Final Exam

Class Room: Online

Assignment Points: 15 points

Thursday 7/23/2020

Name: Sahib Bajwa

Student ID: 107553096

Exam rules:

- You MUST submit this final exam by today, **7/23/2020, 11:59 pm.** There will not be any extension or late submission.
- Submit your assignment in PDF format in Canvas. You can use word, excel or similar tools and convert into pdf.
- This is open book exam and any kind of resource materials are allowed.
- Collaboration and consultation is NOT allowed. Do your own work.

Section 1: 3 points

Normalize the following form into **3NF.** Only your 3rd NF will be graded.

University Departments Sample Form

Dept Name	……..	
Building Num	……..	
Phone 1	xxx-xxx-xxxx	
Phone 2	xxx-xxx-xxxx	
Phone 3	xxx-xxx-xxxx	
Instructor Name	Subject	Gender
……. …….	…….	X
……. …….	…….	X
……. …….	…….	X
……. …….	…….	X

This is the University departments sample form used by many departments.

If there is no concatenated key or many to many relationship, many times you can put directly into 3rd NF, i.e., do not carried away with unnecessary normalization. There is not always 1st, 2nd and 3rd NF needed.
Normalize, as you did in HW-1 Normalization.

Hints: List all attributes.
Identify the repeating group of attributes.
Create entities and keys e.g. PK/FK.
You don't need more than 3 entities in your 3 NF.

Unnormalized Form	1st NF	2nd NF	3rd NF
Dept Name	Department	Department	Department
Dept ID	Dept ID	Dept ID	Dept ID (PK)
Building Num	Dept Name	Dept Name	Dept Name
Phone 1	Building Num	Building Num	Building Num (FK)
Phone 2	Phone 1	Phone 1	
Phone 3	Phone 2	Phone 2	BuildingInfo
Instructor First Name	Phone 3	Phone 3	Building Num (PK)
Instructor Last Name			Phone 1
Instructor ID	Instructor	DepartmentInstructor	Phone 2
Instructor Gender	Instructor ID	Dept ID	Phone 3
Subject	Instructor First Name	Instructor ID	
	Instructor Last Name	Subject	DepartmentInstructor
	Instructor Gender		Dept ID (PK)
	Subject	Instructor	InstructorID (PK)
		Instructor ID	Subject
		Instructor First Name	
		Instructor Last Name	Instructor
		Instructor Gender	Instructor ID (PK)
			Instructor First Name
			Instructor Last Name
			Instructor Gender

Section 2: 3 points

Create **ERD design** for following scenario:

Your data model design (ERD) should include relationships between tables with primary keys, foreign keys, optionality and cardinality relationships. Captions are NOT required.

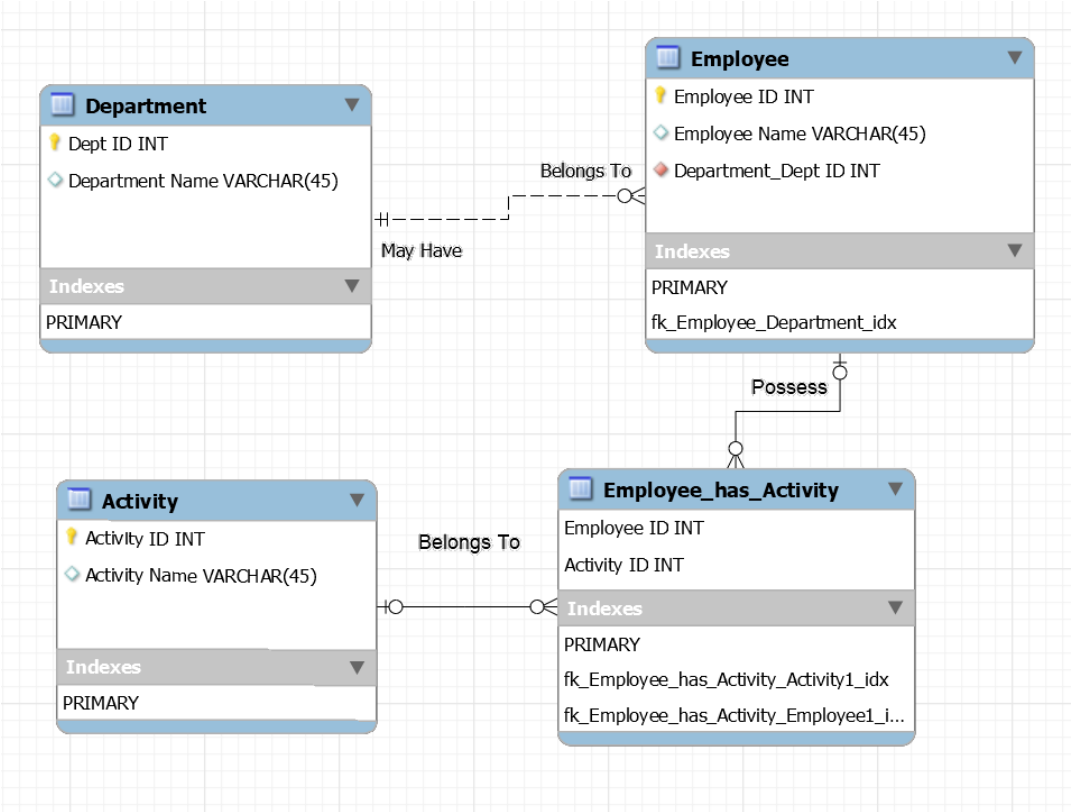
Scenario: There are 3 tables with 2 columns in each table:

Department (Dept ID, Department Name)

Employee (Employee ID, Employee Name)

Activity (Activity ID, Activity Name)

Each Employee must belongs to ONLY ONE Department.
Department may have ZERO, ONE OR MORE Employees, i.e. Department may exists without any employee.
Each Employee may participate in ZERO, ONE OR MORE Activities
Each Activity may be performed by ZERO, ONE OR MORE Employees.



Section 3: 2 points

a. Create table **T1** with following columns and constraints.
Note: DO NOT use alter table, list all constraints while creating table.

C1 INT (10) Primary key
C2 INT (10)
C3 INT (10)
C4 VARCHAR (40)

Constraints:
C3 NON-ZERO
C2 greater than C3
C4 default value of 'HR'

```
# Question 3. a)
CREATE TABLE T1 (
  `C1` INT(10) PRIMARY KEY,
  `C2` INT(10) CHECK (C2 > C3),
  `C3` INT(10) NOT NULL CHECK (C3 > 0),
  `C4` VARCHAR(40) DEFAULT 'HR'
  );
```

b. Create table **T2** with following columns and Foreign Key.
Note: DO NOT use alter table, create FK while creating table.

C5 INT (10) Primary key
C6 INT (10)
FK on C6 column referencing to C1 column in table T1 above.

```
CREATE TABLE T2 (
  `C5` INT(10) PRIMARY KEY,
  `C6` INT(10),
  FOREIGN KEY (C6) REFERENCES T1(C1)
  );
```

c. Explain, in short, the meaning and importance of Referential Integrity (RI).

Referential integrity is important because it helps us make sure that we do not add or remove something that would not make sense. For instance, you do not want to add a product to a database if the supplier does not exist. Likewise, you may not want to delete a supplier from the database if they still have existing products. It is a way of checking that we are not tampering with the database in a way that would make it less sensible.

Section 4: 4 points

All questions are based on below **Employees table**:

EmpId	ManagerId	Name	Department	Salary	City
1	0	Alex Smith	Admin	$90,000	Boulder
2	1	Amy Mars	Admin	$50,000	Longmont
3	1	Logan Mars	Admin	$70,000	Longmont
4	1	James Mont	Marketing	$55,000	
5	6	John Smith	Marketing	$60,000	Boulder
6	1	Lily Mars	Marketing	$95,000	
7	6	Ravi Grace	Database	$75,000	Longmont
8	6	Tara Frank	Database	$80,000	Longmont
9	6	Tom Ford	Database	$65,000	
10	6	William Cruze	Database	$85,000	Longmont

a. Write a SQL statement to find the Name and Salary who has **5th HIGHEST** Salary in the entire Employee table.

```
SELECT Name, Salary FROM Employees ORDER BY Salary DESC LIMIT 4, 1;
```

b. Write a SQL statement to find the Department and their count whose count is more than 3.

```
SELECT Department, COUNT(Department) AS DeptCount FROM Employees
        GROUP BY Department
        WHERE DeptCount > 3;
```

c. Write a SQL statement to show Name, Department and City.
However, if City is NULL, then display 'Broomfield' otherwise display City itself.

```
SELECT Name, Department,
        CASE
                WHEN City IS NULL THEN ‘Broomfield’
                ELSE City
        END AS City
        FROM Employees;
```

d. Write a SQL statement to find distinct employee Name who is also a Manager

```
SELECT Name FROM Employees
    WHERE (EmpID IN (SELECT ManagerID FROM Employees);
```

e. Write a SQL statement to find Maximum, Minimum and Average Salary from the entire Employee table.

```
SELECT MAX(Salary), MIN(Salary), AVG(Salary) FROM Employees;
```

f. Write a SQL statement to show Name, Department and Salary who earn MORE THAN the Average Salary in **THEIR department**. You must use sub-query.

```
SELECT Name, Department, Salary FROM Employees e
        WHERE Salary > (SELECT AVG(Salary) FROM Employees p
        WHERE p.Department = e.Department);
```

g. Write a SQL statement to show Name, Department, Salary and their Rank **WITHIN Department** from highest to lowest salary.
i.e, Salary rank must reset and re-rank start from 1 for EACH Department.

```
SELECT Name, Department, Salary,
        RANK() OVER (
                PARTITION BY Department
                ORDER BY Salary DESC
        ) AS 'Salary Rank'
        FROM Employees
```

h. Write a SQL statement to find HIGHEST paying employee's Name and Salary from the entire Employee table. You must use sub-query.

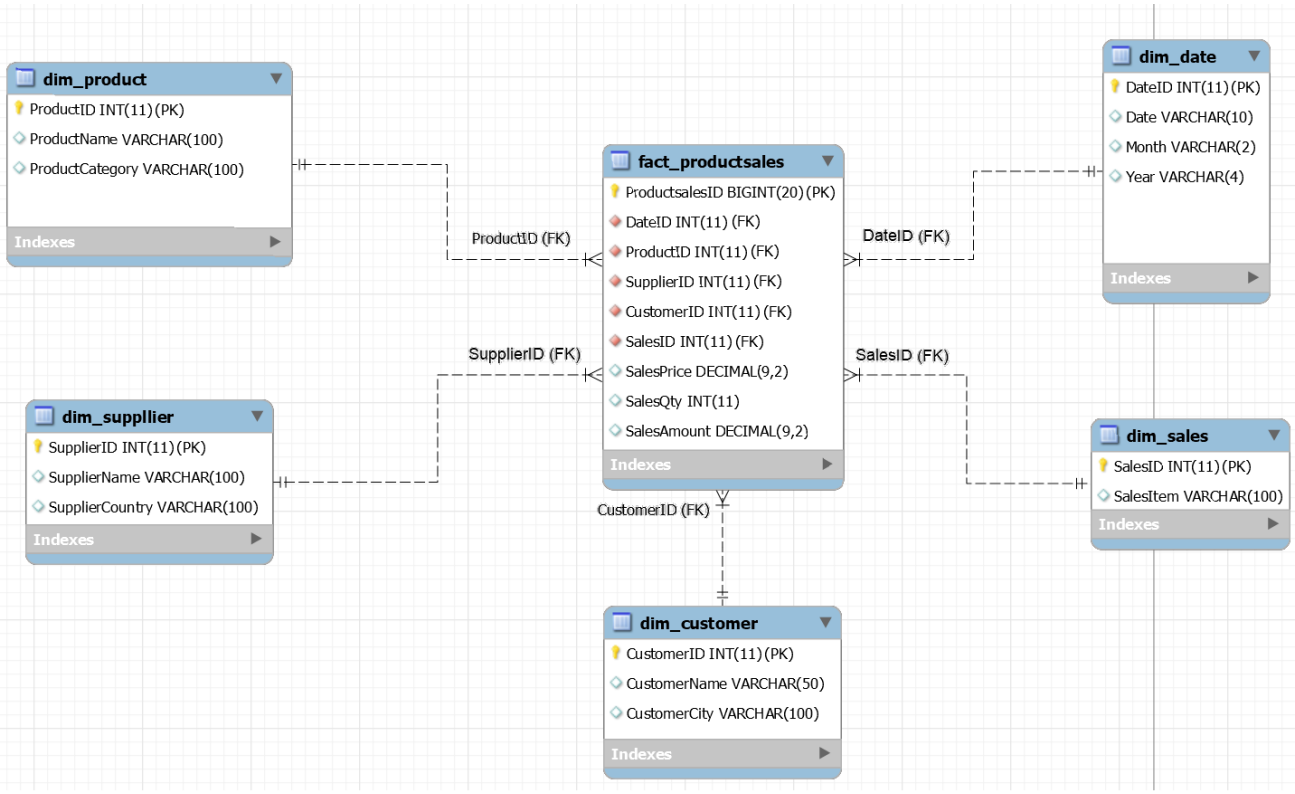
```
SELECT Name, Salary FROM Employees e
        WHERE Salary = (SELECT MAX(Salary) FROM Employees p);
```

Section 5: 3 points

Create a Retail Sales Company **Data Warehouse design** using **STAR schema** from following info. Make sure to indicate proper _DIM and _Fact tables and their PKs/FKs. You need to join those tables using JUST straight lines (optionality and cardinality relationships are NOT required).

Date, Month, Year, SupplierName, SupplierCountry, ProductName, ProductCategory, CustomerName, CustomerCity, SalesItem, SalesPrice, SalesQty, SalesAmount

Note: You may use MySQL workbench or just handwritten to create STAR schema Data Warehouse design.



☺ ☺ ☺ The END ☺ ☺ ☺