[0830] Interfaces as Types

Like classes, Interfaces are data types. You can use the name of an interface anywhere you would use a class name. However, unlike classes, variables of type "interface" can contain objects from any class that implements that interface. We saw that happen in the "birds" area of the ISquawkable code presented on the previous page.

Here's another example:

```
public interface IRollable {
    void Roll();
}

public class Die : IRollable {
    public int Faces { get; private set; }
    public int Value { get; private set; }

    private static Random rng = new Random();

    public Die(int faces) {
        this.Faces = faces;
        this.Value = 1;
    }

    public void Roll() {
        Value = rng.Next(Faces) + 1;
    }

    public override string ToString() {
        return "[" + Value + "]";
    }
}

public class Cup : IRollable {
    Die[] dice;

    public Cup(int numDies) {
        dice = new Die[numDies];
        for (int i = 0; i < numDies; i++)
            dice[i] = new Die(6);
    }

    public void Roll() {
        foreach (Die d in dice)
            d.Roll();
    }

    public override string ToString() {
        string s = "[ ";
        foreach (Die d in dice)
            s += d.ToString();
        s += " ]";
        return s;
    }
}
```