

# 1 Cheat Sheet

## 1.1 Unit Tests

- Teste *Verhalten* (nicht Code)  
Prüfe *Rückgabewerte* und *Kommunikation mit Abhängigkeiten* (anderen Units).
- Minimiere Logik im Test-Code  
Nutze die Fähigkeiten des Frameworks
- Nur **eine** Erwartung je Testmethode  
in der Regel ein einzelnes **assert**
- Aufbau der Testmethode: AAA (*arrange*, *act*, *assert*)
- FIRST RTFM  
Fast **I**solated **R**epeatable **S**elf-validating **T**imely  
Readable **T**rustworthy **F**ast **M**aintainable

## 1.2 TDD - Mirco Cycle

1. Schreibe gerade so viel Test- Code, dass er fehl schlägt.  
Test Code "schlägt fehl" wenn er nicht kompiliert oder ein **assert** Aufruf mit einer Exception endet.
2. Schreibe gerade so viel Produktiv- Code, dass alle Tests erfüllt werden (also entweder compiliert, oder kein **assert** Aufruf fehl schlägt.)  
Dabei gilt: *quick and dirty*, keine lange Planung, einfach so schnell wie möglich so billig wie möglich fertig werden.
3. Refaktorisiere Produktiv-Code und Test-Code (aber nicht gleichzeitig!)  
Beseitige Code-Duplikate, wende Entwurfsmuster dort an, wo es sich anbietet, Überprüfe Transitions-Prioritäten. **In dieser Phase darf kein Test fehl schlagen** oder (außer durch automatisierte Refactorings der IDE) gleichzeitig mit dem Produktivcode geändert werden und wieder grün zu werden!

### 1.3 automatisierte Refactorings

refactoring	Beschreibung	Shortcut Eclipse	Shortcut Deine IDE
Refactoring menu	Liste aller möglichen automatisierten Refactorings, zum Teil abhängig von der Cursorposition bzw. dem markierten Text	Refactor right Mouse >> Refactor ↑ + Alt + T	
context dependend Proposal	Liste aller im aktuellen Kontext möglichen automatisierten Refactorings, abhängig von der Cursorposition bzw. dem markierten Text	ctrl + 1	
change Method signature	Name, Rückgabewert und Anzahl, Reihenfolge oder Typ der Parameter ändern	↑ + ctrl + C	
rename in file	Name von Variablen oder Methoden in deren Sichtbarkeits-Scope innerhalb der Datei ändern	ctrl + 2 → R ctrl + 1 → Rename in file	
extract variable	markierten Ausdruck einer neuen Variablen zuweisen	ctrl + 1 → Extract to local variable	
rename in Workspace	Name von Variablen, Methoden oder Klassen in allen Dateien des Projekts ändern, wenn sie sich auf diese Stelle beziehen	ctrl + R ctrl + 1 → Rename in workspace	
duplicate lines	kopiert die markierten Zeilen innerhalb der Datei	↑ + ctrl + ↑ ↑ + ctrl + ↓	
move lines	bewegt die markierten Zeilen innerhalb der Datei	Alt + ↑ Alt + ↓	
extract method	verschiebt die markierten Instruktionen (wenn möglich) in eine neue Methode	↑ + ctrl + M	
move method	verschiebt die aktuelle Methode (in der der Cursor steht) in die Klasse eines ihrer Parameter	↑ + ctrl + V	