

# LU07a - Spring & Spring Boot

## Einführung

Spring und Spring Boot bieten eine Menge gebräuchlicher Funktionen an, welche in herkömmlichen Java manuell implementiert werden müssen. In der nachfolgenden Tabelle sind einige Beispiele gelistet inklusive einer Alternative in herkömmlichen Java.

Use Case	Beispiel in Plain Java	Spring	Spring Boot
Konfigurationen	Manuelle Implementation mit Properties	Annotations: <code>@Configuration</code> <code>@PropertySource("classpath:application.properties")</code> ... <code>@Value("\${jakarta.persistence.jdbc.user}")</code>	Automatische Erkennung von <code>application.properties</code> <code>@Value("\${jakarta.persistence.jdbc.user}")</code>
Datenbankanbindung	Implementation mit JDBC, JPA/Hibernate	Templates <code>org.springframework.jdbc</code> , um Hibernate einfacher zu nutzen	Zusätzlich <code>org.springframework.data.jpa.repository.JpaRepository</code> , um einfach Entitäten zu mappen.
Logging	SLF4J	Eigene SLF4J-Bindings, einfache Konfiguration	SLF4J automatisch vorkonfiguriert, out-of-the-box
REST-API	Javalin	-	Umfassendes Framework mit vielen fixfertigen Funktionalitäten.
...	...	...	...

From:  
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:  
<https://wiki.bzz.ch/de/modul/ffit/3-jahr/java/learningunits/lu07/a?rev=1760998879>

Last update: **2025/10/21 00:21**

