# Introduction to Computers, the Internet and Java

# 1

## Objectives

In this chapter you'll:

- Learn about exciting recent developments in the computer field.
- Learn computer hardware, software and networking basics.
- Understand the data hierarchy.
- Understand the different types of programming languages.
- Understand the importance of Java and other leading programming languages.
- Understand object-oriented programming basics.
- Learn Internet and web basics.
- Learn a typical Java program-development environment.
- Test-drive a Java application.
- Learn some key recent software technologies.

## Self-Review Exercises

**1.1** Fill in the blanks in each of the following statements:

a) Computers process data under the control of sets of instructions called _____.

**ANS:** programs.

b) The key logical units of the computer are the _____, _____, _____, _____, _____ and _____.

**ANS:** input unit, output unit, memory unit, central processing unit, arithmetic and logic unit, secondary storage unit.

c) The three types of languages discussed in the chapter are _____, _____ and _____.

**ANS:** machine languages, assembly languages, high-level languages.

d) The programs that translate high-level language programs into machine language are called _____.

**ANS:** compilers.

e) _____ is an operating system for mobile devices based on the Linux kernel and Java.

**ANS:** Android.

f) _____ software is generally feature complete, (supposedly) bug free and ready for use by the community.

**ANS:** Release candidate.

g) The Wii Remote, as well as many smartphones, use a(n) _____ which allows the device to respond to motion.

**ANS:** accelerometer.

**1.2** Fill in the blanks in each of the following sentences about the Java environment:

a) The _____ command from the JDK executes a Java application.

**ANS:** `java`.

b) The _____ command from the JDK compiles a Java program.

**ANS:** `javac`.

c) A Java source code file must end with the _____ file extension.

**ANS:** `.java`.

d) When a Java program is compiled, the file produced by the compiler ends with the _____ file extension.

**ANS:** `.class`.

e) The file produced by the Java compiler contains _____ that are executed by the Java Virtual Machine.

**ANS:** bytecodes.

**1.3** Fill in the blanks in each of the following statements (based on Section 1.5):

a) Objects enable the design practice of _____—although they may know how to communicate with one another across well-defined interfaces, they normally are not allowed to know how other objects are implemented.

**ANS:** information hiding.

b) Java programmers concentrate on creating _____, which contain fields and the set of methods that manipulate those fields and provide services to clients.

**ANS:** classes.

c) The process of analyzing and designing a system from an object-oriented point of view is called _____.

**ANS:** object-oriented analysis and design (OOAD).

d) A new class of objects can be created conveniently by _____ —the new class (called the subclass) starts with the characteristics of an existing class (called the superclass), possibly customizing them and adding unique characteristics of its own.

**ANS:** Inheritance.

e) _____ is a graphical language that allows people who design software systems to use an industry-standard notation to represent them.

**ANS:** The Unified Modeling Language (UML).

f) The size, shape, color and weight of an object are considered _____ of the object's class.

**ANS:** attributes.

## Exercises

**1.4** Fill in the blanks in each of the following statements:

a) The logical unit that receives information from outside the computer for use by the computer is the _____.

**ANS:** input unit.

b) The process of instructing the computer to solve a problem is called _____.

**ANS:** computer programming.

c) _____ is a type of computer language that uses Englishlike abbreviations for machine-language instructions.

**ANS:** Assembly language.

d) _____ is a logical unit that sends information which has already been processed by the computer to various devices so that it may be used outside the computer.

**ANS:** The output unit.

e) _____ and _____ are logical units of the computer that retain information.

**ANS:** The memory unit, the secondary storage unit.

f) _____ is a logical unit of the computer that performs calculations.

**ANS:** The arithmetic and logic unit (ALU).

g) _____ is a logical unit of the computer that makes logical decisions.

**ANS:** The arithmetic and logic unit (ALU).

h) _____ languages are most convenient to the programmer for writing programs quickly and easily.

**ANS:** High-level.

i) The only language a computer can directly understand is that computer's _____.

**ANS:** machine language.

j) _____ is a logical unit of the computer that coordinates the activities of all the other logical units.

**ANS:** The central processing unit (CPU).

**1.5** Fill in the blanks in each of the following statements:

a) The _____ programming language is now used to develop large-scale enterprise applications, to enhance the functionality of web servers, to provide applications for consumer devices and for many other purposes.

**ANS:** Java.

b) _____ initially became widely known as the development language of the UNIX operating system.

**ANS:** C.

c) The _____ ensures that messages, consisting of sequentially numbered pieces called bytes, were properly routed from sender to receiver, arrived intact and were assembled in the correct order.

**ANS:** Transmission Control Protocol (TCP).

d) The _____ programming language was developed by Bjarne Stroustrup in the early 1980s at Bell Laboratories.

**ANS:** C++.

**1.6** Fill in the blanks in each of the following statements:

a) Java programs normally go through five phases—_____, _____, _____, _____ and _____.

**ANS:** edit, compile, load, verify, execute.

b) A(n) _____ provides many tools that support the software development process, such as editors for writing and editing programs, debuggers for locating logic errors in programs, and many other features.

**ANS:** integrated development environment (IDE).

c) The command `java` invokes the _____, which executes Java programs.

**ANS:** Java Virtual Machine (JVM).

d) A(n) _____ is a software application that simulates a computer, but hides the underlying operating system and hardware from the programs that interact with it.

**ANS:** virtual machine (VM).

e) The _____ takes the `.class` files containing the program's bytecodes and transfers them to primary memory.

**ANS:** class loader.

f) The _____ examines bytecodes to ensure that they're valid.

**ANS:** bytecode verifier.

**1.7** Explain the two compilation phases of Java programs.

**ANS:** The two compilation phases that Java programs typically go through include one in which source code is translated into bytecodes which are portable across JVMs and a second in which the bytecodes are translated into machine language for the actual computer on which the program executes. In early Java versions, the JVM was simply an interpreter for Java bytecodes. This caused most Java programs to execute slowly because the JVM would interpret and execute one bytecode at a time. Today's JVMs typically execute bytecodes using a combination of interpretation and so-called just-in-time (JIT) compilation. In this process, The JVM analyzes the bytecodes as they are interpreted, searching for hot spots—parts of the bytecodes that execute frequently. For these parts, a just-in-time (JIT) compiler—known as the Java HotSpot compiler—translates the bytecodes into the underlying computer's machine language. When the JVM encounters these compiled parts again, the faster machine-language code executes.

**1.8** One of the world's most common objects is a wrist watch. Discuss how each of the following terms and concepts applies to the notion of a watch: object, attributes, behaviors, class, inheritance (consider, for example, an alarm clock), modeling, messages, encapsulation, interface and information hiding.

**ANS:** The entire watch is an object that is composed of many other objects (such as the moving parts, the band, the face, etc.) Watch attributes are time, color, band, style (digital or analog), etc. The behaviors of the watch include setting the time and getting the time. A watch can be considered a specific type of clock (as can an alarm clock). With that in mind, it is possible that a class called `Clock` could exist from which other classes such as watch and alarm clock could inherit the basic features in

the clock. The watch is an abstraction of the mechanics needed to keep track of the time. The user of the watch does not need to know the mechanics of the watch in order to use it; the user only needs to know that the watch keeps the proper time. In this sense, the mechanics of the watch are encapsulated (hidden) inside the watch. The interface to the watch (its face and controls for setting the time) allows the user to set and get the time. The user is not allowed to directly touch the internal mechanics of the watch. All interaction with the internal mechanics is controlled by the interface to the watch. The data members stored in the watch are hidden inside the watch and the member functions (looking at the face to get the time and setting the time) provide the interface to the data.

## Making a Difference

The Making-a-Difference exercises will ask you to work on problems that really matter to individuals, communities, countries and the world.

**1.9** *(Test Drive: Carbon Footprint Calculator)* Some scientists believe that carbon emissions, especially from the burning of fossil fuels, contribute significantly to global warming and that this can be combated if individuals take steps to limit their use of carbon-based fuels. Various organizations and individuals are increasingly concerned about their "carbon footprints." Websites such as TerraPass

```
http://www.terrapass.com/carbon-footprint-calculator-2/
```

and Carbon Footprint

```
http://www.carbonfootprint.com/calculator.aspx
```

provide carbon-footprint calculators. Test drive these calculators to determine your carbon footprint. Exercises in later chapters will ask you to program your own carbon-footprint calculator. To prepare for this, research the formulas for calculating carbon footprints.

**1.10** *(Test Drive: Body-Mass-Index Calculator)* By recent estimates, two-thirds of the people in the United States are overweight and about half of those are obese. This causes significant increases in illnesses such as diabetes and heart disease. To determine whether a person is overweight or obese, you can use a measure called the body mass index (BMI). The United States Department of Health and Human Services provides a BMI calculator at `http://www.nhlbi.nih.gov/guidelines/obesity/BMI/bmicalc.htm`. Use it to calculate your own BMI. An exercise in Chapter 3 will ask you to program your own BMI calculator. To prepare for this, research the formulas for calculating BMI.

**1.11** *(Attributes of Hybrid Vehicles)* In this chapter you learned the basics of classes. Now you'll begin "fleshing out" aspects of a class called "Hybrid Vehicle." Hybrid vehicles are becoming increasingly popular, because they often get much better mileage than purely gasoline-powered vehicles. Browse the web and study the features of four or five of today's popular hybrid cars, then list as many of their hybrid-related attributes as you can. For example, common attributes include city-miles-per-gallon and highway-miles-per-gallon. Also list the attributes of the batteries (type, weight, etc.).

**ANS:**

- Manufacturer
- Type of Hybrid—Battery hybrid (Hybrid Electric Vehicles), Plug-in hybrid, Fuel cell etc.
- Driver feedback system—so the driver can monitor fuel efficiency based on their driving
- Energy recovery—for example, regenerative breaking
- Carbon footprint—tons of $CO_2$ per year
- Fuel capacity
- City-miles-per-gallon

- Highway-miles-per-gallon
- Two-mode hybrid propulsion system
- Engine size—V6, V8, etc.
- Vehicle type—SUV, crossover, compact, mid-size, etc.
- Seating capacity
- Horse power
- Drive train (front wheel drive, all wheel drive)
- Top speed
- Torque
- Price

**1.12** ***(Gender Neutrality)*** Some people want to eliminate sexism in all forms of communication. You've been asked to create a program that can process a paragraph of text and replace gender-specific words with gender-neutral ones. Assuming that you've been given a list of gender-specific words and their gender-neutral replacements (e.g., replace "wife" with "spouse," "man" with "person," "daughter" with "child" and so on), explain the procedure you'd use to read through a paragraph of text and manually perform these replacements. How might your procedure generate a strange term like "woperchild?" In Chapter 4, you'll learn that a more formal term for "procedure" is "algorithm," and that an algorithm specifies the steps to be performed and the order in which to perform them.

**ANS:** Search through the entire paragraph for a word such as "wife" and replace every occurrence with "spouse." Repeat this searching process for every gender specific word in the list. You could accidentally get a word like "woperchild" if you are not careful about how you perform replacements. For example, the word "man" can be part of a larger word, like "woman." So, replacing every occurrence of "man" can yield strange results. Consider the process of replacing "man" with "person" then replacing "son" with "child." If you encounter the word "woman," which contains the word "man," you'd replace "man" with "person" resulting in the word "woperson." In a subsequent pass you'd encounter "woperson" and replace "son" with "child" resulting in the "woperchild."

**1.13** ***(Intelligent Assistants)*** Developments in the field of artificial intelligence have been accelerating in recent years. Many companies now offer computerized intelligent assistants, such as IBM's Watson, Amazon's Alexa, Apple's Siri, Google's Google Now and Microsoft's Cortana. Research these and others and list uses that can improve people's lives.

**1.14** ***(Big Data)*** Research the rapidly growing field of big data. List applications that hold great promise in fields such as healthcare and scientific research.

**1.15** ***(Internet of Things)*** It's now possible to have a microprocessor at the heart of just about any device and to connect those devices to the Internet. This has led to the notion of the Internet of Things (IoT), which already interconnects tens of billions of devices. Research the IoT and indicate the many ways it's improving people's lives.

# Introduction to Java Applications; Input/Output and Operators

# 2

## Objectives

In this chapter you'll:

- Write simple Java applications.
- Use input and output statements.
- Learn about Java's primitive types.
- Understand basic memory concepts.
- Use arithmetic operators.
- Learn the precedence of arithmetic operators.
- Write decision-making statements.
- Use relational and equality operators.

## Self-Review Exercises

**2.1** Fill in the blanks in each of the following statements:

a) A(n) _____ and a(n) _____ begin and end the body of every method.

**ANS:** left brace ({), right brace (}).

b) You can use the _____ statement to make decisions.

**ANS:** `if`.

c) _____ begins an end-of-line comment.

**ANS:** `//`.

d) _____, _____ and _____ are called white space.

**ANS:** Space characters, newlines and tabs.

e) _____ are reserved for use by Java.

**ANS:** Keywords.

f) Java applications begin execution at method _____.

**ANS:** `main`.

g) Methods _____, _____ and _____ display information in a command window.

**ANS:** `System.out.print`, `System.out.println` and `System.out.printf`.

**2.2** State whether each of the following is *true* or *false*. If *false*, explain why.

a) Comments cause the computer to display the text after the `//` on the screen when the program executes.

**ANS:** False. Comments do not cause any action to be performed when the program executes. They're used to document programs and improve their readability.

b) All variables must be given a type when they're declared.

**ANS:** True.

c) Java considers the variables `number` and `NuMbEr` to be identical.

**ANS:** False. Java is case sensitive, so these variables are distinct.

d) The remainder operator (`%`) can be used only with integer operands.

**ANS:** False. The remainder operator can also be used with noninteger operands in Java.

e) The arithmetic operators `*`, `/`, `%`, `+` and `-` all have the same level of precedence.

**ANS:** False. The operators `*`, `/` and `%` have higher precedence than operators `+` and `-`.

f) The identifier _ (underscore) is valid in Java 9.

**ANS:** False. As of Java 9, _ (underscore) by itself is no longer a valid identifier.

**2.3** Write statements to accomplish each of the following tasks:

a) Declare variables `c`, `thisIsAVariable`, `q76354` and `number` to be of type `int` and initialize each to `0`.

**ANS:**
```
int c = 0;
int thisIsAVariable = 0;
int q76354 = 0;
int number = 0;
```

b) Prompt the user to enter an integer.

**ANS:**
```
System.out.print("Enter an integer: ");
```

c) Input an integer and assign the result to `int` variable `value`. Assume `Scanner` variable `input` can be used to read a value from the keyboard.

**ANS:**
```
int value = input.nextInt();
```

d) Print `"This is a Java program"` on one line in the command window. Use method `System.out.println`.

**ANS:**
```
System.out.println("This is a Java program");
```

e) Print `"This is a Java program"` on two lines in the command window. The first line should end with `Java`. Use method `System.out.printf` and two `%s` format specifiers.

**ANS:**
```
System.out.printf("%s%n%s%n", "This is a Java", "program");
```

f) If the variable `number` is not equal to `7`, display `"The variable number is not equal to 7"`.

**ANS:**
```
if (number != 7) {
   System.out.println("The variable number is not equal to 7");
}
```

**2.4** Identify and correct the errors in each of the following statements:

a)
```
if (c < 7); {
   System.out.println("c is less than 7");
}
```

**ANS:** Error: Semicolon after the right parenthesis of the condition `(c < 7)` in the `if`. As a result, the output statement executes regardless of whether the condition in the `if` is true.
Correction: Remove the semicolon after the right parenthesis.

b)
```
if (c => 7) {
   System.out.println("c is equal to or greater than 7");
}
```

**ANS:** Error: The relational operator `=>` is incorrect.
Correction: Change `=>` to `>=`.

**2.5** Write declarations, statements or comments that accomplish each of the following tasks:

a) State that a program will calculate the product of three integers.

**ANS:**
```
// Calculate the product of three integers
```

b) Create a `Scanner` called `input` that reads values from the standard input.

**ANS:**
```
Scanner input = new Scanner(System.in);
```

c) Prompt the user to enter the first integer.

**ANS:**
```
System.out.print("Enter first integer: ");
```

d) Read the first integer from the user and store it in the `int` variable `x`.

**ANS:**
```
int x = input.nextInt();
```

e) Prompt the user to enter the second integer.

**ANS:**
```
System.out.print("Enter second integer: ");
```

f) Read the second integer from the user and store it in the `int` variable `y`.

**ANS:**
```
int y = input.nextInt();
```

g) Prompt the user to enter the third integer.

**ANS:**
```
System.out.print("Enter third integer: ");
```

h) Read the third integer from the user and store it in the `int` variable `z`.

**ANS:**
```
int z = input.nextInt();
```

i) Compute the product of the three integers contained in variables `x`, `y` and `z`, and store the result in the `int` variable `result`.

**ANS:**
```
int result = x * y * z;
```

j) Use `System.out.printf` to display the message `"Product is"` followed by the value of the variable `result`.

**ANS:**
```
System.out.printf("Product is %d%n", result);
```

**2.6** Using the statements you wrote in Exercise 2.5, write a complete program that calculates and prints the product of three integers.

**ANS:**

```
1  // Ex. 2.6: Product.java
2  // Calculate the product of three integers.
3  import java.util.Scanner; // program uses Scanner
4
5  public class Product {
6     public static void main(String[] args) {
```

```
        // create Scanner to obtain input from command window
        Scanner input = new Scanner(System.in);

        System.out.print("Enter first integer: "); // prompt for input
        int x = input.nextInt(); // read first integer

        System.out.print("Enter second integer: "); // prompt for input
        int y = input.nextInt(); // read second integer

        System.out.print("Enter third integer: "); // prompt for input
         int z = input.nextInt(); // read third integer

         int result = x * y * z; // calculate product of numbers

        System.out.printf("Product is %d%n", result);
    } // end method main
} // end class Product
```

```
Enter first integer: 10
Enter second integer: 20
Enter third integer: 30
Product is 6000
```

## Exercises

*NOTE: Solutions to the programming exercises are located in the* `ch02solutions` *folder. Each exercise has its own folder named* `ex02_##` *where ## is a two-digit number representing the exercise number. For example, Exercise 2.14's solution is located in the folder* `ex02_14`.

**2.7** Fill in the blanks in each of the following statements:

a) _____ are used to document a program and improve its readability.

**ANS:** Comments.

b) A decision can be made in a Java program with a(n) _____.

**ANS:** `if` statement.

c) Calculations are normally performed by _____ statements.

**ANS:** assignment statements.

d) The arithmetic operators with the same precedence as multiplication are _____ and _____.

**ANS:** division (/), remainder (%)

e) When parentheses in an arithmetic expression are nested, the _____ set of parentheses is evaluated first.

**ANS:** innermost.

f) A location in the computer's memory that may contain different values at various times throughout the execution of a program is called a(n) _____.

**ANS:** variable.

**2.8** Write Java statements that accomplish each of the following tasks:

a) Display the message `"Enter an integer: "`, leaving the cursor on the same line.

**ANS:** `System.out.print( "Enter an integer: " );`

b) Assign the product of variables `b` and `c` to the `int` variable `a`.

**ANS:** `=a = b * c;`

c) Use a comment to state that a program performs a sample payroll calculation.

**ANS:** `// This program performs a simple payroll calculation.`

**2.9** State whether each of the following is *true* or *false*. If *false*, explain why.

a) Java operators are evaluated from left to right.
**ANS:** False. Some operators (e.g., assignment, =) evaluate from right to left.
b) The following are all valid variable names: `_under_bar_`, `m928134`, `t5`, `j7`, `her_sales$`, `his_$account_total`, `a`, `b$`, `c`, `z` and `z2`.
**ANS:** True.
c) A valid Java arithmetic expression with no parentheses is evaluated from left to right.
**ANS:** False. The expression is evaluated according to operator precedence.
d) The following are all invalid variable names: `3g`, `87`, `67h2`, `h22` and `2h`.
**ANS:** False. Identifier `h22` is a valid variable name.

**2.10** Assuming that `x = 2` and `y = 3`, what does each of the following statements display?
a) `System.out.printf("x = %d%n", x);`
**ANS:** `x = 2`
b) `System.out.printf("Value of %d + %d is %d%n", x, x, (x + x));`
**ANS:** `Value of 2 + 2 is 4`
c) `System.out.printf("x =");`
**ANS:** `x =`
d) `System.out.printf("%d = %d%n", (x + y), (y + x));`
**ANS:** `5 = 5`

**2.11** Which of the following Java statements contain variables whose values are modified?
a) `int p = i + j + k + 7;`
b) `System.out.println("variables whose values are modified");`
c) `System.out.println("a = 5");`
d) `int value = input.nextInt();`
**ANS:** (a), (d).

**2.12** Given that $y = ax^3 + 7$, which of the following are correct Java statements for this equation?
a) `int y = a * x * x * x + 7;`
b) `int y = a * x * x * (x + 7);`
c) `int y = (a * x) * x * (x + 7);`
d) `int y = (a * x) * x * x + 7;`
e) `int y = a * (x * x * x) + 7;`
f) `int y = a * x * (x * x + 7);`
**ANS:** (a), (d), (e)

**2.13** State the order of evaluation of the operators in each of the following Java statements, and show the value of `x` after each statement is performed:
a) `int x = 7 + 3 * 6 / 2 - 1;`
**ANS:** *, /, +, -; Value of `x` is `15`.
b) `int x = 2 % 2 + 2 * 2 - 2 / 2;`
**ANS:** %, *, /, +, -; Value of `x` is `3`.
c) `int x = (3 * 9 * (3 + (9 * 3 / (3))));`
**ANS:**
```
x = ( 3 * 9 * ( 3 + ( 9 * 3 / ( 3 ) ) ) );
      4   5     3     1   2
```
Value of `x` is `324`.

**2.19** What does the following code print?
```
System.out.printf("*%n**%n***%n****%n*****%n");
```

**ANS:**

```
*
**
***
****
*****
```

**2.20** What does the following code print?

```
System.out.println("*");
System.out.println("***");
System.out.println("*****");
System.out.println("****");
System.out.println("**");
```

**ANS:**

```
*
***
*****
****
**
```

**2.21** What does the following code print?

```
System.out.print("*");
System.out.print("***");
System.out.print("*****");
System.out.print("****");
System.out.println("**");
```

**ANS:**

```
***************
```

**2.22** What does the following code print?

```
System.out.print("*");
System.out.println("***");
System.out.println("*****");
System.out.print("****");
System.out.println("**");
```

**ANS:**

```
****
*****
******
```

**2.23** What does the following code print?

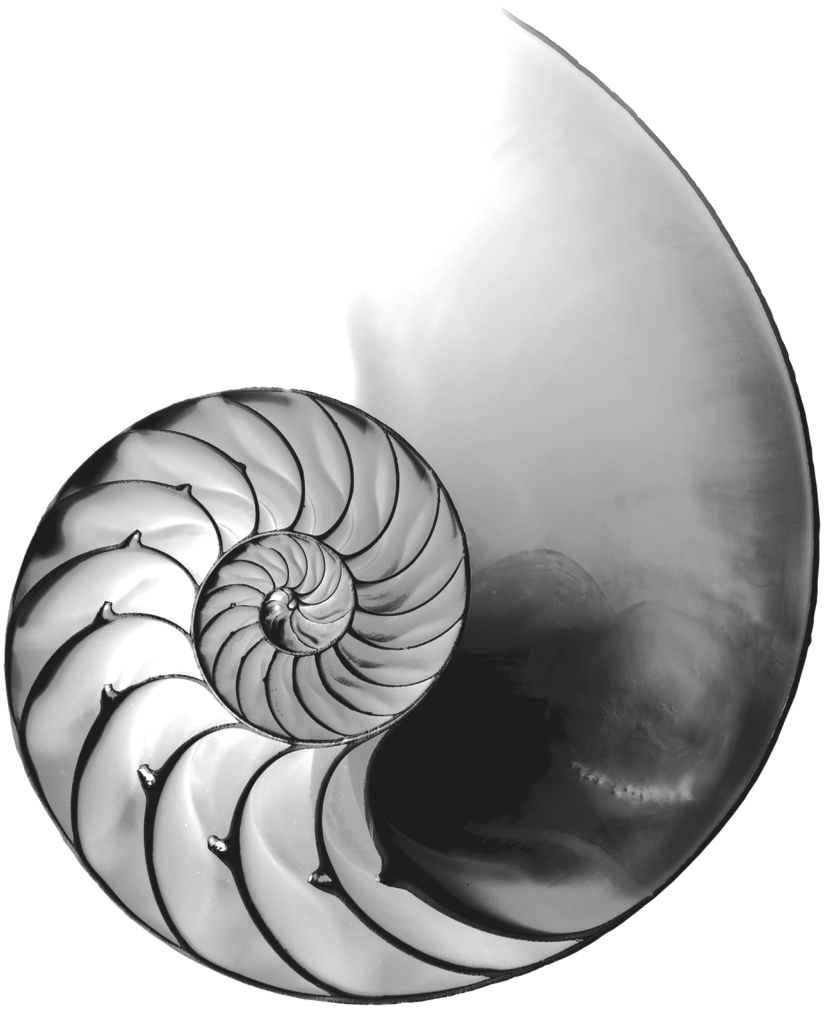
```
System.out.printf("%s%n%s%n%s%n", "*", "***", "*****");
```

**ANS:**

```
*
***
*****
```

# Control Statements: Part 1; Assignment, ++ and -- Operators

# 3



## Objectives

In this chapter you'll:

- Learn basic problem-solving techniques.
- Develop algorithms through the process of top-down, stepwise refinement.
- Use the `if` and `if...else` selection statements to choose between alternative actions.
- Use the `while` iteration statement to execute statements in a program repeatedly.
- Use counter-controlled iteration and sentinel-controlled iteration.
- Use the compound assignment operator and the increment and decrement operators.
- Learn about the portability of primitive data types.

## Self-Review Exercises


**3.1** Fill in the blanks in each of the following statements:

a) All programs can be written in terms of three types of control structures: _____, _____ and _____.

**ANS:** sequence, selection, iteration.

b) The _____ statement is used to execute one action when a condition is true and another when that condition is false.

**ANS:** `if...else`.

c) Repeating a set of instructions a specific number of times is called _____ iteration.

**ANS:** counter-controlled (or definite).

d) When it's not known in advance how many times a set of statements will be repeated, a(n) _____ value can be used to terminate the iteration.

**ANS:** sentinel, signal, flag or dummy.

e) The _____ is built into Java; by default, statements execute in the order they appear.

**ANS:** sequence structure.

f) If the increment operator is _____ to a variable, first the variable is incremented by 1, then its new value is used in the expression.

**ANS:** prefixed.

g) When the declaration `int y = 5;` is followed by the assignment `y += 3.3;` the value of `y` is _____.

**ANS:** 8 [*Note:* You might expect a compilation error on the assignment statement. The Java Language Specification says that compound assignment operators perform an implicit cast on the right-hand expression's value to match the type of the variable on the operator's left side. So the calculated value `5 + 3.3 = 8.3` is actually cast to the `int` value `8`.].

**3.2** State whether each of the following is *true* or *false*. If *false*, explain why.

a) An algorithm is a procedure for solving a problem in terms of the actions to execute and the order in which they execute.

**ANS:** True.

b) A set of statements contained within a pair of parentheses is called a block.

**ANS:** False. A set of statements contained within a pair of braces (`{` and `}`) is called a block.

c) A selection statement repeats an action while a condition remains true.

**ANS:** False. An iteration statement specifies that an action is to be repeated while some condition remains true.

d) A nested control statement appears in the body of another control statement.

**ANS:** True.

e) Java provides the arithmetic compound assignment operators `+=`, `-=`, `*=`, `/=` and `%=` for abbreviating assignment expressions.

**ANS:** True.

f) The primitive types (`boolean`, `char`, `byte`, `short`, `int`, `long`, `float` and `double`) are portable across only Windows platforms.

**ANS:** False. The primitive types (`boolean`, `char`, `byte`, `short`, `int`, `long`, `float` and `double`) are portable across all computer platforms that support Java.

g) Specifying the order in which statements execute in a program is called program control.

**ANS:** True.

h) The unary cast operator `(double)` creates a temporary integer copy of its operand.

**ANS:** False. The unary cast operator `(double)` creates a temporary floating-point copy of its operand.

i) Instance variables of type `boolean` are given the value `true` by default.

**ANS:** False. Instance variables of type `boolean` are given the value `false` by default.

j) Pseudocode helps you think out a program before attempting to write it in a programming language.

**ANS:** True.

**3.3** Write four different Java statements that each add 1 to integer variable `x`.

**ANS:** The four ways to add `1` to `x` are:

```
x = x + 1;
x += 1;
++x;
x++;
```

**3.4** Write Java statements to accomplish each of the following tasks:

a) Use one statement to assign the sum of `x` and `y` to `z`, then increment `x` by 1.

**ANS:** `z = x++ + y;`

b) Test whether variable `count` is greater than 10. If it is, print `"Count is greater than 10"`.

**ANS:**

```
if (count > 10) {
   System.out.println("Count is greater than 10");
}
```

c) Use one statement to decrement the variable `x` by 1, then subtract it from variable `total` and store the result in variable `total`.

**ANS:** `total -= --x;`

d) Calculate the remainder after `q` is divided by `divisor`, and assign the result to `q`. Write this statement in two different ways.

**ANS:**

```
q %= divisor;
q = q % divisor;
```

**3.5** Write a Java statement to accomplish each of the following tasks:

a) Declare variable `sum` of type `int` and initialize it to `0`.

**ANS:** `int sum = 0;`

b) Declare variable `x` of type `int` and initialize it to `1`.

**ANS:** `int x = 1;`

c) Add variable `x` to variable `sum`, and assign the result to variable `sum`.

**ANS:** `sum += x;` or `sum = sum + x;`

d) Print `"The sum is: "`, followed by the value of variable `sum`.

**ANS:** `System.out.printf("The sum is: %d%n", sum);`

**3.6** Combine the statements that you wrote in Exercise 3.5 into a Java application that calculates and prints the sum of the integers from 1 to 10. Use a `while` statement to loop through the calculation and increment statements. The loop should terminate when the value of `x` becomes 11.

**ANS:** The answer to Exercise 3.6 is as follows:

```
// Exercise 3.6: Calculate.java
// Calculate the sum of the integers from 1 to 10
public class Calculate {
   public static void main(String[] args) {
      int sum = 0;
      int x = 1;

```

```
8          while (x <= 10) { // while x is less than or equal to 10
9             sum += x; // add x to sum
10            ++x; // increment x
11         }
12
13         System.out.printf("The sum is: %d%n", sum);
14      }
15   }
```

```
The sum is: 55
```

**3.7** Determine the value of the variables in the statement `product *= x++;` after the calculation is performed. Assume that all variables are type `int` and initially have the value 5.

**ANS:** `product = 25, x = 6`

**3.8** Identify and correct the errors in each of the following code segments—assume that all variables have been properly declared and initialized:

a)

```
1  while (c <= 5) {
2     product *= c;
3     ++c;
```

**ANS:** Error: The closing right brace of the `while` statement's body is missing.
Correction: Add a closing right brace after the statement `++c;`.

b)

```
1  if (gender == 1) {
2     System.out.println("Woman");
3  }
4  else; {
5     System.out.println("Man");
6  }
```

**ANS:** Error: The semicolon after `else` results in a logic error. The second output statement will always be executed.
Correction: Remove the semicolon after `else`.

**3.9** What is wrong with the following `while` statement?

```
1  while (z >= 0) {
2     sum += z;
3  }
```

**ANS:** The value of the variable `z` is never changed in the `while` statement. Therefore, if the loop-continuation condition `(z >= 0)` is true, an infinite loop is created. To prevent an infinite loop from occurring, `z` must be decremented so that it eventually becomes less than 0.

## Exercises

**3.10** Compare and contrast the `if` single-selection statement and the `while` iteration statement. How are these two statements similar? How are they different?

**ANS:** The `if` single-selection statement and the `while` iteration statement both perform an action (or set of actions) based on whether a condition is true or false. However, if

the condition is true, the `if` single-selection statement performs the action(s) once, whereas the `while` iteration statement repeatedly performs the action(s) until the condition becomes false.

**3.11** Explain what happens when a Java program attempts to divide one integer by another. What happens to the fractional part of the calculation? How can you avoid that outcome?

**ANS:** Dividing two integers results in integer division—any fractional part of the calculation is lost (i.e., truncated). For example, 7 ÷ 4, which yields 1.75 in conventional arithmetic, truncates to 1 in integer arithmetic, rather than rounding to 2. To obtain a floating-point result from dividing integer values, a programmer must temporarily treat these values as floating-point numbers in the calculation by using the unary cast operator `(double)`. As long as the `(double)` cast operator is applied to any variable in the calculation, the calculation will yield a `double` result which can be assigned to a `double` variable.

**3.12** Describe the two ways in which control statements can be combined.

**ANS:** Control statements can be "attached" (that is, stacked) to one another by connecting the exit point of one to the entry point of the next. Control statements also may be nested by placing one inside another.

**3.13** What type of iteration would be appropriate for calculating the sum of the first 100 positive integers? What type would be appropriate for calculating the sum of an arbitrary number of positive integers? Briefly describe how each of these tasks could be performed.

**ANS:** Counter-controlled iteration would be appropriate for calculating the sum of the first 100 positive integers because the number of repetitions is known in advance. The program that performs this task could use a `while` iteration statement with a counter variable that takes on the values 1 to 100. The program could then add the current counter value to the total variable in each repetition of the loop. Sentinel-controlled iteration would be appropriate for calculating the sum of an arbitrary number of positive integers. The program that performs this task could use a sentinel value of –1 to mark the end of data entry. The program could use a `while` iteration statement that totals positive integers from the user until the user enters the sentinel value.

**3.14** What is the difference between preincrementing and postincrementing a variable?

**ANS:** Preincrementing a variable causes it to be incremented by 1, and then the new value of the variable is used in the expression in which it appears. Postincrementing a variable causes the current value of the variable to be used in the expression in which it appears, and then the variable's value is incremented by 1. Preincrementing and postincrementing a variable have the same effect when the incrementing operation appears in a statement by itself.

**3.15** Identify and correct the errors in each of the following pieces of code. [*Note:* There may be more than one error in each piece of code.]

a)

```
1  int x = 1, total;
2  while (x <= 10) {
3     total += x;
4     ++x;
5  }
```

**ANS:** The variable `total` should be initialized to zero.

b)

```
1  while (x <= 100)
2     total += x;
3     ++x;
```

**ANS:** The two statements should be enclosed in braces to properly group them into the body of the `while`; otherwise the loop will be an infinite loop.

c)

```
1  while (y > 0) {a
2     System.out.println(y);
3     ++y;
```

**ANS:** The ++ operator should be changed to --; otherwise the loop will be an infinite loop if y starts with a value greater than 0. The closing brace for the `while` loop is missing.

**3.16** What does the following program print?

```
1  // Exercise 3.16: Mystery.java
2  public class Mystery {
3     public static void main(String[] args) {
4        int x = 1;
5        int total = 0;
6
7        while (x <= 10) {
8           int y = x * x;
9           System.out.println(y);
10          total += y;
11          ++x;
12       }
13
14       System.out.printf("Total is %d%n", total);
15    }
16 }
```

**ANS:**

```
1
4
9
16
25
36
49
64
81
100
Total is 385
```

**3.25** What does the following program print?

```
1  // Exercise 3.25: Mystery2.java
2  public class Mystery2 {
3     public static void main(String[] args) {
4        int count = 1;
```

```
5
6           while (count <= 10) {
7              System.out.println(count % 2 == 1 ? "****" : "++++++++");
8              ++count;
9           }
10       }
11    }
```

**ANS:**

```
****
++++++++
****
++++++++
****
++++++++
****
++++++++
****
++++++++
```

**3.26** What does the following program print?

```
1   // Exercise 3.26: Mystery3.java
2   public class Mystery3 {
3      public static void main(String[] args) {
4         int row = 10;
5
6         while (row >= 1) {
7            int column = 1;
8
9            while (column <= 10) {
10              System.out.print(row % 2 == 1 ? "<" : ">");
11              ++column;
12           }
13
14           --row;
15           System.out.println();
16        }
17     }
18  }
```

**ANS:**

```
>>>>>>>>>>
<<<<<<<<<<
>>>>>>>>>>
<<<<<<<<<<
>>>>>>>>>>
<<<<<<<<<<
>>>>>>>>>>
<<<<<<<<<<
>>>>>>>>>>
<<<<<<<<<<
```