

# Lernstandserhebung 1

im Fach Programmieren – II

Fachlich Inhalte:  
Grundlagen von Java-Klassen,  
elementare Beziehungen zwischen Klassen,  
einfache UML-Klassendiagramme,  
Anwendung von ArrayLists

Das vorliegende Programmbeispiel modelliert wesentliche Funktionen einer Bibliothek. Dafür werden die Klassen Bibo, Buch und Nutzer verwendet, deren Quelltext hier angegeben ist.

Lesen Sie den Code und verstehen Sie seine Funktion.

```
package lse.bib;

public class Nutzer {

    String name;

    public Nutzer(String name) {
        this.name = name;
    }

    public String getName() {
        return this.name;
    }

}
```

```
package lse.bib;

public class Buch {

    public static String REGAL = "-Regal-";

    String autor;        // Name des Autos
    String titel;        // Buchtitel
    String signatur;     // Bibliotheksnotation
    String status;       // Bei Ausleihe: Nutzernamen, sonst -Regal-

    public Buch(String autor, String titel, String signatur) {
        this.autor = autor;
        this.titel = titel;
        this.signatur = signatur;
        this.status = REGAL;
    }

    public void ausleihen(String name) {
        this.status = name;
    }

    public void rueckgeben() {
        this.status = REGAL;
    }

    public String getTitel() {
        return this.titel;
    }

    public String getStatus() {
        return this.status;
    }

}
```

```

package lse.bib;

import java.util.ArrayList;

public class Bibo {

    ArrayList<Nutzer> kunden;
    ArrayList<Buch> buecher;

    public Bibo()
    {
        this.kunden = new ArrayList<Nutzer>();
        this.buecher = new ArrayList<Buch>();
    }

    public void verwalteBuch(Buch b) {
        this.buecher.add(b);
    }

    public Buch holeBuch(String titel) {
        for (Buch b : this.buecher) {
            if (b.getTitel().equalsIgnoreCase(titel)
                && b.getStatus().equalsIgnoreCase(Buch.REGAL))
                return b;
        }
        return null;
    }

    public Nutzer neuerNutzer(String name) {
        Nutzer n = new Nutzer(name);
        this.kunden.add(n);
        return n;
    }

    public void ausleihen(Nutzer n, Buch b) {
        if (this.kunden.contains(n)) {
            int idx = this.buecher.indexOf(b);
            if (idx >= 0) {
                b.ausleihen(n.getName());
            }
        }
    }

    public void rueckgeben(Nutzer n, Buch b) {
        if (this.kunden.contains(n)) {
            int idx = this.buecher.indexOf(b);
            if (idx >= 0) {
                b.rueckgeben();
            }
        }
    }

    public static void main(String[] args) {

        Buch b1 = new Buch("Marx", "Das Kapital", "K12-1");
        Buch b2 = new Buch("Tolstoi", "Krieg und Frieden", "K21-1");

        Bibo bib = new Bibo();
        bib.verwalteBuch(b1);
        bib.verwalteBuch(b2);

        Nutzer n1 = bib.neuerNutzer("Pia");

        Buch b = bib.holeBuch("Krieg und Frieden");
        System.out.println(b.getTitel()+" "+b.getStatus());

        bib.ausleihen(n1, b);
        System.out.println(b.getTitel()+" "+b.getStatus());

        bib.rueckgeben(n1, b);
        System.out.println(b.getTitel()+" "+b.getStatus());
    }
}

```

### Aufgabe 1

Ergänzen Sie die Inhalte der Klasse Nutzer im UML-Klassendiagramm.

<b>Bibo</b>
kunden:ArrayList<Nutzer> buecher:ArrayList<Buch>
Bibo() verwalteBuch(Buch b) holeBuch(String titel):Buch neuerNutzer(String name):Nutzer ausleihen(Nutzer n, Buch b) rueckgeben(Nutzer n, Buch b) main(String[] args)

<b>Buch</b>
autor:String titel:String signatur:String status:String
Buch(String autor, String titel, String signatur) ausleihen(String name) rueckgeben() getTitel():String getStatus():String

<b>Nutzer</b>

Bewertung: PFNu: .... / 6 Pkt.

### Aufgabe 2

Ergänzen Sie die Relationen für die Klassen im obigen UML-Klassendiagramm. Begründen Sie die Entscheidung zur Darstellung der Relationen.

... ihre Begründung ...

Bewertung: PFRel: .... / 4 Pkt. PFAg: .... / 3 Pkt. PFCo: .... / 4 Pkt.

### Aufgabe 3

Ändern Sie die Implementierung der Methode "verwalteBuch" in der Klasse Bibo derart, dass eine Warnmeldung ausgegeben wird, wenn dass neu zu verwaltende Buch bereits verwaltet wird.

```
public void verwalteBuch(Buch b) {
```

Bewertung: PSvB: .... / 5 Pkt.

### Aufgabe 4

Was ist ein Konstruktor? Erklären Sie kurz, präzise in wenigen Sätzen.

Bewertung: CUKo: .... / 3 Pkt.

### Aufgabe 5

Was bedeutet das Schlüsselwort „this“? Erklären Sie kurz, präzise in wenigen Sätzen.

Bewertung: CUth: .... / 1 Pkt.

### Aufgabe 6

Schreiben Sie Zeilenkurzkommentare hinter jede Zeile der Methode "ausleihen" der Klasse Bibo.

Bewertung: PFNo: .... / 4 Pkt.

### Aufgabe 7

Welche Ausgabe liefert die erste System.out.println-Anweisung in der main-Methode der Klasse Bibo?

Bewertung: PFEx: .... / 2 Pkt.

### Aufgabe 8

Welche Bedeutung hat die Anweisung:

```
public static String REGAL = "-Regal-";
```

in der Klasse Buch?

... ihre Antwort (1 Satz) ...

Bewertung: PFSt: .... / 1 Pkt.

### Aufgabe 9

Programmieren Sie eine Methode "zeigeStatistik", die die Anzahl Nutzer und die Anzahl Bücher der Klasse Bibo anzeigt.

```
public void zeigeStatistik() {
```

Bewertung: XFSz: .... / 4 Pkt.

**Erreichte Gesamtpunktzahl:** ..... / 37 Pkt.