```
69 package bikeshop;
70
71     public class Bike {
72
73     // create the BikeList
74     protected static Bike[] bikeList = new Bike[5];[a]
75     //    set up member variables
76     protected int deposit = 0;
77     protected int rate = 0;
78     protected int bikeNumber = 0;
79
80     /* This block is run when the class is loaded and sets up our bike store.
81      * It arbitrarily populates the attributes: deposit, rate and bikeNumber */
82     static{
83         int j = 0;
84         for(int i=10;i<15;i++){
85             Bike b = new Bike(i, i, (j*100));
86             bikeList[j] = b;
87             j++;
88         }
89     }
90
91     public Bike(int dep, int rat, int num){
92         // set the member variables
93         deposit = dep;
94         rate = rat;
95         bikeNumber = num;
96     }
97
98     public int getDeposit( ){
99         return deposit;
100    }
101
102    public int getRate( ){
103        return rate;
104    }
105
106    public int getBikeNumber( ){
107        return bikeNumber;
108    }
109
110    public static Bike findBikeByNumber(int bikeNum){
111        int numberOfBikes = bikeList.length;
112
113        // iterate over the list of bikes
114        for(int i=0;i<numberOfBikes;i++){
115            // if we find the bike with the correct number...
116            if(bikeList[i].getBikeNumber( ) == bikeNum){
117                // tell user that we've found it
118                System.out.println("Bike with number '" + bikeNum + "' found" + "\n");
119                // and return it to the UI
120                return bikeList[i];
```

Figure 11.8 Code for the Bike class