UNIVERSITY OF TORONTO SCARBOROUGH

Final Exam
CSCC43
Introduction to Database
August 2017
Instructor: Dr. Marzieh Ahmadzadeh

Student Name: ……. **Student Number: ……..**

Mark: ……...

Please read the following before you start writing.

Write the answers in front of the questions in this booklet.

Use page 10 of this question booklet as your draft, or if an extra space is needed.

If you still needed more space for the answers or for your rough work, use the other provided booklet.

Write the answers neatly. If your answer is not readable, no mark will be awarded.

If you find a question vague, make an assumption, write down the assumption and provide the answer based on that.

Do not separate any paper from question papers.

You have 120 minutes to finish the exam.

No question will be answered in last 15 minutes of the exam.

This exam is a closed book exam therefore NO aid including textbook, handout etc. are allowed.

Mobile phone or any other electronic device is not allowed in the exam.

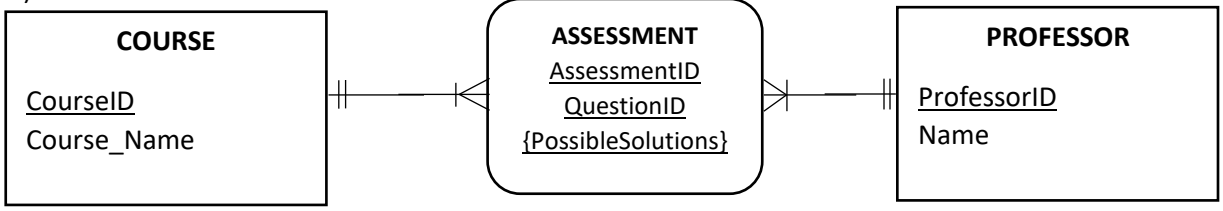
Part	Mark
1	
2	
3	
4	
5	
6	
7	
Total	

Part 1: Write the corresponding terms for the given definitions. (11 Marks)

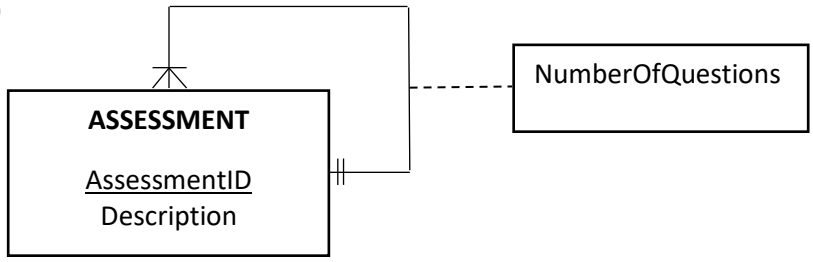
Hard-coded SQL statements included in a program written in another language, such as C or Java	
A join that is the same as an equi-join except that one of the duplicate columns is eliminated in the result table.	
A virtual table that is created dynamically upon request by a user.	
Multiple values returned from an SQL query that includes an aggregate function.	
Commands used to control a database, including those for administering privileges and committing data.	
A table or other data structure used to determine in a file the location of records that satisfy some condition.	
A alternative name used for an attribute.	
A relation that has a primary key and in which there are no repeating groups.	
A rule that states that no primary key attribute (or component of a primary key attribute) may be null.	
The maximum number of instances of one entity that may be associated with each instance of another entity	
A model that has resulted from extending the original E-R model with new modeling constructs.	

Part 2: Transform each EER diagram to corresponding relation(s) using short hand notation. Primary keys should be underlined. (9 Marks)

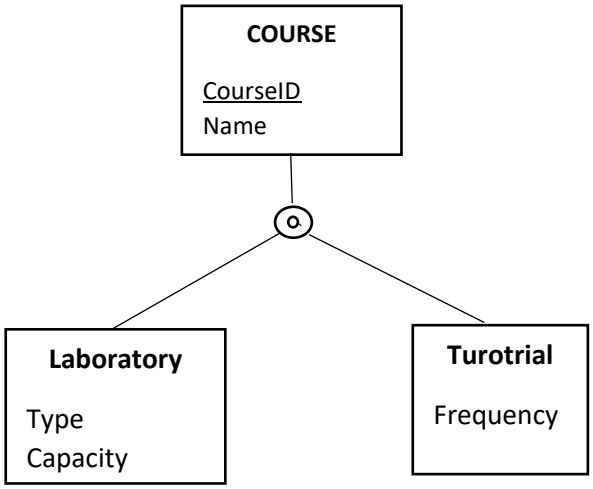
A)



B)



C)



Part 3: (10 Marks)

A) Convert the following relation to third normal form by underlining the primary keys.

PlayID	CoachID	TeamID	CoachName	Specialty	TeamPlayerID	PlayerName	outfitNo
A	11	100	Peter	Tennis	A1	John	10
A	11	100	Peter	Tennis	A2	Mike	20
A	12	200	Brian	Tennis	B1	Andrew	10
A	12	200	Brian	Tennis	B2	Murray	13
B	27	230	Alex	Soccer	C1	Brian	11
B	27	230	Alex	Soccer	C1	Nick	12
B	28	240	David	Soccer	D1	John	20
B	28	240	David	Soccer	D2	Adam	23

B) Suppose that each year, we enroll 10,000 students whose student identification number, name and their top skill are stored in a table similar to the following table. What would you change to have a better design?

StudentID	Name	TopSkill
1	Sue	Python Programming
2	Liz	Database Design
3	Beth	Python Programming
4	Rose	Python Programming
5	Andrea	Database Design

Part 4: What would be the output of running the following SQL code snippet? (10 Marks)

A)

```
CREATE TABLE t_1 (
  ID INT(1) NOT NULL,
  Name VARCHAR(45) NULL,
  PRIMARY KEY (ID));
INSERT INTO t_1 (ID, Name) VALUES (1, 'John');
INSERT INTO t_1 (ID, Name) VALUES (2, 'Sue');

CREATE TABLE t_2 (
  ID INT(1) NOT NULL,
  Name VARCHAR(45) NULL,
  PRIMARY KEY (ID));

INSERT INTO t_2 (ID, Name) VALUES (1, 'John');
INSERT INTO t_2 (ID, Name) VALUES (2, 'Liz');

select * from t_1 union select * from t_2;
```

B)

```
set autocommit = 0;
CREATE TABLE _names (
      id INT(2) NOT NULL,
      name VARCHAR(10) NULL);
insert into _names (id, name) values (10, "john");
rollback;
commit;
insert into _names (id, name) values (20, "jane");
commit;
rollback;
select * from  names;
```

C)

```
Select region, count(region)as cnt
from countries
where indepYear > 1900
group by region
having cnt < 4;
```

Countries

name	region	surfaceArea	IndepYear
Belarus	Eastern Europe	207600.00	1991
Estonia	Baltic Countries	45227.00	1991
Lithuania	Baltic Countries	65301.00	1991
Latvia	Baltic Countries	64589.00	1991
Moldova	Eastern Europe	33851.00	1991
Romania	Eastern Europe	238391.00	1878
Russian Federation	Eastern Europe	17075400.00	1991
Ukraine	Eastern Europe	603700.00	1991

D)

```
select _name.id, name, surname
from _name left outer join _surname
on _name.id;
```

_name

id	name
10	Jane
20	John
30	Pete

_surname

id	surname
10	Doe
20	Smith

E)

```
select name
from _name
where id in
(select id from _surname
where surname = 'Smith');
```

_name

id	name
10	Jane
20	John
30	Pete

_surname

id	surname
10	Doe
20	Smith

Part 5: Fill in the gaps. (10 Marks)

A) It is desired to fire a trigger to set the 'updated' field to *true,* if the *grade* is altered (e.g. `update Grades set grade = 95 where id = 4;`).

```
create table Grades (
    id int(2) primary key not null,
    name varchar(20),
    grade decimal (4,2),
    updated boolean default false
);


create trigger updateGrade before update on Grades
for each row

      ……;
```

B) *Grades* table is defined in part 5-A.

```
create index grade_idx on grades(……);

select name, grade, updated from Grades
where grade > 50
group by updated, id order by grade desc;
```

Part 6: The following SQL codes generate an error. Explain about the source of the error. (20 Marks)

A)

```
create table sample1 (
   sample1_id int(2),
   sample1_description char(10),
   constraint s1_pk primary key (sample1_id));

create table sample2 (
   sample2_id int(2),
   sample1_id char(2),
   sample2_description char(10),
   constraint s2_pk primary key (sample2_id),
   constraint s2_fk Foreign key (sample1_id) references sample1(sample1_id)
);
```

B)

```
create table sample1_1(
    id1 int (1) primary key,
    id2 int (2) primary key,
    description varchar(20));
```

C)

```
create view someCountries as
select code, name , language
from country, countrylanguage
where country.Code = countrylanguage.CountryCode
and countrylanguage.Language in ('english', 'french')
with check option;

update someCountries set language= 'chinese' where code = 'can';
```

D)

```
delimiter DD
create function most_populous_city (in con_name varchar(13))
returns char(35)
begin

    declare cityname char(35);
    set cityname = (select city.Name
    from world.city, world.country
    where country.Continent = con_name and
          city.CountryCode = country.code and
           city.population = (select max(city.population)
               from world.city, world.country
               where country.Continent = con_name and
               city.CountryCode = country.code));

     return cityname;

end DD
delimiter ;


select most_populous_city('asia');
```

Part 7: Write <u>one</u> SQL command for the requested report and the given tables. (30 Marks)

StudentInfo

id	name
1	John
2	Jane
3	Sue
4	Peter
5	Mike
6	Rose

Course

id	Description
CSCA08	Introduction to Programming
CSCC43	Introduction to Database

StudentGrade

StudentID	CourseID	grade
1	CSCA08	60
1	CSCC43	50
2	CSCA08	78
2	CSCC43	55
3	CSCA08	90
3	CSCC43	65
4	CSCC43	70
5	CSCA08	60

A) Find the name of the students who received a maximum mark in each course.

B) Find the names of the students who have taken all the offered courses.

C) Find the names of the students who have taken no course.