**CIS 265 Database assignment: Teen jobs**

**Overview**

In this assignment you create a database for a non-profit organization that connects teenagers with temporary jobs such as babysitting and yard work. The data is already in a workbook, so you will be import all the data from Excel, modifying the resulting tables, create a relationship, and create some queries.

You can start putting all of the following SQL commands in a file called teenjobs.sql and run that using phpMyAdmin, or you can use the phpMyAdmin online panel to work on your queries.

**Create Employees table**

1. Download and save the starting Excel workbook which you can get from the assignment files.
2. Start apache and MySQL and using phpMyAdmin create a new database called TeenJobs. The first step is to create three tables in the database by importing data from the excel workbook you downloaded.
3. Import the data from the Employees worksheet in the teenwork.xlsx workbook. The first row of the worksheet contains headers for the data. Check the settings for each field in the import. The WorkerID should be an integer and indexed with no duplicates. The names and phone should be varchars. The rest of the fields (Weekend through Clerical) should be Yes/No fields. Choose to have the primary key set to be the WorkerID. This data will become the Employees table in the database.

**Create Jobs table**

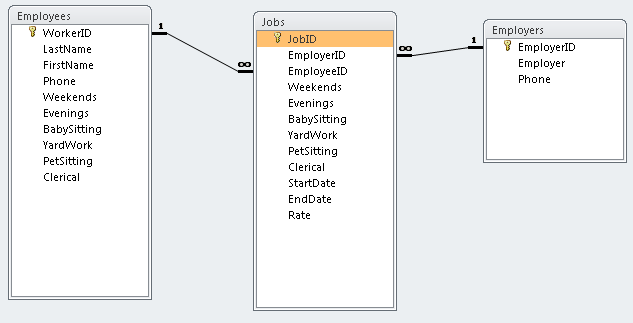
1. Import the data from the Jobs worksheet in the teenwork.xlsx workbook. The first row of the worksheet contains headers for the data. Check the settings for each field in the import. The JobID should be an integer and indexed with no duplicates. The EmployerID should be a text field. The EmployeeID should be an integer. The next six fields (Weekends through Clerical) should be Yes/No fields. The date fields should be Date/Time fields. The rate field should be a double data type. Choose to have the primary key set to be the JobID. This data will become the Jobs table in the database.

**Create Employers table**

1. Import the data from the Employers worksheet in the teenwork.xlsx workbook. The first row of the worksheet contains headers for the data. Check the settings for each field in the import. The EmployerID should be a varchar, indexed with no duplicates. The name and phone should also be varchars. Choose to have the primary key set to be the EmployerID. This data will become the Employers table in the database.

**Create table relationships**

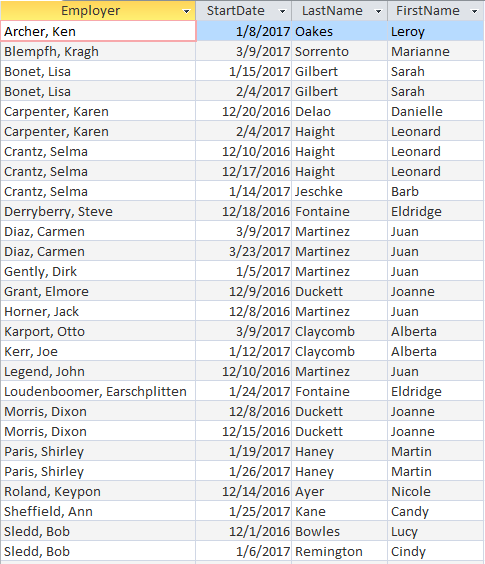
Here are the relationships between the tables:



**Do a EmployeeJobList query**

1. Use the Employees and Jobs tables.
2. Display the worker ID, employee last name, employee first name, job ID, start date, and end date fields.
3. Sort the data in ascending order based on the employee last name, then employee first name, and then start date. 

**Do a JobsByEmployer query**

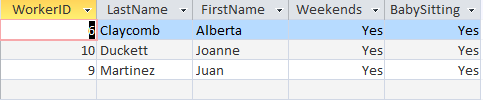
1. Use the Employers, Jobs, and Employees tables.
2. Display the employer name, the job start date, and the employee last name and first name fields.
3. Sort the data in ascending order based on the employer name and then by start date. 

**Overview of part II**

Now its time to find out if we can fill a new job request that came in. The job is for a babysitter on 3/9/13, which is a weekend. We will do this in steps. First we will find out what teens are listed as being able to handle babysitting on the weekend. Then we will find out which of those teens currently has a job on the date in question. Finally we will ask for any teens that are in the first list, but not in the second list. In other words, they are qualified and do not already have a job scheduled for that time.

**Do a WeekendBabySitting query**

1. Use the Employees table.
2. Display the worker ID, employee last name, employee first name, weekends, and babysitting fields
3. Sort the data in ascending order based on the employee last name, and then employee first name. The sort is required to be done in design view (NOT datasheet view).
4. Only show those records where the Weekends field is Yes and the BabySitting field is Yes



**Do JobsForMarch9 query**

Create a new query using design view.

1. Use the WeekendBabySitting query and Jobs table.
2. You will need to tell the query that the WorkerID and EmployeeID fields are related, so click on the WorkerID field in the WeekendBabySitting query and drag to the EmployeeID field in the Jobs table to set up the connection.
3. Display the worker ID, employee last name, employee first name, start date, and end date.
4. Sort the data in ascending order based on the employee last name, then employee first name, and then start date. The sort is required to be done in design view (NOT datasheet view).
5. Only show those records where the start date is <= 3/9/17 and the end date is >= 3/9/17

list of jobs for March 9 for weekend babysitters

**Do AvailableForMarch9 query**

1. Combine the WeekendBabySitting and JobsForMarch9 queries.

Employees available for March 9 babysitting job

**Submit assignment**

Save your database setup and queries in a file called teenwork.sql. Submit this file to the Desire2Learn assignment.