

# LU03c - Klassen und Tabellen synchronisieren

## Ausgangslage

In der vorhergehenden Lektion haben Sie die Klasse Book händisch implementiert. Bei einer Änderung müssen Sie dadurch immer sowohl die Tabelle als auch die Klasse anpassen, da diese immer aufeinanderpassen müssen.

SQL books	Java Book					
<table border="1"> <tr> <td>publication_year</td> <td>INT</td> </tr> </table>	publication_year	INT	<table border="1"> <thead> <tr> <th>Book</th> </tr> </thead> <tbody> <tr> <td>           -id: int            -title: String            -author: String            -isbn: String            -publicationYear: int         </td> </tr> <tr> <td>           +Book(id: int, isbn: String, title: String, author: String, publicationYear: int)            +getId(): int            +getTitle(): String            +setTitle(title: String): void            +getAuthor(): String            +setAuthor(author: String): void            +getIsbn(): String            +setIsbn(isbn: String): void            +getPublicationYear(): int            +setPublicationYear(publicationYear: int): void            +toString(): String         </td> </tr> </tbody> </table>	Book	-id: int -title: String -author: String -isbn: String -publicationYear: int	+Book(id: int, isbn: String, title: String, author: String, publicationYear: int) +getId(): int +getTitle(): String +setTitle(title: String): void +getAuthor(): String +setAuthor(author: String): void +getIsbn(): String +setIsbn(isbn: String): void +getPublicationYear(): int +setPublicationYear(publicationYear: int): void +toString(): String
publication_year	INT					
Book						
-id: int -title: String -author: String -isbn: String -publicationYear: int						
+Book(id: int, isbn: String, title: String, author: String, publicationYear: int) +getId(): int +getTitle(): String +setTitle(title: String): void +getAuthor(): String +setAuthor(author: String): void +getIsbn(): String +setIsbn(isbn: String): void +getPublicationYear(): int +setPublicationYear(publicationYear: int): void +toString(): String						

Um unnötige Arbeitsschritte zu vermeiden, können wir die eine Seite aus der anderen generieren:

- Java-first: Aus den Java-Klassen werden die entsprechenden SQL-Tabellen generiert.
- SQL-first: Aus den SQL-Tabellen werden die entsprechende Java-Klassen generiert.

## JPA/Hibernate

Wir wählen den Java-first-Ansatz mit JPA/Hibernate.

...

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

Permanent link:  
<https://wiki.bzz.ch/de/modul/ffit/3-jahr/java/learningunits/lu03/main?rev=1756462495>

Last update: **2025/08/29 12:14**

