Lesson 9….The *if* Statement

Now that we understand *boolean* quantities, let’s put them to use in an *if* statement, one of Java’s most useful “decision-making” commands. Consider the following code:

Example 1:

```
//Get a grade from the keyboard
Scanner kbReader = new Scanner(System.in);
System.out.print(“What is your grade? ”);
int myGrade = kbReader.nextInt( );

//Make a decision based on the value of the grade you entered
if (myGrade >= 70)
{
        //Execute code here if the test above is true
        System.out.println(“Congratulations, you passed.”);
}
else
{
        //Execute code here if the test above is false
        System.out.println(“Better luck next time.”);
}
```

Leave off the *else*:

We do not necessarily always need the ***else*** part. Consider the following code without an *else*.

Example 2:

```
Scanner kbReader = new Scanner(System.in);
System.out.print(“What state do you live in? ”);
String state = kbReader.nextLine( );  //get state from keyboard

System.out.print(“What is the price? ”);
double purchasePrice = kbReader.nextDouble( );  //get price from keyboard

double tax = 0;
if (  (state = = “Texas”) || (state = = “Tx”)  )
{
        //Execute code here if test above is true
        tax = purchasePrice *.08;  //8% tax
}
double totalPrice = purchasePrice + tax;
System.out.println(“The total price is ” + totalPrice + “.”);
```

It won’t work!

There is just one difficulty with the above code in Example 2. It won’t work! The problem is with how we are trying to compare two *String*s. It **cannot** be as follows:

```
state = = “Texas”
```

Rather, we must do it this way:

```
state.equals(“Texas”)
```

A good way to cover all the bases in the event someone mixes upper and lower case on the input is as follows:

```
        ( state.equalsIgnoreCase(“Texas”) | | state.equalsIgnoreCase(“Tx”) )
```

What? No braces?

Braces are not needed if only **one line of code** is in the *if* or *else* parts. Likewise, the absence of braces implies only one line of code in *if* or *else* parts.

Example 3:

```
int groovyDude = 37;
if (groovyDude = =37)
        groovyDude++;  //this line is executed if test is true
System.out.println(groovyDude);  //38
```

Example 4:

```
int groovyDude = 105;
if (groovyDude = =37)
        groovyDude++;  //this line is not executed if test is false
System.out.println(groovyDude);  //105
```

The *else if*:

Multiple *if*s can be used in the same structure using *else if.*

Example 5:

```
//Get a grade from the keyboard
Scanner kbReader = new Scanner(System.in);
System.out.println(“What is your grade? ”);
int theGrade = kbReader.nextInt( );

if (theGrade>=90)
{
        System.out.println(“You made an A.”);
}
else if (theGrade>=80)
{
        System.out.println(“You made a B.”);
}
else if (theGrade>=70)
{
        System.out.println(“You made a C.”);
}
else if (theGrade>=60)
{
        System.out.println(“You made a D.”);
}
else
{
        System.out.println(“Sorry, you failed.”);
}
```

Exercise on Lesson 9

Use the following code for problems 1 – 10 and give the value of *true_false* for each:

```
int i = 10, j = 3;
boolean true_false;
```

1. true_false = (j > i);

2. true_false = (i > j);

3. true_false = (i= = j);

4. true_false = ((j <= i) | | (j >= i));

5. true_false = ((i > j) && (j = = 0));

6. true_false = ((j < 50) | | (j != 33));

7. true_false = (!(j >= 0) | | (i <= 50));

8. true_false = (!(! (!true)));

9. true_false = (5 < = 5);

10. true_false = (j != i);

11. Write a statement that will store a true in *boolean b* if the value in the variable *m* is 44 or less.

12. Write a statement that will store a false in *boolean b* if the value in *r* is greater than 17.

13. What is returned by the following expression? (Recall that the precedence order of logical operators is !, &&, and finally | |.)

```
        !( (2>3) | | (5= =5) && (7>1) && (4<15) | | (35<=36) && (89!=34) )
```

In problem 14 – 16 what is the output?

14.

```
String s1 = “school BUS”;
if ( s1.equals(“school bus”) )
        System.out.println(“Equal”);
else
        System.out.println(“Not equal”);
```

15.

```
String s1 = “school BUS”;
if ( s1.equalsIgnoreCase(“school bus”) )
System.out.println(“Equal”);
else
System.out.println(“Not equal”);
```

16.
```
int j = 19, m = 200;
if (j= =18)
        m++;
        j++;
System.out.println(m);
System.out.println(j);
```

17. Write a statement that will store a *false* in *boolean b* if the value in *g* is not equal to 34.

18. Write a statement that will store a *true* in *boolean b* if integer *k* is even, *false* if it is odd.

19. Write a program that inputs a *String* from the keyboard after the prompt, "Enter your password". If it's entered exactly as "XRay", printout "Password entered successfully."; otherwise, have it printout "Incorrect password."

20. What is output by the following "nested *if*s" code?

```
int k = 79;
if (k>50)
{
        if (k<60)
        {System.out.println("One");}
        else
        { System.out.println("Two");}
}
else
{
        if (k>30)
                System.out.println("Three");
        else
                System.out.println("Four");
}
```

Project… Even or Odd?

Create a new project called *EvenOrOdd* containing a class called *Tester*. In the *main* method of *Tester* print a prompt that says, "Enter an integer:" Input the user's response from the keyboard, test the integer to see if it is even or odd (use the modulus operator % to do this), and then print the result as shown below (several runs are shown).

```
Enter an integer: 28
The integer 28 is even.

Enter an integer: 2049
The integer 2049 is odd.

Enter an integer: -236
The integer -236 is even.
```