Własny wygląd kontrolek

Czas: 2 × 45 min | praca indywidualna | Wykład 2.10

Cel: Zaprojektujesz własne szablony przycisku i przełącznika ze wszystkimi stanami oraz pokolorujesz wiersze tabeli wyzwalaczami danych.

Po zajęciach potrafisz:

- tworzy ControlTemplate z TemplateBinding i ContentPresenter
- obsługuje stany IsMouseOver, IsPressed, IsKeyboardFocused, IsEnabled
- stosuje DataTrigger i MultiDataTrigger
- przechowuje style w słowniku zasobów aplikacji

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 Przycisk

Projekt WPF `Szablony`, plik `Styles.xaml` dołączony w App.xaml.

1. Styl `PrzyciskZaokraglony` z szablonem: Border z CornerRadius i ContentPresenter.
2. Wyzwalacze: najechanie (jaśniejszy), wciśnięcie (ciemniejszy), fokus klawiatury (obramowanie), nieaktywny (przezroczystość 0,4).
3. Sprawdź, że kolor tła da się zmienić na przycisku (`Background="#B42318"`) dzięki TemplateBinding.
4. Przejdź po przyciskach klawiszem Tab — fokus musi być widoczny.

2 Przełącznik

Styl `Przelacznik` dla CheckBox.

```
<Trigger Property="IsChecked" Value="True">
  <Setter TargetName="Tor" Property="Background" Value="#0E7490" />
  <Setter TargetName="Galka" Property="HorizontalAlignment" Value="Right" />
</Trigger>
```

1. Tor (Border 44 × 22, zaokrąglony) i gałka (Ellipse).
2. IsChecked=True: kolor toru i gałka po prawej.
3. Etykieta przez ContentPresenter; stan nieaktywny.

3 Wiersze danych

Rekord `Zadanie(Tytul, Status, Priorytet)` z enumami.

1. Styl `WierszZadania` dla DataGridRow: Opóźnione — czerwone tło, W trakcie — żółte, Zakończone — szary tekst.
2. MultiDataTrigger: opóźnione o wysokim priorytecie — pogrubienie.
3. Przełącznik „Tylko opóźnione” filtruje listę.
4. Wyeksportuj kopię domyślnego szablonu Button (projektant → Edytuj szablon → Edytuj kopię) i zapisz nazwy 3 elementów, które w nim znalazłeś.
5. Zatwierdź pracę w Git.

W sprawozdaniu umieść

- ☐ Styles.xaml
- ☐ zrzuty przycisków w różnych stanach i przełącznika
- ☐ zrzut tabeli z kolorowaniem
- ☐ elementy znalezione w domyślnym szablonie

Pytania kontrolne

1. Czym różni się styl od szablonu?
2. Do czego służy TemplateBinding?
3. Dlaczego trzeba samodzielnie obsłużyć fokus klawiatury w szablonie?
4. Kiedy użyjesz MultiDataTrigger?

Kryteria oceny

Kryterium	Punkty
Szablon przycisku ze stanami	6
Przełącznik	4
DataTrigger i MultiDataTrigger	5
Analiza domyślnego szablonu i pytania	5
Razem	**20**

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