```
58              // Create a customer and associated hire and payment
59              customer = new Customer(name, postcode, telephone);
60              payment = new Payment(customer);
61              hire = new Hire(new Date( ), numberOfDays, chosenBike, customer);
62          }
63
64          public void calculateTotalPayment( ){
65              // get the total payment from the payment object
66              payment.calculateTotalPayment(hire);
67          }
68      }
```

Bike class

```
69      package bikeshop;
70
71      public class Bike {
72
73      // create the BikeList
74      protected static Bike[] bikeList = new Bike[5];
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;
```

Figure 11.10 The code listing for the Wheels system (continued)