****

**Relational Database**

Wednesday, April 17, 2019, (11:00 am – 2:00 pm)

Name: \_\_\_\_\_Jeehyuep Kwon\_\_\_\_ Student Number: \_\_\_\_200393004\_\_\_\_\_\_

**Please read the instructions:**

1. No textbooks are allowed.

2. Write down your name and student number on this exam page.

3. Communication with other people is not allowed during the exam. You may only communicate with the instructor or the invigilator(s).

4. Allowed Applications: MySQL Workbench, a single text editor for practice, Word for the submission of the assignment and a page \***ONLY\*.**

5. **No** web browsing, chatting, emailing or any internet-related activity is allowed!

**MIDTERM EXAMINATION**

This test is divided into two parts (Section A and Section B). In Section A, you should develop your database and in Section B, you should syntax the Joins and any other Queries. You should review the requirements of both Sections in order to develop the appropriate tables. Save this document in your PC and change the name to your full name + studentid e.g. GeorgeKougioumtzoglou100684468. Then type or copy and paste all the commands one by one in the Answers section at the bottom of this document. Save the document one last time and submit on the Blackboard BEFORE the deadline.

**Section A**

Read all the requirements and follow the steps.

1. Create a new database named GameofThronesSet (1 point).
2. Start a new simple Transaction (2 points).
3. Create AT LEAST the tables below, creating AT LEAST the columns in the brackets (15 points):
   1. Actors: Actor’s names and info (ActorID, ActorFName, ActorLName, AvailabilityStart, AvailabilityEnd, LocationID, LocationName).
   2. Locations: Setting’s Location (LocationID, LocationName, FilmingDateStart, FilmingDateEnd).
   3. Scenes (SceneID, SceneName, LocationID, LocationName).
   4. Directors (DirectorID, DirectorFname, DirectorLname, AvailabilityStart, AvailabilityEnd, LocationID, LocationName).
4. Add at least 3 complete records into each table (5 points).
5. All tables should pass the 1st, 2nd and 3rd Normal Forms Test 🡪 Add Primary Keys, etc. and you should ensure that the data is consistent (20 points).
6. Create Composite Keys, where is needed (5 points).
7. Commit the changes of the Transaction (2 points).

**Section B**

Syntax five Joins retrieving data from your database (10 points each):

1. Select all the actors that do not participate into any one Scene (pick one).
2. Select all the actors working in the same scene (any scene) and in the same scene.
3. Select all the scenes filmed in a specific location.
4. Select all the Directors, filming a specific scene during a specific date.
5. Develop a Natural Join connecting two appropriate tables.

**GOOD LUCK**

**ANSWERS**

**SECTION A**

**1.create database GameofThronesSetj;**

**use GameofThronesSetj;**

**2.start transaction ;**

**3.create table Actors**

**( ActorID int auto\_increment primary key,**

**ActorFName varchar(50),**

**ActorLName varchar(50),**

**AvailabilityStart date,**

**AvailabilityEnd date,**

**LocationID int ,**

**LocationName varchar(50)**

**);**

**create table Locations**

**( LocationID int auto\_increment primary key,**

**LocationName varchar(50),**

**FilmingDateStart date,**

**FilmingDateEnd date**

**);**

**create table Scenes**

**( SceneID int auto\_increment primary key,**

**SceneName varchar(50),**

**LocationID int,**

**LocationName varchar(50)**

**);**

**create table Directors**

**( DirectorID int auto\_increment primary key,**

**DirectorFname varchar(50),**

**DirectorLname varchar(50),**

**AvailabilityStart date,**

**AvailabilityEnd date,**

**LocationID int,**

**LocationName varchar(50)**

**);**

**4. insert into Actors(ActorFName, ActorLName, AvailabilityStart, AvailabilityEnd, LocationID, LocationName) values ( 'jeehyuep', 'kwon', '2010-10-10', '2012-10-10', 1, 'toronto');**

**insert into Actors (ActorFName, ActorLName, AvailabilityStart, AvailabilityEnd, LocationID, LocationName) values ( 'Yuki', 'miwazawa', '2008-02-10', '2015-08-10', 2, 'barrie');**

**insert into Actors (ActorFName, ActorLName, AvailabilityStart, AvailabilityEnd, LocationID, LocationName) values ( 'shinhee', 'kim', '2010-10-10', '2012-10-10', 3, 'america');**

**insert into Locations(LocationName, FilmingDateStart, FilmingDateEnd) values ('toronto', '2010-08-12', '2012-10-12');**

**insert into Locations(LocationName, FilmingDateStart, FilmingDateEnd) values ('barrie', '2011-08-12', '2012-10-12');**

**insert into Locations(LocationName, FilmingDateStart, FilmingDateEnd) values ('america', '2009-08-12', '2012-10-12');**

**insert into Scenes(SceneName, LocationID, LocationName) values ('car', 1, 'toronto');**

**insert into Scenes(SceneName, LocationID, LocationName) values ('romance', 2, 'berrie');**

**insert into Scenes(SceneName, LocationID, LocationName) values ('fight', 3, 'america');**

**insert into Directors(DirectorFname, DirectorLname, AvailabilityStart, AvailabilityEnd, LocationID, LocationName) values ('captain', 'marvel', '2010-08-12', '2012-10-12', 1, 'toronto');**

**insert into Directors(DirectorFname, DirectorLname, AvailabilityStart, AvailabilityEnd, LocationID, LocationName) values ('thonos', '??', '2011-08-12', '2012-10-12', 2, 'barrie');**

**insert into Directors(DirectorFname, DirectorLname, AvailabilityStart, AvailabilityEnd, LocationID, LocationName) values ('Jone', 'brick', '2009-08-12', '2012-10-12', 3, 'america');**

**6.commit;**

**SECTION B**

**1.**

**select A.ActorFName, A.ActorLName**

**from Actors as A inner join Scenes as B**

**on A.LocationID not in(1,2,3);**

**2.**

**select A.ActorFName, A.ActorLName, A.LocationID**

**from Actors as A inner join Scenes as B**

**on A.LocationID = B.LocationID;**

**3.**

**select A.SceneName from Scenes as A inner join Locations as B**

**on A.LocationName in ('toronto');**

**4.**

**select A.DirectorFname, A.DirectorLname**

**from Directors as A inner join Scenes as B**

**on A.LocationName = B.LocationID in (1) and A.AvailabilityStart = '2010-08-12';**

**5.**

**select A.LocationName, B.LocationName from Actors as A natural join Scenes as B;**