

Übungen zu TDD mit TYPO3 CMS 7.6

1. Installiert euch die Extensions `phpunit` und `extension_builder`.
2. Lasst die Tests der `phpunit`-Extension im Backend laufen.
3. Legt mit dem Extension-Builder eine leere Extension `stock` an. Legt zwei leere Models `Product` und `Color` an und löscht die automatisch generierten Tests wieder.
4. Öffnet die Extension in eurer IDE und bindet den TYPO3-Source sowie die PHPUnit-Extension ein (oder die komplette TYPO3-Installation).
5. Schreibt ein `Test/Unit/MathTest.php` einen Test, der überprüft, dass $1 + 1 = 2$ ist.¹ Führt die Tests aus.
6. Schreibt einen Test für die beiden Model-Klasse, dass sie instanziiert werden können und Unterklassen der Model-Basisklasse `AbstractEntity` sind.
7. Fügt (testgetrieben) ein Feld `$title` mit dem Getter und Setter zu beiden Models hinzu. Werft eine `UnexpectedValueException`, falls bei `setTitle` ein leerer String übergeben wird.
8. Legt (testgetrieben) eine 1:n-Assoziation `Product.colors` an (inklusive aller nötigen Methoden).
9. Schreibt (testgetrieben) eine Methode `Product.getFullTitle`, die den Title inklusive aller Farben zurückgibt.
10. Schreibt ein `ProductRepository` und testet, dass ihr es instanziiieren könnt und dass es eine Unterklasse der Repository-Basisklasse ist. Ihr benötigt dafür wahrscheinlich ein gemockten `ObjectManager`. Schaut euch dafür den Konstruktor der Basisklasse an.
11. Schreibt einen `Service/ImportService.php`, der Produkt-Titel aus einer Textdatei liest (eine pro Zeile) und diese im Repository neu anlegt. Der Service braucht nicht zu überprüfen, ob es schon Produkte mit demselben Titel gibt.

¹Es ist gut, das ab und an zu überprüfen, weil wir ein echtes Problem bekommen, falls das irgendwann nicht mehr stimmen sollte.