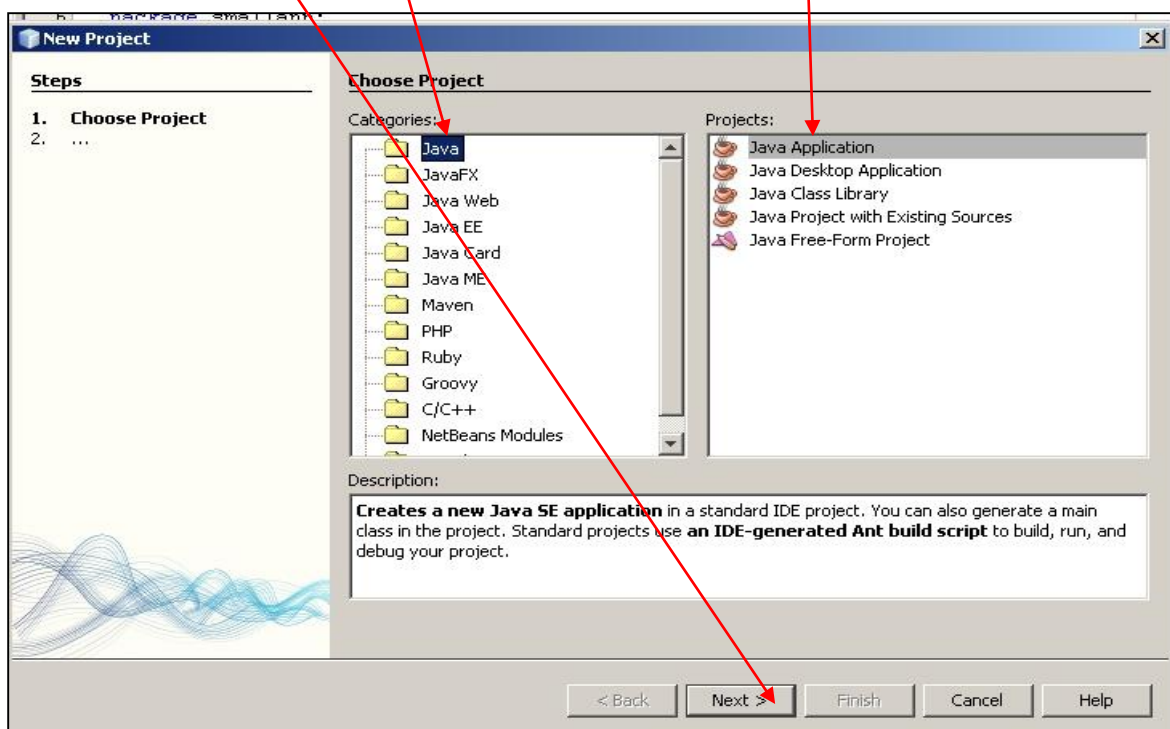


# The Programming Environment and Variables

1. Before we even begin with *NetBeans*, it is important that you make a folder in your directory where you will store your *Java* assignments. Create a folder called **JAVAPROJECTS** either on your desktop or in your network area. Remember where this folder is located; it will be important later on.



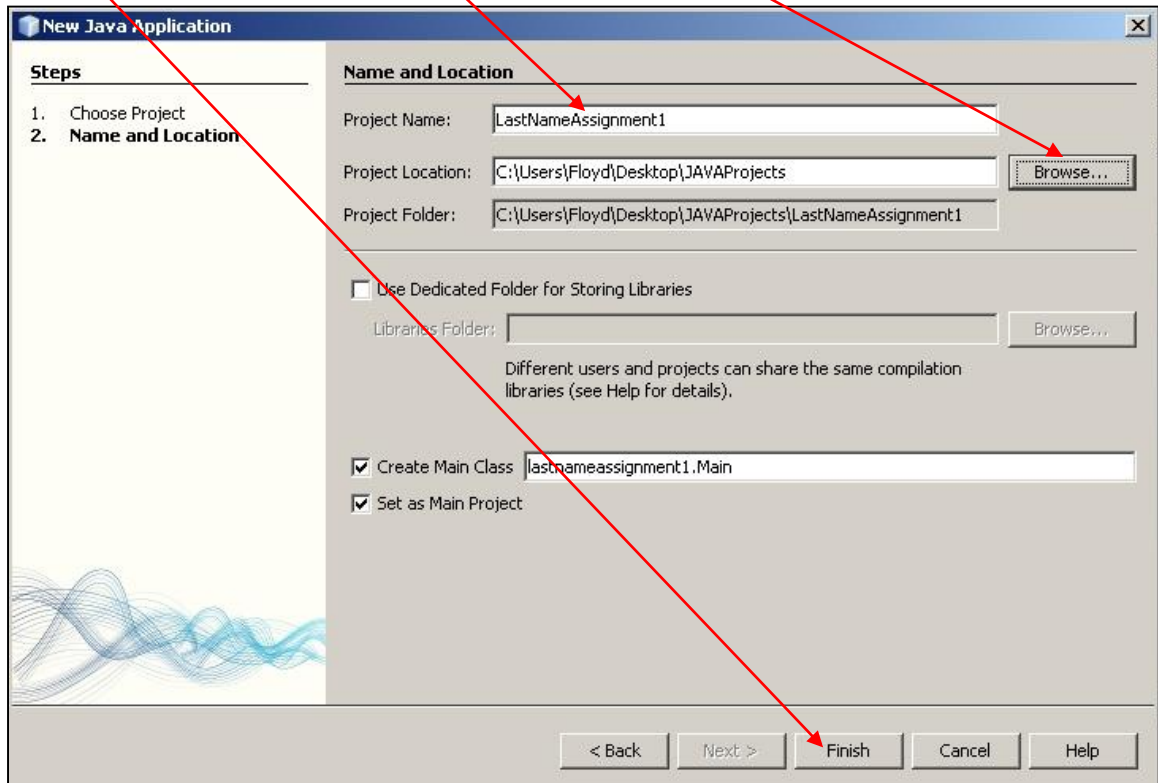
2. Open the *NetBeans IDE*.
3. If the following window is not displayed then click **File... New Project...** Select **Java** from the **Categories** section and **Java Application** from the **Projects:** section, and then click **Next**



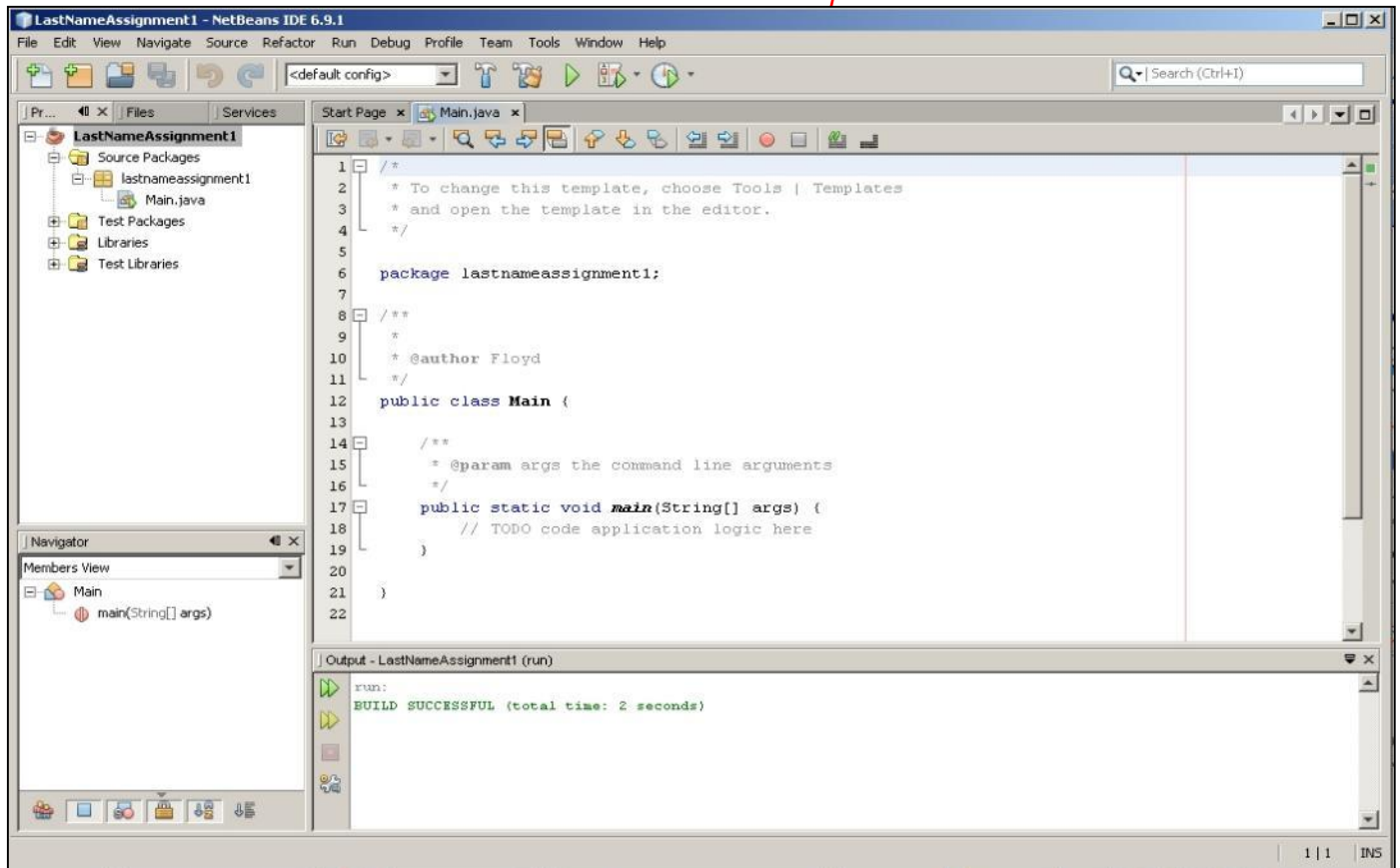
4. In the **Project Name** textbox, type **your last name** and then **Assignment1** (no spaces).

**Important:** Click the **Browse** button and find the folder you created in **Step 1** of this tutorial. By doing this, you will ensure that your *Java* project will be stored in the right place. It is important to ensure that your projects are always going to this folder.

Click **Finish**.



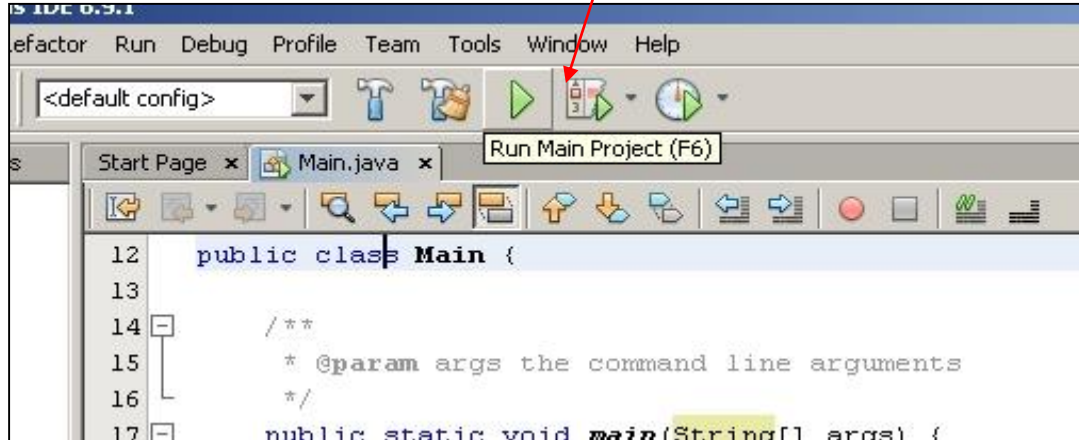
5. You will now see a screen similar to the one below. This is where you will be inputting your code and running your programs. There are several sections to the code area; however, you will just be using the bottom section. It is very important that your code goes into this area; otherwise, your program will not run.



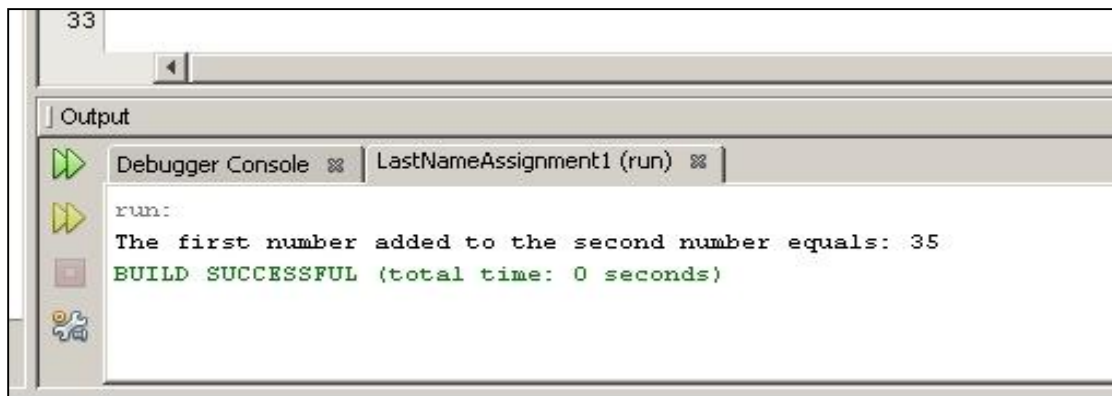
6. Add the following code in the appropriate area and make sure you include all of the appropriate spaces, periods and semi colons (these are **VERY** important).

```
14  /**
15  * @param args the command line arguments
16  */
17  public static void main(String[] args) {
18      // TODO code application logic here
19
20      int firstNum;
21      int secondNum;
22      int thirdNum;
23
24      firstNum = 13;
25      secondNum = 22;
26      thirdNum = firstNum + secondNum;
27
28      System.out.println ("The first number added to the second number equals: " + thirdNum);
29
30  }
31
32 }
```

- Once you have inputted your code, click the **Run Main Project** button, or just press **F6** on your keyboard.



You should see the output generated by our program at the bottom of your screen. It should appear similar to this:



If the same output is not generated, then you may have to go back into your code and make some changes. Double check the code and make sure you have typed it **EXACTLY** as shown.

Congratulations, you have created your first *Java* application!

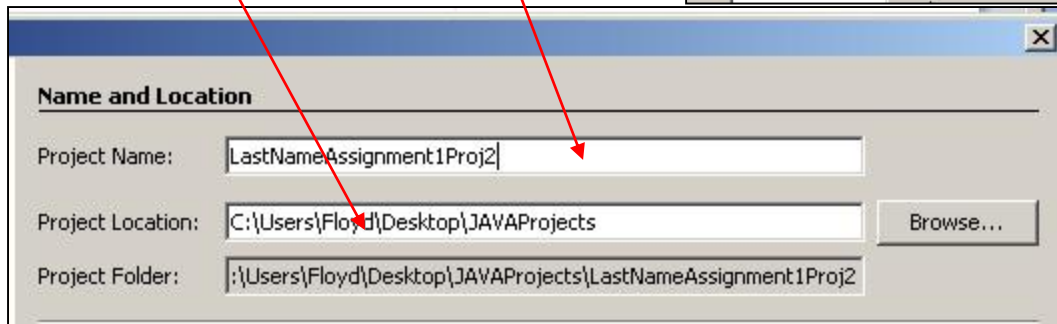
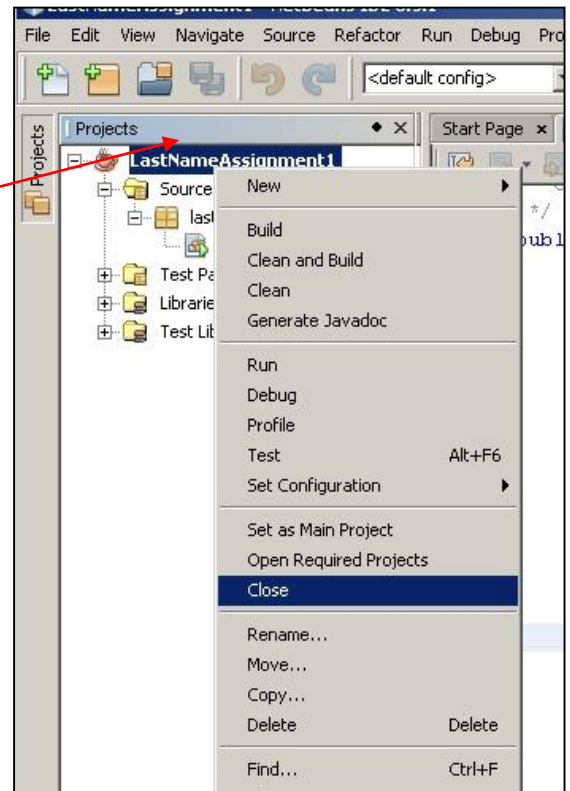
When you run your project (by clicking the **Run Main Project** button or **F6**), your project is automatically saved. If you make any more changes to it, but you don't run it, then you will need to click **File... Save** in order to save the latest version.

8. Now that you've created a simple project with some of the code provided, it's time to create a second project, and this time you are going to write the code yourself.

To get rid of your first project, in the **Projects** area right click on the project name, which is **LastNameAssignment1**, and then select **Close**.

9. Click **File... New Project...** and select **JAVA** and **JAVA Application** and then click **Next**.

10. Input **LastNameAssignment1Proj2** as the **Project Name**, and make sure that the **Project Location** textbox contains the path for your **JAVAProjects** folder. Click **Finish**.



11. For this program you will not be given the code. Instead you will be given an outline of the specific for each line of code. Then you will have to create the code yourself. If you aren't sure, look back at your first program in order to get some hints.

**Important:** Make sure your code goes immediately after the following line:

```
// TODO code application logic here
```

The following pseudocode explains what should occur in each of your lines of code:

Line 20: Declare a variable called value1 that is of type double

*(hint: in the first program you created, the variables were all of type int)*

Line 21: Declare a variable called value2 that is of type double

Line 22: Declare a variable called answer that is of type double

Line 23: Nothing

Line 24: Assign the value of 34.1 to value1

Line 25: Assign the value of 12.3 to value2

Line 26: Assign the value of answer to value1 minus value2

*(hint: the subtraction sign is just a dash "-")*

Line 27: Nothing

Line 28: Create an output statement that will display the following output:

The second number subtracted from the first number equals: 21.8

12. When you have completed this task, and the program runs effectively, close the project and create a new project called **LastNameAssignment1Proj3**.
13. You should now create a program that will take two numbers (12.1 and 13.6), multiply them and create an output similar to the output in the other two programs you created. This third program should be very similar in structure to the other two programs you created.
14. Well done! You are doing great so far. If you would like to really get ahead of the game, and complete a few other programs, here are some other challenges for you:

Create a program that adds 34, 54, 12, 34 and 11 together, and then outputs the result. This should be stored as **LastNameAssignment1Proj4**

Create a program that subtracts 23, 87, and 11 from 150, and then outputs the result. This should be stored as **LastNameAssignment1Proj5**