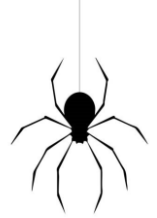# Refactoring Gilded Rose...

...on autopilot

What does autopilot mean?

Let the IDE do all the work, so:

- No typing!
- No thinking!

What's the use of this?

- Getting some practice

- Discovering new shortcuts and refactoring actions of the IDE

- Having fun!

- ...and, ok, a presentation 'for free'

The modus operandi:

1. Some code   ## 2. Highlighting POIs   ## 3. IDE Support   ## 4. Changed code

5. New stuff

ctrl+alt+shift+insert+cursor-up → knotted fingers

```
1    package com.gildedrose;
2
3    class GildedRose {
         36 usages
4        Item[] items;
5
6        public GildedRose(Item[] items) {
7            this.items = items;
8        }
9
10       public void updateQuality() {
11           for (int i = 0; i < items.length; i++) {
12               if (!items[i].name.equals("Aged Brie")
13                   && !items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
14                   if (items[i].quality > 0) {
15                       if (!items[i].name.equals("Sulfuras, Hand of Ragnaros")) {
16                           items[i].quality = items[i].quality - 1;
17                       }
18                   }
19               } else {
20                   if (items[i].quality < 50) {
21                       items[i].quality = items[i].quality + 1;
22
23                       if (items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
24                           if (items[i].sellIn < 11) {
25                               if (items[i].quality < 50) {
26                                   items[i].quality = items[i].quality
27                               }
28                           }
29
30                           if (items[i].sellIn < 6) {
31                               if (items[i].quality < 50) {
32                                   items[i].quality = items[i].quality
33                               }
34                           }
35                       }
36                   }
37               }
38
39               if (!items[i].name.equals("Sulfuras, Hand of Ragna
40                   items[i].sellIn = items[i].sellIn - 1;
41               }
42
43               if (items[i].sellIn < 0) {
44                   if (!items[i].name.equals("Aged Brie")) {
45                       if (!items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
46                           if (items[i].quality > 0) {
47                               if (!items[i].name.equals("Sulfuras, Hand of Ragnaros")) {
48                                   items[i].quality = items[i].quality - 1;
49                               }
50                           }
51                       } else {
52                           items[i].quality = items[i].quality - items[i].quality;
53                       }
54                   } else {
55                       if (items[i].quality < 50) {
56                           items[i].quality = items[i].quality + 1;
57                       }
58                   }
59               }
60           }
61       }
62   }
```

1. Unused code → no tests

2. ctrl+shift+t

```
8
9    public GildedRose(Item[] items) {
10       this.items        Choose Test for GildedRose (0 found)
11   }                     💡 Create New Test...
12
13   public void updateQuality() {
14       for (int i = 0; i < items.length; i++) {
```

GildedRose.java   GildedRoseTest.java

```
1    package com.gildedrose;
2
3    import static org.junit.jupiter.api.Assertions.*;
4
5    class GildedRoseTest {
6
7    }
```

ctrl+shift++t → Create New Test

```java
package com.gildedrose;

import static com.gildedrose.GildedRose.ITEM_NAME_AGED_BRIE;
import static com.gildedrose.GildedRose.ITEM_NAME_BACKSTAGEPASSES;
import static com.gildedrose.GildedRose.ITEM_NAME_SULFURAS;

import org.approvaltests.combinations.CombinationApprovals;
import org.junit.jupiter.api.Test;

class GildedRoseTest {

    @Test
    void updateQuality() {
        String[] names = new String[]{"", ITEM_NAME_AGED_BRIE, ITEM_NAME_BACKSTAGEPASSES, ITEM_NAME_SULFURAS};
        Integer[] sellIns = new Integer[]{-1, 0, 1, 5, 6, 10, 11, 49, 50, 60};
        Integer[] qualities = new Integer[]{-1, 0, 1, 49, 50, 51};
        CombinationApprovals.verifyAllCombinations(
            (name, sellIn, quality) -> {
                Item item = new Item(name, sellIn, quality);
                GildedRose testee = new GildedRose(new Item[] {item});
                testee.updateQuality();
                return item;
            },
                names,
                sellIns,
                qualities
        );
    }
}
```

```java
static final String ITEM_NAME_SULFURAS = "Sulfuras, Hand of Ragnaros";

Item[] items;

public GildedRose(Item[] items) {
    this.items = items;
}

public void updateQuality() {
    for (int i = 0; i < items.length; i++) {
        if (!items[i].name.equals(ITEM_NAME_AGED_BRIE)
                && !items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
            if (items[i].quality > 0) {
                if (!items[i].name.equals(ITEM_NAME_SULFURAS)) {
                    items[i].quality = items[i].quality - 1;
                }
            }
        } else {
            if (items[i].quality < 50) {
                items[i].quality = items[i].quality + 1;

                if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
                    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.equals(ITEM_NAME_SULFURAS)) {
            items[i].sellIn = items[i].sellIn - 1;
        }

        if (items[i].sellIn < 0) {
            if (!items[i].name.equals(ITEM_NAME_AGED_BRIE)) {
                if (!items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
                    if (items[i].quality > 0) {
                        if (!items[i].name.equals(ITEM_NAME_SULFURAS)) {
                            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;
                }
            }
        }
    }
}
```
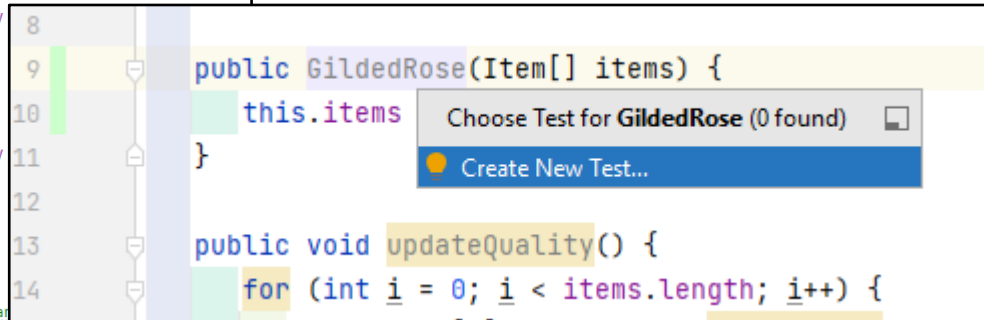
Coverage: GildedRoseTest

Element	Class, %	Method, %	Line, %
com.gildedrose	100% (2/2)	100% (4/4)	100% (33/33)
GildedRose	100% (1/1)	100% (2/2)	100% (28/28)
Item	100% (1/1)	100% (2/2)	100% (5/5)

Shortcut to a working test. You know how to do that...

Make sure you start off with 100%
(if there is no unreachable code...)

```
[Aged Brie, 49, 50] => Aged Brie, 48, 50
[Aged Brie, 49, 51] => Aged Brie, 48, 51
[Aged Brie, 50, -1] => Aged Brie, 49, 0
[Aged Brie, 50, 0] => Aged Brie, 49, 1
[Aged Brie, 50, 1] => Aged Brie, 49, 2
[Aged Brie, 50, 49] => Aged Brie, 49, 50
[Aged Brie, 50, 50] => Aged Brie, 49, 50
[Aged Brie, 50, 51] => Aged Brie, 49, 51
[Aged Brie, 60, -1] => Aged Brie, 59, 0
[Aged Brie, 60, 0] => Aged Brie, 59, 1
[Aged Brie, 60, 1] => Aged Brie, 59, 2
[Aged Brie, 60, 49] => Aged Brie, 59, 50
[Aged Brie, 60, 50] => Aged Brie, 59, 50
[Aged Brie, 60, 51] => Aged Brie, 59, 51
[Backstage passes to a TAFKAL80ETC concert, -1, -1] => Backstage passes to a TAFKAL80ETC concert, -2, 0
[Backstage passes to a TAFKAL80ETC concert, -1, 0] => Backstage passes to a TAFKAL80ETC concert, -2, 0
[Backstage passes to a TAFKAL80ETC concert, -1, 1] => Backstage passes to a TAFKAL80ETC concert, -2, 0
[Backstage passes to a TAFKAL80ETC concert, -1, 49] => Backstage passes to a TAFKAL80ETC concert, -2, 0
[Backstage passes to a TAFKAL80ETC concert, -1, 50] => Backstage passes to a TAFKAL80ETC concert, -2, 0
[Backstage passes to a TAFKAL80ETC concert, -1, 51] => Backstage passes to a TAFKAL80ETC concert, -2, 0
[Backstage passes to a TAFKAL80ETC concert, 0, -1] => Backstage passes to a TAFKAL80ETC concert, -1, 0
[Backstage passes to a TAFKAL80ETC concert, 0, 0] => Backstage passes to a TAFKAL80ETC concert, -1, 0
[Backstage passes to a TAFKAL80ETC concert, 0, 1] => Backstage passes to a TAFKAL80ETC concert, -1, 0
[Backstage passes to a TAFKAL80ETC concert, 0, 49] => Backstage passes to a TAFKAL80ETC concert, -1, 0
[Backstage passes to a TAFKAL80ETC concert, 0, 50] => Backstage passes to a TAFKAL80ETC concert, -1, 0
[Backstage passes to a TAFKAL80ETC concert, 0, 51] => Backstage passes to a TAFKAL80ETC concert, -1, 0
[Backstage passes to a TAFKAL80ETC concert, 1, -1] => Backstage passes to a TAFKAL80ETC concert, 0, 2
[Backstage passes to a TAFKAL80ETC concert, 1, 0] => Backstage passes to a TAFKAL80ETC concert, 0, 3
[Backstage passes to a TAFKAL80ETC concert, 1, 1] => Backstage passes to a TAFKAL80ETC concert, 0, 4
[Backstage passes to a TAFKAL80ETC concert, 1, 49] => Backstage passes to a TAFKAL80ETC concert 0, 51
[Backstage passes to a TAFKAL80ETC concert, 1, 50] => Backstage passes to a TAFKAL80ETC concert 0, 51
[Backstage passes to a TAFKAL80ETC concert, 1, 51] => Backstage passes to a TAFKAL80ETC concert 0, 51
[Backstage passes to a TAFKAL80ETC concert, 5, -1] => Backstage passes to a TAFKAL80ETC concert, 4, 2
[Backstage passes to a TAFKAL80ETC concert, 5, 0] => Backstage passes to a TAFKAL80ETC concert, 4, 3
[Backstage passes to a TAFKAL80ETC concert, 5, 1] => Backstage passes to a TAFKAL80ETC concert, 4, 4
[Backstage passes to a TAFKAL80ETC concert, 5, 49] => Backstage passes to a TAFKAL80ETC concert, 4, 50
[Backstage passes to a TAFKAL80ETC concert, 5, 50] => Backstage passes to a TAFKAL80ETC concert, 4, 50
[Backstage passes to a TAFKAL80ETC concert, 5, 51] => Backstage passes to a TAFKAL80ETC concert, 4, 51
[Backstage passes to a TAFKAL80ETC concert, 6, -1] => Backstage passes to a TAFKAL80ETC concert, 5, 1
[Backstage passes to a TAFKAL80ETC concert, 6, 0] => Backstage passes to a TAFKAL80ETC concert, 5, 2
[Backstage passes to a TAFKAL80ETC concert, 6, 1] => Backstage passes to a TAFKAL80ETC concert, 5, 3
[Backstage passes to a TAFKAL80ETC concert, 6, 49] => Backstage passes to a TAFKAL80ETC concert, 5, 50
[Backstage passes to a TAFKAL80ETC concert, 6, 50] => Backstage passes to a TAFKAL80ETC concert, 5, 50
[Backstage passes to a TAFKAL80ETC concert, 6, 51] => Backstage passes to a TAFKAL80ETC concert, 5, 51
[Backstage passes to a TAFKAL80ETC concert, 10, -1] => Backstage passes to a TAFKAL80ETC concert, 9, 1
[Backstage passes to a TAFKAL80ETC concert, 10, 0] => Backstage passes to a TAFKAL80ETC concert, 9, 2
[Backstage passes to a TAFKAL80ETC concert, 10, 1] => Backstage passes to a TAFKAL80ETC concert, 9, 3
[Backstage passes to a TAFKAL80ETC concert, 10, 49] => Backstage passes to a TAFKAL80ETC concert, 9, 50
[Backstage passes to a TAFKAL80ETC concert, 10, 50] => Backstage passes to a TAFKAL80ETC concert, 9, 50
[Backstage passes to a TAFKAL80ETC concert, 10, 51] => Backstage passes to a TAFKAL80ETC concert, 9, 51
[Backstage passes to a TAFKAL80ETC concert, 11, -1] => Backstage passes to a TAFKAL80ETC concert, 10, 0
[Backstage passes to a TAFKAL80ETC concert, 11, 0] => Backstage passes to a TAFKAL80ETC concert, 10, 1
```

```
10:43:55 PIT >> INFO : Verbose logging is disabled. If you encounter a problem, please enable it before reporting an issue.
10:43:56 PIT >> INFO : Incremental analysis reduced number of mutations by 0
10:43:56 PIT >> INFO : Created  2 mutation test units in pre scan
10:43:56 PIT >> INFO : Sending 3 test classes to minion
10:43:56 PIT >> INFO : Sent tests to minion
/10:43:57 PIT >> INFO : Calculated coverage in 1 seconds.
10:43:57 PIT >> INFO : Incremental analysis reduced number of mutations by 0
10:43:57 PIT >> INFO : Created  2 mutation test units
-================================================================
10:44:01 PIT >> INFO : Completed in 6 seconds
- Mutators
================================================================

> org.pitest.mutationtest.engine.gregor.mutators.ConditionalsBoundaryMutator
>> Generated 10 Killed 10 (100%)
> KILLED 10 SURVIVED 0 TIMED_OUT 0 NON_VIABLE 0
> MEMORY_ERROR 0 NOT_STARTED 0 STARTED 0 RUN_ERROR 0
> NO_COVERAGE 0
----------------------------------------------------------------
> org.pitest.mutationtest.engine.gregor.mutators.MathMutator
>> Generated 7 Killed 7 (100%)
> KILLED 7 SURVIVED 0 TIMED_OUT 0 NON_VIABLE 0
> MEMORY_ERROR 0 NOT_STARTED 0 STARTED 0 RUN_ERROR 0
> NO_COVERAGE 0
----------------------------------------------------------------
> org.pitest.mutationtest.engine.gregor.mutators.returns.EmptyObjectReturnValsMutator
>> Generated 1 Killed 1 (100%)
> KILLED 1 SURVIVED 0 TIMED_OUT 0 NON_VIABLE 0
> MEMORY_ERROR 0 NOT_STARTED 0 STARTED 0 RUN_ERROR 0
> NO_COVERAGE 0
----------------------------------------------------------------
> org.pitest.mutationtest.engine.gregor.mutators.NegateConditionalsMutator
>> Generated 18 Killed 18 (100%)
> KILLED 18 SURVIVED 0 TIMED_OUT 0 NON_VIABLE 0
> MEMORY_ERROR 0 NOT_STARTED 0 STARTED 0 RUN_ERROR 0
> NO_COVERAGE 0
----------------------------------------------------------------
================================================================
- Timings
================================================================
> pre-scan for mutations : < 1 second
> scan classpath : < 1 second
> coverage and dependency analysis : 1 seconds
> build mutation tests : < 1 second
> run mutation analysis : 4 seconds
----------------------------------------------------------------
> Total  : 5 seconds
----------------------------------------------------------------
================================================================
- Statistics
================================================================
>> Line Coverage: 36/36 (100%)
>> Generated 36 mutations Killed 36 (100%)
>> Mutations with no coverage 0. Test strength 100%
>> Ran 36 tests (1 tests per mutation)
Enhanced functionality available at https://www.arcmutate.com/

Process finished with exit code 0
Open report in browser
```

Shortcut to characterization test (using ApprovalTests)
Plus mutation testing (using Pitest)

Removing clutter

1. Mouse hover → IDE hints at duplication

```
10    public void updateQuality() {
11        for (int i = 0; i < items.length; i++) {
12            if (!items[i].name.equals("Aged Brie")
13                && !items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
14                if (items[i].quality > 0) {
15                    if (!items[i].name.equals("Sulfuras, Hand of
16                        items[i].quality = items[i].quality - 1;
17                    }
18                }
19            } else {
20                if (items[i].quality < 50) {
21                    items[i].quality = items[i].quality + 1;
```

Define a constant instead of duplicating this literal "Backstage passes to a TAFKAL80ETC concert" 3 times.

SonarLint: Show issue locations   Alt+Umschalt+Eingabe   More actions...   Alt+Eingabe

```
9
      1 usage
10    public void updateQuality() {
11        for (int i = 0; i < items.length; i++) {
12            if (!items[i].name.equals("Aged Brie")
13                && !items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
14                if (items[i].quality > 0) {
15                    if (!items[i].name.equals("Sulfuras, Hand of Ragnaros")) {
16                        items[i].quality = items[i].quality - 1;
17                    }
18                }
19            } else {
20                if (items[i].quality < 50) {
21                    items[i].quality = items[i].quality + 1;
22                    if (items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
23                        if (items[i].sellIn < 11) {
24                            if (items[i].quality < 50) {
25                                items[i].quality = items[i].quality + 1;
26                            }
27
```

2. ctrl+alt+c → extract constant

```
7     public GildedRose(Item[] items) {
8         this.items = items;
9     public static final String ITEM_NAME_AGED_BRIE = "Aged Brie";
10    ☑ Replace all occurrences
11    ☐ Move to another class
12        for (int i = 0; i < items.length; i++) {
13            if (!items[i].name.equals(ITEM_NAME_AGED_BRIE)
14                && !items[i].name.equals("Backstage passes to a TAFKAL80ETC concert")) {
15                if (items[i].quality > 0) {
16                    if (!items[i].name.equals("Sulfuras, Hand of Ragnaros")) {
17                        items[i].quality = items[i].quality - 1;
18                    }
19                }
20            } else {
21                if (items[i].quality < 50) {
22                    items[i].quality = items[i].quality + 1;
```

💡 Mouse hover → IDE hints
ctrl+alt+c → extract constant

Removing clutter continued...

1. Mouse hover
→ IDE hints

'item.quality - item.quality' can be replaced with '0'

Result of 'item.quality - item.quality' is always '0'

Replace with '0' Alt+Umschalt+Eingabe   More actions...

2. alt+Enter
→ IDE actions

Simplify
Cleanup code
Encapsulate field
Remove braces from 'else' statement
Replace '=' with '-='
Press Strg+Q to toggle preview

3. Decluttered code

alt+Enter → IDE actions

Dealing with the conditional statements

```java
if (!item.name.equals(ITEM_NAME_AGED_BRIE)
    && !item.name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    if (item.quality > 0) {
        if (!item.name.equals(ITEM_NAME_SULFURAS)) {
            item.quality = item.quality - 1;
        }
    }
} else {
    if (item.quality < 50) {
        item.quality = item.quality + 1;


        if (item.name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
            if (item.sellIn < 11) {
                if (item.quality < 50) {
                    item.quality = item.quality + 1;
                }
            }


            if (item.sellIn < 6) {
                if (item.quality < 50) {
                    item.quality = item.quality + 1;
                }
            }
        }
    }
}
```

1. Mouse hover → IDE hints

```
17    if (item.quality > 0) {
18        if (!item.name.equals(ITEM_NAME_SULFURAS)) {
19    ┌─────────────────────────────────────────────────────────────┐
      │ Merge this if statement with the enclosing one.          ⋮ │
20    ├─────────────────────────────────────────────────────────────┤
      │ SonarLint: Show issue locations   Alt+Umschalt+Eingabe    More actions... Alt+Eingabe │
21        }                          └───────────────────────────────┘
22    } else {
```

2. Refactored code

```
15    if (!item.name.equals(ITEM_NAME_AGED_BRIE)
16        && !item.name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
17        if (item.quality > 0 && !item.name.equals(ITEM_NAME_SULFURAS)) {
18            item.quality = item.quality - 1;
19        }
20    } else {
21        if (item.quality < 50) {
22            item.quality = item.quality + 1;
23
24            if (item.name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
25                if (item.sellIn < 11 && item.quality < 50) {
26                    item.quality = item.quality + 1;
27                }
28
29                if (item.sellIn < 6 && item.quality < 50) {
30                    item.quality = item.quality + 1;
31                }
32            }
33        }
34    }
```
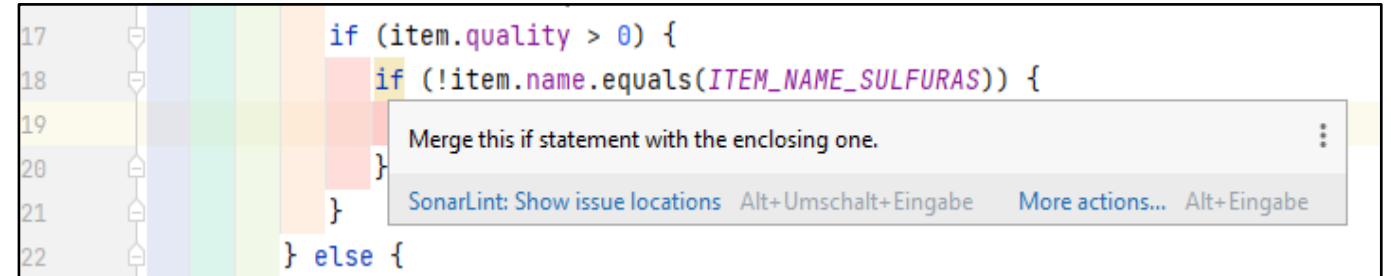
Dealing with the conditional statements continued...

```java
15    if (!item.name.equals(ITEM_NAME_AGED_BRIE)
16        && !item.name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
17        if (item.quality > 0 && !item.name.equals(ITEM_NAME_SULFURAS)) {
18            item.quality = item.quality - 1;
19        }
20    } else {
21        if (item.quality < 50) {
22            item.quality = item.quality + 1;
23
24            if (item.name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
25                if (item.sellIn < 11 && item.quality < 50) {
26                    item.quality = item.quality + 1;
27                }
28
29                if (item.sellIn < 6 && item.quality < 50) {
30                    item.quality = item.quality + 1;
31                }
32            }
33        }
34    }
```

Is the code any better?
- Too much code-juggling (deleting brackets, moving code, ...)
- Mixing differen things in if-Statements
- Code is not easier to understand...

💡 ctrl+z / git rollback is your friend

Getting rid of duplication

```
if (items[i].quality < 50) {
    items[i].quality = items[i].quality + 1;
    if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
        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;
            }
        }
    }
}
}
```

1. ctrl+shift+Cursor-down
   → move code down one line

```
if (items[i].quality < 50) {
    items[i].quality = items[i].quality + 1;
}
if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    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;
        }
    }
}
}
```

ctrl+shift+Cursor-up/down → move code up/down one line

Getting rid of duplication continued...

```
if (items[i].quality < 50) {
    items[i].quality = items[i].quality + 1;
}
if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    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;
        }
    }
}
}
```

1. ctrl+shift+m → extract method

```
increaseQuality (i);
if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    if (items[i].sellIn < 11) {
        if (items[i].quality < 50) {
            items[i].quality = items[i].quality + 1;
        }
    }

    if (items
        if (it
            ite
        }
    }
}
}
```

Process Duplicates

Do you want to replace this occurrence?
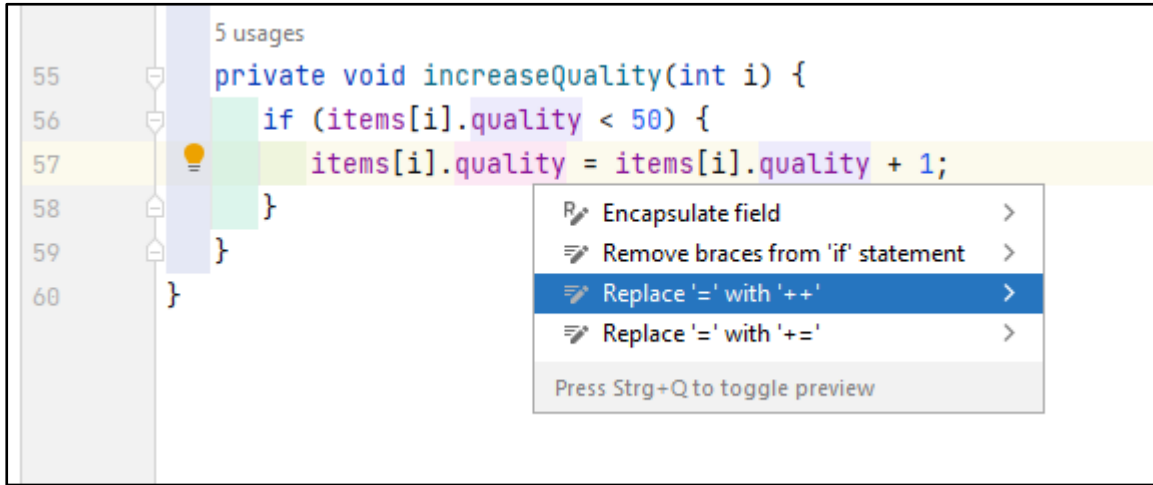
[Replace] [Skip] [All] [Cancel]

ctrl+alt+m → extraxct method

Refactoring new code as well...

1. alt enter → simplify
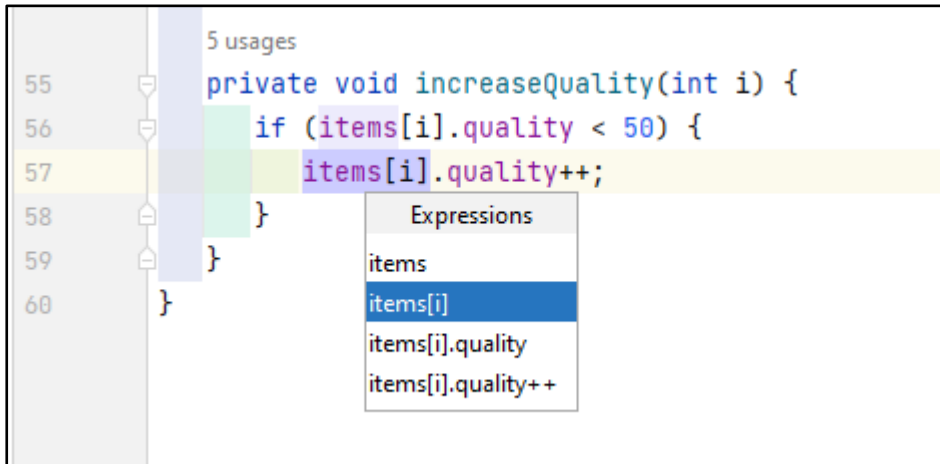
```
5 usages
55  private void increaseQuality(int i) {
56      if (items[i].quality < 50) {
57          items[i].quality = items[i].quality + 1;
58      }
59  }
60  }
```

```
 ⌨ Encapsulate field                        >
 ⌨ Remove braces from 'if' statement        >
 ⌨ Replace '=' with '++'                     >
 ⌨ Replace '=' with '+='                     >
   Press Strg+Q to toggle preview
```

2. ctrl+alt+p → extract method parameter

```
5 usages
55  private void increaseQuality(int i) {
56      if (items[i].quality < 50) {
57          items[i].quality++;
58      }
59  }
60  }
```
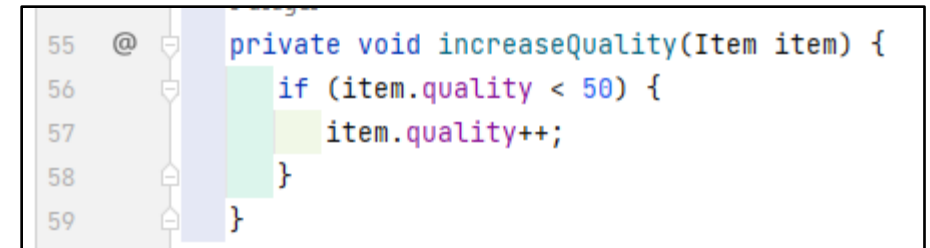
```
Expressions
items
items[i]
items[i].quality
items[i].quality++
```

3. Refactored code

```
55 @  private void increaseQuality(Item item) {
56      if (item.quality < 50) {
57          item.quality++;
58      }
59  }
```

ctrl+alt+p → extract method parameter

Dealing with the conditional statements , second attempt

```
if (!items[i].name.equals(ITEM_NAME_AGED_BRIE)
    && !items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    if (items[i].quality > 0) {
        if (!items[i].name.equals(ITEM_NAME_SULFURAS)) {
            items[i].quality = items[i].quality - 1;
        }
    }
} else {
    increaseQuality(items[i]);
    if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
        if (items[i].sellIn < 11) {
            increaseQuality(items[i]);
        }

        if (items[i].sellIn < 6) {
            increaseQuality(items[i]);
        }
    }
}
```

1. alt+Enter → IDE actions

```
if (!item.name.equals(ITEM_NAME_AGED_BRIE)) {
    if (!item.name.equals(ITEM_NA        Flip '.equals()'                    >
        if (item.quality > 0) {          Invert 'if' condition               >
            if (!item.name.equals(I      Remove braces from 'if' statement   >
                item.quality = item.     Replace with 'Objects.equals()'     >
            }
        }                                Press Strg+Q to toggle preview
    }
```

2. Refactored Code

```
if (items[i].name.equals(ITEM_NAME_AGED_BRIE)
    || items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    increaseQuality(items[i]);
    if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
        if (items[i].sellIn < 11) {
            increaseQuality(items[i]);
        }

        if (items[i].sellIn < 6) {
            increaseQuality(items[i]);
        }
    }
} else {
```

Dealing with the conditional statements continued

```
if (items[i].name.equals(ITEM_NAME_AGED_BRIE)
    || items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
    increaseQuality(items[i]);
    if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
        if (items[i].sellIn < 11) {
            increaseQuality(items[i]);
        }

        if (items[i].sellIn < 6) {
            increaseQuality(items[i]);
        }
    }
} else {
```

1. alt+Enter → IDE actions

```
for (int i = 0; i < items.length; i++) {
    if (items[i].name.equals(ITEM_NAME_AGED_BRIE)
        || items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
        increaseQua┌─────────────────────────────────────────────────┐
        if (items[i│ ℞  Encapsulate field                    >    │ EPASSES)) {
            if (item│ ☰  Extract Set from comparison chain    >    │
                incre│ ☰  Extract if (items[i].name.equals(...))  > │
            }        │ ☰  Flip '.equals()'                     >    │
                     │ ☰  Invert 'if' condition                >    │
        if (item     │ ☰  Replace '||' with '&&'               >    │
            increaseQu│ Press Strg+Q to toggle preview              │
        }            └─────────────────────────────────────────────────┘
    }
}
```

IDE action extract if statement

Wow, unused code!

```
13  public void updateQuality() {
14      for (int i = 0; i < items.length; i++) {
15          if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
16              increaseQuality(items[i]);
17              if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
18                  if (items[i].sellIn < 11) {
19                      increaseQuality(items[i]);
20                  }
21
22                  if (items[i].sellIn < 6) {
23                      increaseQuality(items[i]);
24                  }
25              }
26          } else if (items[i].name.equals(ITEM_NAME_AGED_BRIE)) {
27              increaseQuality(items[i]);
28              if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
29                  if (items[i].sellIn < 11) {
30                      increaseQuality(items[i]);
31                  }
32
33                  if (items[i].sellIn < 6) {
34                      increaseQuality(items[i]);
35                  }
36              }
37          } else {
38              if (items[i].quality > 0) {
39                  if (!items[i].name.equals(ITEM_NAME_SULFURAS)) {
40                      decreaseQuality(items[i]);
41                  }
42              }
43          }
44
```

⚠4 ⚠2 ✔5

com.gildedrose   100% (2/2) 100% (6/6) 89% (35/.
GildedRose   100% (1/1) 100% (4/4) 88% (30/.
Item   100% (1/1) 100% (2/2) 100% (3/.

Let's get rid of it!

So after repeating these steps…

…the code got more readable…

…and most importantly: contidional statements refer to the same field!

```java
public void updateQuality() {
    for (int i = 0; i < items.length; i++) {
        if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
            increaseQuality(items[i]);
            if (items[i].sellIn < 11) {
                increaseQuality(items[i]);
            }
            if (items[i].sellIn < 6) {
                increaseQuality(items[i]);
            }
        } else if (items[i].name.equals(ITEM_NAME_AGED_BRIE)) {
            increaseQuality(items[i]);
        } else if (items[i].name.equals(ITEM_NAME_SULFURAS)) {
        } else {
            reduceQuality(items[i]);
        }

        if (!items[i].name.equals(ITEM_NAME_SULFURAS)) {
            items[i].sellIn = items[i].sellIn - 1;
        }
        if (items[i].sellIn < 0) {
            if (items[i].name.equals(ITEM_NAME_AGED_BRIE)) {
                increaseQuality(items[i]);
            } else if (items[i].name.equals(ITEM_NAME_SULFURAS)) {
                continue;
            } else if (items[i].name.equals(ITEM_NAME_BACKSTAGEPASSES)) {
                items[i].quality = 0;
            } else {
                reduceQuality(items[i]);
            }
        }
    }
}
```

After some more juggling…

…I ended up with one switch statement!

This is WIP, so the presentation stops here.

The next step could be introducing polymorphism

```java
public void updateQuality() {
    for (int i = 0; i < items.length; i++) {
        switch (items[i].name) {
            case ITEM_NAME_BACKSTAGEPASSES:
                decreaseSellIn(items[i]);
                increaseQuality(items[i]);
                if (items[i].sellIn < 10) {
                    increaseQuality(items[i]);
                }
                if (items[i].sellIn < 5) {
                    increaseQuality(items[i]);
                }
                if (items[i].sellIn < 0) {
                    items[i].quality = 0;
                }
                break;
            case ITEM_NAME_AGED_BRIE:
                decreaseSellIn(items[i]);
                increaseQuality(items[i]);
                if (items[i].sellIn < 0) {
                    increaseQuality(items[i]);
                }
                break;
            case ITEM_NAME_SULFURAS:
                break;
            default:
                decreaseSellIn(items[i]);
                reduceQuality(items[i]);
                if (items[i].sellIn < 0) {
                    reduceQuality(items[i]);
                }
                break;
        }
    }
}
```

So did my experiment work?

- ## No typing!
  Methodnames, Tests, some deleting and copying → Pretty much, yeah!

- ## No thinking!
  I had to think only twice, but I did not have to really understand, what the code does!

? Suggestions ? Head scratches

Questions ? Kudos ¿ More Questions ?

Thank you for your attention!

Contact: philipp.eichenberger@css.ch

Used Resources:
- IntelliJ IDE https://www.jetbrains.com/idea
- Pitest for Mutation Testing https://pitest.org
- Apporoval Tests for Characterization Testing https://approvaltests.com