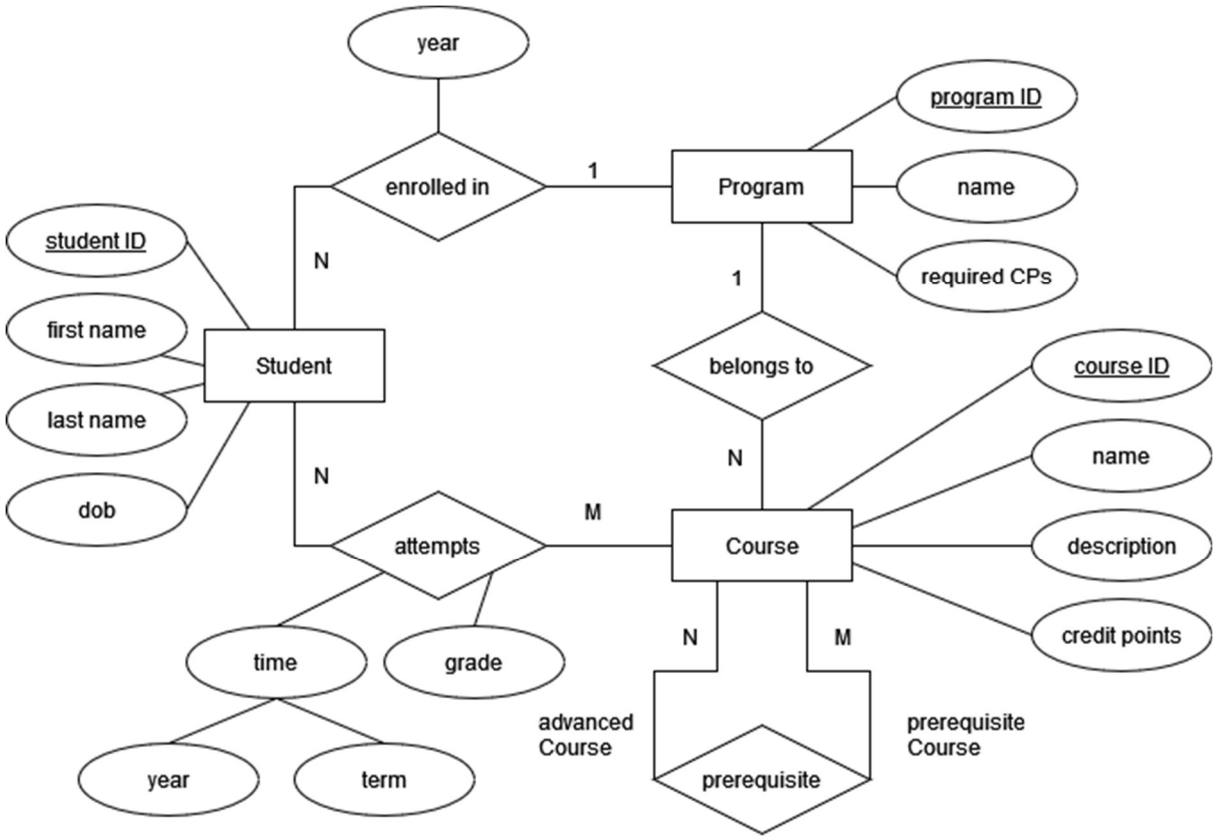
You should work on the following assignments in fixed teams of two. Please note that *every* team member must be able to explain *all* solutions of the team of two. Please submit only one solution for each team of two.

Deadline to upload your solution for assignments 1, 2, 3, and 4:
Sunday, 11:59 pm bevor the laboratory.
The remaining assignments can be done during the laboratory.

If you have questions or need any help, use the forum in our EMIL room und help each other.

Assignment 1: Relational Model of a Student Information System

Consider the following ERD for the Student Information System (analogous to Lab1 Assignment 1):

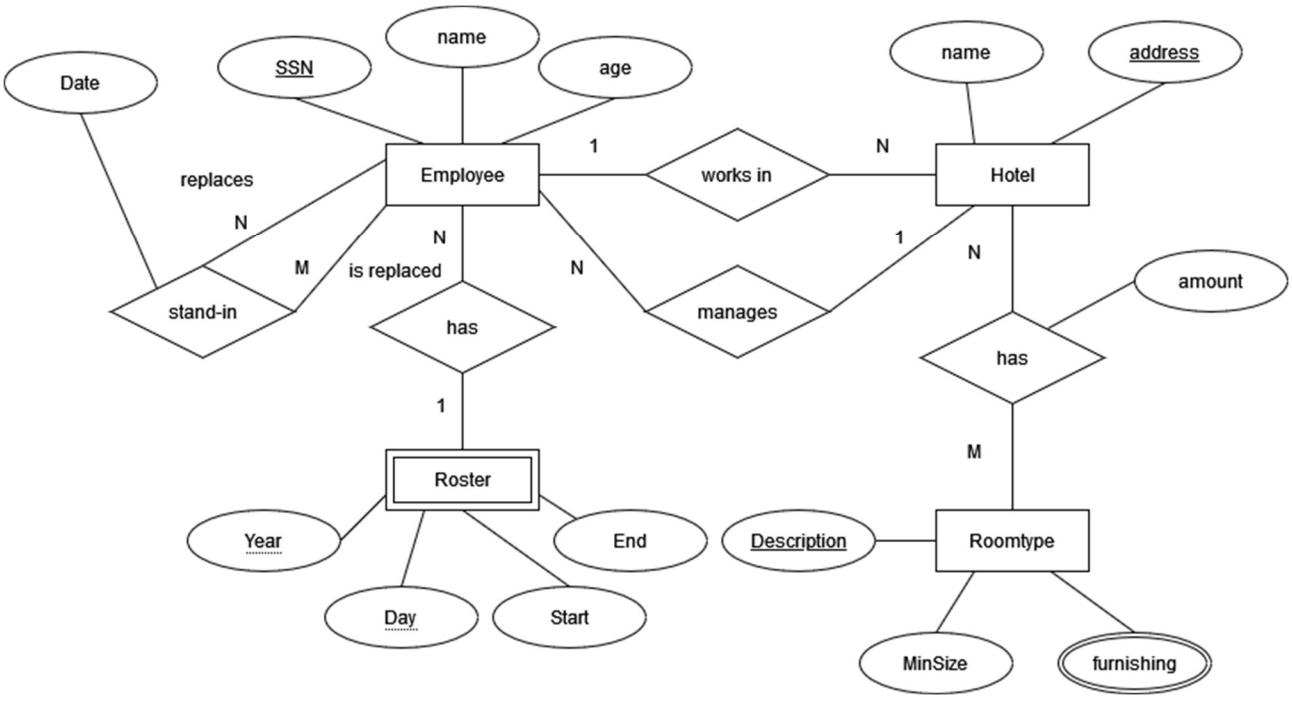


1.1. Convert the ERD into a relational schema. Follow the design guidelines from the lecture.

1.2. Give example data for the relational schema you created (you could for example give rows for your own study program / courses / attempts or any fictional data).

Assignment 2: Relational Model of a Hotel

A hotel chain wants to use a database to keep track of their hotels and employees, described in the following ERD:



In addition, take account of the following constraint: “Every stand-in is uniquely identifiable by the combination of the employee who replaces, the employee who is replaced and the date of the stand-in.”

Convert the ERD into a relation schema. Follow the design guidelines from the lecture.

Assignment 3: Functional Dependencies and Normalization

A university stores information about the quantities of copies of lecture notes sold by each professor. Lectures are taught by different professors, using different lecture notes.
The following sample database relation **SELLS** is used:

SELLS	lectId	lectName	profId	profName	noteId	price	quantity
	24	DB	47	Miller	5	32	12
	24	DB	272	Adams	1	35	15
	24	DB	251	Meyer	5	32	17
	25	Java	47	Miller	3	22	19

3.1 Determine the (full) functional dependencies. Keep in mind that FDs are determined by the model, not just by the actual data in the database relations.

3.2. Identify a primary key for the given relation **SELLS**. Explain your answer.

3.3. Transform the relational schema to 2NF. Your relation(s) should indicate PKs & FKs and contain all the data.

3.4. Transform the relational schema to 3NF. Your relation(s) should indicate PKs & FKs and contain all the sample data.

3.5. Create an ERD of the 3NF schema.

Assignment 4: Relational Algebra

The following excerpt of a database schema models a database of chess players. The relation **PLAYER** models a chess player, the relation **GAME** a game between two chess players. The attribute `remis` indicates whether the game is draw, it can be `true` or `false`.

PLAYER(PID, firstName, lastName, dateOfBirth)
GAME(GID, winnerID (FK), loserId (FK), remis)

The attributes `winnerID` and `loserID` are foreign keys to `PID` of **PLAYER**. Specify the following natural language queries as relational algebra queries.

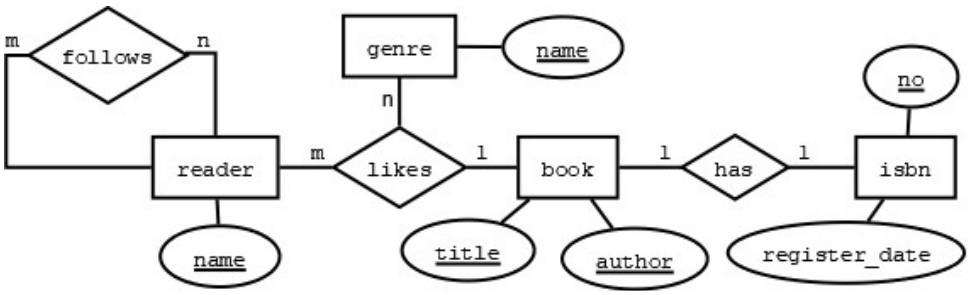
4.1 Which chess players have won at least once? (Output: first name and last name)

4.2 What different chess players were born on the same day? (Output: PID1, PID2)

4.3 Which chess players have won in at least two games? (Output: WinnerID)

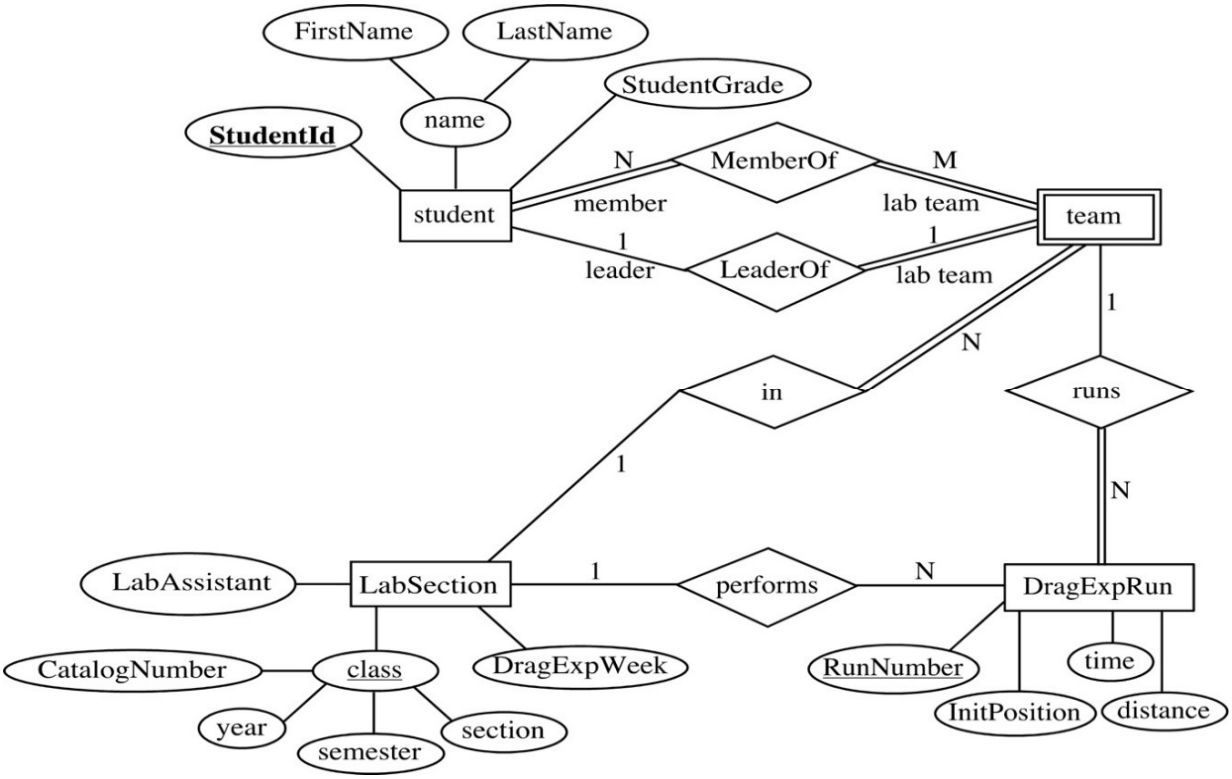
Assignment 5 - Relational Model for Favorite Books

Transform given ERD to an equivalent relational model.



Assignment 6 - Relational Model for the Organization of Laboratory

Transform given ERD to an equivalent relational model. Note that a double line indicates optional participation. The team may be identified by an id.



Assignment 7 – Normalization of a Shop

Consider the following relation **SHOP**:

```
SHOP(customerId, customerName, orderNr, orderedAt, articleNumber,
description, quantity, price, articleName)
```

and functional dependencies **F** that hold over this relation:

```
customerId              →    customerName
orderNr                 →    orderedAt, customerId, customerName
orderNr, articleNumber  →    quantity, customerId, customerName,
                             orderedAt, articleName, price, description
articleNumber           →    articleName, price, description
articleName             →    articleNumber
```

7.1 Determine all candidate keys (possible primary keys) of **SHOP**.

7.2 In which normal form is the relation **SHOP** (recall that a relation can be in multiple normal forms), and why?

7.3 If relation **SHOP** is not in 1NF, 2NF or 3NF then decompose it, accordingly, taking care to denote possible keys (both PKs and FKs).

Assignment 8 – Normalization for a General Example

Consider the following relation **ANY**:

```
ANY (aid, bid, cid, aName, bName, cName, since, description, location)
```

and functional dependencies **F** that hold over the relation **ANY**:

```
aid              →    aName
aName            →    aid
bid              →    bName, location, cid, cName, description
cid              →    cName, description
cName            →    cId
aName, bid       →    aid, bid, cid, aName, bName, cName, location,
                      since, description
```

8.1 Determine all candidate keys (possible primary keys) of **ANY**.

8.2 In which normal form is the relation **ANY** (recall that a relation can be in multiple normal forms), and why?

8.3 If relation **ANY** is not in 1NF, 2NF or 3NF then decompose it, accordingly, taking care to denote possible keys (both PKs and FKs).