

# UNITTESTING

By Emad Easa



# Einführung in Unittesting

- Unittesting ist eine Softwaretestmethode, bei der einzelne Einheiten oder Komponenten einer Anwendung isoliert getestet werden.
  - Ziel ist es, sicherzustellen, dass jede Komponente wie erwartet funktioniert.
  - Unittests helfen dabei, Fehler früh im Entwicklungsprozess zu identifizieren und zu beheben.
-

# Grundlagen des Unittestings in Java

- JUnit ist eine weit verbreitete Unittest-Bibliothek in Java, die eine einfache Erstellung und Ausführung von Tests ermöglicht.
  - Ein typischer JUnit-Test besteht aus Annotationen wie `@Test`, die dem Testframework signalisieren, welche Methoden als Tests auszuführen sind.
-

# Triple-A Pattern im Unittesting

- Triple-A steht für Arrange, Act, Assert und ist ein Muster, das zur Strukturierung von Unittests verwendet wird.
  - Es hilft dabei, Tests übersichtlich und wartbar zu gestalten
  - Arrange: In diesem Schritt werden alle notwendigen Vorbereitungen getroffen
    - Erstellen von Objekten*
    - Initialisieren von Variablen*
  - Act: Dieser Schritt umfasst die Ausführung der Funktionalität
  - Assert: Im letzten Schritt werden Verifizierungen durchgeführt, um sicherzustellen, dass die im Act-Schritt ausgeführte Aktion die erwarteten Ergebnisse oder Zustandsänderungen produziert hat.
-

# Einstellungen für JUnit

- Download der Mavendependency von <https://mvnrepository.com/artifact/junit/junit>
  - Kopieren der Dependency und einfügen im pom file innerhalb des <dependencies> Tag (wenn dieser Tag noch nicht existiert, muss man ihn anlegen)
-

# Parametrisierte Tests mit JUnit

- ... ermöglichen es dieselbe Testmethode mit verschiedenen Daten zu wiederholen
  - Testen des Verhaltens mit unterschiedlichen Eingabewerten
  - `@ValueSource`: Angabe von verschiedenen Literalen wie Strings oder Integer
  - `@MethodSource`: Hier kann eine Methode angegeben werden, die einen Stream, Collection oder Array von Werten liefert
  - `@CsvSource` (`@CsvFileSource`): Ermöglicht es, Daten in Form von CSV (Comma-Separated Values) anzugeben. `@CsvSource` nimmt die Daten direkt als String-Annotation, während `@CsvFileSource` es erlaubt, die Daten aus einer Datei zu lesen.
-