Chapter 2 Elementary Programming

Section 2.2 Writing a Simple Program
1. _____ is the code with natural language mixed with Java code.
a. Java program
b. A Java statement
c. Pseudocode
d. A flowchart diagram
key:c

#
2. What is the exact output of the following code?

```
double area = 3.5;
System.out.print("area");
System.out.print(area);
```

a. 3.53.5
b. 3.5 3.5
c. area3.5
d. area 3.5
Key:c The first print statement prints a string followed by the second print statement that prints a number.

#
Section 2.3 Reading Input from the Console
3. Suppose a Scanner object is created as follows:

```
Scanner input = new Scanner(System.in);
```

What method do you use to read a real number?

a. input.nextDouble();
b. input.nextdouble();
c. input.double();
d. input.Double();
Key:a The correct method to read a real number is nextDouble().

#
4. The following code fragment reads in two numbers:

```
Scanner input = new Scanner(System.in);
int i = input.nextInt();
double d = input.nextDouble();
```

What is the incorrect way to enter these two numbers?

a. Enter an integer, a space, a double value, and then the Enter key.
b. Enter an integer, two spaces, a double value, and then the Enter key.
c. Enter an integer, an Enter key, a double value, and then the Enter key.
d. Enter a numeric value with a decimal point, a space, an integer, and then the Enter key.
Key:d

#
5. If you enter 1 2 3, when you run this program, what will be the output?

```
import java.util.Scanner;

public class Test1 {
  public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.print("Enter three numbers: ");
    double number1 = input.nextDouble();
    double number2 = input.nextDouble();
    double number3 = input.nextDouble();

    // Compute average
    double average = (number1 + number2 + number3) / 3;

    // Display result
    System.out.println(average);
  }
}
```

a. 1.0
b. 2.0
c. 3.0
d. 4.0
Key:b (1.0 + 2.0 + 3.0) / 3 is 2.0

#
Section 2.4 Identifiers
6. Every letter in a Java keyword is in lowercase?
a. true
b. false
Key:a It is true that the keywords in Java are in lowercase. For example, public, static, int, double, and void are the keywords.

#
7. Which of the following is a valid identifier?
a. $343
b. class
c. 9X
d. 8+9
e. radius
Key:ae class is a keyword, which cannot be used as an identifier. Identifiers cannot start with a number.

#
Section 2.5 Variables
8. Which of the following are correct names for variables according to Java naming conventions?
a. radius
b. Radius
c. RADIUS

d. findArea
e. FindArea
Key:ad A single-word variable is in lowercase. In a multiple-word variable, the words are concatenated with the first word in lowercase and the first letter of each subsequent word in uppercase.

#
9. Which of the following are correct ways to declare variables?
a. int length; int width;
b. int length, width;
c. int length; width;
d. int length, int width;
Key:ab Note that a semicolon ends a statement. In B, length and width are both declared as int.

#
Section 2.6 Assignment Statements and Assignment Expressions
10. _____ is the Java assignment operator.
a. ==
b. :=
c. =
d. =:
Key:c

#
11. To assign a value 1 to variable x, you write
a. 1 = x;
b. x = 1;
c. x := 1;
d. 1 := x;
e. x == 1;
Key:b

#
12. Which of the following assignment statements is incorrect?
a. i = j = k = 1;
b. i = 1; j = 1; k = 1;
c. i = 1 = j = 1 = k = 1;
d. i == j == k == 1;
Key:cd

#
Section 2.7 Named Constants
13. To declare a constant MAX_LENGTH inside a method with value 99.98, you write
a. final MAX_LENGTH = 99.98;
b. final float MAX_LENGTH = 99.98;
c. double MAX_LENGTH = 99.98;
d. final double MAX_LENGTH = 99.98;
Key:d

#
14. Which of the following is a constant, according to Java naming conventions?

a. MAX_VALUE
b. Test
c. read
d. ReadInt
e. COUNT
Key:ae All letters in a constant are in uppercase. In a multiple-word constant, the words are connected using underscores.

#
15. To improve readability and maintainability, you should declare _____ instead of using literal values such as 3.14159.
a. variables
b. methods
c. constants
d. classes
Key:c A constant gives a literal a descriptive name and makes the code more readable.

#
Section 2.8 Naming Conventions
16. According to Java naming convention, which of the following names can be variables?
a. FindArea
b. findArea
c. totalLength
d. TOTAL_LENGTH
e. class
Key:bc The first word in a variable is in lowercase. So B and C are correct.

#
Section 2.9 Numeric Data Types and Operations
Section 2.9.1 Numeric Types
17. Which of these data types requires the most amount of memory?
a. long
b. int
c. short
d. byte
Key:a long takes 8 bytes. int 4 bytes. short 2 bytes. byte 1 byte.

#
18. When assigning a literal to a variable of the byte type, if the literal is too large to be stored as a byte value, it _____.
a. causes overflow
b. causes underflow
c. causes no error
d. cannot happen in Java
e. receives a compile error
Key:e For example, byte b = 23232 will cause a compile error.

#
Section 2.9.3 Numeric Operators
19. What is the result of 45 / 4?
a. 10

b. 11
c. 11.25
d. 12
Key:b 45 / 4 is an integer division, which results in 11

#
20. Which of the following expression results in a value 1?
a. 2 % 1
b. 15 % 4
c. 25 % 5
d. 37 % 6
Key:d 2 % 1 is 0, 15 % 4 is 3, 25 % 5 is 0, and 37 % 6 is 1

#
21. 25 % 1 is _____
a. 1
b. 2
c. 3
d. 4
e. 0
Key:e

#
22. -25 % 5 is _____
a. 1
b. 2
c. 3
d. 4
e. 0
Key:e

#
23. 24 % 5 is _____
a. 1
b. 2
c. 3
d. 4
e. 0
Key:d

#
24. -24 % 5 is _____
a. -1
b. -2
c. -3
d. -4
e. 0
Key:d

#
25. -24 % -5 is _____
a. 3
b. -3
c. 4

d. -4
e. 0
Key:d

#
Section 2.9.4 Exponent Operations
26. How do you write 2.5 ^ 3.1 in Java?
a. 2.5 * 3.1
b. Math.pow(2.5, 3.1)
c. Math.pow(3.1, 2.5)
d. 2.5 ** 3.1
e. 3.1 ** 2.5
Key:b

#
27. Math.pow(2, 3) returns _____.
a. 9
b. 8
c. 9.0
d. 8.0
Key:d It returns a double value 8.0.

#
28. Math.pow(4, 1 / 2) returns _____.
a. 2
b. 2.0
c. 0
d. 1.0
e. 1
Key:d Note that 1 / 2 is 0.

#
29. Math.pow(4, 1.0 / 2) returns _____.
a. 2
b. 2.0
c. 0
d. 1.0
e. 1
Key:b Note that the pow method returns a double value, not an integer.

#
30. The _____ method returns a raised to the power of b.
a. Math.power(a, b)
b. Math.exponent(a, b)
c. Math.pow(a, b)
d. Math.pow(b, a)
Key:c

#
Section 2.10 Numeric Literals
31. To declare an int variable number with initial value 2, you write
a. int number = 2L;
b. int number = 2l;
c. int number = 2;

d. int number = 2.0;
Key:c

#
32. Analyze the following code.

```
public class Test {
  public static void main(String[] args) {
    int month = 09;
    System.out.println("month is " + month);
  }
}
```

a. The program displays month is 09.
b. The program displays month is 9.
c. The program displays month is 9.0.
d. The program has a syntax error, because 09 is an incorrect literal value.
Key:d Any numeric literal with the prefix 0 is an octal value. But 9 is not an octal digit. An octal digit is 0, 1, 2, 3, 4, 5, 6, or 7.

#
33. Which of the following is incorrect?
a. 1_2
b. 0.4_56
c. 1_200_229
d. _4544
Key:d You can use the digit separator _ for intergers or floating point numbers. The separator must be placed between the digits.

#
34. Which of the following are the same as 1545.534?
a. 1.545534e+3
b. 0.1545534e+4
c. 1545534.0e-3
d. 154553.4e-2
Key:abcd

#
35. Which of the following is incorrect?
a. int x = 9;
b. long x = 9;
c. float x = 1.0;
d. double x = 1.0;
Key:c

#
Section 2.11 Evaluating Expressions and Operator Precedence
36. The expression 4 + 20 / (3 - 1) * 2 is evaluated to
a. 4
b. 20
c. 24
d. 9
e. 25
Key:c

#
Section 2.12 Case Study: Displaying the Current Time
37. The System.currentTimeMillis() returns _____ .
a. the current time.
b. the current time in milliseconds.
c. the current time in milliseconds since midnight.
d. the current time in milliseconds since midnight, January 1, 1970.
e. the current time in milliseconds since midnight, January 1, 1970 GMT (the Unix time).
Key:e

#
38. To obtain the current second, use _____.
a. System.currentTimeMillis() % 3600
b. System.currentTimeMillis() % 60
c. System.currentTimeMillis() / 1000 % 60
d. System.currentTimeMillis() / 1000 / 60 % 60
e. System.currentTimeMillis() / 1000 / 60 / 60 % 24
Key:c

#
39. To obtain the current minute, use _____.
a. System.currentTimeMillis() % 3600
b. System.currentTimeMillis() % 60
c. System.currentTimeMillis() / 1000 % 60
d. System.currentTimeMillis() / 1000 / 60 % 60
e. System.currentTimeMillis() / 1000 / 60 / 60 % 24
Key:d

#
40. To obtain the current hour in UTC, use _____.
a. System.currentTimeMillis() % 3600
b. System.currentTimeMillis() % 60
c. System.currentTimeMillis() / 1000 % 60
d. System.currentTimeMillis() / 1000 / 60 % 60
e. System.currentTimeMillis() / 1000 / 60 / 60 % 24
Key:e

#
Section 2.13 Augmented Assignment Operators
41. To add a value 1 to variable x, you write
a. 1 + x = x;
b. x += 1;
c. x := 1;
d. x = x + 1;
e. x = 1 + x;
Key:bde

#
42. To add number to sum, you write (Note: Java is case-sensitive)
a. number += sum;
b. number = sum + number;
c. sum = Number + sum;

d. sum += number;
e. sum = sum + number;
Key:de

#
43. Suppose x is 1. What is x after x += 2?
a. 0
b. 1
c. 2
d. 3
e. 4
Key:d

#
44. Suppose x is 1. What is x after x -= 1?
a. 0
b. 1
c. 2
d. -1
e. -2
Key:a

#
45. What is x after the following statements?

```
int x = 2;
int y = 1;
x *= y + 1;
```

a. x is 1.
b. x is 2.
c. x is 3.
d. x is 4.
Key:d (y + 1) is executed first and its result is multiplied with x and assigned to x.

#
46. What is x after the following statements?

```
int x = 1;
x *= x + 1;
```

a. x is 1.
b. x is 2.
c. x is 3.
d. x is 4.
Key:b

#
47. Which of the following statements are the same?

(A) x -= x + 4
(B) x = x + 4 - x
(C) x = x - (x + 4)

a. (A) and (B) are the same
b. (A) and (C) are the same
c. (B) and (C) are the same
d. (A), (B), and (C) are the same
Key:b

#
Section 2.14 Increment and Decrement Operators
48. Are the following four statements equivalent?

```
number += 1;
number = number + 1;
number++;
++number;
```

a. Yes
b. No
Key:a

#
49. What is i printed?

```
public class Test {
  public static void main(String[] args) {
    int j = 0;
    int i = ++j + j * 5;

    System.out.println("What is i? " + i);
  }
}
```

a. 0
b. 1
c. 5
d. 6
Key:d Operands are evaluated from left to right in Java. The left-hand operand of a binary operator is evaluated before any part of the right-hand operand is evaluated. This rule takes precedence over any other rules that govern expressions. Therefore, ++j is evaluated first, and j is now 1. Then j * 5 is evaluated, returns 5. So, i is 6.

#
50. What is i printed in the following code?

```
public class Test {
  public static void main(String[] args) {
    int j = 0;
    int i = j++ + j * 5;

    System.out.println("What is i? " + i);
  }
}
```

a. 0
b. 1
c. 5

d. 6
Key:c Operands are evaluated from left to right in Java. The left-hand operand of a binary operator is evaluated before any part of the right-hand operand is evaluated. This rule takes precedence over any other rules that govern expressions. Therefore, j++ is evaluated first. j is now 1. Since j++ is postincrement, the old value of j is returned for j++. So j++ + j * 5 equals 0 + 1 * 5. So, the result is 5.

#
51. What is y displayed in the following code?

```
public class Test {
  public static void main(String[] args) {
    int x = 1;
    int y = x++ + x;
    System.out.println("y is " + y);
  }
}
```

a. y is 1.
b. y is 2.
c. y is 3.
d. y is 4.
Key:c When evaluating x++ + x, x++ is evaluated first, which does two things: 1. returns 1 since it is post-increment. x becomes 2. Therefore y is 1 + 2.

#
52. What is y displayed?

```
public class Test {
  public static void main(String[] args) {
    int x = 1;
    int y = x + x++;
    System.out.println("y is " + y);
  }
}
```

a. y is 1.
b. y is 2.
c. y is 3.
d. y is 4.
Key:b When evaluating x + x++, x is evaluated first, which is 1. X++ returns 1 since it is post-increment and 2. Therefore y is 1 + 1.

#
Section 2.15 Numeric Type Conversions
53. To assign a double variable d to a float variable x, you write
a. x = (long)d
b. x = (int)d;
c. x = d;
d. x = (float)d;
Key:d

#
54. Which of the following expressions will yield 0.5?

a. 1 / 2
b. 1.0 / 2
c. (double) (1 / 2)
d. (double) 1 / 2
e. 1 / 2.0
Key:bde 1 / 2 is an integer division, which results in 0.

#
55. What is the output of the following code:

```
double x = 5.5;
int y = (int)x;
System.out.println("x is " + x + " and y is " + y);
```

a. x is 5 and y is 6
b. x is 6.0 and y is 6.0
c. x is 6 and y is 6
d. x is 5.5 and y is 5
e. x is 5.5 and y is 5.0
Key:d The value is x is not changed after the casting.

#
56. Which of the following assignment statements is illegal?
a. float f = -34;
b. int t = 23;
c. short s = 10;
d. int t = (int)false;
e. int t = 4.5;
Key:de

#
57. What is the value of (double)5/2?
a. 2
b. 2.5
c. 3
d. 2.0
e. 3.0
Key:b

#
58. What is the value of (double)(5/2)?
a. 2
b. 2.5
c. 3
d. 2.0
e. 3.0
Key:d

#
59. Which of the following expression results in 45.37?
a. (int)(45.378 * 100) / 100
b. (int)(45.378 * 100) / 100.0
c. (int)(45.378 * 100 / 100)
d. (int)(45.378) * 100 / 100.0
Key:b

#
60. The expression (int)(76.0252175 * 100) / 100 evaluates to _____.
a. 76.02
b. 76
c. 76.0252175
d. 76.03
Key:b In order to obtain 76.02, you have divide 100.0.

#
61. If you attempt to add an int, a byte, a long, and a double, the result will be a _____ value.
a. byte
b. int
c. long
d. double
Key:d

#
Section 2.16 Software Life Cycle
62. _____ is a formal process that seeks to understand the problem and document in detail what the software system needs to do.
a. Requirements specification
b. Analysis
c. Design
d. Implementation
e. Testing
Key:a

#
63. _____ seeks to analyze the data flow and to identify the system's input and output. When you do analysis, it helps to identify what the output is first, and then figure out what input data you need in order to produce the output.
a. Requirements specification
b. Analysis
c. Design
d. Implementation
e. Testing
Key:b

#
Section 2.18 Common Errors and Pitfalls
64. Analyze the following code:

```
public class Test {
  public static void main(String[] args) {
    int n = 10000 * 10000 * 10000;
    System.out.println("n is " + n);
  }
}
```

a. The program displays n is 1000000000000.
b. The result of 10000 * 10000 * 10000 is too large to be stored in an int variable n. This causes an overflow and the program is aborted.

c. The result of 10000 * 10000 * 10000 is too large to be stored in an int variable n. This causes an overflow and the program continues to execute because Java does not report errors on overflow.
d. The result of 10000 * 10000 * 10000 is too large to be stored in an int variable n. This causes an underflow and the program is aborted.
e. The result of 10000 * 10000 * 10000 is too large to be stored in an int variable n. This causes an underflow and the program continues to execute because Java does not report errors on underflow.
Key:c