```
        /* This little program will run through the methods on IssueBikeUI
         * calling each in turn, like a user with a front end would do. */

        // First, create the UI
        IssueBikeUI ui = new IssueBikeUI( );

        // 1. Show details for chosen bike
        ui.showBikeDetails(100);

        // 2. Calculate cost of hiring this bike for 5 days
        ui.calculateCost(5);

        // 3. Create new customer, payment and hire
        ui.createCustomer("Les Hargreaves", "PW2 6TR", 01462501339);

        // 4. Calculate the total cost
        ui.calculateTotalPayment( );
    }
}
```

IssueBikeUI class

```
/*IssueBikeUI Class*/

package bikeshop;

import java.util.Date;

public class IssueBikeUI {

    // Set up the member (or class-level variables)
    private Bike chosenBike = null;
    private Customer customer = null;
    private Payment payment = null;
    private Hire hire = null;
    private int numberOfDays = 0;

    public void showBikeDetails(int bikeNum){
        // Find the bike by its number
        chosenBike = Bike.findBikeByNumber(bikeNum);
        if(chosenBike !=null){
            // then ask it for its details
            chosenBike.showDetails( );
        }
    }
    public void calculateCost(int numDays){
        // set the member variable so it can be used later
        numberOfDays = numDays;
        // then ask the bike for the cost
        chosenBike.calculateCost(numDays);
    }

    public void createCustomer(String name, String postcode, int telephone){
```

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