- Use different line-breaking strategies to display text data.
- Plot the same data in different formats: line graph, bar chart, or pie chart.

In each case, the client program could tell a driver module (Context) which of these strategies to use and then ask it to carry out the operation.

The idea behind Strategy is to encapsulate the various strategies in a single module and provide a simple interface to allow a choice between them. Each strategy must have the same programming interface, but they do not need to be members of the same class hierarchy. However, they do have to implement the same programming interface.

Sample Code

Let's consider a simplified graphing program that can present data as a line graph or a bar chart. We start with an abstract PlotStrategy class and derive the two plotting classes from it, as illustrated in Figure 24.1.

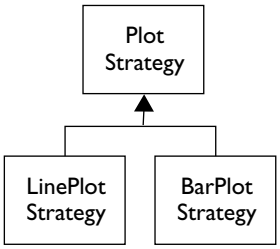

Figure 24.1 Two instances of a PlotStrategy Class.

Each plot will appear in its own frame, so the base PlotStrategy class will be derived from JFrame.

```
public abstract class PlotStrategy extends JFrame {
    protected float[]    x, y;
    protected float      minX, minY, maxX, maxY;
    protected int        width, height;
    protected Color      color;

    public PlotStrategy(String title) {
        super(title);
        width = 300;
        height = 200;
        color = Color.black;
    addWindowListener(new WindAp(this));
    }
    //--------------
    public abstract void plot(float xp[], float yp[]);
```