us where in Norway the department is located.

d) Explain why the table is not in 3NF.
e) Normalize the table so that we obtain 3NF.

Good luck!