Tentamen Databases 1, 29 mei 2009

1. UML: The Webshop

A web shop has three divisions, Books, Music and Games. The web shop sells a number of products, which are divided in three categories, Books, (music) CD's and games. Books are sold by the Books division, etc. Each product has a unique internal product code (IPC), a description and a price. Books may have an ISBN number. Books also have one or more authors, CD's one or more artists and games have one or more creators. The products are furthermore divided into subcategories, like classic, rock and techno for music, etc. Each product is associated to exactly one such subcategory.

For each division its (potential) customers must be maintained. Each customer has a firstname, lastname, address (street, house number city ans zip-code) and one or more telephone numbers. A customer may be known by more than one division (at least one).

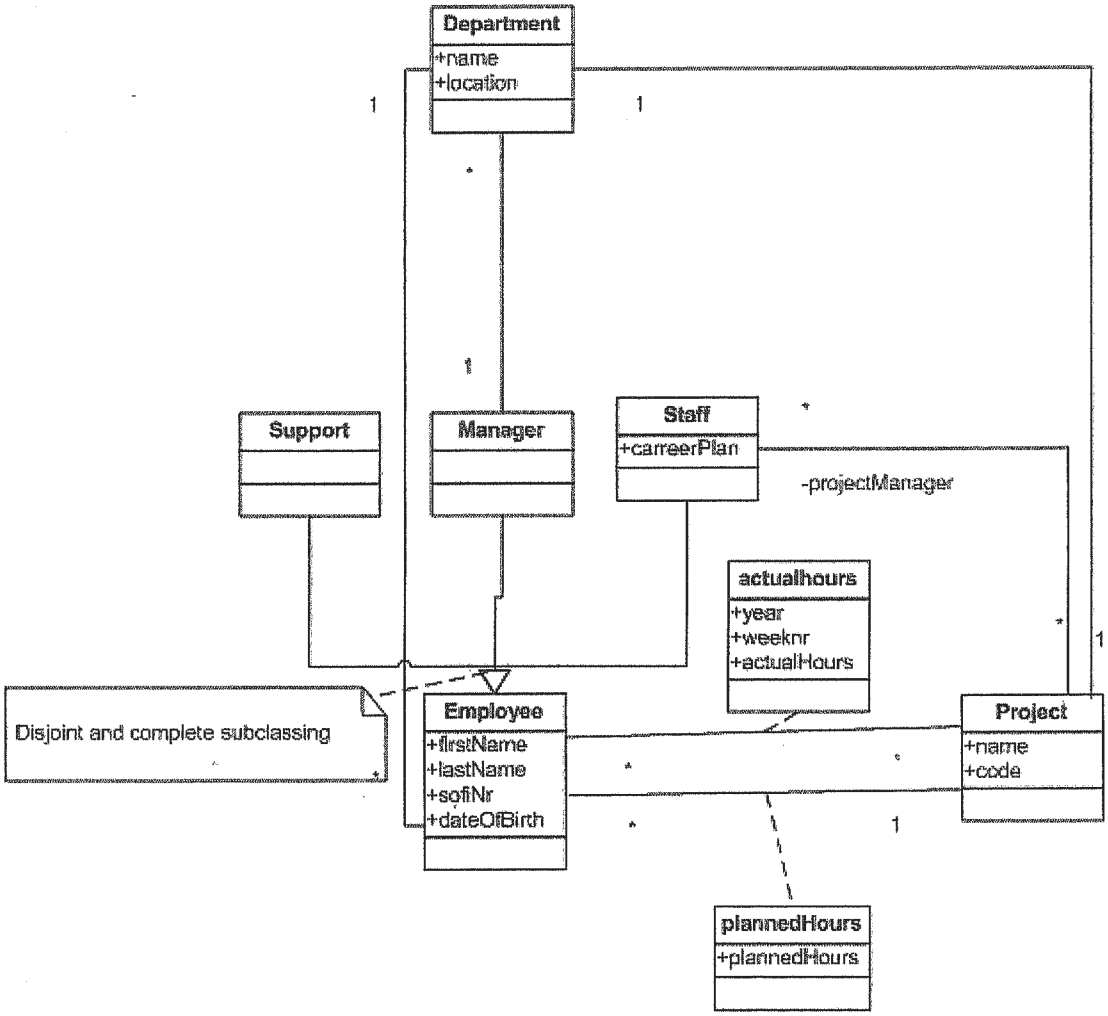
Each division has a manager. Employees can be divided into sales, after-sales and administrative employees. Each employee (including the managers) has a first name, last name, Social Security Number, telephone number and date of birth. A sales manager is responsible for a number of subcategories of products, a subcategory falls under exactly one sales manager. Sales managers also have a target. An after-sales employee has one or more specializations, which correspond with the above mentioned subcategories.

Customers can book an order. Each order has an order date, one or more product codes and per product code a number of ordered items.Directly after the booking the order is assigned to an after-sales employee, who will be responsible for handling the order.

Customers can issue complaints, which must be connected to an order. Furthermore, a complaint has date (at which it was issued), a closing date (initially unknown), a description and possibly one ore more product (one of the products of the associated order).

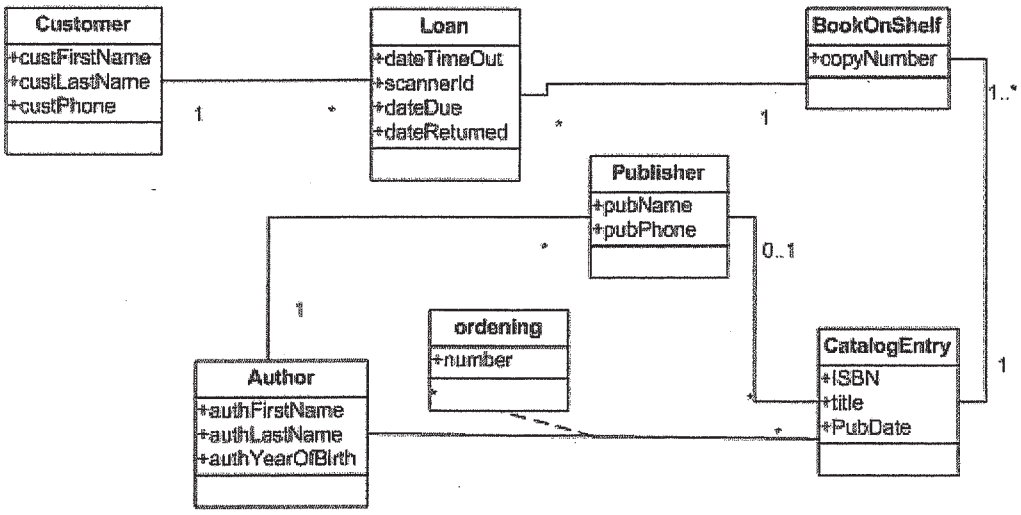
2. Relational Model

Give the relational model of the following UML model. I.e., give a list of tables, for each table give all the attributes, the primary key and possible candidate keys and foreign keys.



3. SQL

Consider the following UML model and its relational model, as we have used in our practical as well. Note that not all details are relevant for the questions below.




```
create table Customer (
    custId                      INT NOT NULL AUTO_INCREMENT,
    custFirstName               VARCHAR (100),
    custLastName                VARCHAR (100),
    custPhone                   VARCHAR (100),
    PRIMARY KEY (custId));

create table Loan (
    custId                      INT NOT NULL,
    copyId                      INT NOT NULL,
    dateTimeOut                 DATETIME,
    scannerId                   INT,
    dateDue                     DATE,
    dateReturned                DATE,
    PRIMARY KEY (dateTimeOut, scannerId));

create table BookOnShelf (
    copyId                      INT NOT NULL AUTO_INCREMENT,
    bookId                      INT NOT NULL,
    copyNumber                  INT,
    PRIMARY KEY (copyId));

create table Publisher (
    pubId                       INT NOT NULL AUTO_INCREMENT,
    pubName                     VARCHAR (255) NOT NULL,
    pubPhone                    BIGINT,
    PRIMARY KEY (pubId));

create table CatalogEntry (
    bookId                      INT NOT NULL AUTO_INCREMENT,
    pubId                       INT,
```

```
    ISBN                        VARCHAR(20),
    title                       VARCHAR (255),
    pubYear                     int,
    PRIMARY KEY (bookId));

create table Author (
    authId                      INT NOT NULL AUTO_INCREMENT,
    authFirstName               VARCHAR (100) NOT NULL,
    authLastName                VARCHAR (100) NOT NULL,
    authDateOfBirth             DATE NOT NULL,
    pubId                       INT,
    PRIMARY KEY (authId));

create table AuthorOfBook (
    bookId                      INT NOT NULL,
    authId                      INT NOT NULL,
    number                      INT,
    PRIMARY KEY (bookId, authId));
```

Answer the following queries by formalizing them in SQL.

3.A. Give the names of customers who have never borrowed any book.

3.B. Give for each book the number of overdue outstanding loans (i.e., the number of loaned copies which have not yet returned although they should have been returned already). . (Hint, use `dateDue < sysdate()`. Be sure that you limit the output to books that have at least one such overdue outstanding loan

3.C Give the customers that have borrowed all books of the publisher Addison Wesley (not necessarily at the same time of course).

4. Normalisation

Consider the the table with the following attributes

- (CC) CourseCode
- (CN) CourseName
- (YR) Year
- (L C) Lecturer
- (RN) Rank
- (CR) ClassRoom
- (ST) StudentNumber
- (GR) Grade

The key of this table is (CC, YR, ST), it has the following functional dependencies

- CC $\rightarrow$ CN
- CC, YR $\rightarrow$ LC, CR
- LC $\rightarrow$ RN
- CC, YR, ST $\rightarrow$ GR

4.A Explain the design problems of this table

4.B Explain the normal form (NF) of this table

4.C Decompose the table (if needed) such that the resulting tables are in 3 NF

4.D Show that this decomposition satisfies the lossless join property

5. Concurrency

Given the two processes ("*r* " stands for "read", "*w*" stands for "write"):

- P1: r1(X); X := X+1; w1(X); r1(Y); Y := Y -1; w1(Y)
- P2: r2(X); X := 2 * X; w2(X)

These two processes can be executed according to different schedules, (aschedule is a sequence of actions of P1 and P2).

5.A..Give an example of a serial schedule

5.B. Give an example of a serializable schedule, give also the dependency graph and the associated serial schedule.

5.C Give a conflicting schedule that suffers from the lost update problem, show the conflict in the dependency graph.

Ranking

1		20
2		10
3a	10	
3b	12	
3c	13	
3		35
4a	3	
4b	3	
4c	5	
4d	5	
4		16
5a	3	
5b	3	
5c	3	
5		9
for free		10
total		100