

In *Effective Java*, Joshua Bloch recommends creating an abstract method which would be implemented for each constant. For example:

```
enum Color {  
    RED { public String getName() {return "Red";} },  
    GREEN { public String getName() {return "Green";} },  
    BLUE { public String getName() {return "Blue";} };  
    public abstract String getName();  
}
```

This would function as a safe switch, forcing you to implement the method if you add a new constant.

EDIT: To clear up some confusion, here's the equivalent using a regular `switch`:

```
enum Color {  
    RED, GREEN, BLUE;  
    public String getName() {  
        switch(this) {  
            case RED: return "Red";  
            case GREEN: return "Green";  
            case BLUE: return "Blue";  
            default: return null;  
        }  
    }  
}
```