

## ¿Qué hace el código a simple vista?

El código permite agregar Items con nombres específicos y características de valor de calidad(quility) y vencimiento(sell\_in), dichos item al pasar de los días aumentan o disminuyen sus características según su tipo.

## ¿Te parece un mal código?

A lo largo de mi vida laboral he visto muchos códigos peores, pero si hubiese podido encontrar una mejor solución, ya que se debe siempre pensar en que las cosas van a crecer y los cambios y mejoras estarán a la orden del día.

- o Justifica por qué (con pequeños ejemplos de código si hace falta)
  - ¿Qué harías para mejorarlo? (especificando el orden de los pasos que seguirías)
    - o Pienso que la mejor solución es realizar Polimorfismo a cada posible variable que tienen de If o método así se tendría separada cada tipo de Item.

```
class Bird {  
    //...  
    double getSpeed() {  
        switch (type) {  
            case EUROPEAN:  
                return getBaseSpeed();  
            case AFRICAN:  
                return getBaseSpeed() - getLoadFactor();  
            case NORWEGIAN_BLUE:  
                return (isNailed) ? 0 : getBaseSpeed();  
        }  
        throw new RuntimeException("Should be unreachable");  
    }  
}
```

```
abstract class Bird {  
    //...  
    abstract double getSpeed();  
}  
  
class European extends Bird {  
    double getSpeed() {  
        return getBaseSpeed();  
    }  
}  
  
class African extends Bird {  
    double getSpeed() {  
        return getBaseSpeed() - getLoadFactor();  
    }  
}  
  
class NorwegianBlue extends Bird {  
    double getSpeed() {  
        return (isNailed) ? 0 : getBaseSpeed();  
    }  
}  
  
// Somewhere in client code  
speed = bird.getSpeed();
```