# Die Klasse GildedRose

Diese Klasse beinhaltet die Logik zur Bewertung von Items.

Die Bewertung aktualisiert über die Methode updateQuality() die übergebenen Items.

Ein Item kann in verschiedenen Ausprägungen vorliegen und verhält sich entsprechend unterschiedlich. Jedes Item besitzt neben einer Art und einem Namen 2 Eigenschaften: sellIn und quality. SellIn beschreibt hierbei die Dauer in Tagen bis zum Mindesthaltbarkeitsdatum. Quality beschreibt, wie viel das Item wert ist.

Jeden Abend findet eine Neubewertung statt und der sellIn wird reduziert.

Die unterschiedlichen Items verhalten sich wie folgt:

* Die Qualität wird niemals negativ.
* Die Qualität steigt nie über 50.
* Die Qualität normaler Items sinkt um eins pro Tag. Mit Erreichen der Mindesthaltbarkeit sinkt sie doppelt so schnell.
* Die Qualität von "Aged Brie" steigt anstatt zu sinken.
* "Sulfuras" ist ein konstantes Item ohne Mindesthaltbarkeit und mit gleichbleibendem Wert von 80.
* Ein Backstage Pass verhält sich ähnlich wie "Aged Brie". Allerdings steigt seine Qualität mit dem Näherrücken des Datums (in diesem Fall das Konzertdatum) schneller:
  + Ab 10 Tagen vor dem Konzert steigt die Qualität um 2 Punkte pro Tag.
  + Ab 5 Tagen sogar um 3 Punkte pro Tag.
  + Nach dem Konzert ist ein Backstage Pass nichts mehr wert.

Es soll eine neue Art von Items eingebaut werden; sogenannte "Conjured" Items. Diese verfallen doppelt so stark wie normale Items.

Wir empfehlen *dringend*, den vorhandenen Code vorher aufzuräumen!

Nähere Infos zu dieser Kata gibt es beispielsweise unter: https://github.com/emilybache/Refactoring-Katas/tree/master/GildedRose

Der Code:

**class** GildedRose {

Item[] items;

**public** **void** updateQuality() {

**for** (**int** i = 0; i < items.length; i++) {

**if** (items[i].name != "Aged Brie"

&& items[i].name != "Backstage passes to a TAFKAL80ETC concert") {

**if** (items[i].quality > 0) {

**if** (items[i].name != "Sulfuras, Hand of Ragnaros") {

items[i].quality = items[i].quality - 1;

}

}

} **else** {

**if** (items[i].quality < 50) {

items[i].quality = items[i].quality + 1;

**if** (items[i].name == "Backstage passes to a TAFKAL80ETC concert") {

**if** (items[i].sellIn < 11) {

**if** (items[i].quality < 50) {

items[i].quality = items[i].quality + 1;

}

}

**if** (items[i].sellIn < 6) {

**if** (items[i].quality < 50) {

items[i].quality = items[i].quality + 1;

}

}

}

}

}

**if** (items[i].name != "Sulfuras, Hand of Ragnaros") {

items[i].sellIn = items[i].sellIn - 1;

}

**if** (items[i].sellIn < 0) {

**if** (items[i].name != "Aged Brie") {

**if** (items[i].name != "Backstage passes to a TAFKAL80ETC concert") {

**if** (items[i].quality > 0) {

**if** (items[i].name != "Sulfuras, Hand of Ragnaros") {

items[i].quality = items[i].quality - 1;

}

}

} **else** {

items[i].quality = items[i].quality - items[i].quality;

}

} **else** {

**if** (items[i].quality < 50) {

items[i].quality = items[i].quality + 1;

}

}

}

}

}

}

# Gilded Rose Requirements Specification

Hi and welcome to team Gilded Rose. As you know, we are a small inn with a prime location in a prominent city ran by a friendly innkeeper named Allison. We also buy and sell only the finest goods.

Unfortunately, our goods are constantly degrading in quality as they approach their sell by date. We have a system in place that updates our inventory for us. It was developed by a no-nonsense type named Leeroy, who has moved on to new adventures.

Your task is to add the new feature to our system so that we can begin selling a new category of items.

First an introduction to our system:

* All items have a SellIn value which denotes the number of days we have to sell the item
* All items have a Quality value which denotes how valuable the item is
* At the end of each day our system lowers both values for every item

Pretty simple, right? Well, this is where it gets interesting:

* Once the sell by date has passed, Quality degrades twice as fast
* The Quality of an item is never negative
* "Aged Brie" actually increases in Quality the older it gets
* The Quality of an item is never more than 50
* "Sulfuras", being a legendary item, never has to be sold or decreases in Quality
* "Backstage passes", like aged brie, increases in Quality as its SellIn value approaches; Quality increases by 2 when there are 10 days or less and by 3 when there are 5 days or less but Quality drops to 0 after the concert
* Just for clarification, an item can never have its Quality increase above 50, however "Sulfuras" is a legendary item and as such its Quality is 80 and it never alters.

We have recently signed a supplier of conjured items. This requires an update to our system:

* "Conjured" items degrade in Quality twice as fast as normal items

Feel free to make any changes to the UpdateQuality method and add any new code as long as everything still works correctly. However, do not alter the Item class or Items property as those belong to the goblin in the corner who will insta-rage and one-shot you as he doesn't believe in shared code ownership (you can make the UpdateQuality method and Items property static if you like, we'll cover for you).