Katalog w DataGrid

Czas: 2 × 45 min | praca indywidualna | Wykład 2.03

Cel: Zbudujesz katalog książek w DataGrid z polskimi nagłówkami, formatowaniem walut, edycją, filtrowaniem tekstem i podsumowaniem pod tabelą.

Po zajęciach potrafisz:

- definiuje kolumny DataGrid różnych typów
- formatuje waluty i daty po polsku w wiązaniach
- filtruje i sortuje dane przez ICollectionView
- aktualizuje podsumowania po zmianach danych

Sprzęt i materiały

- komputer z Windows 11 i dostępem do internetu
- pendrive lub konto w repozytorium do zapisania projektu

Oprogramowanie

- **Visual Studio Community 2026 (obciążenie „Programowanie aplikacji klasycznych platformy .NET")** — darmowe dla uczniów i nauczycieli; w pracowni zainstalowane visualstudio.microsoft.com/pl/downloads
- **.NET 10 SDK (LTS)** — instalowany razem z Visual Studio; sprawdzenie: `dotnet --version` dotnet.microsoft.com/pl-pl/download/dotnet/10.0

Bezpieczeństwo

- Zapisuj projekt w swoim folderze (np. `C:\Users\<login>\source\repos`), a na koniec zajęć skopiuj go na pendrive lub wyślij do repozytorium.
- Nie uruchamiaj programów i pakietów z nieznanych źródeł; pakiety NuGet instaluj tylko te, które wskazuje zadanie.
- Rób przerwy dla oczu (zasada 20-20-20) — praca z projektantem okien wymaga długiego patrzenia w ekran.

Przebieg zadania

1 Dane

Utwórz projekt WPF `Katalog`.

1. Klasa `Ksiazka` z powiadomieniami: Tytuł, Autor, Rok, Cena (decimal), Dostępna (bool), Gatunek (enum).
2. Metoda statyczna `Przyklady()` z co najmniej 8 książkami.
3. W `App.xaml.cs` ustaw język formatowania pl-PL (lekcja 2.03) — bez tego cena pokaże się w dolarach.

2 DataGrid

Kolumny ręcznie (`AutoGenerateColumns="False"`).

1. Kolumny tekstowe Tytuł (szerokość *), Autor, Rok; kolumna Cena ze `StringFormat={}{0:C}`.
2. Kolumna `DataGridComboBoxColumn` dla gatunku i `DataGridCheckBoxColumn` dla dostępności.
3. Naprzemienne kolory wierszy, `CanUserAddRows="False"`, `SelectedItem` związany z modelem.
4. Sprawdź sortowanie kliknięciem nagłówka i edycję komórki dwuklikiem.

3 Filtrowanie

Model okna `KatalogModel`.

```
Widok = CollectionViewSource.GetDefaultView(Ksiazki);
Widok.Filter = o => o is Ksiazka k &&
    (string.IsNullOrWhiteSpace(Szukaj) ||
     k.Tytul.Contains(Szukaj, StringComparison.CurrentCultureIgnoreCase) ||
     k.Autor.Contains(Szukaj, StringComparison.CurrentCultureIgnoreCase));
```

1. `Widok = CollectionViewSource.GetDefaultView(Ksiazki)` i `Widok.Filter` po tytule i autorze (bez rozróżniania wielkości liter).
2. Właściwość `Szukaj` wywołuje `Widok.Refresh()`; DataGrid wiąże się z `Widok`.
3. Domyślne sortowanie po tytule: `SortDescriptions`.

4 Podsumowanie i usuwanie

Pod tabelą liczba książek i łączna wartość (widocznych po filtrze).

1. Zgłaszaj zmianę podsumowań po zmianie kolekcji, filtra i ceny w wierszu.
2. Przycisk Usuń zaznaczoną z pytaniem MessageBox.
3. Sprawdź: filtr „lem”, zmiana ceny w tabeli, usunięcie książki — podsumowanie zawsze aktualne.
4. Zatwierdź pracę w Git.

W sprawozdaniu umieść

- ☐ kod klas i XAML
- ☐ zrzuty: cała lista, filtr „lem”, po edycji ceny
- ☐ wyjaśnienie poprawki języka pl-PL

Pytania kontrolne

1. Dlaczego bez poprawki języka cena wyświetla się w dolarach?
2. Czym różni się filtrowanie widoku od tworzenia nowej listy?
3. Jak dodać kolumnę z wyborem gatunku z listy?
4. Dlaczego zmiana ceny w wierszu nie zgłasza CollectionChanged?

Kryteria oceny

Kryterium	Punkty
Klasa danych i przykładowe dane	3
Kolumny DataGrid i formatowanie pl-PL	5
Filtr i sortowanie przez ICollectionView	5
Podsumowanie i usuwanie	4
Pytania	3
Razem	**20**

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).