

Chapter 3 Exercise Solutions

- EX 3.1.** Write a statement that prints the number of characters in a `String` object called `overview`.

```
System.out.println(overview.length());
```

- EX 3.2.** Write a statement that prints the 8th character of a `String` object called `introduction`.

```
System.out.println(introduction.charAt(7));
```

- EX 3.3.** Declare a `String` variable named `str` and initialize it to contain the same characters as a `String` object called `name`, except in all uppercase characters.

```
String str = name.toUpperCase();
```

- EX 3.4.** Write a declaration for a `String` variable called `change` and initialize it to the characters stored in another `String` object called `original` with all 'e' characters changed to 'j'.

```
String change = original.replace('e', 'j');
```

- EX 3.5.** What output is produced by the following code fragment?

```
String m1, m2, m3;  
m1 = "Quest for the Holy Grail";  
m2 = m1.toLowerCase();  
m3 = m1 + " " + m2;  
System.out.println(m3.replace('h', 'z'));
```

The output produced is:

```
Quest for tze Holy Grail quest for tze zoly grail
```

The original string is concatenated with a lowercase version of itself, then all lowercase 'h' characters are replaced with 'z'.

- EX 3.6.** What is the effect of the following import statement?

```
import java.awt.*;
```

*This statement allows the program in which it is written to access all classes (because of the wildcard *) in the package `java.awt` without any further reference to the package name.*

- EX 3.7.** Assuming that a `Random` object has been created called `generator`, what is the range of the result of each of the following expressions?

- a. `generator.nextInt(20)`

0 to 19, inclusive

- b. `generator.nextInt(8) + 1`

1 to 8, inclusive

- c. `generator.nextInt(12) + 2`

2 to 13, inclusive

- d. `generator.nextInt(35) + 10`

10 to 44, inclusive

- e. `generator.nextInt(100) - 50`

-50 to 49, inclusive

- EX 3.8.** Write code to declare and instantiate an object of the `Random` class (call the object reference variable `rand`). Then write a list of expressions using the `nextInt` method that generate random numbers in the following specified ranges, including the endpoints. Use the version of the `nextInt` method that accepts a single integer parameter.

```
Random rand = new Random();
```

- a. 0 to 10

```
rand.nextInt(11)
```

- b. 0 to 400

```
rand.nextInt(401)
```

- c. 1 to 10

```
rand.nextInt(10) + 1
```

- d. 1 to 400

```
rand.nextInt(400) + 1
```

- e. 25 to 50

```
rand.nextInt(26) + 25
```

- f. -10 to 15

```
rand.nextInt(26) - 10
```

- EX 3.9.** Write an assignment statement that computes the square root of the sum of `num1` and `num2` and assigns the result to `num3`.

```
num3 = Math.sqrt(num1 + num2);
```

- EX 3.10.** Write a single statement that computes and prints the absolute value of `total`.

```
System.out.println(Math.abs(total));
```

- EX 3.11.** Write code statements to create a `DecimalFormat` object that will round a formatted value to 4 decimal places. Then write a statement that uses that object to print the value of `result`, properly formatted.

```
DecimalFormat fmt = new DecimalFormat("0.####");  
System.out.println(fmt.format(result));
```

- EX 3.12.** Write code statements that prompt for and read a double value from the user, and then print the result of raising that value to the fourth power. Output the results to 3 decimal places.

```
Scanner scan = new Scanner(System.in);  
DecimalFormat fmt = new DecimalFormat("0.###");  
System.out.println("Enter a value: ");  
double num = scan.nextDouble();  
System.out.println(fmt.format(Math.pow(num, 4)));
```

EX 3.13. Write a declaration for an enumerated type that represents the days of the week.

```
enum Days {sunday, monday, tuesday, wednesday, thursday,  
friday, saturday}
```

EX 3.14. Compare and contrast a traditional coordinate system and the coordinate system used by Java graphical components.

A traditional coordinate system has the origin in the lower-left corner, with x increasing to the right and y increasing upward. The coordinate system used by Java has the origin in the upper-left corner with x increasing to the right and y increasing downward.

EX 3.15. Write a declaration for each of the following:

a. **A line that extends from point (60, 100) to point (30, 90)**

```
Line line = new Line(60, 100, 30, 90);
```

b. **A rectangle that is 20 pixels wide, 100 pixels high, and has its upper-left corner at point (10, 10).**

```
Rectangle rect = new Rectangle(10, 10, 20, 100);
```

c. **A circle that is centered at point (50, 75) and has a radius of 30.**

```
Circle circle = new Circle(50, 75, 30);
```

d. **An ellipse that is centered at point (150, 180) and is 100 pixels wide and 80 pixels high.**

```
Ellipse ellipse = new Ellipse(150, 180, 50, 40);
```

EX 3.16. Are the following lines horizontal, vertical, or neither?

a. **new Line(30, 90, 30, 10)**

vertical

b. **new Line(85, 70, 70, 85)**

neither

c. **new Line(20, 40, 150, 40)**

horizontal

EX 3.17. Is each of the following ellipses wider than it is tall or taller than it is wide?

a. **new Ellipse(300, 100, 50, 10)**

wider than it is tall

b. **new Ellipse (100, 200, 20, 40)**

taller than it is wide

c. **new Ellipse (150, 220, 60, 30)**

wider than it is tall

EX 3.18. How do you make a shape that has no fill color, so tht you can see the elements behind it?

Set its fill color to null:

```
circle.setFill(null);
```

EX 3.19. Write a line of code that rotates an ellipse called myEllipse 45 degrees clockwise.

```
myEllipse.setRotate(45);
```