

ICT & Business Intelligence & CRM 2024/2025
MSc in Innovation Management – MAIN

EXAM

Andrea Vandin, Scuola Superiore Sant'Anna
Paolo Ferragina, Scuola Superiore Sant'Anna
Anna Monreale, Università di Pisa

Instructions

When:

- Deadline: 21 May 2025, before 23:59

Where:

- Submit on in the shared folder available on our github wiki

Who:

- This is a group project according to the groups agreed during the classes.

Exercise-specific instructions

Ex.1 (Vandin): you need to deliver:

- A picture or pdf of the conceptual model. File name should contain all students' surnames, and 'Ex1Conceptual'. E.g.: **RossiVerdi_Ex1Conceptual.png**
- A picture or pdf of the logical model. File name: same as point 1, but with Logical. E.g.: **RossiVerdi_Ex1Logical.png**

Ex.2 (Vandin): you need to deliver:

- File with text of query and SQL query. File name: all students' surnames, and EX2. E.g.: **RossiVerdi_Ex2.sql**
- DBeaver's screenshot of query result. File name, same as previous point. E.g.: **RossiVerdi_Ex2.png**

Ex.3 (Monreale): you need to deliver:

- ...
- ...

Ex.4 (Monreale): you need to deliver:

- ...
- ...

Ex.5 (Ferragina):

- *There currently is an ongoing discussion on the evaluation method for this part.*

All these files must be inserted in a folder with the name equal to the list of students names and the folder must be zipped (e.g., RossiVerdi.zip). This zip file must be uploaded on model.

Exercise 1 (10 points)

Design the **conceptual and the logical model of a database** respecting the following requirements:

The database will register information about a Burger restaurant which just sells burgers (no drinks, no sides). The database must contain information about Burgers, and about the customers who bought them (this information is collected via the fidelity card). In particular, the processing system for the restaurant is based on an operational database with the following characteristics.

Burgers are characterized by a name, a bun/bread type, a sauce (or more sauces). Burgers can be meat-based, in which case we are interested in knowing what type meat it contains (chicken, turkey, etc), and how much. Alternatively, burgers can be plant-based, in which case we are interested in knowing the allergens it contains (if any).

A bun/bread type is characterized by a name, the list of allergens it contains, and whether it is vegan and/or suitable for celiacs.

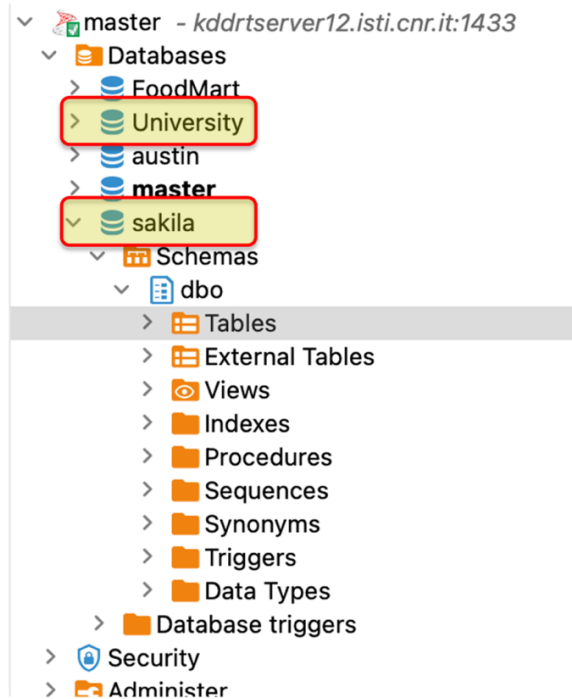
For each customer, the information of interest is the fidelity card code, the name and surname, the email address (if any), the city of residence, the year of birth.

For each order, we are interested in having information about the date and time, the customer, the burgers (one or more) bought in the order, the payment method.

Exercise 2 (6 points)

Write and solve the queries below on the database **Sakila**, available online. It can be accessed using DBeaver in the same way in which we accessed the online **University** database in class:

- If during the classes you have configured DBeaver to access the **University** database, you will find the **Sakila** one next to it in the Database navigator.



- If not, follow the DBeaver instructions on our website
- In the queries you might have to write 'sakila.dbo.' before table names (e.g., sakila.dbo.film)

These are queries to be considered:

1. List of all customers (first_name, last_name) satisfying the following condition:
 - have first name with "o" as second letter and surname ending with "a".Avoid repetitions in the results.
2. List of actors (first_name, last_name) of movies which have 'Italian' as language (i.e., any actor who has done at least one movie with Italian language).
3. List of actors (first_name, last_name) that worked in movies of language 'English', but not in movies of language Italian (actors who appeared in at least a movie with language Italian shall be excluded from those appearing in movies of language English). Avoid repetitions in the results. Sort the results by first name
4. Overall and average duration of rentals of movies, grouped per category. Also, compute the number of movies of each category.
5. List the movies (title) with 3 or more actors, including the length and rating. Sort the results by title of the movie.

Deliver a script file with text of queries (in a comment) and corresponding SQL query. Also provide a screenshot of DBeaver's output of the query (do not need to include all rows. The default view in DBeaver is fine).

Exercise 3 (7 points)

Exercise 4 (3 points)

Exercise 5 (6 points)