# PiGE, Lab11: WPF Data Binding

The goal of this task is to create a simple graphical application for displaying information about fantasy novels from the Harry Potter series. Sample data is retrieved from the public API PotterDB and includes details about the novels in the series as well as their individual chapters.

## Overview of the Starter Code

The provided solution consists of two projects:

- `MiNIPotter` a WPF Application and
- `Potter.API` a Class Library project.

Additionally, the folder containing the instructions includes the files `potter.png` and `HarryPotter-Regular.ttf`. References to these resources can be found in the Scoring section.

### Potter.API

The library provides the `IBooksRepository` interface along with an implementation in the form of the `BooksRepository` class:

```
public interface IBooksRepository
{
    Task<IEnumerable<Book>> GetBooksAsync(CancellationToken cancellationToken = default);
}
```

It also defines two simple models that will be used by the graphical part of the application:

```
public sealed record Book(
    Guid Id,
    string Author,
    string Cover,                   // link to the book cover image
    string Dedication,              // author's dedication
    int Pages,                      // number of pages
    DateTime ReleaseDate,           // release date
    string Summary,                 // short description
    string Title,
    ICollection<Chapter> Chapters   // collection of chapters
);

public sealed record Chapter(
    Guid Id,
    string Summary,                 // short description
    string Title,
    double Rating                   // reader rating
);
```

### MiNIPotter

The project contains two key elements that should be used when designing the graphical interface:

- An instance of the `BookViewModel` class should serve as the `DataContext` for the main application window.
- After calling the `LoadBooksAsync` method, its internal `Books` collection will contain data about the novels and chapters.
- The `Styles/Colors/Colors.xaml` file includes `SolidColorBrush` and `Color` resources.
- References to these resources can be found in the Scoring section.

Note:

- The entire task should be completed using `.xaml` files.
- The only exceptions are the implementation of necessary converters and setting the `DataContext` attribute for the main application window.

# Scoring

## Main Window Parameters (`0.5pts`)

- Size: Default is `1200 × 800`, minimum is `1000 × 600`;
- Title: MiNIPotter Library;
- Background: `PrimaryBackgroundBrush`;
- Foreground: `PrimaryForegroundBrush`;
- Icon: `potter.jpg`;

## Main Window Layout

Banner (`0.5pts`)

- Font: `HarryPotter-Regular.ttf`;
- Text: Size `40`, bold, and centered;
- Color: `PrimaryForegroundBrush`;

Loading button (`0.5pts`)

- Background: `SecondaryBackgroundBrush`;
- Foreground: `SecondaryForegroundBrush`;
- Text: Includes 🗲 character, bold, size `16`;
- Clicking the button causes the data to reload;

## Book List (`3pts`)

- Width: Fixed at `400px`;
- Background: `SecondaryBackgroundBrush`;
- Scrollbars:
  - Vertical scrollbar is always visible;
  - Horizontal scrollbar is never visible;
- List Item:
  - Image: Fixed width of `100px`, height maintains original aspect ratio;
  - Title: The text wraps, is bolded and has font size `18`;
  - Author: Font size `10`;
  - Page count & release date:
    - Always positioned at the bottom of the list tile, font size `12`;
    - Release date has a following format: `d MMMM yyyy`;

## Book Details (`1.5pts`)

- Layout: Divided into two equal sections;
- Left Column:
  - A vertical scrollbar appears if the window height is too small (text must not be cut off);
  - Author's dedication: Font size `14`, italic, and `DemiBold`;
  - Short description: Text wraps, font size `14`, italic;
  - Average rating: The number of full stars corresponds to `(int)Math.Clamp(Rating, 0, 10)`, where `Rating` is the arithmetic mean of the individual chapter ratings.

- Right Column:
  - Image: Fills the entire available space;

### Chapter List (**2pts**):

- Background: `SecondaryBackgroundBrush`;
- List Header: `PrimaryForegroundBrush`, font size `20`, bold;
- List Item:
  - Header: Displays the chapter title, font size `14`, `Demibold`;
  - Expandable/Collapsible: Chapters can be expanded or collapsed. Initially, all items are collapsed;
  - Short Description: Text wraps, font size `12`, italic;
  - Rating:
    - Symbol: ★ (empty star ☆ can be found at the bottom of the linked page);
    - Number of full stars: Determined similarly to the method described in Book Details;
    - Stars color calculation:
      - `red = (byte)((1 - (Rating / 10)) * 192);`
      - `green = (byte)((Rating / 10) * 192);`
      - `blue = 0;`

### Hints:

- Packaging Fonts with Applications;
- ListBox Control;
- Expander Control;
- ScrollViewer Control
- Image Control;
- Data conversion
- Merged resource dictionaries