

Dokumentacja projektu – zaawansowany kalkulator ze zmiennymi

1. Forma główna - **CalculatorMain**

- 1.1. Zawiera główny interfejs programu, który wywołuje formę pomocniczą, gdy jest to potrzebne.
- 1.2. Zmienne w nim występujące to zmienne pomocnicze, służące do operowania przyciskami:
 - 1.2.1. FinalValue – zmienna typu double, przechowuje końcowy wynik,
 - 1.2.2. ResultValue – zmienna typu double, przechowuje pierwszy argument - operanda,
 - 1.2.3. CalcOperator – zmienna typu string, przechowuje operator,
 - 1.2.4. IsCalcOperatorActive – zmienna typu boolean, true - jeżeli wcisnęliśmy operator, false w. p. p.,
 - 1.2.5. Button – zmienna typu Button, używana do przekazania wartości przycisku do ResultValue i FinalValue,
 - 1.2.6. Variable – zmienna typu Variable – jak wyżej.
- 1.3. Kontrolki Windows Forms:
 - 1.3.1. LabelCurrentExpr – label, przechowuje aktualne wyrażenie,
 - 1.3.2. TextBoxRes - pole tekstowe, w którym pojawiają się operandy, operatory i wynik,
 - 1.3.3. Przyciski - ogólnie, wszystkie przyciski z formy, do których przypisane są konkretne eventy (niektóre wspólne, uniwersalne).
- 1.4. Metody:
 - 1.4.1. BtnArg_Click - służy do wczytywania przycisków,
 - 1.4.2. BtnVar_Click, btnConst_Click, btnOp_Click, btnBinOp_Click – j. w.,
 - 1.4.3. BtnEq_Click - służy do zwracania wyniku,
 - 1.4.4. BtnCE_Click, btnC_Click – czyszczenie okna.

2. Forma pomocnicza – CalculatorValueForVariables

- 2.1. Zawiera pomocniczy interfejs, wywoływany podczas naciśnięcia któregoś z przycisków dla zmiennych, służy do wpisania wartości, jaką zmienna ma przyjąć.
- 2.2. Zmienne w nim występujące - jest tylko jedna taka zmienna: varValue – publiczna, typu double. Przekazywana jest do głównej formy, aby móc już operować na wartości, a nie symbolu.
- 2.3. Kontrolki WF:
 - 2.3.1. TextBoxVarValue – pole tekstowe, gdzie wpisujemy wartość zmiennej,
 - 2.3.2. BtnEnter - służy do zatwierdzania, przekazywania i zamykania formy,
 - 2.3.3. BtnDelete – usuwa błędny wpis,
 - 2.3.4. BtnComa – przecinek, aby wpisać liczby zmiennoprzecinkowe. Zabezpieczony przed wielokrotnym klikaniem,
 - 2.3.5. BtnNeg – dla ujemnych wartości, również zabezpieczony.
- 2.4. Metody:
 - 2.4.1. VarValEntry_Click – dopisuje do pola TextBoxVarValue wartość przycisku,
 - 2.4.2. BtnVarValueEnter_Click – przechowuje wartość z pola tekstowego w zmiennej double (parsowanie) i przekazuje, zamykając formę.

- 2.4.3. BtnDelete_Click – usuwa niechciany (ostatni wpisany) znak,
- 2.4.4. BtnComa_Click – dodaje przecinek,
- 2.4.5. BtnNeg_Click – dodaje minus z przodu.

3. Klasa abstrakcyjna - Expression

- 3.1. Klasa, po której dziedziczą wszystkie klasy w programie, ma dwie abstrakcyjne metody – Evaluate i Result, które są nadpisywane przez subklasy.
Obiekty tej klasy mają postać Expression *expression*, gdzie *expression* to dowolne zestawienie subklas takich jak Addition, Sinus, Log, Multiplication itp.
Jej metody są używane w metodach formy CalculatorMain w odpowiednich kategoriach.
- 3.2. Zmienne są pomocnicze, wykorzystywane przez konstruktory dziedziczących subklas, np. dla stałej zmiennej typu double, która jest zwracana przy Evaluate() i zwracana.ToString() w Result().
- 3.3. Metody:
 - 3.3.1. Evaluate() - wylicza wartość wyrażenia i zwraca jako double,
 - 3.3.2. Result() - zwraca wyrażenie w formie tekstowej.