

## Aufgaben zum Workshop

### Exploration

1. Verschieben Sie die statische innere Klasse *SchedulerOverviewPage.DateUtils* in eine separate Datei.
2. Ergründen Sie, wie die Klassen *DateUtils* und *ProductField* funktionieren, indem Sie den Code sukzessive formatieren, verständliche Bezeichner für Klassen, Methoden und Variablen wählen und diese mit zweckmäßigen JavaDocs versehen.

### Code Qualität

3. Entfernen Sie toten Code, Magic Numbers und andere Code Smells.
4. Loggen Sie das Auftreten nicht gezielt behandelter Exceptions.
5. *DateUtils.format* wirft offenbar eine checked Exception, deklariert diese aber nicht. Wie kann das sein?
6. Entzerren Sie die Streams, indem Sie Teile davon in separate Methoden extrahieren.

### Testbarkeit

7. Ersetzen Sie das Interface *ProductField* durch eine einfache Datenklasse mit den Attributen *label* und *property*.
8. Extrahieren Sie die Erstellung des Headers in eine separate Methode mit "funktionaler" Schnittstelle und ohne interne Abhängigkeiten.
9. Schreiben Sie einen Regressionstest für die neu extrahierte Methode.
10. Es stellt sich heraus, dass die Speicherung mehrerer Properties in einem einzelnen Feld fachlich gar nicht benötigt wird. Entfernen Sie diese Funktion und führen Sie Ihren Regressionstest erneut aus.
11. Verschieben Sie die Erstellung der *CharSequence* aus der *SchedulerOverviewPage* in den *SchedulerService* und schreiben Sie einen Integrationstest.

### Reflektion

12. Führen Sie ein Code Review mit einem Partner durch:
  - a. Welche Anpassungen habe ich vorgenommen und warum?
  - b. Was sind die Konsequenzen aus meinen Entscheidungen?
  - c. Welche Alternativen hätte es gegeben?

### Optional: Testcontainers

13. Aktuell nutzen die Tests eine HSQL in-Memory Datenbank. Um Fehler in Produktion leichter nachstellen zu können, sollen diese künftig ebenfalls gegen PostgreSQL laufen. Stellen Sie die Tests so um, dass vor der Ausführung immer ein neuer Docker-Container mit entsprechend vorbereiteter Datenbank hochgefahren wird. Nutzen Sie hierzu die bereitgestellten Klassen *TestContainersPersistenceConfig*, *DbContainerExtension* und *DatabaseContainerHolder*.