Lab Exercise 1. Writing a program to enter numbers one digit at a time in the GUI (development for Assignment 1)

1. Login to lab computers

2. Open NetBeans and open the project TestArith

Go to the Program menu and find NetBeans and start it. Either find the project that you did last Thursday called TestArith or open it from the Project menu. Now create another jFrameForm in the testarith package by doing a right click on the package name and selecting New -> JFrameForm. Name this form something like TestDigitsUI. You may set the Properties under Run to have this form as the main class, or just plan to right click on the program name and use Run File to run it.

3. **The program specification (statement of problem)**

The goal of this program is just to demonstrate how to add digits to a TextField in the way that a calculator would in order to prepare for assignment 1.

4. **Create the Form Design.**
   - **Add a label to the form** by dragging it from the Palette pane (upper right hand side).
     - o Scroll down to find the text property and change it to label the textfield "Display Number"
   - **Add a textfield to the form** by dragging it from the Palette pane.
     - o Change the text property to be 0 (and hit enter). Then resize the textfield on the form with your mouse.
     - o Make the 0 appear on the right of the TextField by finding the Horizontal property and selecting Right as its value.
   - **Add another label** to the form that gives instructions to the user "Enter a digit" **with another TextField** to hold that digit.
   - **Add a button to the form** by dragging it from the Palette pane.
     - o Change the text property of the button to be an instruction to the user, e.g. "Place the digit in the display number".
   - **Add a separator** below this from the component palette.
   - **Add another button** and edit its text to be "Show the display number".
   - **Add another label** to the form below the button to say "Current Number:" and **another TextField** to hold the number.

The resulting form will look something like:

Display Number: |_____0_|

Enter a digit: |_0_|
                          | Place the digit in the display number | (Button)
                          _____   (Separator)

                          | Show the display number | (Button)

        Current Number: |__0_____|

- **To add a program to be executed** when the user clicks the button, right click on the both buttons on your form and select Event -> Action -> actionPerformed.

Form component names

As a form gets more complex, it is helpful to rename the components to have more meaningful names to the program than jButton1, jTextField1, etc. So today we will rename our buttons. For each Button, right click on the Button and select Change Variable Name. A box will come up where you can type a more meaningful name, for example, you may call the Buttons "addDigitButton", and"showNumberButton". In the future, we will also change the names of TextFields and other components.

Now go to the Source tab, and notice that the name of the button methods contains the new name of the button.

5. Add the Program

Go to the source tab and in the program, **before the actionPerformed methods**, we will add a variable and a method.

Adding a variable outside a method in the class makes it a **class level variable**, or **a "global" variable** in the class, and it can be used in all the methods in the class. We are going to demonstrate this because it will be needed in assignment 1, although we are not actually going to use this variable in two different methods in this lab. Add this variable declaration:
        // class level variable to hold the value of the current display number
            int current = 0;

Now add a comment to the addDigitButton event method to document its actions:

/* Get the digit typed by the user and append (catenate) it to the right of the string
 *   for the display number
*/

Add the code for addDigitButton right after the comment with TODO:

For each step, add a comment and then on the next line(s), add the programming statement(s) that perform the computational step described.

      Step 1:   // declare variables for the new digit and the display digit string
              Add the declarations for two String variables
      Step 2:   // get the new digit from the second textfield
              Add the statement that puts the digit textfield into its variable.
      Step 3:   // get the display number from the first textfield as a String
              Add the statement that puts the display textfield into its variable.
      Step 4:   // catenate the digit to the display string
              Add the statement that uses String + to put the digit on the right
      Step 5: // put the display string back into its textfield
              Add the statement that uses setText to put the display string back.

Now add a comment to the showNumberButton event method to document its actions:

/* Get the display number as an integer and then put it back to the third textfield
*/

Add the code for showNumberButton right after the comment with TODO:

Add statements for the following two steps:
      Step 1: Get the value of the display textfield as an integer and put it into the current variable.

      Step 2: Put the value of current as a String into the third textfield.

- **Run this program** by the little green triangle on the top bar (or right click and Run File).
    - When the program executes, it should bring up the user interface form, and you should type in a digit into the second text field and click the Add Digit button for each digit
    - Use the show number button to show the number in the last textfield.

Optional: Go to the addDigitButton and add code that will not put the leading 0 on the display string. This can be done by testing if the display string is equal to "0", and if it is, only put the digit.

There is nothing to hand in for this exercise because it is to be used in your assignment.

Lab Exercise 2 – Shipping program

The goal of this exercise is to develop an application with fewer explicit instructions and to get more experience with "if" statements.

1. Program Specification

Suppose that you are shopping on-line and want to check how much the shipping will be on your purchase. There is a form where you can enter the amount of your purchases and whether you want "expedited" or "regular" shipping and then click a button to find out how much your shipping will be. Shipping is computed as follows: For purchases under $100.00, the regular shipping rate is $ .10 per dollar of purchase and the expedited shipping rate is $ .25 per dollar. For purchases $100 or over, the regular shipping rate is $ .05 per dollar and the expedited shipping rate is $ .15 per dollar.

2. **Create a project with a form:**

Create a project called Shipping and with a JFrame Form called ShippingUI. If you need to look at the details, refer back to the instructions from last week. But remember the three main steps:
- create the new project,
- create the new JFrameForm, name it something like ShippingUI, and put it into the package for shipping and
- set the Properties under Run to have the UI as the main class.

Create a form for this application that looks something like:

```
Enter total amount of purchases:              |__0_____|
Enter shipping type, either regular or expedited:   |_____|

                | Compute Shipping  |              (Button)

                Result                (Label or TextField)
```

Create an actionPerformed method for the button.

3. Add code to your program.

Go to the actionPerformed button method and write a program for the following steps. For each step, add a comment and then add the programming statement(s) that perform the computational step described.

Step 1: Declare variables to hold the input for purchase amount and shipping type. Also declare variables for the shipping rate and the shipping amount.

Step 2: Get the input values from the textfields.

Step 3: Compute the shipping rate (using appropriate "if" statements, using the description from the program specification.

Step 4: Compute the shipping amount by multiplying the shipping rate times the purchase amount.

Step 5: Display the shipping amount in the last textfield.

4. Run this program.

5. Test your program:

Design 4 sets of test input and show the result

Test Run	Purchase amt	Shipping type	Shipping
1			
2			
3			
4			

To hand in this lab, copy the program that you wrote from the button method into a text file and print it.

Hand in this lab sheet, with the program from the Shipping problem, either at the end of class today or by the beginning of class on Tuesday, January 31.