**BRAC University**

**Department of Computer Science and Engineering**

**CSE 110/162**

**Lab 2**

**Objective**

In 1st part of today’s lab, we will look at the basic building blocks of all computer programs – variables and literals. We will see the different data types available in java, and how to create variables and literals for each data type. We will conclude by writing a few simple programs using the java programming language.

In 2nd part,

We will learn about the different data types in java, and how to create variables of the different data types and use them in our programs. We will also learn variable naming conventions, and try declaring some variable names to see whether or not they are allowed in java. We will conclude with writing and testing a few small programs that take inputs from the user and perform calculations.

**Problem Statement**

Let us begin by getting students familiar with the software we will be using to write and test our programs.

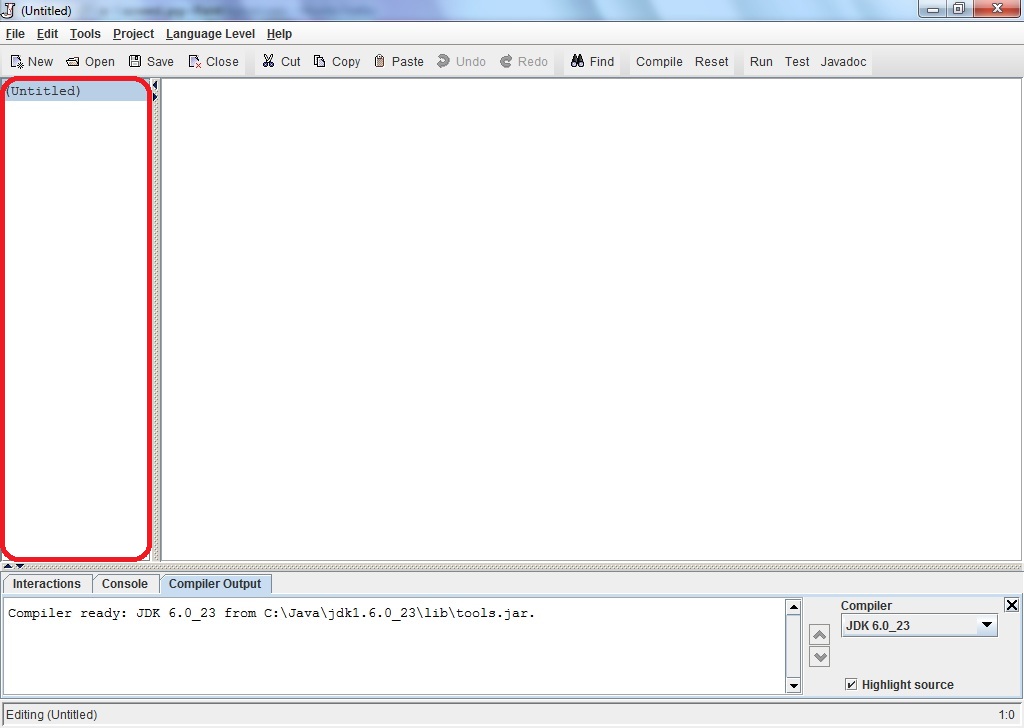
**Open Dr. Java:**

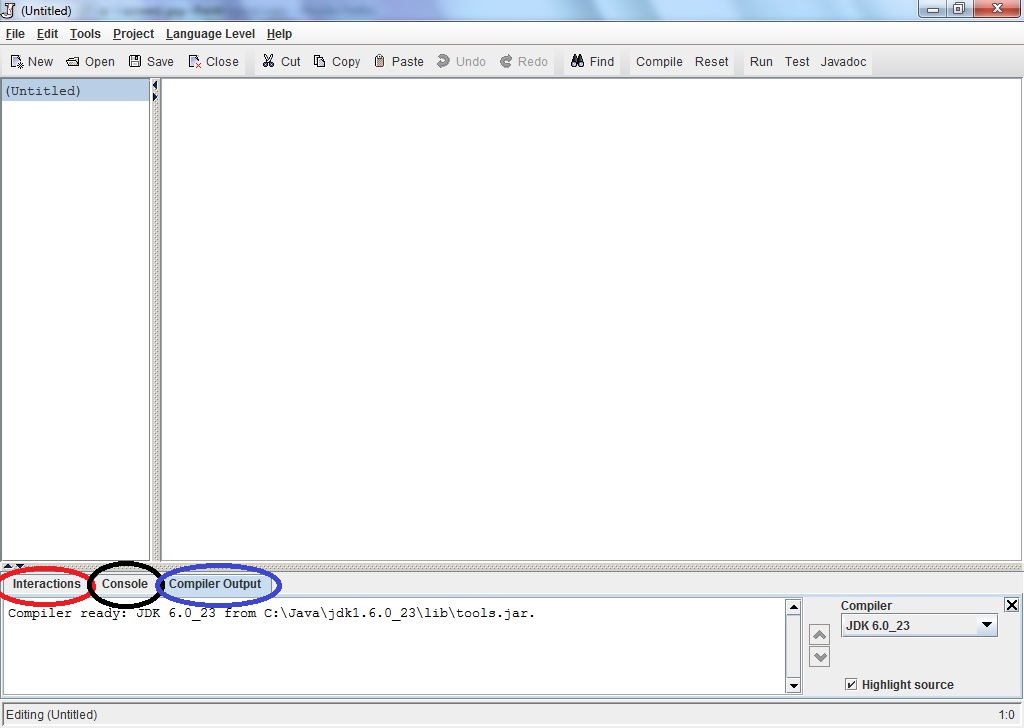
**On the Desktop**, you should see an icon for DrJava (it looks like a big "J", ).

****

Double click on it. The main window of DrJava is broken up into three panes.

On the left side, you should see a narrow pane containing the word "(Untitled)." This is shown in the screenshot below, circled in red.



The large pane to the right of that is where you should type your program. At the bottom of the window is a pane with three tabs "Interactions", "Console", and "Compiler Output." These are shown in the image below, circled in red, black and blue respectively. Please check with the image and make sure you see these tabs on your Dr. Java window.

**Write Java statements in the Interactions pane: [21]**

Click on the "Interactions" tab in the bottom pane. This is a great feature of Dr. Java.

Here, you can type any Java expression and Dr Java will instantaneously compile, interpret, and evaluate it. Try it by typing the following Java expressions. Type them in exactly as shown, including the quotation marks. Record the result of these interactions.

1. 3
2. 3+5
3. 4/6
4. 4.0/6
5. 4/6.0
6. 4.0/6.0
7. 9 - 4 \* 3
8. (9 - 4) \* 3
9. 3 /4 \* 2
10. "hello" + "goodbye"
11. "3" + "5"
12. "3 + 5 is " + 3 + 5
13. "3 + 5 is " + (3 + 5)
14. 3 + 5 + “ is 3 + 5”
15. (3 + 5) + “ is same as 3 + 5”
16. Math.pow(2,3)
17. Math.pow(15, 0.5)
18. Math.sqrt(16) + 8.3
19. Math.pow(2,3) + Math.sqrt(64)
20. Math.pow(Math.pow(2,3),2)
21. Math.toRadians(3)
22. Math.sin(90)
23. Now, based on your current knowledge, experiment and find out how to calculate the value of sin 90 degree. Your answer should be 1.0 or very close.

You may be discouraged by Java's poor math skills in examples l and m. In fact, the "+" operator does different things depending on its arguments. If both arguments are numbers, it adds them; if either argument is a String of characters, it concatenates them into a single String. **We will talk more about this in lecture later**. In the Interactions pane, you can also type more complex Java statements. For instance, to write "Hello World" to the console, we can use the println method. Type the following in the **Interactions** pane:

System.out.println("Hello World");

Now print your name to the console. Click on the **Console** tab and you should see:

Hello World

For most of the programs that you write in this class, the output of your program will be a series of statements printed to the console.

Try to print your name following the previous method.

**Write your first program:**

***[ Note to Instructor:* Please discuss/explain flowcharts and construct the flowchart for the “Hello World” program before discussing the code. ]**

Your first program will be a simple one: it prints "Hello World!" to the console.

Create a class called HelloWorld which System.out.println("Hello World!"); As a template, here is a class called HelloBRACU which prints "Hello BRAC University!" to the console. The class has one method called main. Inside the body of main, there is a single statement, a call to the println method of the System.out object (We will discuss about classes and objects more in class lectures). Change the name of this class to HelloWorld and also change what it prints to the screen. Also, change the author listed in the comment at the top of the program.

**// My first program**

**// Author: Matin Abdullah**

**public class HelloBRACU**

**{**

**public static void main(String[] args)**

**{**

**System.out.println("Hello BRAC University!");**

**}**

**}**

Save your program to a file on the **Desktop** with the filename

HelloWorld.java

**Note: Java requires that the file name match the name of the class defined in the file**.

Click the “**Compile**” button (or under the Tools menu, select "**Compile All Documents**"). Once DrJava has compiled your document, you should see the message Last compilation completed successfully. in the "Compiler Output" tab of the bottom window pane. Now if you look on the Desktop, you should see another file called **HelloWorld.class**

This file contains the bytecode into which your program was compiled. When you run your program, it is this bytecode that is interpreted by the Java virtual machine.

Now it is time to run your program. Go to **Tools->Run Document's Main Method**. Again in the bottom pane under the Interactions tab you should see:

Welcome to DrJava.

> java HelloWorld

Hello World!

>

Under the Console tab, you should see only: "Hello World!" Congratulations, you wrote your first of many programs in Java! Notice that double quotes (“”) are not visible in the output.

**2nd part**

**Problem Statement**

All the chapter numbers and section headings that follow are from the following textbook:

Java 2: The Complete Reference, Fifth Edition, by Herbert Schildt

A pdf copy of this book can be found at:

[\\tsr\Fall\CSE\Annajiat\Books\Java 2 The Complete Reference.pdf](file:///\\tsr\Fall\CSE\Annajiat\Books\Java%202%20The%20Complete%20Reference.pdf)

**Data types**

Read Chapter 3, starting from the section labeled “The Simple Types” (page 42 of the book, page 72 in the soft copy). You need to read only pages 42-49, and only the sections listed below.

1. The Simple Types
2. Integers
3. Floating-Point Types
4. Characters
5. Booleans

For each section, read all of its sub-sections. Reading the above topics should give you an idea of what data types are, how many there are in java and what they are called. At the end of your reading, please answer the following questions and bring them in with you to the lab.

1. How many simple data types are there in java?
2. List all the simple data types.
3. What are the different Integer and Floating-Point data types available in java?

**Variables**

Read Chapter 3, page 52. The required sections are listed below:

1. Variables, p. 52
   1. Declaring a variable
   2. Dynamic initialization

You need only read the two sub-sections listed.

**Also read the following resources:**

* the “Naming part” of http://java.sun.com/docs/books/tutorial/java/nutsandbolts/variables.html
* <http://java.sun.com/docs/books/tutorial/java/nutsandbolts/_keywords.html>
* **“Rules for Naming”** for the **Identifier Type, “Variables”** from <http://java.sun.com/docs/codeconv/html/CodeConventions.doc8.html>

**Exercise on variables**

## ***Find out which of the followings are legal identifiers in Java, and which are not. Also take note why the invalid ones are invalid. You can try to define each of these (items a to k below) as a variable in Dr. Java’s interaction pane and find out. The first one is done for you as an*** *Example. If you want to define hungry, you have to try int hungry;*

**a) hungry b) 2AB c) 312.2 d) MOBILE e) “Ans” f) $30**

**g) Yes/No h) student-id i) A+3 j) ‘X’ k) return**

**Exercises on data types**

Unless otherwise specified, these exercises are to be done in the Dr Java interactions pane as shown in the previous lab.

1. Declare an integer variable. Initialize it with some value of your choice and print it to check the value has been stored properly.
2. Declare and initialize another integer variable. Add this to the first one and print out the result. Verify that the addition has been done correctly.
3. Repeat exercise 1 and 2 for variables of data type double. Verify your answers.
4. Repeat exercise 1 and 2 for variables of data type String. How does the addition operator work for Strings?

**Java programming exercises**

Before moving on to the exercises in this section, please read the materials provided in the folder named “How to take input”.

To solve these exercises, you will have to write, save and run complete programs similar to the one we did in the previous lab. Review the last section of that lab if you need help with this step.

1. Write a java program that reads one integer from the user, and prints it back to show which integer was entered by the user.
2. Repeat exercise 1 for a floating point number.
3. Write a java program that reads two integers from the user, and prints their sum, product and difference.
4. Repeat exercise 3 for two floating point numbers.
5. Write a java program that reads the radius of a circle and prints its circumference and area.