**Assignment 1**

**Part 1: Creating Contacts**

**Part 2: Form Binding and View Contacts**

**Part 3: Database Connectivity**

**Part 4: Edit and Delete**

**Part 5: Registration Page**

**Part 6: Data Manipulation By Role**

This assignment will take place in parts and will span the course. It will require you to contain components of various lessons throughout the course. Each part will be graded like the exercises in the course. Each part must be done within 1 week of being posted for full marks. Parts that are presented after 1 week will only be awarded half marks. Exceptions will only be considered around holidays and examinations.

What we are going to do for this part is modify the assignment so that it uses security and roles to alter your list of contacts.

Step 1:

Modify your Contact bean so that it contains a 5th field called role.

Create a dropdown list in your HTML page for adding a contact so that the client can select either "guest", "member", or "admin".

Modify your table in your database so that it can support the role.

Modify the view HTML so that it can show the role.

Step 2:

Write 3 Database Access methods that will query the database by their role.

**public** ArrayList<Contact> getGuestContacts(){

//Connection code here

String Query = "SELECT \* FROM contacts where role = 'guest'";

//More code here

**public** ArrayList<Contact> getMembersContacts(){

//Connection code here

String Query = "SELECT \* FROM contacts where role = 'member'";

//More code here

**public** ArrayList<Contact> getAdminContacts(){

//Connection code here

String Query = "SELECT \* FROM contacts where role = 'admin'";

//More code here

Step 3:

Modify your Home Controller method that is used to display the list of roles. Display only the contacts who's roles match the user's roles.

ArrayList<Contact> contacts = **new** ArrayList<Contact>();

**if** ("User has role guest") {

contacts.addAll(da.getGuestContacts());

}

**if** ("User has role member") {

contacts.addAll(da.getMemberContacts());

}

**if** ("User has role admin") {

contacts.addAll(da.getAdminContacts());

}

You would then display the contacts list in the HTML page. You should only be able to see contacts where the roles match the roles of the user.