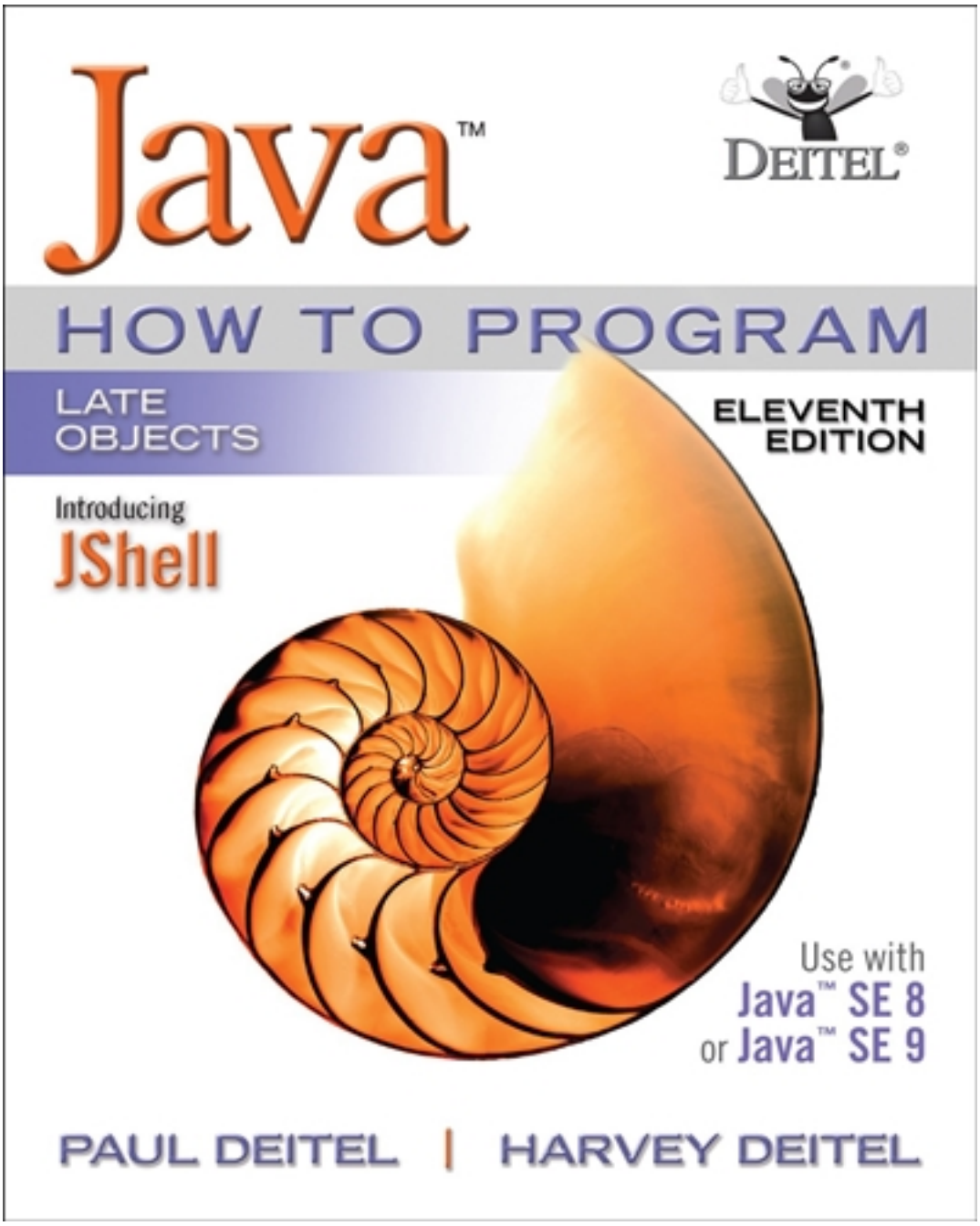
Solutions for Java How To Program Late Objects 11th Edition by Deitel

CLICK HERE TO ACCESS COMPLETE Solutions



Solutions

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) {
```

```
7         // create Scanner to obtain input from command window
8         Scanner input = new Scanner(System.in);
9
10        System.out.print("Enter first integer: "); // prompt for input
11        int x = input.nextInt(); // read first integer
12
13        System.out.print("Enter second integer: "); // prompt for input
14        int y = input.nextInt(); // read second integer
15
16        System.out.print("Enter third integer: "); // prompt for input
17         int z = input.nextInt(); // read third integer
18
19         int result = x * y * z; // calculate product of numbers
20
21        System.out.printf("Product is %d%n", result);
22     } // end method main
23  } // 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?

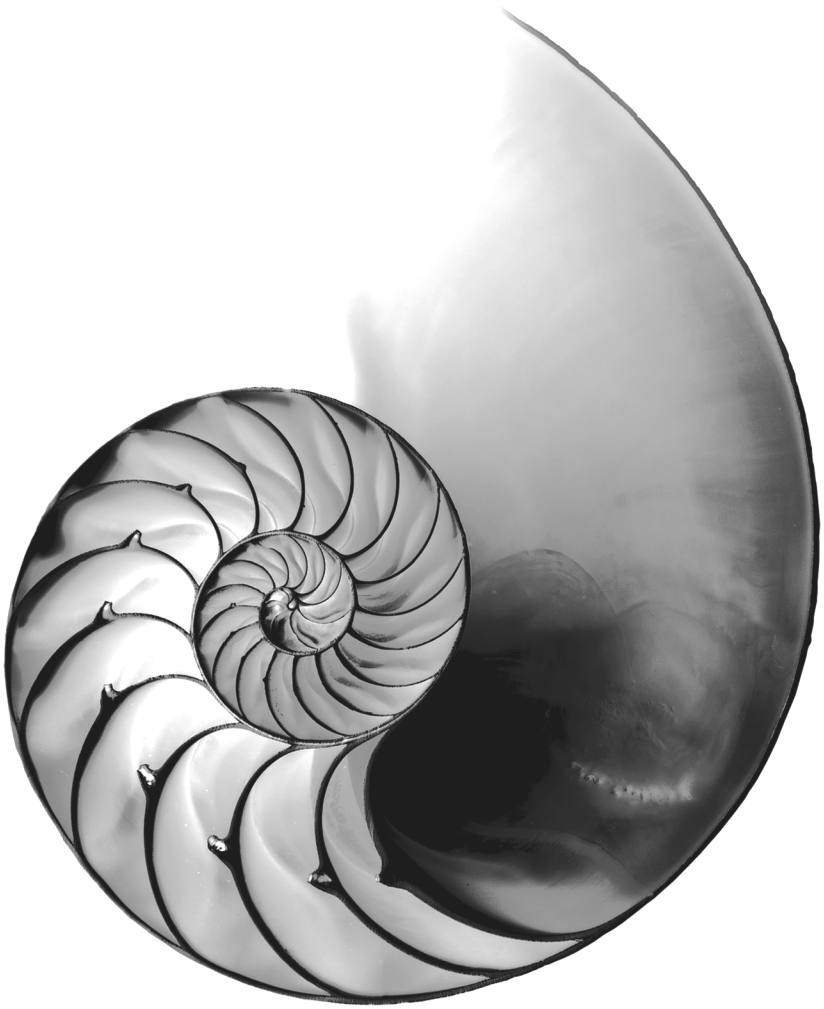
```
System.out.printf("%s%n%s%n%s%n", "*", "***", "*****");
```

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.