CSIT115 Data Management and Security
Laboratory 2

Scope

This laboratory includes the following:

- tasks related to logical modelling and relational data model
- tasks related to quality evaluation of relational database design

Specification of each task starts from a new page.

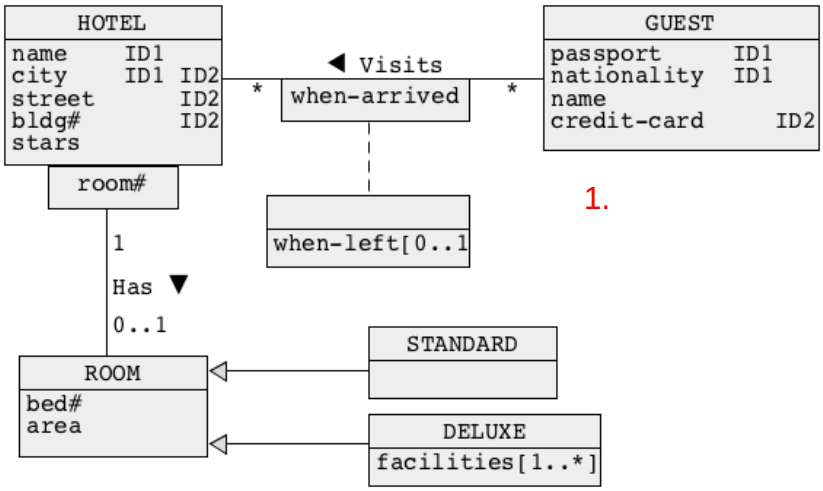
It is recommended to solve the problems before attending an enrolled laboratory class in order to efficiently use supervised laboratory time.

Task 1

Which situation need foreign key?

Consider a conceptual schema given below.

no need to add foreign key



1.

Your task is to perform a step of logical database design, i.e. to transform a conceptual schema given above into a collection of relational schemas.

For each relational schema created clearly list the names of attributes, primary key, candidate keys (if any), and foreign keys (if any). Assume, that **subset method** must be used to implement a generalization (if any). A way how a conceptual schema can be transformed into a collection of relational schemas is explained in a presentation 06 Logical Design.

The relational schemas must be listed in a format presented in the last few slides in a presentation 06 Logical Design.

1. Hotel (name, city, street, bldg#, stars)
   PK: (name, city)
   CK: (city, street, bldg#)

   Guest (passport, nationality, name, credit)
   PK: (passport, nationality)
   CK: (credit-card)

   Visits (name, city, passport, nationality, when-arrived, when-left)
   PK: (name, city, password, nationality, when-arrived)
   FK1: (name, city) references Hotel (name, city)
   FK2: (password, nationality) references Guest (passport, nationality)

   h1 c1 p1 n1 23-10-2022 25-10-2022
   h1 c1 p1 n1 23-10-2023 25-12-2023
   h1 c1 p2 n2 13-01-2024 15-01-2024

many rooms can have same facilities cannot be identifier

2. Standard(name, city, room#, bed#, area)
   PK: (name, city, room#)
   FK: (name, city) references Hotel (name, city)

   Deluxe(name, city, room#, bed#, area)
   PK:(name, city, room#)
   FK: (name, city) references Hotel (name, city)

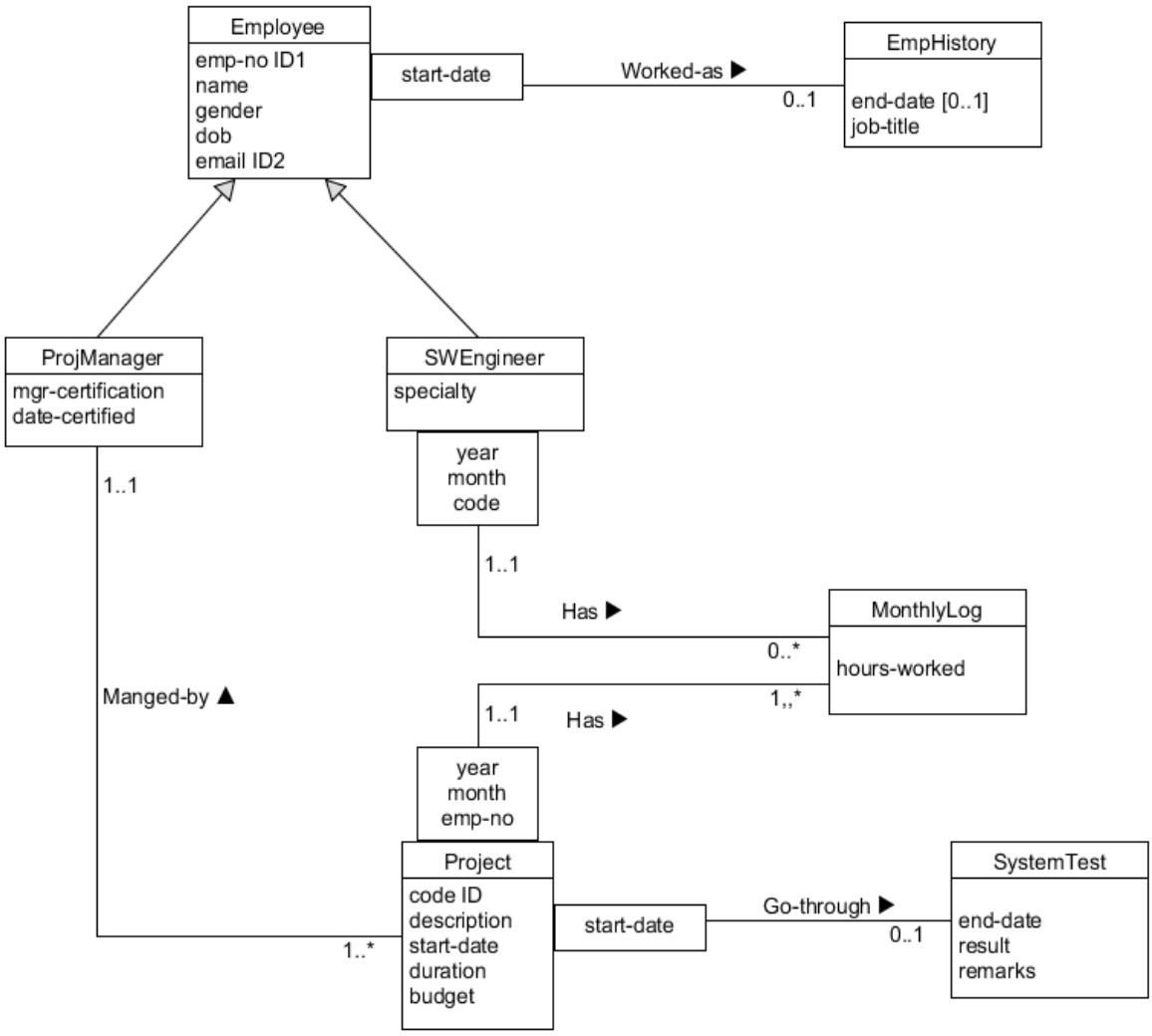
   Deluxe-facilities (name, city, room#, facility-type)
   PK: (name, city, room#, facility-type)
   FK: (name, city, room#) references Deluxe (name, city, room#)

   name city room# bed# facilities
   h1 c 1 r1 b1 pool/bar counter/big tv
   h1 c1 r2 b1 pool/bar counter/big tv

   name city room# facilities
   h1 c1 r1 Pool
   h1 c1 r1 Bar counter
   h1 c1 r1 Big TV
   h1 c1 r2 Big TV

Task 2

Consider a conceptual schema given below.



must managed by some person, should mention, so add foreign key

Your task is to perform a step of logical database design, i.e. to transform a conceptual schema given above into a collection of relational schemas.

For each relational schema created clearly list the names of attributes, primary key, candidate keys (if any), and foreign keys (if any). Assume, that **association method** must be used to implement a generalization (if any). A way how a conceptual schema can be transformed into a collection of relational schemas is explained in a presentation 06 Logical Design.

The relational schemas must be listed in a format presented in the last few slides in a presentation 06 Logical Design.

Task 3

Consider the following collection of relational schemas.

```
PROPERTY(PNum, City, Street, HouseNum, OwnerPhone, Price)
Primary key = (PNum)
Candidate key = (City, Street, HouseNum)
Foreign key = (OwnerPhone) references OWNER(OwnerPhone)

OWNER(OwnerPhone, OwnerName)
Primary key = (OwnerPhone)

BUYER(BuyerPhone, BuyerName, City, Street, HouseNum)
Primary key = (BuyerPhone)
Candidate key = (City, Street, HouseNum)

PREFERENCE(BuyerPhone, City, Street, MaxPrice, MinPrice, PDate)
Primary key = (BuyerPhone, PDate)
Foreign key = (BuyerPhone) references BUYER(BuyerPhone)

INSPECTION(BuyerPhone, PNum, IDate)
Primary key = (BuyerPhone, PNum,  IDate)
Foreign key 1 = (BuyerPhone) references BUYER(BuyerPhone)
Foreign key 2 = (PNum) references PROPERTY(PNum)
```

Your task is to perform *reverse database engineering*, i.e. to find a conceptual schema of a database that has a collection of relational schemas given above. Use UMLetlet to draw a conceptual schema found. Use an option File->Export as… to export your diagram into a file `solution2.bmp` in `BMP` format.

Task 4

Analyze the relational schemas listed below and find the schemas that are incorrectly designed. A schema is incorrectly designed when the insertions of rows into the schema creates the redundancies. Consider a method of row insertions explained in a presentation 07 Database Design Quality to find out which relational schemas are incorrectly designed.

Next, for each incorrectly designed relational schema propose its decomposition into a pair of correctly designed relational schemas and equivalent to the original one. A pair of relational schemas is equivalent to other relational schemas when the respective pair of relational tables contains exactly the same information as a relational table built on the original schema.

(1) `PLAYER(pnum, pname, team, position, team-address)`
A relational schema `PLAYER` is a header of a relational table that contains information about the players and teams the players belong to. A player number (`pnum`) uniquely identifies each player in a team; and team name (`team`) uniquely identifies the team address (`team-address`). The attribute `position` describes the player's position in a team.

(2) `WAREHOUSE(name, address, item, price, quantity )`
A relational schema `WAREHOUSE` is a header of a relational table that contains information about the warehouses and items kept there. An attribute `address` is an address of a warehouse and it uniquely identifies each warehouse. An attribute `name` is name of a warehouse and it also uniquely identifies each warehouse. An attribute `item` is a unique name of an item. A pair of attributes (`address`, `item`) uniquely identify the item's unit price (attribute `price`) and quantity (attribute `quantity`) of the item.

(3) `FLIGHT(passenger, flightNum, seatType, departurePlace, departureTime)`
A relational schema `FLIGHT` is a header of a relational table that contains information about the flights booked by the passengers. A passenger is uniquely identified by a value of an attribute `passenger`. A passenger books seats (`seatType`) on many different flights (`flightNum`). Only one flight (`flightNum`) takes off from a given departure place (`departurePlace`) at a given departure time (`departureTime`). A flight number (`flightNum`) uniquely determines departure place (`departurePlace`) and departure time (`departureTime`).

(4) `ENROLMENT(studentNum, subjectCode, enrolDate, IP)`
A relational schema `ENROLMENT` is a header of a relational table that contains information about the students enrolled in the subjects (`subjectCode`). A triple of attributes (`studentNum`, `subjectCode`, `enrolDate`) uniquely identifies each row in the relational table. An attribute `IP` is an IP address of a computer from where an enrolment has been done.

(5) `CAR(regoNum, manufacturer, model, year, colour)`
A relational table `CAR` contains the descriptions of cars. A registration number (`regoNum`) uniquely identifies each car. A car is described by a manufacturer name (`manufacturer`), model name (`model`), year when manufactured (`year`), and colour (`colour`).

No new attributes can be added to the relational schemas listed above !

The relational schemas must be listed in a format presented in the slides 44-45 in a presentation 06 Logical Design.

For each one of the cases from (1) to (5) write either a statement:

A relational schema <schema-name> is designed correctly

or a statement:

A relational schema <schema-name> is designed incorrectly

<schema-name> must be replaced with a name of a respective relational schema.

For each incorrectly designed schema, show the results from an insertion test that reveal the redundancies in a relational table that has incorrectly designed relational schema. Insert precisely 3 rows into a relational table.

Additionally, for each incorrectly designed relational schema list a correctly designed relational schemas obtained from a decomposition of the incorrect schema. The relational schemas must be listed in a format presented in the last few slides in a presentation 06 Logical Design.