```
1    // Example 5.2 joi/examples/OverridingDemo.java
2    //
3    //
4    // Copyright 2003 Bill Campbell and Ethan Bolker
5    //
6    // Small program to illustrate overriding and toString()
7    //
8    // Here's what the output looks like:
9    //
10   // %> java Overriding Jessica benjamin
11   // Terminal t = new Terminal();
12   // nobj = new NamedObject( args[0] );
13   // nobj.toString();
14   // nobj:                                Jessica
15   // nobj.toStringfromObject():NamedObject@206fdf64
16   // nobj = new NamedObject( args[1] );
17   // nobj.toString():                     benjamin
18   // nobj:                                benjamin
19   // nobj.toStringfromObject():NamedObject@2103df64
20   //
21   // toString():
22   // t:                                   Terminal@201bdf64
23   //                                      Terminal@201bdf64
24
25
26   public class OverridingDemo
27   {
28
29       public static void main( String[] args )
30       {
31
32           Terminal t = new Terminal();
33           NamedObject nobj;
34
35           t.println("Terminal t = new Terminal();");
36           t.println("nobj = new NamedObject( args[0] );");
37           nobj = new NamedObject( args[0] );
38           t.println( "nobj.toString();" );
39           t.println( "nobj:" +
40                      nobj.toStringfromObject():       " +
41                      nobj.toStringfromObject());
42           t.println( "nobj:         " + nobj );
43           t.println( "nobj.toStringfromObject():" +
44                      nobj.toStringfromObject());
45           nobj = new NamedObject( args[1] );
46           t.println( "nobj.toString():  " + nobj.toString() );
47           t.println( "nobj:         " + nobj );
48           t.println( "nobj.toStringfromObject():" +
49                      nobj.toStringfromObject());
50
51           t.println( "\ntoString():" );
52           t.println( "t:            " + t.toString() );
53           t.println( "t:            " + t );
54
55       }
56   }
```

```
57
58   class NamedObject // extends Object, by default
59   {
60       private String name;
61
62       // constructor does the obvious thing
63       public NamedObject( String name )
64       {
65           this.name = name;
66       }
67
68       // override toString in class Object
69       public String toString()
70       {
71           return name;
72       }
73
74       // access to the overridden method using super
75       public String toStringfromObject()
76       {
77           return super.toString();
78       }
79   }
80
81
82   }
```