The aim of the second set of exercises is obtain a better understanding in the techniques that can be used to obtain a good database design. To this aim, you will need to design two databases, from the conceptual design to the physical design. This exercise is done in **groups of 3**. The final deadline for the answers to all exercises is **March 29, 23:55**.
Creating a conceptual design for a database is similar to writing a text: an important aim of the conceptual design is to communicate which information needs to be stored in the database, and which constraints can be imposed on the data. The success of a diagram can hence be measured by the clarity of the design in the eyes of people that have to study the diagram. To evaluate the quality of one of your designs, we will rely on peer-reviewing. To support this peer-reviewing, this exercise has several intermediate deadlines.
Please read these instructions fully and carefully before starting; take note of the intermediate deadlines.
Consider the use of software such as Dia (`http://dia-installer.de`), Diagrams.Net (`https://www.diagrams.net/`) or other online tools for creating diagrams; this software is available for all common operating systems.

1 Collecting Pieces of Art

The context of this exercise is the following.

> You are creating an application for the ministry of culture; this application should keep track of various pieces of arts, collections, exhibitions, and locations, such as museums and storage sites.
>
> Within the database for this application we will need to store the following kinds of information:
>
> - Information about artists, such as their birth date, their place of birth, and their name.
> - Information about pieces of art. For all pieces of art, this includes the identity of the artist(s) and when the piece of art was created. For paintings, this should include the width and length of the painting; for sculptures, this should include the width, length and height. Moreover, every time a piece of art moves to a new location, the year, day and target location of this move should be stored; on one day a piece of art can only move to one new location; the same piece of art can be moved to the same location multiple times, but then on different days.
> - Information about collections. This includes the name of the collection and the pieces of art in the collection, and, if relevant, to which museum the collection is associated.
> - Information about exhibitions, such as their begin and end date, the location where the exhibition takes place, and which pieces of art are part of the exhibition.
> - Information about locations, such as the city of the location, the street name of the location and the street number of the location.
> - Information about locations that are museums, such as the name of the museum and the opening hours of the museum;

- Information about locations that are storage sites, such as the size of the location.

Some aspects of the application are not yet clearly specified: are there other pieces of art than paintings and sculptures that could be stored in the database? Should other information be stored, such as the materials used to create the piece of art? Can pieces of art belong to multiple collections? What kind of identifiers do we have for the various locations, exhibitions, etc.?

You are tasked with the creation of a first proposal for all the data that could be collected in this application. This proposal should fill in all the details of the data that will be collected in this application.

The tasks in this exercise are the following.

1. Create a **conceptual** design for your database by creating a diagram for your database using ER Chen notation. Take care that the following requirements are met:

   - Your diagram will reflect all the requirements listed above, and will provide additional relations as you believe to be necessary in your application.
   - Your diagram will have to contain at least 7 entities.
   - Your diagram should reflect your choices in as much detail as possible: hence, make sure that cardinality constraints, participation constraints, and primary keys are indicated.

   Note that your diagram should be **conceptual**. A conceptual diagram **does not** exactly represent all the tables that will be stored in a relational database; it **does not** represent a phyical design of the database. The most important goal of a conceptual diagram is to make clear what is conceptually going to be stored in the database.

2. Once you have finished this diagram, you are required to create a second diagram in ER/Crow's foot notation or UML notation. Also here, you should make sure that the diagram indicates all constraints correctly, and is **conceptual**.

3. Submit both your diagrams on Moodle for review, in PDF form; the deadline for this phase is **March 18, 23:55**. You are only allowed to send the diagrams; hence, the diagrams should be understandable without additional information describing them. You are allowed to add limited textual annotations in the diagram, for instance, labels that indicate the meaning of an edge in the diagram. Each group will receive the diagrams of three other groups, as well as instructions for reviewing these diagrams. Each member of your group is expected to review the diagram of one other group. You have to complete these reviews by **March 25, 23:55**, and will also receive the reviews of your design at this same moment. 20% percent of the grade for this exercise will be determined by the grades that you receive from other students!

4. Once you have created the conceptual diagram, you are required to create a **physical design** of the database. Create the schema for your proposed database using statements in SQL. These statements should be executable in SQLite. Ensure the following:

   - All primary key and foreign key constraints are properly defined;
   - The types of all columns are sensible, including whether or not attributes are NULLable;

- The database is in third normal form;
- The design of your database reflects the conceptual design.

The most important design criterion when creating the physical design is that the quality of the data is ensured; hence, you are not required to take into account considerations related to the performance of the database. You may start with this schema before the reviews are finished; however, the final schema should reflect the choices in your final diagram. Submit all your schemas, your updated conceptual diagrams, and a short description of your choices and the way you processed the other's students comments by **March 29, 23:55**. The latter document should have at most 2 pages.

In total, this exercise determines 60% of your final grade.

2 European Soccer

On Moodle you will find a database for European Soccer, also available from `kaggle.com`: `https://www.kaggle.com/hugomathien/soccer`. We will study the quality of this database.

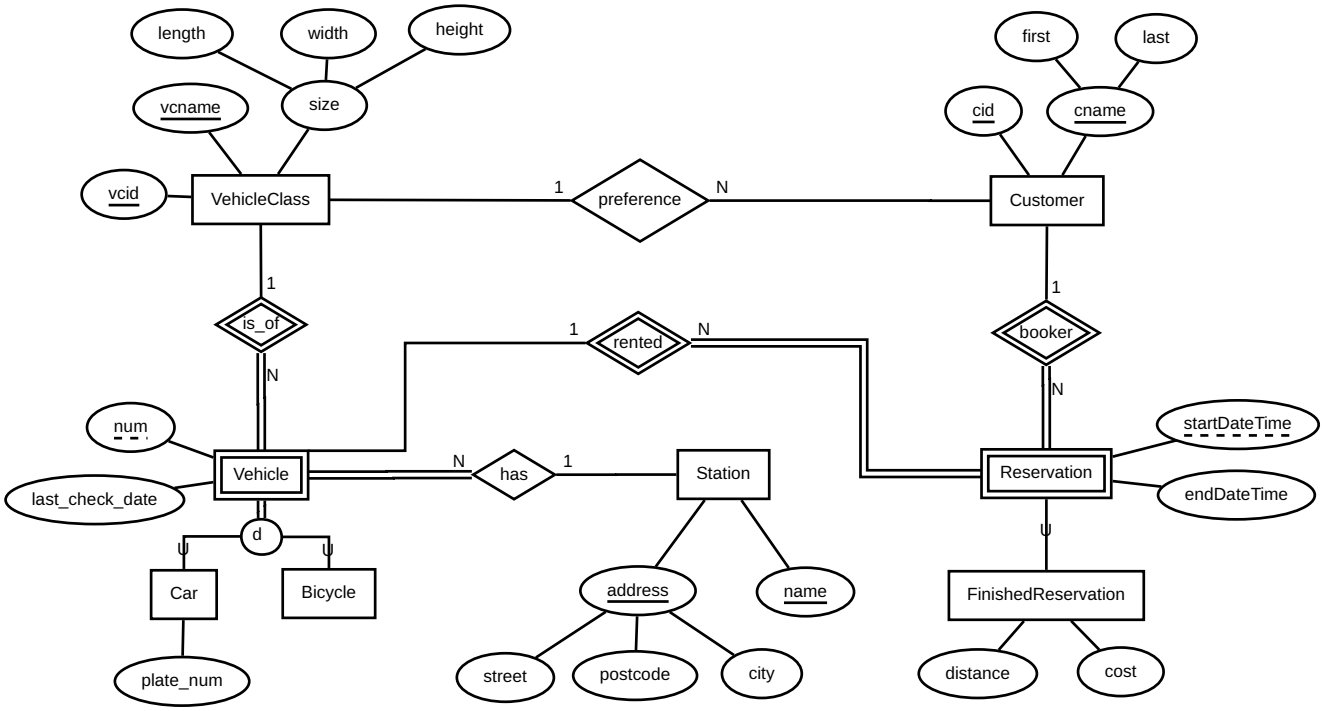
1. Use SQLite's functionality to determine the schemas of the relations (relation names and their attributes); furthermore, determine the primary keys and foreign keys in this database. Note that SQLite uses 'dot commands' for this purpose; see `http://www.w3resource.com/sqlite/sqlite-dot-commands.php`. Create an ER diagram, using Crow's foot notation, that closely reflects the physical design of this database. You may abbreviate numbered attributes using ... notation.

2. This database violates the first normal form. Propose an improved schema (giving relation names, their attribute names, primary keys, and foreign keys). You may use SQL to denote this schema, but this is not required.

This exercise determines 20% of your final grade.

3 Vehicle Sharing

Below is given a conceptual diagram using the ER-Chen notation for a database of a company that manages shared vehicles. These shared vehicles are either cars or bicycles. Each vehicle belongs to a vehicle class and has a fixed station to which it belongs. Customers can make reservations for the vehicles. For finished reservations, it is known which distance the vehicle traversed and at which cost.



Convert this ER diagram into a relational schema (relation names, attributes, primary keys, foreign keys), using the approach discussed in Elmasri's textbook and the lectures. Provide the necessary statements for creating this database in SQLite. Take into account these considerations:

- The database should satisfy **all** constraints specified in the conceptual schema: the database should only allow to store data such as represented in the schema;
- All relations created should be in 3rd Normal Form;
- While satisfying the above constraints, the database should have a minimal number of relations; consider representing relationships by adding attributes to relations, where possible;
- The database should have a minimal number of attributes that are NULLable;
- Explicitly describe which constraints could not be represented in the relational schema, if any (without restorting to assertions);
- Also specify in your schema what happens when a tuple is deleted from a relation.

This exercise determines 20% of your final grade.