

## Aufgabe: Zoo

### *Ziel:*

Erstelle eine Java-Anwendung, die einen Zoo simuliert. Die Anwendung soll die Möglichkeit bieten, verschiedene Tiere hinzuzufügen und zu entfernen und Informationen über die Tiere abzurufen. Verwende Spring Boot und JUnit-Tests, um die Anwendung zu implementieren. Verwende außerdem eine Collection deiner Wahl, um die Tiere zu speichern und zu verwalten.

### *Anforderungen:*

1. Damit wir uns gut auf das Gespräch vorbereiten können, benötigen wir den Sourcecode der Anwendung. Du kannst den Sourcecode als Archiv zum vereinbarten Zeitpunkt über den Link <https://drive.google.com/<folder-id>> in Google-Drive hochladen, dafür benötigst du einen Google-Account. Wenn du kein Google-Konto hast bzw. erstellen möchtest, kannst du das Projekt auch in GitHub einchecken oder uns das Archiv als Downloadlink zur Verfügung stellen.
2. Es wird ein Build Tool wie z.B. Maven eingesetzt, damit der Code kompiliert und ausgeführt werden kann. Anforderungen wie z.B. die benötigte Java Version oder besondere Setup-Anweisungen sind in einer README-Datei dokumentiert.
3. Erstelle eine Klasse `Animal`, die die Eigenschaften eines Tieres enthält (Name, Gewicht, Alter, usw.).
4. Erstelle eine Klasse `Zoo`, die die Tiere verwaltet. Die Klasse soll Methoden zum Hinzufügen, Entfernen und Abrufen von Tieren enthalten.
5. Verwende Spring Boot, um eine REST-API zu erstellen, mit der Tiere hinzugefügt, entfernt und abgerufen werden können.
6. Erstelle JUnit-Tests für die Klassen `Animal` und `Zoo`.

### *Bonus:*

- Füge eine Methode hinzu, um die Tiere nach bestimmten Kriterien zu suchen (z.B. nach Gewicht, Alter, etc.).
- Füge eine Methode hinzu, um ein Tier zu füttern und sein Gewicht zu ändern.

### *Hinweise:*

- Wir besprechen deine Umsetzung mit dir im Bewerbungsgespräch. Dabei kannst du uns deinen Code vorstellen und deine Implementierungsentscheidungen begründen.
- Es ist nicht erforderlich, die Bonusaufgaben zu lösen. Es hilft uns allerdings, einen besseren Einblick über dein Vorwissen zu erhalten.