Figure 2.0 Second version of the washing-machine program.

```
/**
 * Second version of the washing-machine program.
 */
public class TestWash {
// Notice that TestWash is another class; TestWash will create WashingMachine objects.

    public static void main (String[] args) {
    // Why is 'static' used here? Notice that methods of the WashingMachine class don't have
    // 'static' in their headers.

        WashingMachine myWashingMachine;
        // We declare a variable, ready to hold a reference to a WashingMachine object.

        myWashingMachine = new WashingMachine ();
        // We create a new WashingMachine object and save the reference to it in our variable.

        /**
        * Notice that the statements here are executed in order. The first statement is a declaration
        * that sets aside a space where an object reference can be stored. The second statement creates
        * an object with the 'new' operator and stores the reference to the new object in the variable
        * declared in the first statement. Because the second statement assigns a value to a variable it
        * is called an assignment statement.
        */

        System.out.println ("Your washing machine will now speak to you.");

        myWashingMachine.printIdentity ();
        // A call of myWashingMachine's printIdentity method.
    }
    // matches the second '{'
}
// matches the first '{'
```