

## CMPSCI 445 — Homework 2

80 Points

Due October 6th, at the beginning of class.

Typed solutions preferred. If hand-written, solutions must be **legible**.

1. (5 pts) **Views**

Briefly answer the following questions based on this schema:

Emp(eid: integer, ename: string, age: integer, salary: real)

(a) Suppose you have a view SeniorEmp defined as follows:

```
CREATE VIEW SeniorEmp (sname, sage, salary)
AS SELECT E.ename, E.age, E.salary
FROM Emp E
WHERE E.age > 50
```

Explain what the system will do to process the following query. That is, write a query equivalent to the query below, but without mentioning the view SeniorEmp:

```
SELECT S.sname
FROM SeniorEmp S
WHERE S.salary > 100,000
```

2. (10 points) **More on updating views**

Given two tables Students(sid, name, age) and Enrolled(studid, cid, grade) consider the view  $V$  defined as follows:

```
CREATE VIEW Bgrades () AS
SELECT name, sid, cid
FROM Students as S, Enrolled as E
WHERE S.sid = E.studid AND grade = "B"
```

The view result consists of tuples of the form (name, sid, cid) and suppose  $t = (Joe, 8250, 445)$  is an example tuple in the result. For each of the following update operations on  $V$ , describe how they can be translated into operations on Students and/or Enrolled, and any complications involved. Mention any side effects of operations you propose on the base relations.

- (a) Insert a new tuple (Mary, 8251, 645) into  $V$ .
- (b) Delete tuple  $t$  from  $V$ .

3. (25 pts) **ER Diagrams**

Although you always wanted to be an artist, you ended up being an expert on databases because you love to cook data and you somehow confused database with data baste. Your old love is still there, however, so you set up a database company, ArtBase, that builds a product for art galleries. The core of this product is a database with a schema that captures all the information that galleries need to maintain. Galleries keep information about artists, their names (which are unique), birthplaces, age, and style of art. For each piece of artwork, the artist, the year it was made, its unique title, its type of art (e.g., painting, lithograph, sculpture, photograph), and its price must be stored. Pieces of artwork are also classified into groups of various kinds, for example, portraits, still lifes, works by Picasso, or works of the 19th century; a given piece may belong to more than one group. Each group is identified by a name (like those just given) that describes the group. Finally, galleries keep information about customers. For each customer, galleries keep that persons unique name, address, total amount of dollars spent in the gallery (very important!), and the artists and groups of art that the customer tends to like.

Draw the ER diagram for the database. Indicate one-to-many or one-to-one relationships with arrows where appropriate.

4. (20 pts) **ER Diagram to Relations**

Write SQL statements to create the corresponding relations to the ER diagram you designed above. You should write SQL CREATE TABLE commands and specify primary keys and foreign keys.

5. (20 points) **Schema Refinement**

Consider a schema  $\text{Student}(\underline{\text{ssn}}, \text{name}, \text{groupID}, \text{group-name}, \text{project-grade})$ . A natural functional dependency on this table is  $\text{groupID} \rightarrow \text{group-name}, \text{project-grade}$ . This means that the value of  $\text{groupID}$  determines a unique value for both  $\text{group-name}$  and  $\text{project-grade}$ .

- (a) Construct a database instance for this schema (with say 6 tuples) that satisfies this functional dependency (and also helps you to complete the remaining problems below).
- (b) Indicate a single change to one value in the database that would violate the functional dependency and explain the violation.
- (c) Give an example of an update anomaly on this table.
- (d) Give an example of a deletion anomaly on this table.
- (e) Propose a decomposition of the schema above to resolve these anomalies. Indicate the schemas of the new tables and show the decomposed instances of the table in part (a).