Open for Extension, Closed for Modification

Simply put, **classes should be open for extension, but closed for modification. In doing so, we stop ourselves from modifying existing code and causing potential new bugs** in an otherwise happy application.

Of course, the **one exception to the rule is when fixing bugs in existing code.**

Imagine we've implemented a *Guitar* class.

```
public class Guitar {
private String make;
private String model;
private int volume;
//Constructors, getters & setters
}
```

We launch the application, and everyone loves it. However, after a few months, we realize the *Guitar* is a little bit boring and could do with an awesome flame pattern to make it look a bit more 'rock and roll'.

At this point, it might be tempting to just open up the *Guitar* class and add a flame pattern – but who knows what errors that might throw up in our application.

Instead, let's **stick to the open-closed principle and simply extend our *Guitar* class**:

```
public class SuperCoolGuitarWithFlames extends Guitar {
private String flameColor;
//constructor, getters + setters
}
```

By extending the *Guitar* class we can be sure that our existing application won't be affected.