

Seminar 9

In cadrul seminarului se va implementa paginarea unei liste de filme (atat la nivel de repository, cat si la nivel de GUI).

The screenshot displays a web application interface for managing a movie list. At the top, there are search filters: a 'Year' dropdown menu, a 'Year>' text input, a 'Director' text input, and a 'Title' text input. Below these is a table with the following data:

Id	Title	Director	Year
1	Movie1	D1	1990
2	Movie2	D2	2001
3	Movie3	D3	2024
4	Movie4	D4	2001
5	Movie5	D5	2010

Below the table, there is a pagination section highlighted with a red rectangle. It contains three elements: a 'Previous' button, the text 'Page 1 of 2', and a 'Next' button. At the bottom center, there is a 'Delete' button.

Cerinte:

1. Descarcati arhiva *schelet-seminar9*.
2. Creati tabelul **Movie**(id, title, director, year).
3. Asigurati ca programul ruleaza si in tabel sunt afisate filmele din baza de date.
4. Implementati clasele **Page** si **Pageable** (in pachetul *util/paging*)
5. Implementati clasele si relatiile la nivel de repository conform diagramei UML.
6. Update **MovieService** conform diagramei UML (adaugarea metodelor care returneaza listele paginate)
7. Adaugare elemente grafice pentru paginare (butoane **Previous/Next**, label pentru **Page x of y**) si logica pentru afisarea rezultatelor paginate.

Page<E>	Pageable
m Page (Iterable<E>, int)	m Pageable (int, int)
f elementsOnPage Iterable<E>	f pageSize int
f totalNumberOfElements int	f pageNumber int
p elementsOnPage Iterable<E>	p pageNumber int
p totalNumberOfElements int	p pageSize int

Page: Reprezintă un set de rezultate paginat, care include lista entităților și metadate (de exemplu, numărul total de elemente).

Pageable: Înglobează informații despre paginare, cum ar fi dimensiunea paginii și numărul paginii.

Exemplu de utilizare

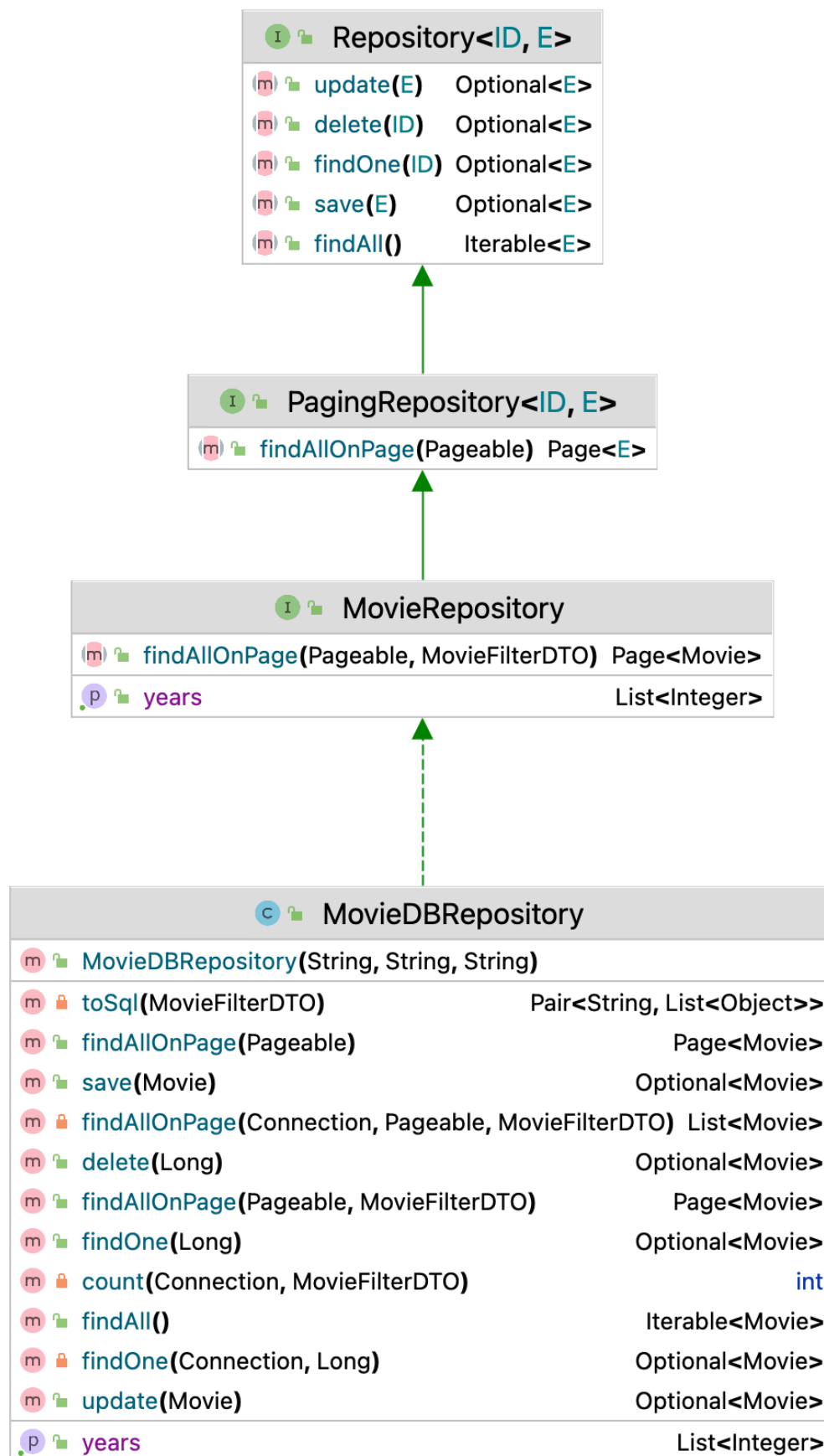
Să presupunem că paginezi o listă de cărți:

1. **Se dorește pagina 2 cu o dimensiune a paginii de 5 elemente:**
Folosește `new Pageable(2, 5)` pentru a descrie cererea.
2. **O interogare extrage cărțile pentru acea pagină:**
De exemplu, numărul total de cărți este 20, iar cărțile de pe pagina 2 sunt: `[Book6, Book7, Book8, Book9, Book10]`.

Creezi un obiect `Page<Book>`:

```
new Page<>(books, 20);
```

3. **Obiectul `Page` oferă:**
 - Cele 5 cărți de pe pagina solicitată (`elementsOnPage`).
 - Numărul total de cărți (`totalNumberOfElements = 20`).



Implementarea paginarii la nivelul repository se face prin intermediul SQL query

SELECT * FROM movies WHERE director = ? LIMIT 5 OFFSET 10;

⇒ Fetches 5 rows (Page Size = 5).

⇒ Starts at row 11 (Offset = $5 \times \text{Page Number} = 2$).

