

## Praktikum 3 zur OOS

### Aufgabe 1: (Installation und Integration von JUnit 5)

In der ersten Aufgabe sollen Sie JUnit 5 installieren und in Ihre (Entwicklungs-) Umgebung integrieren, damit Sie die selbst geschriebenen JUnit-Tests ausführen können. Als Java-Version brauchen Sie mind. Java 8. Da die Installation (und Integration) von externen Libraries/Frameworks von Ihrer Umgebung/Plattform abhängt, werden hier lediglich **zwei mögliche Vorgehensweisen** vorgestellt (Weitere sind denkbar und werden bei korrekter Lauffähigkeit der Tests akzeptiert):

- Möglichkeit 1 - *IntelliJ mit Maven*  
Recherchieren Sie, wie Sie ein Maven-Projekt in IntelliJ erzeugen können. Sie können die von uns im Ilias hochgeladene pom.xml nutzen.
- Möglichkeit 2 - *Eclipse*  
Siehe z.B. [diesen Link](#) für weitere Infos.

### Aufgabe 2: (JUnit-Test der Klassen aus Praktikumsversuch 2)

Geben Sie eine ausreichende Menge von Tests der Klassen aus Praktikum 2 an und implementieren Sie diese in JUnit 5.

### Aufgabe 3: (Persistenz der Nutzerverwaltung)

Im Folgenden sollen die Datenstrukturen der Nutzerverwaltung persistent als Datei auf dem Dateisystem des Rechners abgelegt werden.

- Erweitern Sie die Klasse `BenutzerVerwaltungAdmin` um eine Methode `dbInit()`, die ein neues leeres Objekt Ihrer Datenstruktur anlegt, serialisiert und dieses im Dateisystem ablegt. Der Dateipfad-/name soll programmatisch beim Erzeugen einer Instanz der Klasse `BenutzerVerwaltungAdmin` gesetzt werden können. Diese Methode wird wie die Methode `benutzerLoeschen` nicht im Interface `BenutzerVerwaltung` zur Verfügung gestellt.
- Erweitern Sie die Implementierungen der drei anderen Methoden der Klasse `BenutzerVerwaltungAdmin`, so dass Änderungen und Überprüfungen in der Datenstruktur immer auf der von der Methode `dbInit()` angelegten Datei durchgeführt werden und somit persistent gemacht werden. Lesen Sie hierzu jeweils die aktuelle Datenstruktur aus dem File, deserialisieren Sie diese und verweisen Sie mit dem `private`-Attribut auf das erhaltene Objekt. Führen Sie dann die Änderungen oder Überprüfungen darauf aus. Im Falle einer durchgeführten Änderung serialisieren Sie danach die veränderte Datenstruktur und speichern/überschreiben Sie diese in Ihrem Dateisystem.
- Erweitern Sie die Testfälle aus Aufgabe 1 zum Test der Methoden aus den Teilen a) und b). Testen Sie insbesondere, ob sich Ihr Programm sowohl mit einer initialisierten als auch mit einer nicht-initialisierten Datenstruktur korrekt verhält.