

1. Stwórz nowy projekt o typie WPF Application – Visual C#.
2. Przejrzyj pliki projektu z poziomu Visual Studio.
 - App.xaml i App.xaml.cs
 - MainPage.xaml i MainPage.xaml.cs
3. Przejdź do pliku MainPage.xaml w trybie Design. Otwórz okno Toolbox i przypnij je dziś do końca ćwiczeń na stałe do VS (View->Toolbox lub Ctrl+Alt+X).
4. Metodą przeciągnij i upuść umieść Button na środku okna MainPage.xaml. Przejdź do zakładki XAML i sprawdź jaki kod został wygenerowany automatycznie.
5. Zmień właściwość Content na „Hello World”. Skompiluj i uruchom program.
6. Za pomocą kursora myszy zmień położenie elementu Button. Potem ręcznie zmodyfikuj w kodzie XAML właściwość Margin. Co się stanie jak będą tylko dwa parametry? a co będzie jeśli tylko jeden?
7. Naciśnij Button prawym klawiszem myszy. Wybierz opcję Layout i Reset Margin. Gdzie przemieści się przycisk?
8. Teraz zmodyfikuj właściwość Margin z okna Properties, Layout.
9. Dla elementu Button zmodyfikuj poćwicz zmianę parametru VerticalAlignment i HorizontalAlignment na 3 sposoby (Designer, kod XAML, okno Properties). Na koniec wyśrodkuj Button i wyzeruj marginesy.
10. Dla elementu Button w oknie Properties w polu Name wpisz Przycisk. Co się zmieniło w kodzie?
11. Na oknie Design kliknij dwa razy na Button. Powinna wygenerować Ci się metoda Przycisk_Click. W jej wnętrzu wpisz:
`MessageBox.Show("Witaj Świecie");`
Potem skompiluj i uruchom program.
12. Do aplikacji dodaj dwa elementy typu TextBox. Nazwij je Liczba1 i Liczba2. W właściwości Text ustaw odpowiedniej „Podaj pierwszą liczbę” i „Podaj drugą liczbę”.
13. Zmień właściwość Content elementu Button na Dodaj.
14. Zmodyfikuj wnętrze metody Przycisk_Click następująco:
`double liczba1=Convert.ToDouble(Liczba1.Text);`

```
double liczba2 = Convert.ToDouble(Liczba2.Text);  
double wynik = liczba1 + liczba2;  
MessageBox.Show(wynik.ToString());
```

Następnie skompiluj i uruchom program.

15. Uzupełnione elementy TextBox nie są dobrym rozwiązaniem za każdym razem. Usuń z nich właściwość Text.

16. Z lewej strony TextBox umieść odpowiednie dwa elementy Label. Content ustaw jako Pierwsza Liczba i Druga Liczba. Zadbaj o dobry wygląd aplikacji po kompilacji.

17. Cały czas MessageBox nie wygląda za fajnie. Pod elementem Button Dodaj umieść Label. Usuń w niej Content a nazwę ustaw na Wynik.

18. Zmodyfikuj wewnątrz metody Przycisk_Click. Usuń linijkę z MessageBoxem. Zamiast tego wpisz:

```
Wynik.Content = wynik.ToString();
```

19. Wykorzystaj konsolowy kalkulator stworzony przy ćwiczeniach o wyjątkach i stwórz podobną aplikację z użyciem WPF. Kalkulator ma wyłapywać wyjątki typu:

- dzielenie przez zero i zwrócenie informacji o błędzie (a nie nieskończoności)
- wyjście poza zakres zadeklarowanego typu
- błędny przecinek lub kropka
- wprowadzenie znaków innych niż dopuszczalne np. liter lub zostawienie pustego łańcucha w TextBoxie

Przeanalizuj też co się zmieni, gdybyśmy we wnętrzu metody Przycisk_Click użyć innego typu np. float, decimal, int... Co się zmieni jeśli zamiast konwertowania użylibyśmy parsowania? Przetestuj też konstrukcję z użyciem TryParse np.

```
double liczba1;  
Double.TryParse(Liczba1.Text, out liczba1);
```

20. Rozbuduj kalkulator o funkcję Memory.