Self Check 1.1

What is required to play a music CD on a computer?

Answer: A program that reads the data on the CD and sends output to the speakers and the screen.

Self Check 1.2

Why is a CD player less flexible than a computer?

Answer: A CD player can do one thing – play music CDs. It cannot execute programs.

Self Check 1.3

Can a computer program develop the initiative to execute tasks in a better way than its programmers envisioned?

Answer: No – the program simply executes the instruction sequences that the programmers have prepared in advance.

Self Check 1.4

Where is a program stored when it is not currently running?

Answer: In secondary storage, typically a hard disk.

Self Check 1.5

Which part of the computer carries out arithmetic operations, such as addition and multiplication?

Answer: The central processing unit.

Self Check 1.6

What is the code for the Java virtual machine instruction “Load the contents of memory location 100”?

Answer: `21 100`

Self Check 1.7

Does a person who uses a computer for office work ever run a compiler?

Answer: No – a compiler is intended for programmers, to translate high-level programming instructions into machine code.

Self Check 1.8

What are the two most important benefits of the Java language?

Answer: Safety and portability.

Self Check 1.9

How long does it take to learn the entire Java library?

Answer: No one person can learn the entire library – it is too large.

Self Check 1.10

How would you modify the `HelloPrinter` program to print the words `"Hello,"` and `"World!"` on two lines?

Answer:

```
System.out.println("Hello,");
System.out.println("World!");
```

Self Check 1.11

Would the program continue to work if you omitted the line starting with `//`?

Answer: Yes – the line starting with `//` is a comment, intended for human readers. The compiler ignores comments.

Self Check 1.12

What does the following set of statements print?

```
System.out.print("My lucky number is");
System.out.println(3 + 4 + 5);
```

Answer: The printout is

```
My lucky number is12
```

It would be a good idea to add a space after the `is`.

Self Check 1.13

Can you use a word processor for writing Java programs?

Answer: Yes, but you must remember to save your file as "plain text".

Self Check 1.14

What do you expect to see when you load a class file into your text editor?

Answer: A sequence of random characters, some funny looking. Class files contain virtual machine instructions that are encoded as binary numbers.

Self Check 1.15

Suppose you omit the `//` characters from the `HelloPrinter.java` program but not the remainder of the comment. Will you get a compile-time error or a run-time error?

Answer: A compile-time error. The compiler will not know what to do with the word `Display`.

Self Check 1.16

When you used your computer, you may have experienced a program that “crashed” (quit spontaneously) or “hung” (failed to respond to your input). Is that behavior a compile-time error or a run-time error?

Answer: It is a run-time error. After all, the program had been compiled in order for you to run it.

Self Check 1.17

Why can't you test a program for run-time errors when it has compiler errors?

Answer: When a program has compiler errors, no class file is produced, and there is nothing to run.

Self Check 1.18

Investment Problem: You put $10,000 into a bank account that earns 5 percent interest per year. How many years does it take for the account balance to be double the original?

Algorithm:

Start with a year value of 0 and a balance of $10,000.
Repeat the following steps while the balance is less than $20,000.
  Add 1 to the year value.
  Multiply the balance value by 1.05 (a 5 percent increase).

Suppose the interest rate was 20 percent. How long would it take for the investment to double?

Answer: 4 years:

```
0 10,000
1 12,000
2 14,400
3 17,280
4 20,736
```

Self Check 1.19

Suppose your cell phone carrier charges you $29.95 for up to 300 minutes of calls, and $0.45 for each additional minute, plus 12.5 percent taxes and fees. Give an algorithm to compute the monthly charge for a given number of minutes.

Answer:

Is the number of minutes at most 300?
a. If so, the answer is $29.95 × 1.125 = $33.70.
b. If not,
   1. Compute the difference: (number of minutes) – 300.
   2. Multiply that difference by 0.45.
   3. Add $29.95.
   4. Multiply the total by 1.125. That is the answer.

Self Check 2.1

What is the type of the values `0` and `"0"`?

Answer: `int` and `String`.

Self Check 2.2

Which number type would you use for storing the area of a circle?

Answer: `double`.

Self Check 2.3

Why is the expression `13.println()` an error?

Answer: An `int` is not an object, and you cannot call a method on it.

Self Check 2.4

Write an expression to compute the average of the values `x` and `y`.

Answer: `(x + y) * 0.5`

Self Check 2.5

Which of the following are legal identifiers?

```
Greeting1
g
void
101dalmatians
Hello, World
<greeting>
```

Answer: Only the first two are legal identifiers.

Self Check 2.6

Define a variable to hold your name. Use camel case in the variable name.

Answer:

```
String myName = "John Q. Public";
```

Self Check 2.7

Is `12 = 12` a valid expression in the Java language?

Answer: No, the left-hand side of the `=` operator must be a variable.

Self Check 2.8

How do you change the value of the `greeting` variable to `"Hello, Nina!"`?

Answer:

```
greeting = "Hello, Nina!";
```

Note that

```
String greeting = "Hello, Nina!";
```

is not the right answer – that statement defines a new variable.

Self Check 2.9

How can you compute the length of the string `"Mississippi"`?

Answer: `river.length()` or `"Mississippi".length()`

Self Check 2.10

How can you print out the uppercase version of `"Hello, World!"`?

Answer:

```
System.out.println(greeting.toUpperCase());
```

Self Check 2.11

Is it legal to call `river.println()`? Why or why not?

Answer: It is not legal. The variable `river` has type `String`. The `println` method is not a method of the `String` class.

Self Check 2.12

What are the implicit parameters, explicit parameters, and return values in the method call `river.length()`?

Answer: The implicit parameter is `river`. There is no explicit parameter. The return value is 11.

Self Check 2.13

What is the result of the call `river.replace("p", "s")`?

Answer: `"Missississi"`.

Self Check 2.14

What is the result of the call

```
greeting.replace("World", "Dave").length()
```

?

Answer: `12`.

Self Check 2.15

How is the `toUpperCase` method defined in the `String` class?

Answer: As `public String toUpperCase()`, with no explicit parameter and return type `String`.

Self Check 2.16

How do you construct a square with center (100, 100) and side length 20?

Answer:

```
new Rectangle(90, 90, 20, 20)
```

Self Check 2.17

The `getWidth` method returns the width of a `Rectangle` object. What does the following statement print?

```
System.out.println(new Rectangle().getWidth());
```

Answer:

```
0
```

Self Check 2.18

Is the `toUpperCase` method of the `String` class an accessor or a mutator?

Answer: An accessor – it doesn't modify the original string but returns a new string with uppercase letters.

Self Check 2.19

Which call to `translate` is needed to move the `box` rectangle so that its top-left corner is the origin (0, 0)?

> **Answer:** `box.translate(-5, -10)`, provided the method is called immediately after storing the new rectangle into `box`.

Self Check 2.20

Look at the API documentation of the `String` class. Which method would you use to obtain the string `"hello, world!"` from the string `"Hello, World!"`?

> **Answer:** `toLowerCase`

Self Check 2.21

In the API documentation of the `String` class, look at the description of the `trim` method. What is the result of applying trim to the string `" Hello, Space ! "`? (Note the spaces in the string.)

Answer: `"Hello, Space !"` – only the leading and trailing spaces are trimmed.

Self Check 2.22

The `Random` class is defined in the `java.util` package. What do you need to do in order to use that class in your program?

Answer: Add the statement

```
import java.util.Random;
```

at the top of your program.

Self Check 2.23

Suppose we had called `box.translate(25, 15)` instead of `box.translate(15, 25)`. What are the expected outputs?

Answer:

```
x: 30, y: 25
```

Self Check 2.24

Why doesn't the `MoveTester` program print the width and height of the rectangle?

Answer: Because the `translate` method doesn't modify the shape of the rectangle.

Self Check 2.25

What is the effect of the assignment `greeting2 = greeting`?

Answer: Now `greeting` and `greeting2` both refer to the same `String` object.

Self Check 2.26

After calling `greeting2.toUpperCase()`, what are the contents of `greeting` and `greeting2`?

Answer: Both variables still refer to the same string, and the string has not been modified. Recall that the `toUpperCase` method constructs a new string that contains uppercase characters, leaving the original string unchanged.

Self Check 2.27

How do you display a square frame with a title bar that reads `"Hello, World!”`?

Answer: Modify the `EmptyFrameViewer` program as follows:

```
frame.setSize(300, 300);
frame.setTitle("Hello, World!");
```

Self Check 2.28

How can a program display two frames at once?

Answer: Construct two `JFrame` objects, set each of their sizes, and call `setVisible(true)` on each of them.

Self Check 2.29

How do you modify the program to draw two squares?

Answer:

```
Rectangle box = new Rectangle(5, 10, 20, 20);
```

Self Check 2.30

How do you modify the program to draw one rectangle and one square?

Answer: Replace the call to `box.translate(15, 25)` with

```
box = new Rectangle(20, 35, 20, 20);
```

Self Check 2.31

What happens if you call `g.draw(box)` instead of `g2.draw(box)`?

Answer: The compiler complains that `g` doesn't have a `draw` method.

Self Check 2.32

Give instructions to draw a circle with center (100, 100) and radius 25.

Answer:

```
g2.draw(new Ellipse2D.Double(75, 75, 50, 50));
```

Self Check 2.33

Give instructions to draw a letter "V" by drawing two line segments.

Answer:

```
Line2D.Double segment1 = new Line2D.Double(0, 0, 10, 30);
g2.draw(segment1);
Line2D.Double segment2 = new Line2D.Double(10, 30, 20, 0);
g2.draw(segment2);
```

Self Check 2.34

Give instructions to draw a string consisting of the letter "V".

Answer:

```
g2.drawString("V", 0, 30);
```

Self Check 2.35

What are the RGB color values of `Color.BLUE`?

Answer: `0, 0,` and `255`

Self Check 2.36

How do you draw a yellow square on a red background?

Answer: First fill a big red square, then fill a small yellow square inside:

```
g2.setColor(Color.RED);
g2.fill(new Rectangle(0, 0, 200, 200));
g2.setColor(Color.YELLOW);
g2.fill(new Rectangle(50, 50, 100, 100));
```

Self Check 3.1

Supply the body of a method `public void reset()` that resets the counter back to zero.

Answer:

```
public void reset()
{
   value = 0;
}
```

Self Check 3.2

Suppose you use a class `Clock` with private instance variables `hours` and `minutes`. How can you access these variables in your program?

Answer: You can only access them by invoking the methods of the `Clock` class.

Self Check 3.3

Consider the `Counter` class. A counter's value starts at 0 and is advanced by the `count` method, so it should never be negative. Suppose you found a negative `value` variable during testing. Where would you look for the error?

Answer: In one of the methods of the `Counter` class.

Self Check 3.4

In Chapters 1 and 2, you used `System.out` as a black box to cause output to appear on the screen. Who designed and implemented `System.out`?

Answer: The programmers who designed and implemented the Java library.

Self Check 3.5

Suppose you are working in a company that produces personal finance software. You are asked to design and implement a class for representing bank accounts. Who will be the users of your class?

Answer: Other programmers who work on the personal finance application.

Self Check 3.6

How can you use the methods of the public interface to *empty* the `harrysChecking` bank account?

Answer:

```
harrysChecking.withdraw(harrysChecking.getBalance())
```

Self Check 3.7

What is wrong with this sequence of statements?

```
BankAccount harrysChecking = new BankAccount(10000);
System.out.println(harrysChecking.withdraw(500));
```

Answer: The `withdraw` method has return type `void`. It doesn't return a value. Use the `getBalance` method to obtain the balance after the withdrawal.

Self Check 3.8

Suppose you want a more powerful bank account abstraction that keeps track of an *account number* in addition to the balance. How would you change the public interface to accommodate this enhancement?

Answer: Add an `accountNumber` parameter to the constructors, and add a `getAccountNumber` method. There is no need for a `setAccountNumber` method – the account number never changes after construction.

Self Check 3.9

Provide documentation comments for the `Counter` class of Section 3.1.

Answer:

```
/**
   This class models a tally counter.
*/
public class Counter
{
   private int value;
   /**
      Gets the current value of this counter.
      @return the current value
   */
   public int getValue()
   {
      return value;
   }
```

Continued

Self Check 3.9 (cont.)

```
   /**
      Advances the value of this counter by 1.
   */
   public void count()
   {
      value = value + 1;
   }
}
```

Self Check 3.10

Suppose we enhance the `BankAccount` class so that each account has an account number. Supply a documentation comment for the constructor

```
public BankAccount(int accountNumber, double initialBalance)
```

Answer:

```
/**
   Constructs a new bank account with a given initial balance.
   @param accountNumber the account number for this account
   @param initialBalance the initial balance for this account
*/
```

Self Check 3.11

Why is the following documentation comment questionable?

```
/**
   Each account has an account number.
   @return the account number of this account
*/
public int getAccountNumber()
```

Answer: The first sentence of the method description should describe the method – it is displayed in isolation in the summary table.

Self Check 3.12

Suppose we modify the `BankAccount` class so that each bank account has an account number. How does this change affect the instance variables?

Answer:

An instance variable

```
private int accountNumber;
```

needs to be added to the class.

Self Check 3.13

Why does the following code not succeed in robbing mom's bank account?

```
public class BankRobber
{
   public static void main(String[] args)
   {
      BankAccount momsSavings = new BankAccount(1000);
      momsSavings.balance = 0;
   }
}
```

Answer: Because the `balance` instance variable is accessed from the `main` method of `BankRobber`. The compiler will report an error because `balance` has private access in `BankAccount`.

Self Check 3.14

The `Rectangle` class has four instance variables: `x`, `y`, `width`, and `height`. Give a possible implementation of the `getWidth` method.

Answer:

```
public int getWidth()
{
   return width;
}
```

Self Check 3.15

Give a possible implementation of the `translate` method of the `Rectangle` class.

Answer: There is more than one correct answer. One possible implementation is as follows:

```
public void translate(int dx, int dy)
{
   int newx = x + dx;
   x = newx;
   int newy = y + dy;
   y = newy;
}
```

Self Check 3.16

When you run the `BankAccountTester` program, how many objects of class `BankAccount` are constructed? How many objects of type `BankAccountTester`?

Answer: One `BankAccount` object, no `BankAccountTester` object. The purpose of the `BankAccountTester` class is merely to hold the `main` method.

Self Check 3.17

Why is the `BankAccountTester` class unnecessary in development environments that allow interactive testing, such as BlueJ?

Answer: In those environments, you can issue interactive commands to construct `BankAccount` objects, invoke methods, and display their return values.

Self Check 3.18

What do local variables and parameter variables have in common? In which essential aspect do they differ?

Answer: Variables of both categories belong to methods – they come alive when the method is called, and they die when the method exits. They differ in their initialization. Parameter variables are initialized with the call values; local variables must be explicitly initialized.

Self Check 3.19

Why was it necessary to introduce the local variable `change` in the `giveChange` method? That is, why didn't the method simply end with the statement

```
return payment - purchase;
```

Answer: After computing the change due, `payment` and `purchase` were set to zero. If the method returned `payment - purchase`, it would always return zero.

Self Check 3.20

How many implicit and explicit parameters does the `withdraw` method of the `BankAccount` class have, and what are their names and types?

Answer: One implicit parameter, called `this`, of type `BankAccount`, and one explicit parameter, called `amount`, of type `double`.

Self Check 3.21

In the `deposit` method, what is the meaning of `this.amount`? Or, if the expression has no meaning, why not?

Answer: It is not a legal expression. `this` is of type `BankAccount` and the `BankAccount` class has no variable named `amount`. s

Self Check 3.22

How many implicit and explicit parameters does the `main` method of the `BankAccountTester` class have, and what are they called?

Answer: No implicit parameter – the main method is not ivoked on any object – and one explicit parameter, called `args`.

Self Check 3.23

Which class needs to be modified to have the two cars positioned next to each other?

Answer: `CarComponent`

Self Check 3.24

Which class needs to be modified to have the car tires painted in black, and what modification do you need to make?

Answer: In the `draw` method of the `Car` class, call

```
g2.fill(frontTire);
g2.fill(rearTire);
```

Self Check 3.25

How do you make the cars twice as big?

Answer: Double all measurements in the `draw` method of the `Car` class.

Self Check 4.1

Which are the most commonly used number types in Java?

Answer: `int` and `double`

Self Check 4.2

Suppose you want to write a program that works with population data from various countries. Which Java data type should you use?

Answer: The world's most populous country, China, has about 1.2 x 10^9 inhabitants. Therefore, individual population counts could be held in an int. However, the world population is over 6×10^9. If you compute totals or averages of multiple countries, you can exceed the largest int value. Therefore, double is a better choice. You could also use long, but there is no benefit because the exact population of a country is not known at any point in time.

Self Check 4.3

Which of the following initializations are incorrect, and why?

a. `int dollars = 100.0;`

b. `double balance = 100;`

Answer: The first initialization is incorrect. The right hand side is a value of type `double`, and it is not legal to initialize an `int` variable with a `double` value. The second initialization is correct — an `int` value can always be converted to a `double`.

Self Check 4.4

What is the difference between the following two statements?

```
final double CM_PER_INCH = 2.54;
```

and

```
public static final double CM_PER_INCH = 2.54;
```

Answer: The first definition is used inside a method, the second inside a class.

Self Check 4.5

What is wrong with the following statement sequence?

```
double diameter = . . .;
double circumference = 3.14 * diameter;
```

Answer:

1. You should use a named constant, not the “magic number” 3.14.
2. 3.14 is not an accurate representation of π.

Self Check 4.6

What is the value of `n` after the following sequence of statements?

```
n--;
n++;
n--;
```

Answer: One less than it was before.

Self Check 4.7

What is the value of `1729 / 100`? Of `1729 % 100`?

Answer: `17` and `29`

Self Check 4.8

Why doesn't the following statement compute the average of `s1`, `s2`, and `s3`?

```
double average = s1 + s2 + s3 / 3; // Error
```

Answer: Only `s3` is divided by `3`. To get the correct result, use parentheses. Moreover, if `s1`, `s2`, and `s3` are integers, you must divide by `3.0` to avoid integer division:

```
(s1 + s2 + s3) / 3.0
```

Self Check 4.9

What is the value of `Math.sqrt(Math.pow(x, 2) + Math.pow(y, 2))` in mathematical notation?

Answer: $\sqrt{x^2 + y^2}$

Self Check 4.10

When does the cast `(long) x` yield a different result from the call `Math.round(x)`?

Answer: When the fractional part of `x` is ≥ 0.5

Self Check 4.11

How do you round the `double` value `x` to the nearest `int` value, assuming that you know that it is less than 2 · 109?

Answer: By using a cast: `(int) Math.round(x)`

Self Check 4.12

Why can't you call `x.pow(y)` to compute x^y?

Answer: `x` is a number, not an object, and you cannot invoke methods on numbers.

Self Check 4.13

Is the call `System.out.println(4)` a static method call?

Answer: No – the `println` method is called on the object `System.out`.

Self Check 4.14

Assuming the `String` variable `s` holds the value `"Agent"`, what is the effect of the assignment `s = s + s.length()`?

Answer: `s` is set to the string `Agent5`

Self Check 4.15

Assuming the String variable `river` holds the value `"Mississippi "`, what is the value of `river.substring(1, 2)`? Of `river.substring(2, river.length() - 3)`?

Answer: The strings `"i"` and `"ssissi"`

Self Check 4.16

Why can't input be read directly from `System.in`?

Answer: The class only has a method to read a single byte. It would be very tedious to form characters, strings, and numbers from those bytes.

Self Check 4.17

Suppose `in` is a `Scanner` object that reads from `System.in`, and your program calls

```
String name = in.next();
```

What is the value of name if the user enters `John Q. Public`?

Answer: The value is `"John"`. The `next` method reads the next *word*.

Self Check 5.1

Why did we use the condition `amount <= balance` and not `amount < balance` in the example for the `if/else` statement?

Answer: If the withdrawal amount equals the balance, the result should be a zero balance and no penalty.

Self Check 5.2

What is logically wrong with the statement

```
if (amount <= balance)
   newBalance = balance - amount;
   balance = newBalance;
```

and how do you fix it?

Answer: Only the first assignment statement is part of the `if` statement. Use braces to group both assignment statements into a block statement.

Self Check 5.3

What is the value of `s.length()` if `s` is

a. the empty string `""`?
b. the string `" "` containing a space?
c. `null`?

Answer: (a) 0; (b) 1; (c) an exception occurs.

Self Check 5.4

Which of the following comparisons are syntactically incorrect? Which of them are syntactically correct, but logically questionable?

```
String a = "1";
String b = "one";
double x = 1;
double y = 3 * (1.0 / 3);
```

a. a == "1"
b. a == null
c. a.equals("")
d. a == b
e. a == x
f. x == y
g. x - y == null
h. x.equals(y)

Answer: Syntactically incorrect: e, g, h. Logically questionable: a, d, f.

Self Check 5.5

The `if/else/else` statement for the earthquake strength first tested for higher values, then descended to lower values. Can you reverse that order?

Answer: Yes, if you also reverse the comparisons:

```
if (richter < 3.5)
   r = "Generally not felt by people";
else if (richter < 4.5)
   r =  "Felt by many people, no destruction";
else if (richter < 6.0)
   r =  "Damage to poorly constructed buildings";
...
```

Self Check 5.6

Some people object to higher tax rates for higher incomes, claiming that you might end up with less money after taxes when you get a raise for working hard. What is the flaw in this argument?

Answer: The higher tax rate is only applied on the income in the higher bracket. Suppose you are single and make $31,900. Should you try to get a $200 raise? Absolutely: you get to keep 90 percent of the first $100 and 75 percent of the next $100.

Self Check 5.7

When does the statement

```
system.out.println (x > 0 || x < 0);
```

print `false`?

Answer: When `x` is zero.

Self Check 5.8

Rewrite the following expression, avoiding the comparison with `false`:

```
if (character.isDigit(ch) == false) ...
```

Answer: `if (!Character.isDigit(ch)) ...`

Self Check 5.9

How many test cases do you need to cover all branches of the `getDescription` method of the `Earthquake` class?

Answer: 7.

Self Check 5.10

Give a boundary test case for the `EarthquakeRunner` program. What output do you expect?

Answer: An input of 0 should yield an output of `"Generally not felt by people"`. (If the output is `"Negative numbers are not allowed"`, there is an error in the program.)