Assignment 5 – Working with the JDBC (75 pts)

1. Create a database that can handle customers, products, and orders.
   1. Customer table  
      custID (auto-incremented integer), firstName, lastName, address, city, state, zip, email
   2. Product table  
      prodID( string ##-###), productName, category, unitCost,unitSellingPrice
   3. Order table  
      orderID (auto-incremented integer), custID, prodID, qty

Populate your tables with appropriate data. Make sure that you maintain referential integrity when you populate the order table so that only existing product ids and existing customers are used.

1. Customer table:

Use a RowSet (see pp 885-886 in the Deitel text) to display the entire customer table onto your user interface. Don’t simply write it to the console – put the data onto a GUI form.

1. Product table:

Provide a form that allows the user to browse through the current products (similar to Figure 25.32a on page 903), with Next, Previous, and Browse All Products buttons.  
  
The form should also provide an INSERT NEW PRODUCT button that allows for entry of another product to the database.

1. Orders Table: Similar to product table – provide a form that allows the user to browse through the orders and insert new orders. Use dropdown lists to let the user select a product by productName and select a customer by first and last name when inserting a new order. Make sure that the new order only allows existing customers and existing products to be entered.
2. Create the following parameterized queries for the products. It should display the results of the query in a JTable – create a custom model for the JTable.   
   1. Display the product id, product name, and unitCost for all products less than a user-entered amount.
   2. Display the product id, product name and description for all products for a particular category selected by the user.
   3. Display the product id, name, and description for all products whose product name starts with a certain letter entered by the user.