Figure 4.0 Variables in Java.

There are two places where you can declare variables:

- in the body of a method
  - exist only while the method is "being run"
  - these are called *local* variables
- in a class, outside any method
  - exist for as long as the object that owns them exists
  - often this is the entire time the program is running

An example of this second kind of variable:

```
class WashingMachine {
    private String contents;
}
public class TestWash {
    public static void main (String[] args) {
        WashingMachine myWashingMachine = new WashingMachine ();
    }
}
```

The String variable **contents** is created along with the **myWashingMachine** object. This second kind of variable is often called

- an *instance* variable, or
- a *field* or *data member* of the object.

contents is declared **private**, so it can't be seen or used outside of the **WashingMachine** class. However any method *inside* the class *can* use it. For example,

```
class WashingMachine {
    private String contents;

    // A new method to print the contents of the washing machine.
    public void printContents () {
        System.out.println("Inside you will find " + contents);
    }
}
```

If the value of **contents** was "sock", this new method would print "Inside you will find sock".