

Wir werden nun die Medienverwaltung vollständig als JavaFX-Anwendung realisieren. Die Medienverwaltung soll dabei eine Drei-Schichtenarchitektur besitzen. Treffen Sie zunächst die folgenden Vorbereitungen, um eine JavaFX-GUI-Anwendung realisieren zu können. Gehen Sie bei der Programmierung schrittweise vor. Die ersten vier Arbeitsschritte sind erforderlich, um eine klare Trennung zwischen Fachlogik und Interaktion zu erhalten.

1. Überschreiben Sie in den Klassen `Audio` und `Bild` die `toString()`-Methode der Klasse `Object`. Geben Sie über die `toString`-Methode die Informationen zurück, die auch über die `druckeDaten()`-Methode ausgegeben wird.
2. Ergänzen Sie die Klassen `Audio` und `Bild` um einen Standardkonstruktor und um getter- und setter-Methoden für die fachlichen Attribute.
3. Ändern Sie die Methode „`sucheNeuesMedium()`“ der Klasse `Medienverwaltung`. Die Methode soll das gefundene Objekt als Rückgabewert liefern. Die Methode darf keine direkte Ausgabe vornehmen.
4. Ergänzen Sie die Klasse `Medienverwaltung` um die Methode `public Iterator<Medium> iterator()`. Diese Methode liefert ein Iterator-Objekt, um die interne Medienliste zu durchlaufen.

Erstellen Sie dann die Grundstruktur der 3-Schichten-Architektur und bereiten Sie danach die Integration der in Praktikum 9 erstellten Fensterklassen vor.

5. Erstellen Sie die drei Pakete `pk1.mv.gui`, `pk1.mv.fachlogik` und `pk1.mv.datenhaltung`.
6. Nehmen Sie die Implementierung der Fenster aus dem Praktikum 9 und integrieren Sie den Code in das Projekt der Medienverwaltung (falls noch nicht geschehen). Verteilen Sie alle Klassen der Medienverwaltung auf die geeigneten Pakete.<sup>1</sup>
7. Falls im Konstruktor der Klasse `BildErfassungView` eine Referenz auf eine Instanz der Klasse `Bild` übergeben wird, dann sollen die Attributwerte in die Eingabefelder eingeblendet werden. Falls der Button „Neu“ geklickt wird, dann werden die Eingabedaten in das `Bild`-Objekt übertragen.
8. Verfahren Sie analog mit dem Fenster die Audioerfassung `AudioErfassungView()` aus dem Praktikum 9.

Kümmern Sie sich nun um die Anbindung der Datenhaltung. Beachten Sie dabei die Pakete.

9. Trennen Sie die Fachkonzeptklassen von der Datenhaltung über ein DAO. Verwenden Sie dabei die folgende Schnittstelle:

---

<sup>1</sup>Die alte `Menu`-Klasse werden Sie nicht mehr benötigen.

```
public interface IDao {  
    void speichern(List<Medium> liste) throws PersistenzException;  
    List<Medium> laden() throws PersistenzException;  
}
```

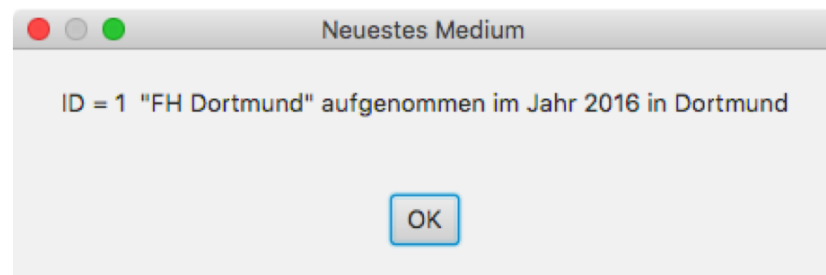
Dabei werden technische Ausnahmen auf die Ausnahme `PersistenzException` abgebildet.

10. Implementieren Sie die Schnittstelle `IDao`, indem Sie die Serialisierung aus dem Praktikum 6 einbinden. Schreiben Sie das statische Attribut für die ID-Verwaltung nicht in den Ausgabestrom. Ermitteln Sie später dieses Attribut über die Länge der geladenen Liste (falls noch nicht geschehen).

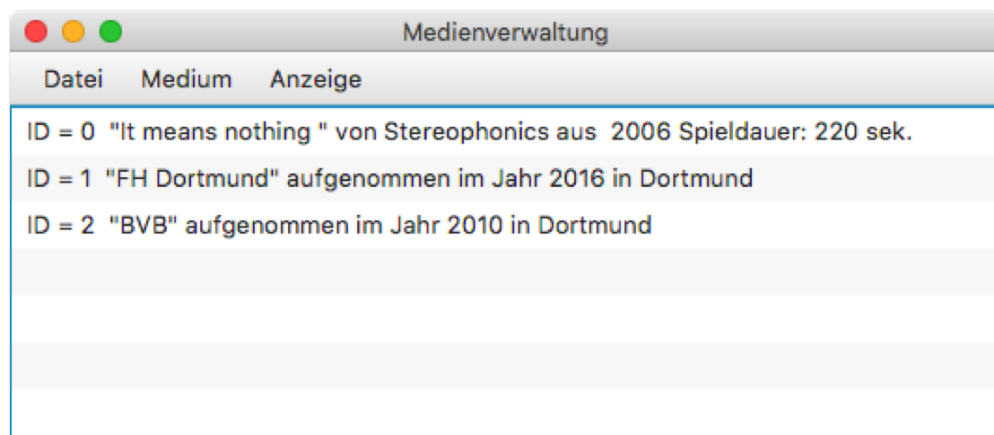
Sorgen Sie abschließend für die Ereignisbehandlung.

11. Versehen Sie das Menü aus dem Praktikum 9 mit einer geeigneten Ereignisbehandlung. Über die Menüs sollen die folgenden Funktionen aufgerufen werden können (die Funktionen haben Sie bereits in den bisherigen Praktika implementiert):
  - (a) *Laden* : Eine gespeicherte Medienliste wird geladen
  - (b) *Speichern* : Die aktuelle Medienliste wird persistent gespeichert (hier: serialisiert).
  - (c) *Medienliste in Datei schreiben* : Die komplette Medienliste wird in lesbarer Form in eine Datei geschrieben. Der Dateiname wird vorher über ein Fenster abgefragt. Sie können die Hilfsklasse `InputView` verwenden (siehe Ilias).
  - (d) *Audio neu* : Ein neuer Audio-Datensatz wird angelegt. Die Eingabe der Daten erfolgt über eine Instanz der Klasse `AudioErfassungView` (siehe Praktikum 8 und 9).

- (e) *Bild neu*: Ein neuer Bild-Datensatz wird angelegt. Die Eingabe der Daten erfolgt über eine Instanz der Klasse `BildErfassungView` (siehe Praktikum 8 und 9).
- (f) *Erscheinungsjahr*: Das durchschnittliche Erscheinungsjahr der Medien wird berechnet und in einem Fenster angezeigt (Sie können die Hilfsklasse `MessageView` verwenden).
- (g) *Neuestes Medium*: Ein Medium mit dem neuesten Erscheinungsjahr wird gesucht und ausgegeben. Beispiel:



- (h) Zeigen Sie alle verwalteten Medien im Hauptfenster als Liste an. Die Liste soll stets den aktuellen Bestand der Medienverwaltung anzeigen.



Sorgen Sie im Fall von Ausnahmen für eine geeignete Meldung an den Anwender. Achten Sie darauf, die Fachkonzeptklassen geeignet an die GUI und die Datenhaltung anzubinden. Sie können in der GUI-Schicht mit einer zusätzlichen Klasse **Controller** arbeiten, welche den Aufruf der einzelnen Fenster (*Views*) steuert. Der *Controller* kann die Fenster und die benötigten Methoden der Klasse `Medienverwaltung` aufrufen.<sup>2</sup>

---

<sup>2</sup>Die Klassen `InputView` und `MessageView` aus dem Ilias bieten einen einfachen Ersatz für die Swing-Methoden `JOptionPane.showInputDialog` und `JOptionPane.showMessageDialog`.