

Import Settings:

Base Settings: Brownstone Default

Information Field: Complexity

Information Field: Ahead

Information Field: Subject

Information Field: Title

Information Field: Taxonomy

Highest Answer Letter: E

Multiple Keywords in Same Paragraph: No

NAS ISBN13: 9781284239928, add to Ahead, Title tags

Chapter: Chapter 02 - Quiz

Multiple Choice

1. Which one of these is *not* a primitive data type?

A) *int*

B) *double*

C) *bool*

D) *long*

E) *char*

Ans: C

Complexity: Easy

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Analysis

2. The result of $3 / 2$ is:

A) 1.

B) 1.5.

C) 3.

D) 2.

E) 0.

Ans: A

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

3. The result of $9 \% 2$ is:

A) 1.

B) 4.5.

C) 4.

D) 2.

E) 0.

Ans: A

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

4. The result of $3.0 / 2$ is:

A) 1.

B) 1.5.

C) 3.

D) 2.

E) 0.

Ans: B

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

5. The result of $(\text{double}) (3 / 2)$ is:

A) 1.0.

B) 1.5.

C) 3.0.

D) 2.0.

E) 0.0.

Ans: A

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

6. The result of $(\text{double}) 3 / 2$ is:

A) 1.0.

B) 1.5.

C) 3.0.

D) 2.0.

E) 0.0.

Ans: B

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

True/False

1. True or False? *helloWorld* is a valid identifier.

Ans: True

Complexity: Moderate

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

2. True or False? *int9* is a valid identifier.

Ans: True

Complexity: Moderate

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

3. True or False? *char* is a valid identifier.

Ans: False

Complexity: Moderate

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

4. True or False? *2008taxRate* is a valid identifier.

Ans: False

Complexity: Moderate

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

5. True or False? *tax Rate* is a valid identifier.

Ans: False

Complexity: Moderate

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

6. True or False? A *double* is more precise than a *float*.

Ans: True

Complexity: Easy

Ahead: Data Types, Variables, and Constants
Subject: Chapter 2
Title: Programming Building Blocks—Java Basics
Taxonomy: Recall

7. True or False? In the += shortcut operator, there is a space between + and =.

Ans: False
Complexity: Easy
Ahead: Expressions and Arithmetic Operators
Subject: Chapter 2
Title: Programming Building Blocks—Java Basics
Taxonomy: Recall

8. True or False? * is a unary operator.

Ans: False
Complexity: Moderate
Ahead: Expressions and Arithmetic Operators
Subject: Chapter 2
Title: Programming Building Blocks—Java Basics
Taxonomy: Recall

Short Answer

1. The Java keyword for a constant is _____.

Ans: final
Complexity: Easy
Ahead: Data Types, Variables, and Constants
Subject: Chapter 2
Title: Programming Building Blocks—Java Basics
Taxonomy: Recall

2. The two floating point data types are _____ and float.

Ans: double
Complexity: Easy
Ahead: Data Types, Variables, and Constants
Subject: Chapter 2
Title: Programming Building Blocks—Java Basics
Taxonomy: Recall

3. *char* values are surrounded by _____ quotes.

Ans: single
Complexity: Easy
Ahead: Data Types, Variables, and Constants
Subject: Chapter 2

Title: Programming Building Blocks—Java Basics
Taxonomy: Recall

4. *String* literals are surrounded by _____ quotes.

Ans: double

Complexity: Easy

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

5. A Java escape sequence starts with a(n) _____.

Ans: \

Complexity: Easy

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

6. The division operator / returns the _____ of the division between two numbers.

Ans: quotient

Complexity: Easy

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall

Essay

1. Describe two ways to introduce a comment.

Ans: 1. Put the text between /* and */. 2. Precede the comment with two forward slashes (//).

Complexity: Easy

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

2. Compare and contrast single-precision and double-precision floating-point data types.

Ans: The single-precision type (*float*) is stored in 32 bits. Double-precision type (*double*) is stored in 64 bits. Because of its greater precision, the *double* data type is usually preferred over the *float* data type. However, for calculations not requiring such precision, *floats* are often used because they require less memory. Both can be positive or negative.

Complexity: Moderate

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2
Title: Programming Building Blocks—Java Basics
Taxonomy: Analysis

3. When you declare a variable, with an assignment operator (=) with the following syntax:

```
dataType variableName = initialValue;
```

How would you read the sentence?

Ans: “*VariableName* gets *initialValue*.”

Complexity: Moderate

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

4. Would it be better to use doubles or floats for calculations to scale up recipes to feed three times as many people as the recipes are written to serve?

Ans: Float; these are simple calculations that do not require the precision or memory that doubles would require.

Complexity: Difficult

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Analysis

5. What are three ways to add 1 to an *int* variable named *number*?

Ans:

```
number++;
```

```
++number;
```

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

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

6. Assuming *number* is an *int* variable, write a statement using a shortcut operator equivalent to this one:

```
number = number * 5;
```

Ans: *number* *= 5;

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Taxonomy: Application

7. Rewrite the following into one statement:

```
int age;
```

```
age = 32;
```

Ans: `int age = 32;`

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

8. You want to insert the comment “Java is better than ice cream.” Identify two ways to include this comment in your code.

Ans: 1. `*Java is better than ice cream.*`. 2. `//Java is better than ice cream.`

Complexity: Easy

Ahead: Java Application Structure

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

9. What are some advantages of using variables?

Ans: You can name a variable, assign it a value, and refer to the name of a variable in an expression rather than hard coding the specific value.

Complexity: Moderate

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

10. Compare the variable identifiers *numberOfStudentsPassing* and *winners*. Describe which would be preferable in a program that evaluates success of a teaching program.

Ans: *numberOfStudentsPassing* would be preferable. It clearly indicates what is being measured.

Although this might be a positive thing, making the students “winners,” the identifier *winners* is too vague and users might not remember its meaning, leading to errors in data entry.

Complexity: Moderate

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Analysis

11. Sam wants to calculate whether his students simply passed or failed. He uses a *char* variable, because student grades are A through F. Is this the best data type to use? Why or why not?

Ans: No, Sam should use a *boolean* data type to capture the two possible values (passed/failed).

Complexity: Difficult

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

12. One way to specify an initial value for a variable (variable 1) is to assign the variable the value of another variable (variable 2). Explain why it is not valid to assign variable 1 as a double and variable 2 as a float.

Ans: Because a float is less precise than a double.

Complexity: Difficult

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Analysis

13. What will happen if your *String* literal is longer than one line?

Ans: The statement will generate compiler errors.

Complexity: Easy

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Analysis

14. List two errors in the following statement:

`System.out.println("Remember the slogan "Click It or Ticket" and buckle
your seatbelt before driving your car.");`

Ans: The *String* concatenation operator (+) is missing and there are illegal quotes within the *String* quotes without escape sequences.

Complexity: Moderate

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Analysis

15. How would you create an indented poem in a *String* literal statement?

Ans: Insert the `\t` escape sequence before each line of the literal to be indented.

Complexity: Difficult

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

16. Would it be better to write `PI` or `3.14159` in an arithmetic expression?

Ans: `PI`, because it is more meaningful.

Complexity: Moderate

Ahead: Data Types, Variables, and Constants

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

17. In which order should the computer do the following calculation? Solve the expression.

`int dimes = 2 * 20 + 5 * 10 + 3 * 5;`

Ans: Do the multiplication and then the addition: $(2 * 20) + (5 * 10) + (3 * 5) = 40 + 50 + 15 = 105$

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Application

18. What does the shortcut `++` do?

Ans: It adds 1 to the value of the variable before or after it, and to the result of the statement.

Complexity: Moderate

Ahead: Expressions and Arithmetic Operators

Subject: Chapter 2

Title: Programming Building Blocks—Java Basics

Taxonomy: Recall