CECS 323 LAB FIRST RECORDS

OBJECTIVE: Give the student first-hand experience in storing rows in the database.

PROCEDURE: The first objective is to install the NetBeans IDE that we will initially use for a front-end to the Derby database. Later we will use more of the NetBeans capabilities when we develop a database application in NetBeans. I have a set of instructions for installing NetBeans and the Derby database here. As an added resource, the NetBeans quick start is here.

1) Build a new Derby database. Call it "First Records".
2) Design a table to hold the following information:
   a) You have a team with a handful of members in it. You want to keep track of each member's:
      i) Last name
      ii) First name
      iii) E-mail address
      iv) Cell phone number
      v) Their current task that they are working on with the rest of the team
      vi) The due date when they expect to finish that task
   b) Decide which column(s) you think that you should put into the primary key
3) Implement that table using a **create table** statement. Remember that the online tutorial has a sample create table statement here.
   a) NetBeans has a wizard that prompts you through the process of creating a table. **Do not use the wizard**. I want you to get experience using the Data Definition Language (DDL). Every relational database management system (RDBMS) supports DDL. You will not always have NetBeans to help you along. Also, if you get to a truly large application, there may be hundreds of tables that you have to create, and then do it all over again when you migrate from Development to Test, and yet again when you migrate from Test to Production. Using a script that has all the create table statements will make that transition much quicker and more reliable.
   b) All the columns in the example are varchar. Look up the date datatype here. Use it for any column(s) that you feel are appropriate.
   c) Also, be sure to use the integer datatype where appropriate.
4) Insert some rows into your new table using the insert statement.
   a) Make at least one attempt to insert a duplicate row. In the database's mind, a duplicate row is any row that agrees with any row already stored in the table on all the columns of any unique key.
   b) At this point, the only unique key that you will have is your primary key. Creating a primary key constraint automatically creates a unique key to enforce the primary key's uniqueness constraint.
5) Select the rows from your team (or whatever you called your table, it is up to you) table by issuing the following statement:

   ```
   select * from team;
   ```

WHAT TO TURN IN:

- The SQL that you used to create the table, including the primary key constraint.
- The SQL that you used to insert the rows into the table.
- A screen shot of the insert that you used to test the primary key, along with the error message that came back from NetBeans.
- A screenshot or a Word table showing the records that you inserted into the table.

- Your team's collaboration document. You can find the template for that at BeachBoard | Content | Student Helps | Lab Collaboration Document.