**ISEM 3006 Data Management in Business**

**Individual Assignment**

**Answers**

**Question 1**

**Use the Trustful Property Rental Company Database in Exercise 3.**

**Use PostgreSQL SELECT statement to answer the following queries.**

**Unless state otherwise, the case of strings is important, i.e. if I ask you to retrieve properties that are located in 'KLN', you only retrieve properties that are located in 'KLN', but not 'kln', 'Kln', 'kLn', etc.**

1. For the three properties that have the most viewings, list the property number and property street. There may be tie, i.e., if the fourth property is having the same number of viewings as the third one, you should also list the ~~third~~ FOURTH property. (15 points)

SELECT P.PropertyNo, PropertyStreet

FROM Viewing V JOIN Property P

ON V.PropertyNo = P.PropertyNo

GROUP BY P.PropertyNo

HAVING COUNT(ViewingNo) IN (SELECT COUNT(ViewingNo)

FROM Viewing

GROUP BY PropertyNo

ORDER BY COUNT(ViewingNo) DESC

FETCH FIRST 3 ROWS ONLY)

1. Explain your logic. (5 points)

First we need to find out the 3 properties that have the most viewings. We need to find out the number of viewings of each property, order the rows by the number of viewings in descending order, then the top three rows will be the properties that have the top three viewing. The data is in the viewing table, if we group the record based on PropertyNo, then count the number of rows in each group, we can find out the number of viewings for each property, then we order the summarized table by number of viewings (COUNT(ViewingNo) in descending order, then fetch the first 3 rows.

Since there may be tie, that’s why we need to find out the number of viewings for the top three properties, if other properties have the same number of viewings, then we also need to list them out. That’s why in the subquery, we project the COUNT(ViewingNo), which will return a column of the top three number of viewings. Then the outer query is using the same logic as in the above paragraph to find out the number of viewings for each property and compare it with the top three numbers got from the subquery using the HAVING clause. Since we need to project the property street, that’s why we need to join the Viewing and the Property tables.

**Question 2 (30 points)**

You are one of the organizer of a poster exhibition on Global Issues and you must design a database to keep track of the administration of the exhibition.

Graphic designers create posters for the exhibition to illustrate one of the chosen global problems. Relevant information on designers includes their name, phone number, email address, and the name of the organizations they work for. A unique designer code will be given to the designer when he or she submits the first poster. The submitted poster must has a title and is assigned an identification number. A poster may be created by several graphic designers. The percentage of the contribution of each designer in creating the poster needs to be recorded. Each individual designer may be involved in at most 5 posters. Where a group of graphic designers create a poster, a designer needs to be designated as the contact person. This information needs to be recorded. A poster may receive an award from the organizer. There are different awards. There may be multiple posters that can receive the same award. However, one poster can only have one award. Each award is identified by an award name. Other information includes the amount of the prize and the maximum number of posters that can receive the award.

Construct an entity-relationship model according to the above description Add any entity types or attributes that are required to support the operations. **Resolve the many-to-many relationships**. List the attributes with the entity type, underline the primary key, and **put the foreign keys into the proper entity type** so that the entity types can be converted to relational tables directly. Be sure to state any **reasonable and necessary** assumptions that you have made.

ER before breaking down the many-to-many relationships and put the foreign keys into the correct entity types. (I did not ask you to provide this, I put it here to make you better understand the process of developing the ER.)



ER after breaking down the many-to-many relationships and put the foreign keys into the correct entity types.



Explanation (I did not ask you to do this.)

Designer and Poster (‘Hand In’ relationship)

A designer can hand in one or more posters and a poster can be created by one or more designers. As the system only record the designer information when he or she submitted the first poster, that’s why a designer must has at least one poster. The Contribution of a designer to a poster in fact is the attribute of the Hand In relationships.

The Poster Designer associative entity type (resulted from the breaking down of the many-to-many relationship) records the information about the designer contributes to a poster. A designer can have one to many Poster Designer records because a designer must submitted at least one poster before his or her information is recorded. A Poster Designer must be submitted by one designer. A poster can be created by one to many poster designers, but a Poster Designer must be referring to a particular poster.

Designer and Poster (‘Is contact person of’ relationship)

A poster must have one contact person, who is one of the designer of that poster. A designer can be a contact person of one or more posters. However, some designers may also not be a contact person of any poster that he or she submitted with teammates as other teammates may be the contact person.

In the Poster entity type, the Contact Designer Code is the foreign key that refers to the primary key, Designer Code, of the Designer entity type for the contact person.

Award and Poster

As stated, one poster can only receive at most one award and some poster may not receive any award. An award may be given to one or more posters. However, before the announcement of the award, no poster will be associated with the award.