Department of Computer Science and Software Engineering
Xi'an Jiaotong-Liverpool University

Lab 2 – The Basics of Java – If Statements and Data Types

Aim

The aim of this lab is to learn about different variable types, integer arithmetic, and if statements. If very important in Java.

Resources

You will find your lecture notes to be very useful. Your practical notes from last week will be a useful guide. This work builds on work that you should have finished last week.

Tips:

1. Remember, ask a TA if you have problems, they are here to help you!

2. This lab sheet is expected take more than the 2 allocated hours. You should complete it in your own time before the next classes.

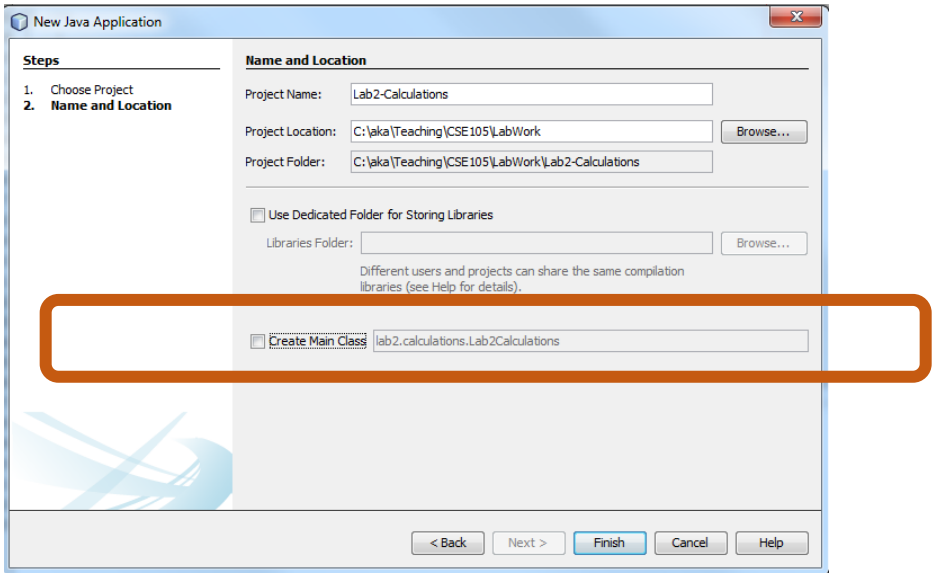
3. Ensure you are saving on your network (M:) folder. If you do not see an M: drive, then ask a TA.

4. When you have completed this sheet, complete the homework sheet.

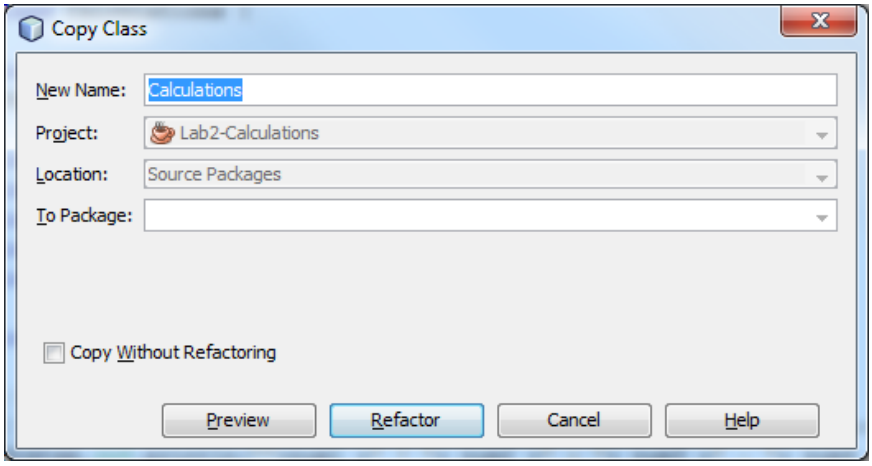
Java Project Management

- This week, we will modify your calculations class from last week

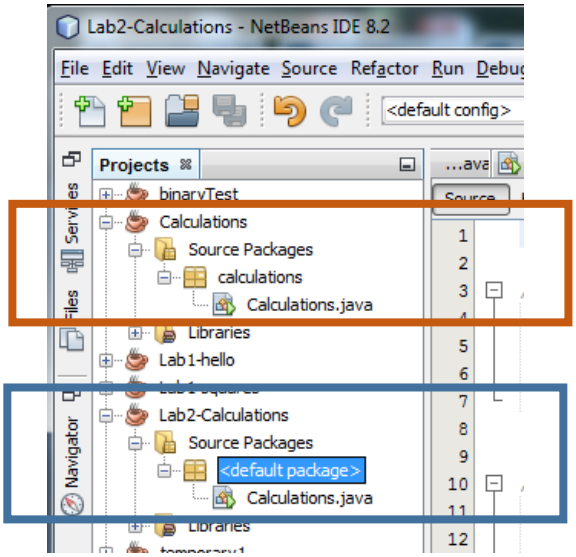
- Create a new project, call it week 2 calculations, and make sure it does not have a new main class (make sure the box is not ticked)



- Copy your Calculations.java file (right click on the Calculations.java file, press copy, and then paste it to your new source package folder in Netbeans) from your old project to your new project. If asked, press the "Refactor" button



- This will create a copy of your Class, while keeping the old class in your old project

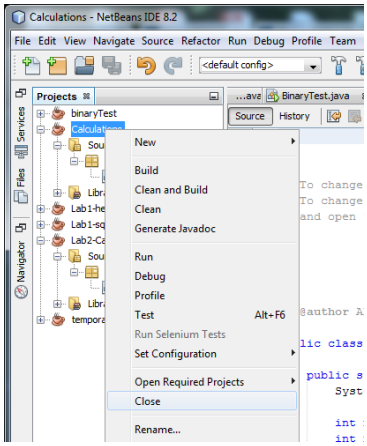


Java Project from Week1

Your new Java Project, with Calculation.java file copied

- Close your project from last week, so that you do not accidentally run the wrong file (see below)

- This is the process of creating a new project with existing Java code, and is useful to ensure that you have a backed up copy of each week's Lab



Variable Types and Calculations – Ints and Mods

- You will have a lot of code from week 1, but do not delete it, as it could be useful! So first, comment it out. You can comment out code, by putting 2 forward slashes at the start of each line, as shown below

```
 * @author ANDREW.ABEL
 */
public class Calculations {

    public static void main(String[] args) {
        // System.out.println("Hello Andrew");

        // int num1 =5;
        // int num2 =7;
        //  int num3 =4;
        // int num4 =2;
        // int num5 =2;
        //  int calculation = num1 * num2 + num3 - num4  /  num5;
        //System.out.println("5 * 7 + 4 - 2 / 2 = 38");
        // System.out.println(""+num1 +" * "+ num2 +" + "+ num3 +" - "+ num
        //  System.out.println(""+num1 +" * "+ num2 +" + "+ num3 +" - "+ nu

    }


}
```

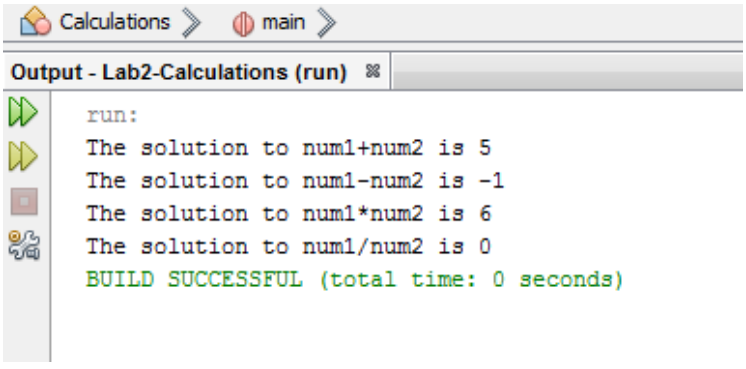
- Using your week 1 work as a guide, declare 2 integer variables called num1 and num2. Assign them values of num1 = 2 and num2 = 3

- Create an integer variable called solution1, and make solution1 = num1+num2

- Use a System.out to output your solution, use text like:

```
System.out.println("The solution to num1+num2 is " +solution );
```

- Create calculations for num1*num2, num1-num2, and num1/num2, and output the answers to your console window



- Check your answer to num1/num2. Is the answer what you expect?

- Add additional code to calculate the moduli (mod), num1 % num2. Output the result on screen. Is the answer what you expect?

- Experiment with different values for num1 and num2 for both div/ and mod%. First work out the expected answer on paper. Then experiment with your program. Is the result what you expect? Try num1 = 5 num2 = 7, num1 = 30 num2 = 4, and others.

Variable Types and Calculations – Doubles and Operator Precedence

- So far you have used integers, which means your answers have been returned in whole numbers

- Change your int values to doubles (instead of int num1, call it double num1). Look at the results of your calculations. Do they match what you expect? What is the difference? Try for the same variables as you tried above. If you do not understand, ask a TA

- Now try a new calculation. Initialise and Assign 3 double variables as shown below

```
double calcVal1 = 4;
double calcVal2 = 8;
double calcVal3 = -1;
```

- Use these variables to create the calculation (replace the numbers with your variables)

```
double complexCalc = 4 +8 * -1 - 4 + 4/8 + -1;
```

- Calculate the answer to this calculation on paper, what do you expect it to be?

- Now add a system.out to display the Java calculation, and compare your result to your result. Are your answers different? If they are, then try to work out why. Ask a TA if you are unsure

- The lecture notes discuss the precedence of operators. Can you add brackets so that the result will be -2.5?

Variable Types and Calculations – Maths and Randoms

- Java can perform much more complex calculations. To do this, you can use "Math", which can be accessed by typing "Math." in Netbeans, and a list of options will appear. For example, to get the value of PI, type in Math.PI

- Comment out your previous code
- Output PI as a System.out (i.e. "The value of Pi is 3.141592653589793")
- Can you calculate the square root of 5 (look at the lecture notes)?
- You can generate a random number, create a double variable to store a random value, and assign a random value to it between 0 and 10. Output a message like "The Random Value is 3.1"
- Compile and run the program several times, does the message change each time? Congratulations, you have generated a random number!

If Statements

Displaying information is of limited use. We want to do something WITH the results of the calculations. One useful thing to do is to add conditions. IF a condition is met, then do something. If it is not met, then do something else.

- Using 2 double variables and one double variable to store the answer, create code to calculate:

```
= 3*numA + (numB-1)
```

- Initially, set numA to 3 and numB to 4. Your answer should be 12
- Using your lecture notes, create an IF statement. IF the answer is greater than 10, then display "Greater than 10"
- However, if it is less than or equal to 10, then you want it to say "10 or less". Add an if statement to handle this
- How can you test to see if this condition is met?
- As these statements are related, it would be better to convert it to an if/else statement. Using your lecture notes, make this change
- Finally, we want to experiment with different numbers, so we want to try to random values. Convert your variables to random numbers between 0 and 10. Test your program!

Checkpoint

Show a TA your IF statement to test if the answer is 10 or less, with the input variables generated randomly

Answer any questions they have, and then continue with the sheet

This will count towards your final grade

Department of Computer Science and Software Engineering
Xi'an Jiaotong-Liverpool University

If Statements - Days of the week

Your lecture notes contained an example of how to compare strings, and days of the weeks. Here, we will create a timetable program that will display your classes for a given day. If you input a String variable such as "Monday", then it will compare that to a series of IF and ELSE-IF statements to produce all your classes.

- Create a new class or a new project for this. Look at your notes from week 1 and this week, and create a new project with a new main class

- First, produce a list of all classes that one of you has on each day. Use the IF-Else statement from the lecture notes, and list all the classes and times using System.outs

- Use a String variable to select the day of the week. For example, if your input variable is String day = "Wednesday", then your output might look something like:

```
Monday's Classes
Lecture CSE105 - 1200
Lab CSE105 - 1400
```

- Test different days and ensure it works.

Nested If Statements – Hours of the week

- We want to improve our program. Previously, we selected a day, and generated all the classes. Now we want to be more specific. As well as entering a day variable, we also want to select an hour integer, and then only display classes that correspond to that hour.

- To do this, we want to use nested ifs, i.e. ifs statement within your days of the week if statements. For example, in the example above, you may want to produce something following the model of:

```
if(condition1 ==true){
      if(condition2 == value){
            Display output
      }   else if(condition2 == value2){
            Display output
      } else {
            No output found
      }
}
```

- Here, condition1 could represent the days of the week, which you already have, and condition2 could represent the hour. Your lecture notes contain a slide on nested ifs. This is intended to be a challenge, so feel free to work together and ask questions!

Next Steps – Homework 2

When you have completed this sheet, move on to homework sheet 2 on ICE!