

BRAC University
Department of Computer Science and Engineering
CSE 110/162 (Programming Language I)
Lab 3

Objective

In today's lab, 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

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\CSE\Annajiat\Books\Java 2 The Complete Reference.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
 - a. Declaring a variable
 - b. 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. Write a java program that reads two integers from the user, and prints their sum, product and difference.
3. Repeat question 2 for two floating point numbers.
4. Write a java program that reads the radius of a circle and prints its circumference and area.
5. Write a java program that reads two integers from the user and prints “first” if the first number is greater than the second number.
6. Write a java program that reads two integers from the user. Your program should then print “first is greater” if the first number is greater, and “first is not greater” otherwise.
7. Write a java program that reads two integers from the user. Your program should then print “first is greater” if the first number is greater, “second is greater” if the second number is greater, and “the numbers are equal” otherwise.
8. Write a java program that reads two integers, subtracts the smaller number from the larger one, and prints the result.
9. Write a java program that reads an integer, and prints “The number is even” or “The number is odd”, depending on whether the number is even or odd. (Hint: use the modulus operator)

Flowchart exercises

Task 1:

Draw the flowchart of a program that reads an integer, and if the number is even and greater than 10, prints “An even number greater than 10”. If the number is even but not greater than 10, print “An even number not greater than 10”. If the number is greater than 10 but odd, print “An odd number greater than 10”. If the number is odd and also less than 10, print “An odd number less than 10”.

Task 2:

Draw the flowchart of a program that reads an integer, and prints the integer if it is a multiple of **either 2 or 5**.

For example, 2, 4, 5, 6, 8, 10, 12, 14, 15, 16, 18, 20, 22 ...

Task 3:

Draw the flowchart of a program that reads an integer, and prints the integer if it is a multiple of either 2 or 5 **but not both**.

For example, 2, 4, 5, 6, 8, 12, 14, 15, 16, 18, 22 ...

Task 4:

Draw the flowchart of a program that reads an integer, and prints the integer if it is a multiple of 2 **and** 5.

For example, 10, 20, 30, 40, 50 ...

Task 5:

Draw the flowchart of a program that reads an integer, and prints the integer if it is a multiple of **NEITHER 2 NOR 5**.

For example, 1, 3, 7, 9, 11, 13, 17, 19, 21, 23, 27, 29, 31, 33, 37, 39 ...

Task 6:

Draw the flowchart of a program that reads an integer, and prints the integer if it is **NOT** a multiple of 2 **OR NOT** a multiple of 5.

Task 7:

Draw the flowchart of a program that reads a student’s mark for a single subject, and prints out “Pass” if the student got more than 50, and “You shall not pass” otherwise.

Task 8:

Draw the flowchart of a program that reads a student’s mark for a single subject, and prints out the corresponding grade for that mark. The mark ranges and corresponding grades are shown in the table below.

Marks	Grade
90 and above	A
80-89	B
70-79	C
60-69	D
50-59	E
Below 50	F