We also create a class called SwimData that maintains a Vector of the Swimmers we read in from the database.

```
public class TimeSwimData extends SwimData
implements Cloneable , Serializable {
    protected Vector swimmers;

    public TimeSwimData(String filename) {
        String s = "";
        swimmers    = new Vector();
        InputFile f = new InputFile(filename);
        s= f.readLine();
        while (s != null) {
            swimmers.addElement(new Swimmer(s));
            s= f.readLine();
        }
        f.close();
    }
```

We also provide a *getSwimmer* method in SwimData and *getName*, *getAge*, and *getTime* methods in the Swimmer class. Once we've read the data into SwimInfo, we can display it in a list box.

Then, when the user clicks on the Clone button, we'll clone this class and sort the data differently in the new class. Again, we clone the data because creating a new class instance would be much slower, and we want to keep the data in both forms.

```
public class SwimInfo extends Frame
    implements ActionListener {
    SwimData sdata;
    List swList, cloneList;
    Button Clone, Refresh, Quit;
    Swimmer sw;

    //_______________________________
    private void cloneAndLoad() {
        SwimData sxdata = null;
        try{
            sxdata = (SwimData)sdata.cloneMe();
            sxdata.sort();
        }
        catch(CloneNotSupportedException e){}

        cloneList.removeAll();
        //now print out sorted values from clone
        for (int i = 0; i < sxdata.size(); i++) {
            sw = sxdata.getSwimmer(i);
            cloneList.add(sw.getName()+" "+sw.getTime());
        }
    }
```