Databases and Web Applications (320302)

Midterm Exam Fall 2008

Logistics

- You have **one hour fifteen minutes** (sharp) for the test.
- Don’t forget to enter your name – we cannot grade if not present or illegible!
- *Different problems test different skills and knowledge, so do not get stuck on one problem.*

Name:

(To be used for correcting, do not write into box below)

Task	1.1	1.2	1.3	2.1	2.2	2.3	3.1	3.2	4.1	4.2	5.1	Total
Total	**7**	**7**	**2**	**2**	**4**	**6**	**3**	**3**	**2**	**2**	**5**	**41**
Reached												

1 Data Modeling

Given the following information about an entertainment store:

1) The stores sell only music and video CD's and each CD has a title, duration.

2) In addition, each music CD has a genre, number of tracks, while each video CD has a box office rating and date it was premiered.

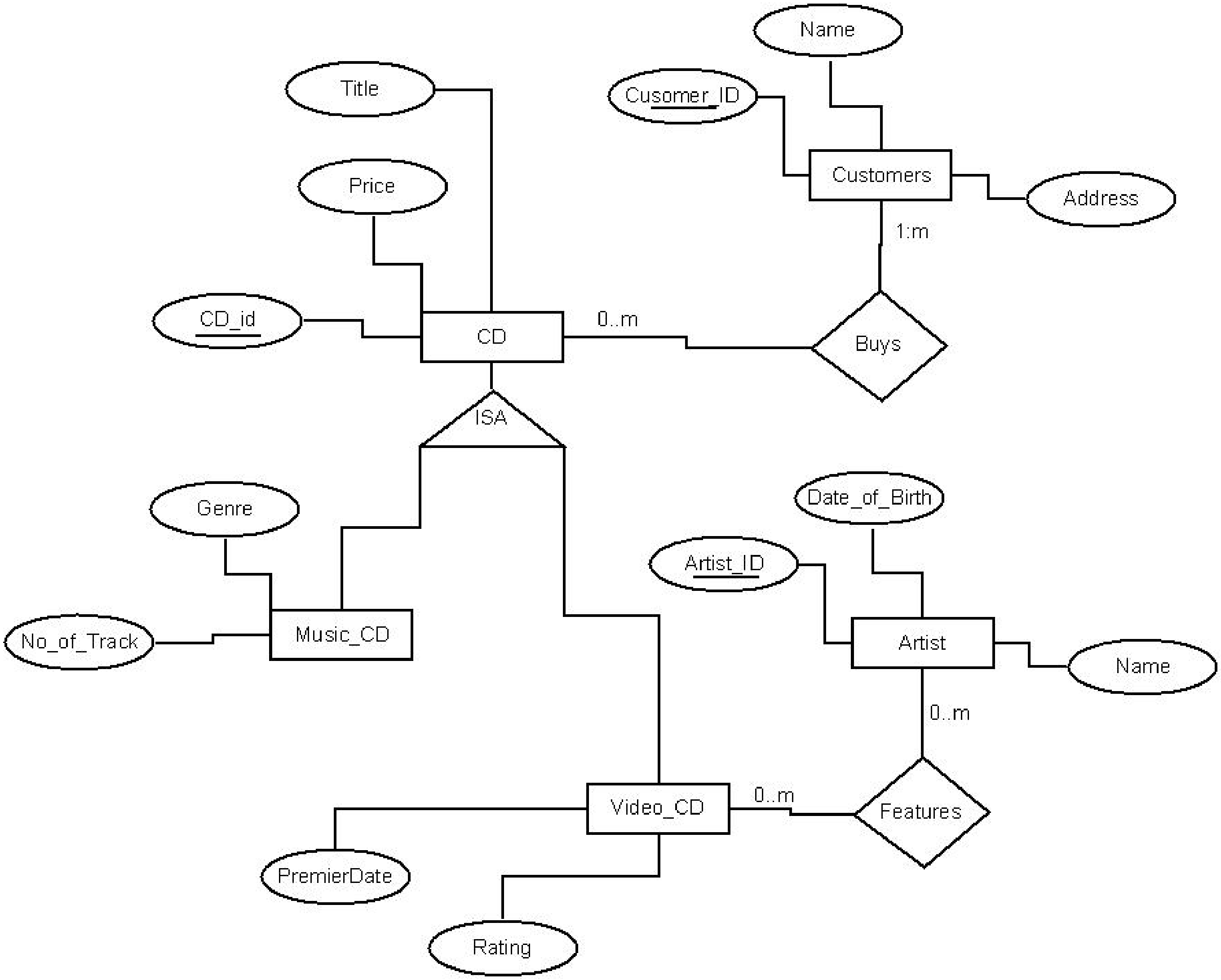
3) Each video CD must feature at least one artist, and an artist can feature in any number video of CD's

4) Each artist has a name, and date of birth.

5) The shop also keeps the list of its customers (with a name and address) and the CD's they buy. Each customer must buy at least one CD for his/er name to be entered in the list.

Task 1.1 (7 pts): Draw the Entity-Relationship diagram for the application above.

Solution:



Task 1.2 (7 pts): Write SQL statements to create the relations corresponding to the ER diagram above. Capture as many of the constraints as possible (including primary key and foreign key information). If you cannot capture some constraints, explain why.

Solution:

```
CREATE TABLE CD(

    CD_ID INT,
    TITLE CHAR(20),
    PRICE INTEGER,
    PRIMARY KEY(CD_ID));

CREATE TABLE MUSIC_CD(
      GENRE CHAR(20),
    NUMBER_OF_TRACK INTEGER,
    CD_ID INTEGER,
   PRIMARY KEY(CD_ID)
   FOREIGN KEY(CD_ID) REFERENCE CD ON DELETE CASCADE);

CREATE VCD (
    BOX_OFFICE_RATING INTEGER,
    PREMIER_DATE DATE,
    CD_ID INTEGER,
    PRIMARY KEY(CD_ID)
    FOREIGN KEY(CD_ID) REFERENCES CD ON DELETE CASCADE);

CREATE FEATURES(
    VCD_ID INT,
    ARTIST_ID INT,
    PRIMARY KEY(CD_ID, ARTSIT_ID),
    FOREIGN KEY(CD_ID) REFERENCES VCD,
    FOREIGN KEY(ARTIST_ID) REFERENCES ARTIST);

CREATE TABLE ARTSIT(
    ARTIST ID INT,
    NAME CHAR(30),
    DATE_OF_BIRTH DATE,
    PRIMARY KEY(ARTIST_ID));

CREATE TABLE CUSTOMER(
    CUSTOMER_ID INTEGER,
    NAME CHAR(20),
    ADDRESS CHAR”(50),
    PRIMARY KEY (CUSTMER_ID)),

CREATE TABLE BUYS(
    CUSTOMER_ID INTEGER,
    VCD_ID INTEGER,
    PRIMARY KEY (CUSTOMER_ID, VCD_ID)
    FOREIGN KEY (CUSTOMER_ID) REFERENCES CUSTOMER,
    FOREIGN KEY(VCD_ID) REFERENCES VCD);
```

Task 1:3 (2 pts): Discuss the two types of relational modeling alternatives for ISA, stating for each one its advantages and disadvantages.

Solution:

Two ways of modeling ISA relationship in relational mode are:

a) Separate relation per entity set,i.e a superclass entity set is created which is always referenced by the entity set of a subclass.

b) Create a relation each for each entity set (subclass) that has instances. All the attributes for each entity set(subclass) is in its corresponding relation and there is no need to referenced to a superclass. Each entity must be in any of the subclasses

2 SQL Queries

The following relations keep track of airline flight information:

Flights (flno: integer, *from:* string, *to:* string, *distance:* integer, *departs:* time, arrives: time, price: integer)

Aircraft (aid*:* integer, *aname:* string, *cruisingrange:* integer)

Certified (*eid:* integer, *aid:* integer)

Employees (*eid:* integer, *ename:* string, *salary:* integer)

Note that the Employees relation describes pilots and other kinds of employees as well; every pilot is certified for some aircraft, and only pilots are certified to fly.

Write each of the following queries in SQL:

Task 2.1 (2 pts): Find the eid's of pilots certified for some Boeing aircraft.

Solution:

```
SELECT C.eid
FROM Aircraft A, Certified C WHERE A.aid = C.aid AND A.aname = `Boeing'
```

Task 2.2 (4 pts): Compute the difference between the average salary of a pilot and the average salary of all employees (including pilots).

Hint: You may want to use sub-queries in the FROM clause.

Solution:

```
SELECT      avg(Temp1) – avg(Temp2)
FROM (SELECT E.salary FROM Employees E, Certified E WHERE E.eid = C.eid) Temp1,
        (SELECT E.salary FROM Employees E) Temp2
```

Task 2.3 (6 pts): Print the enames of pilots who can operate planes with cruisingrange greater than 3000 miles but are not certified on any Boeing aircraft.

Solution:

```
SELECT DISTINCT E.ename
FROM Employees E
WHERE E.eid IN ( ( SELECT C.eid
                FROM Certified C
                WHERE EXISTS
                     ( SELECT A.aid
                     FROM Aircraft A
```

```
            WHERE A.aid = C.aid AND A.cruisingrange > 3000 )
                AND NOT EXISTS
                    ( SELECT A1.aid
                    FROM Aircraft A1
                    WHERE A1.aid = C.aid AND A1.aname LIKE `Boeing%' ))
```

3 Relational Concepts

Task 3.1 (3 Points): What are the differences between relational calculus and relational algebra? Also, compare both in terms of their expressiveness.

Solution:

Relational algebra is a set-based formalization of selection, projection, cross product (no aggregation!), relational calculus is expressed in predicate logic

Relational algebra is operation oriented i.e. it states the procedure to follow to compute a result, but Relational Calculus is declarative i.e. it allows us to declare the answer without being explicit about how they are computed

Both Relational Algebra and Relational Calculus have equal expressive power.

Task 3.2 (3 Points): Given tables S1 and S2, what is the equivalent SQL operator name for these set expressions?

1) $[\ t \mid t \in S1 \vee t \in S2\ ]$
2) $[\ t \mid t \in S1 \wedge t \in S2\ ]$
3) $[\ t \mid t \in S1 \wedge t \notin S2\ ]$

Solution:

1) $[\ t \mid t \in S1 \vee t \in S2\ ]$ = UNION
2) $[\ t \mid t \in S1 \wedge t \in S2\ ]$ = INTERSECTION
3) $[\ t \mid t \in S1 \wedge t \notin S2\ ]$ = EXCEPT

4 Database Application Programming

Task 4.1 (2 point): Explain the following terms: Cursor, Embedded SQL, JDBC, SQLJ, stored procedure.

Solution:

A cursor enables individual row access of a relation by positioning itself at a row and reading its contents. Embedded SQL refers to the usage of SQL commands within a host program. JDBC stands for Java DataBase Connectivity and is an interface that allows a Java program to easily connect to any database system. SQLJ is a tool that allows SQL to be embedded directly into a Java program. A stored procedure is program that runs on the database server and can be called with a single SQL statement.

Task 4.2 (2 points): What are the differences between JDBC and SQLJ? Why do both techniques exist?

Solution:

SQLJ provides embedded SQL statements. These SQL statements are static in nature and thus are preprocessed and precompiled. For instance, syntax checking and schema checking are done at compile time. JDBC allows dynamic queries that are checked at runtime. SQLJ is easier to use than JDBC and is often a better option for static queries. For dynamic queries, JDBC must still be used.

5 Web Applications

A library wants to implement a web interface for searching its book catalogue in the library database. The books can be searched by specifying a maximum of two search fields, and the fields of search can be any combination of Name of Book, Year of Publication, Author and Publisher. Information to be retrieved is the ISBN of the books and abstract of the books.

Your task is to design the HTML page which submits a search form which includes the library_id and password of that user.

Task 5.1 (5 pts): Write a full HTML page which allows the user to input its library ID, password, and two search texts corresponding to the two search criteria selected from a two lists.

Solution:

```
<html>

<head>

  <title>Search Library Catalogue</title>

</head>

<body>

<form method="POST" action="search.psp">

  Library ID: <input type="text" name="ID">

  <br>

Password: <input type="password" name="password">

<br>

<select name="search_field_1">

    <option>Name of Book</option>

    <option>Year of Publication</option>
    <option>Author</option>
    <option>Publisher</option>
</select>
&nbsp; : &nbsp
<input type="text" name="search_field_1_name">
<br>
<select name="search_field_2">
    <option>Name of Book</option>
    <option>Year of Publication</option>
    <option>Author</option>
    <option>Publisher</option>
</select>
&nbsp; : &nbsp
<input type="text" name="search_field_2_name">
<br>
<input type="submit" value = "submit">
</form>
</body>
</html>
```

--*end of exam*--