

Arbeitsblatt: Unmodifiable Collection Decorator

Im Collection Framework gibt es neben dem Synchronisations-Dekorator auch noch Dekoratoren, die verhindern, dass eine Collection verändert wird. Diese Dekoratoren können mit einer der folgenden statischen Methoden aus der Klasse Collections erstellt werden:

```
public static <T> Collection<T> unmodifiableCollection(Collection<? extends T> c)
public static <T> Set<T> unmodifiableSet(Set<? extends T> c)
public static <T> SortedSet<T> unmodifiableSortedSet(SortedSet<T> s)
public static <T> List<T> unmodifiableList(List<? extends T> list)
```

Diese Methoden geben eine „read-only“ Sicht auf jene Collection zurück, die beim Aufruf übergeben wird. Leseoperationen greifen dabei auf die Original-Collection zurück und Operationen, welche die Collection verändern wollen, werfen eine UnsupportedOperationException.

Aufgabe:

Implementieren Sie die Methode

```
public static <T> Collection<T> unmodifiableCollection(Collection<T> c)
```

Diese soll einen Collection-Dekorator zurückgeben, der Methodenaufrufe, welche die Collection potentiell ändern könnten, mit einer UnsupportedOperationException beantwortet. Dies gilt auch für Methoden die über einen von diesem Dekorator zurückgegebenen Iterator aufgerufen werden.

Sie finden im Wochenprojekt im Paket patterns.decorator.util einen Programmrahmen und eine Testklasse.

Bemerkungen:

- Wir haben die Signatur der Methode unmodifiableCollection absichtlich etwas vereinfacht (bezüglich Generics), denn der Fokus soll auf den Dekoratoren liegen.
- Falls Sie Eclipse verwenden: Der Code kann fast ausschliesslich mit Operationen aus dem Eclipse-Menu *Source* generiert werden.