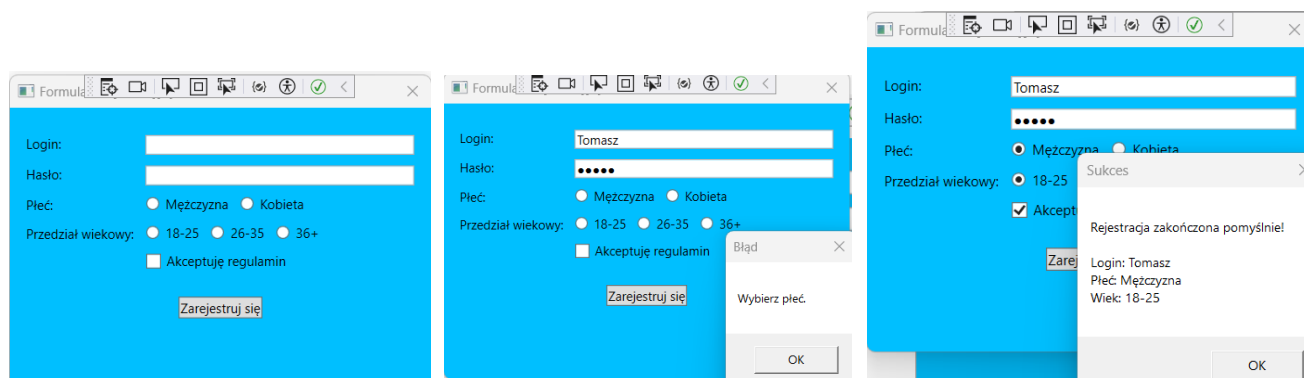


Tworzenie Formularza Rejestracyjnego w WPF

Twoim zadaniem jest stworzenie aplikacji WPF, która wyświetli formularz rejestracyjny z walidacją danych. Formularz będzie posiadał pola tekstowe, przyciski radiowe oraz checkbox, a także przycisk do zatwierdzenia rejestracji. Po wypełnieniu formularza, dane mają zostać przetworzone i zweryfikowane. W przypadku brakujących informacji, aplikacja powinna wyświetlić odpowiednie komunikaty błędów.



Wymagania funkcjonalne

1. Pola Formularza:

- **Pole tekstowe** do wprowadzenia loginu.
- **Pole hasła** do wprowadzenia hasła.
- **Grupa przycisków radiowych** do wyboru płci (opcja "Mężczyzna" i "Kobieta").
- **Grupa przycisków radiowych** do wyboru przedziału wiekowego (opcje: „18-25”, „26-35”, „36+”).
- **Checkbox** do potwierdzenia akceptacji regulaminu.
- **Przycisk rejestracji** do zatwierdzenia formularza.

2. Logika Walidacji:

- Wprowadzenie loginu i hasła jest wymagane.
- Wybór płci jest wymagany.
- Wybór przedziału wiekowego jest wymagany.
- Akceptacja regulaminu jest wymagana.

3. Komunikaty o błędach:

- Jeśli użytkownik nie wprowadzi loginu lub hasła, aplikacja powinna wyświetlić komunikat: „Wprowadź login i hasło.”
- Jeśli nie zostanie wybrana płeć, aplikacja powinna wyświetlić komunikat: „Wybierz płeć.”
- Jeśli nie zostanie wybrany przedział wiekowy, aplikacja powinna wyświetlić komunikat: „Wybierz przedział wiekowy.”
- Jeśli nie zostanie zaznaczona zgoda na regulamin, aplikacja powinna wyświetlić komunikat: „Musisz zaakceptować regulamin.”

4. Zakończenie rejestracji:

- Jeśli wszystkie pola są poprawnie wypełnione, aplikacja powinna wyświetlić komunikat „Rejestracja zakończona pomyślnie!” wraz z podsumowaniem wprowadzonych danych (login, płeć, przedział wiekowy).

Szkielet kodu aplikacji

1. **Interfejs Użytkownika (XAML):** Skonfiguruj formularz z polami tekstowymi, przyciskami radiowymi i checkboxem.
2. **Logika Weryfikacji:** Dodaj obsługę zdarzenia kliknięcia przycisku rejestracji, aby sprawdzić, czy wszystkie pola są wypełnione.

Tworzenie Formularza Zmiany Stylu Tekstu

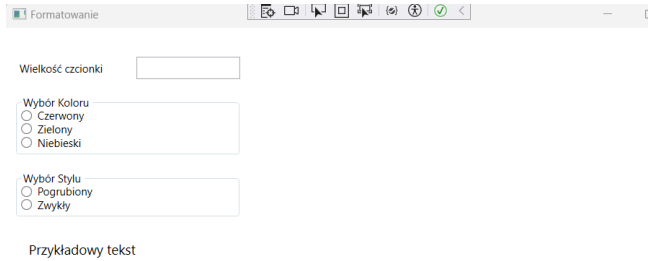
Twoim zadaniem jest stworzenie aplikacji WPF, która umożliwi użytkownikowi zmianę stylu tekstu na podstawie zaznaczonego przycisku radiowego. Zamiast przycisków (Button), aplikacja powinna używać RadioButton do wyboru koloru i stylu tekstu. Dodatkowo, wartość tekstu wprowadzona przez użytkownika w polu tekstowym będzie miała wpływ na wielkość czcionki.

Wymagania funkcjonalne

1. **Pola Formularza:**
 - **Pole tekstowe** do wprowadzenia wartości procentowej, która będzie wpływać na wielkość czcionki (np. 50% dla małej czcionki, 200% dla dużej).
 - **Grupa przycisków radiowych** do wyboru koloru tekstu (opcje: „Czerwony”, „Zielony”, „Niebieski”).
 - **Grupa przycisków radiowych** do wyboru stylu czcionki (opcje: „Pogrubienie” i „Zwykły”).
 - **Pole tekstowe**, które zmienia styl czcionki na podstawie dokonanych wyborów.
2. **Logika Walidacji:**
 - Po zaznaczeniu odpowiednich RadioButton, tekst w TextBlock powinien zmieniać:
 - **Kolor** w zależności od zaznaczonego RadioButton (Czerwony, Zielony, Niebieski).
 - **Styl czcionki** na pogrubiony lub normalny, w zależności od wybranego stylu.
 - **Wielkość czcionki** zależnie od wartości wprowadzonej w polu tekstowym (np. 100% to domyślna wielkość czcionki, 50% to mniejsza czcionka).
3. **Specyfikacja:**
 - Przy wybraniu opcji „Czerwony” w grupie kolorów, tekst powinien stać się czerwony.
 - Przy wybraniu opcji „Zielony” w grupie kolorów, tekst powinien stać się zielony.
 - Przy wybraniu opcji „Niebieski” w grupie kolorów, tekst powinien stać się niebieski.
 - Styl czcionki powinien zmieniać się na **pogrubiony** lub **zwykły**, w zależności od zaznaczonego RadioButton.
 - Wielkość czcionki będzie zależna od wartości wprowadzonej przez użytkownika w polu tekstowym.

Przykładowe Wskazówki Implementacyjne

1. Zamiast przycisków (Button) do zmiany koloru, użyj RadioButton do wyboru koloru.
2. Dodaj dwa grupy RadioButton:
 - o **Pierwsza grupa** do wyboru koloru tekstu: Czerwony, Zielony, Niebieski.
 - o **Druga grupa** do wyboru stylu tekstu: Pogrubiony lub Normalny.



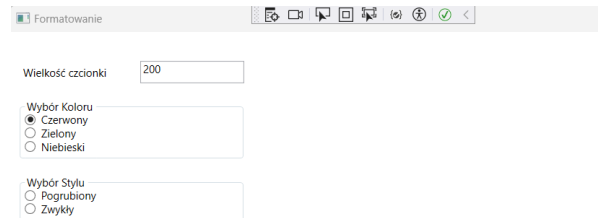
Formatowanie

Wielkość czcionki:

Wybór Koloru:
☐ Czerwony
☐ Zielony
☐ Niebieski

Wybór Stylu:
☐ Pogrubiony
☐ Zwykły

Przykładowy tekst



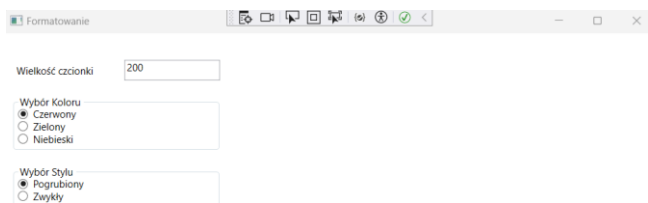
Formatowanie

Wielkość czcionki:

Wybór Koloru:
☒ Czerwony
☐ Zielony
☐ Niebieski

Wybór Stylu:
☐ Pogrubiony
☐ Zwykły

Przykładowy tekst



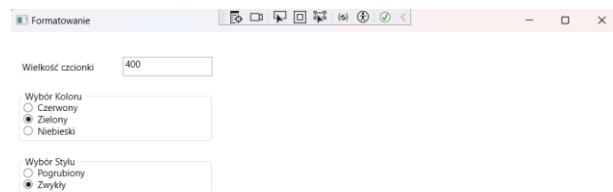
Formatowanie

Wielkość czcionki:

Wybór Koloru:
☒ Czerwony
☐ Zielony
☐ Niebieski

Wybór Stylu:
☒ Pogrubiony
☐ Zwykły

Przykładowy tekst



Formatowanie

Wielkość czcionki:

Wybór Koloru:
☐ Czerwony
☒ Zielony
☐ Niebieski

Wybór Stylu:
☐ Pogrubiony
☒ Zwykły

Przykładowy tekst

Formatowanie prostokąta za pomocą RadioButtonów

W aplikacji WPF (Windows Presentation Foundation) stwórz program, który będzie umożliwiał użytkownikowi zmianę koloru oraz rozmiaru prostokąta (Rectangle) na ekranie za pomocą przycisków radiowych (RadioButton).

Szczegóły zadania:

1. **Stwórz okno główne aplikacji** (plik MainWindow.xaml), które będzie zawierało następujące elementy:
 - Prostokąt (Rectangle)
 - Grupa przycisków radiowych (RadioButton) do wyboru koloru prostokąta:
 - RedRadioButton — zmienia kolor na czerwony.
 - GreenRadioButton — zmienia kolor na zielony.
 - BlueRadioButton — zmienia kolor na niebieski.
 - Grupa przycisków radiowych do wyboru rozmiaru prostokąta:
 - RB200x200 — ustawia rozmiar prostokąta na 200 x 200 pikseli.
 - RB400x200 — ustawia rozmiar prostokąta na 400 x 200 pikseli.
 - RB100x300 — ustawia rozmiar prostokąta na 100 x 300 pikseli.
2. **Zaimplementuj logikę zmiany koloru i rozmiaru** w pliku MainWindow.xaml.cs.
 - Metoda `color_checked()` powinna reagować na zmianę zaznaczenia przycisków radiowych do wyboru koloru i ustawiać kolor prostokąta (`rect.Fill`).
 - Metoda `size_checked()` powinna reagować na zaznaczenie przycisków radiowych do wyboru rozmiaru i ustawiać odpowiednie wymiary prostokąta (`rect.Width` oraz `rect.Height`).
 - Metoda `reset()` przywraca ustawienia wyjściowe dla rectangle i radioButtons

