

Name:

Matrikelnummer:

Aufgabe 8: (8+8=16 Punkte)

Betrachten Sie zunächst das folgende UML-Diagramm, das die Modellierung eines als DVD verfügbaren Films darstellt:



- (a) Implementieren Sie die durch obiges Diagramm vorgegebene Klasse in Java. Es müssen nur die im Diagramm angegebenen Methoden umgesetzt werden.

Die Funktion `freiFuerKinder` soll genau dann `true` zurückliefern, wenn das bei der freiwilligen Selbstkontrolle angegebene Alter (`FSKAlter`) maximal den Wert 6 hat.

Um den Rahmen der Aufgabe nicht zu sprengen, dürfen Sie auf Kommentare verzichten.

- (b) Zu der gegebenen Klasse `DVD` soll eine Unterklasse `ActionDVD` modelliert und implementiert werden, ohne dass hierfür die Klasse `DVD` verändert werden soll.

Jedes Objekt der Klasse `ActionDVD` modelliert einen als DVD verfügbaren Action-Film, dessen `FSKAlter` immer den Wert 18 hat und der somit niemals für Kinder freigegeben sein kann.

- (i) Ergänzen Sie das UML-Diagramm um die Darstellung dieser Unterklasse. Achten Sie auf eine sinnvolle Signatur für beide Konstruktoren.
- (ii) Implementieren Sie diese Unterklasse in Java. Auch hier dürfen Sie auf Kommentare verzichten.

Punktevergabe:

- 1 Attribute
- 2 Standard-Konstruktor
- 2 Kopier-Konstruktor
- 1 Methode zum Setzen des FSK-Alters
- 1 Methode zum Lesen des FSK-Alters
- 1 Methode zur Abfrage der Freigabe

8

- 1 Vererbung (UML)
- 1 Standard-Konstruktor (UML)
- 1 Kopier-Konstruktor (UML)
- 1 Verdecken einer passenden Methode (UML)
- 1 Vererbung (Java)
- 1 Standard-Konstruktor (Java)
- 1 Kopier-Konstruktor (Java)
- 1 Verdecken einer passenden Methode (Java)

8

Punkteabzüge:

- 1 Attribute nicht private (UML)
- 1 Attribute nicht private (Java)
- 1 Kopierkonstruktor legt keine Kopie von `titel` an
- 1 `int`-Konstante ohne explizite Typkonvertierung

Name:

Matrikelnummer:

Lösung:

(a) Implementierung:

```

public class DVD {
    private String titel;
    private int laufzeit;
    private byte FSKAlter;

    public DVD(String titel, int laufzeit, byte FSKAlter) {
        this.titel = titel;
        this.laufzeit = laufzeit;
        this.FSKAlter = FSKAlter;
    }

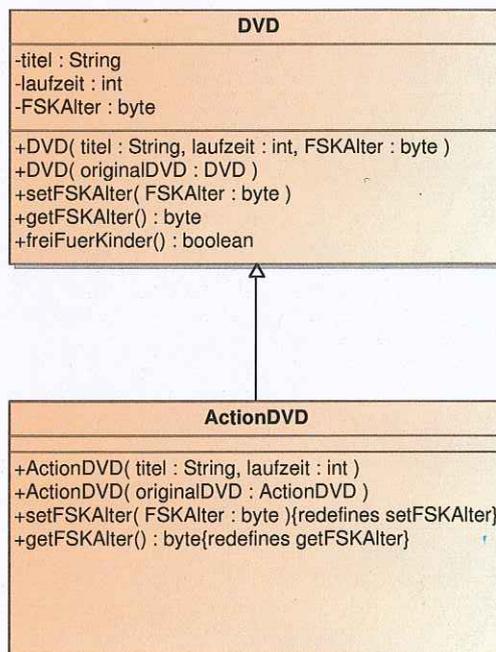
    public DVD(DVD originalDVD) {
        this.titel = new String(originalDVD.titel);
        this.laufzeit = originalDVD.laufzeit;
        this.FSKAlter = originalDVD.FSKAlter;
    }

    public void setFSKAlter(byte FSKAlter) {
        this.FSKAlter = FSKAlter;
    }

    public byte getFSKAlter() {
        return this.FSKAlter;
    }

    public boolean freiFuerKinder() {
        return (this.FSKAlter <= 6);
    }
}
    
```

(b) Modellierung



Name:

Matrikelnummer:

Implementierung:

```
public class ActionDVD extends DVD {  
  
    public ActionDVD(String titel, int laufzeit) {  
        super(titel, laufzeit, (byte)18);  
    }  
  
    public ActionDVD(ActionDVD originalActionDVD) {  
        super(originalActionDVD);  
    }  
  
    public byte getFSKAlter() {  
        return (byte)18;  
    }  
  
    public void setFSKAlter(byte FSKAlter) {  
        super.setFSKAlter((byte)18);  
    }  
}
```