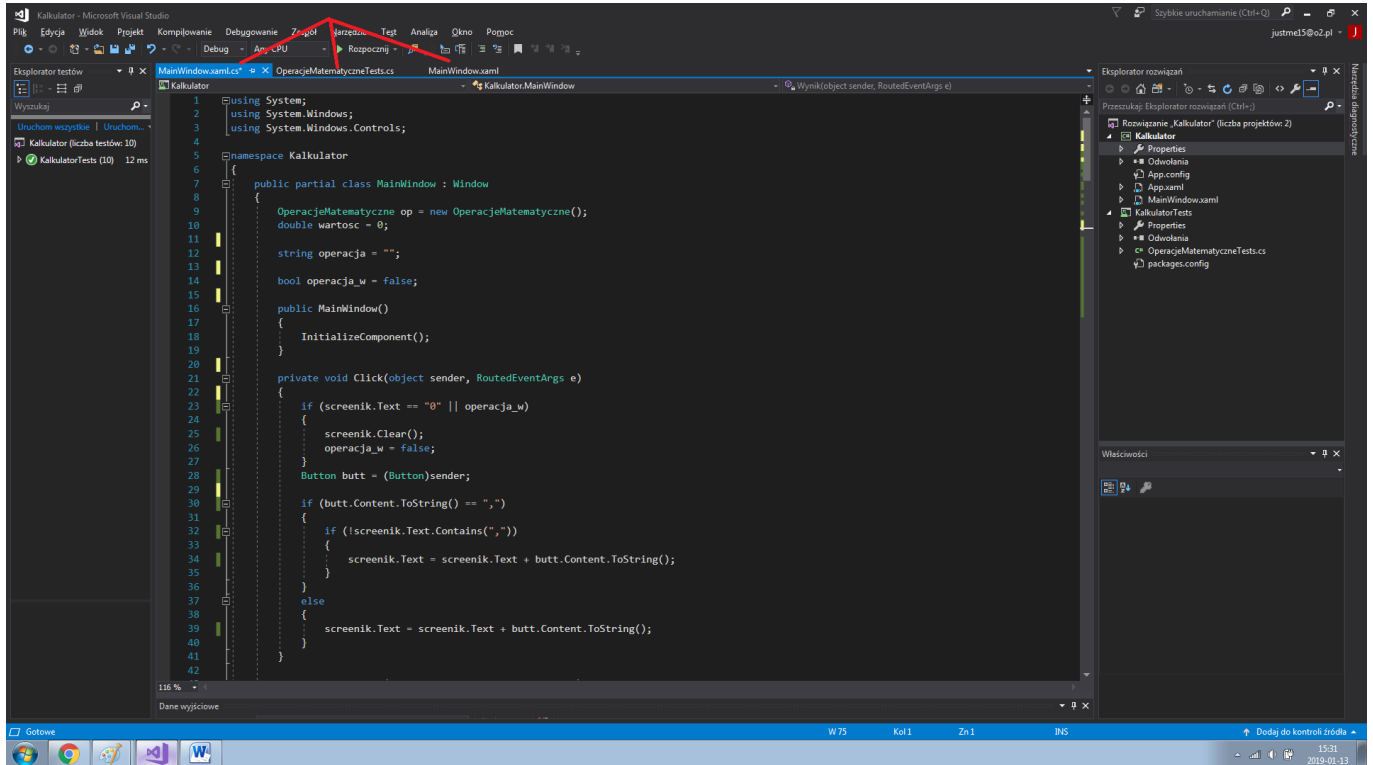


Kalkulator z Graficznym Interfejsem

Tematem naszego projektu było stworzenie zaawansowanego kalkulatora z graficznym interfejsem użytkownika.

Projekt został wykonany w programie Microsoft Visual Studio z pomocą języka programowania C#



W projekcie możemy wyróżnić trzy główne okna zawierające: kod z opisanymi funkcjami, testy funkcji oraz kod definiujący wygląd aplikacji, czyli rozłożenie buttonów, labeli, textboxóww napisy itd.

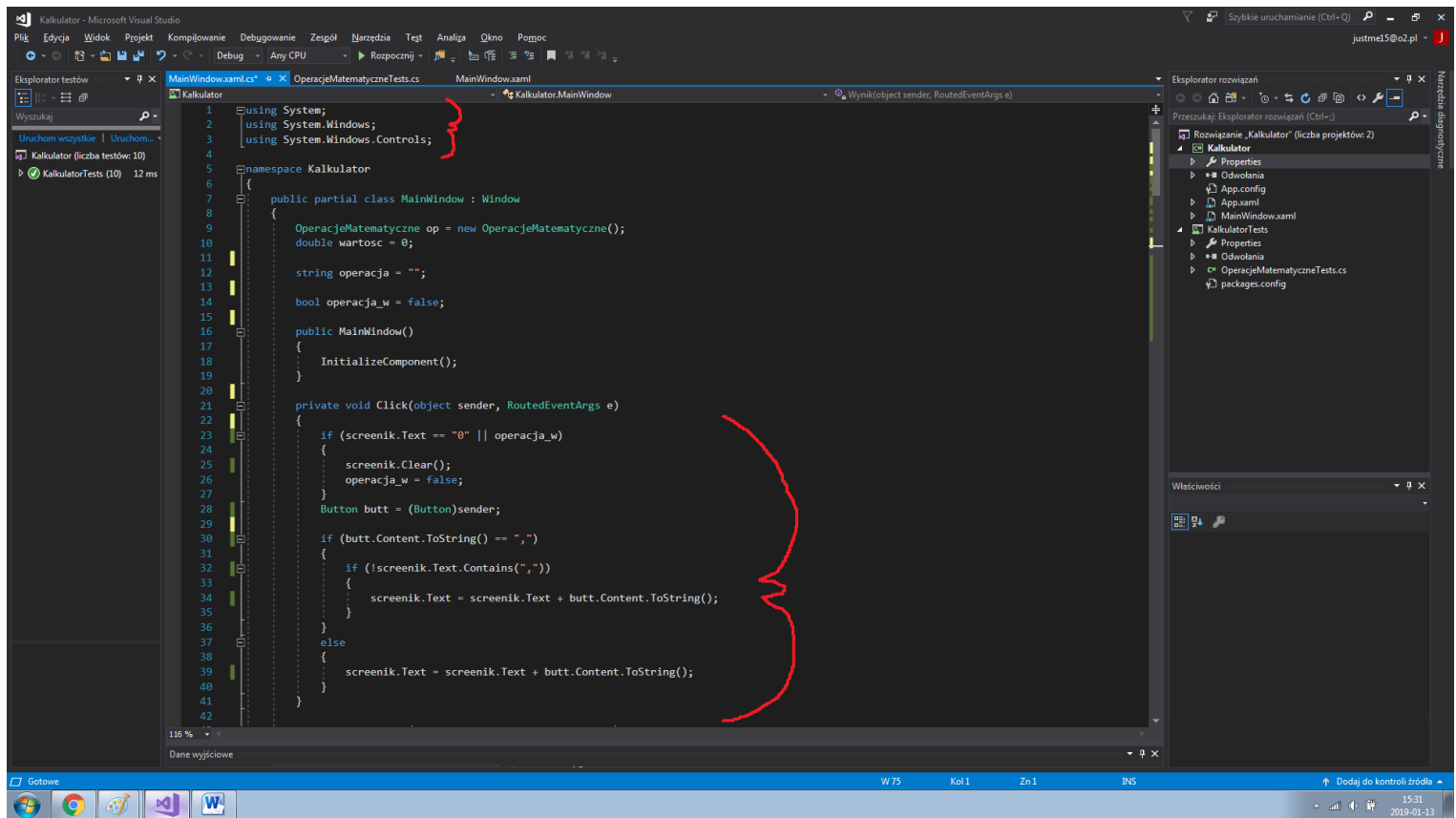
W pierwszej klamrze(↓) mamy przedstawione użyte biblioteki, zaraz potem definiujemy klasę częściową MainWindow w której jest opisane co robi każdy button.

//„String operacja” służy do deklarowania operacji matematycznych.

//„bool operacja_w” sprawdza czy przycisk jest aktywowany.

//Funkcja „Click” obsługuje kliknięcie w przycisk. Jej pierwszy if zajmuje się czyszczeniem wyświetlacza, drugi nie pozwala wprowadzić więcej niż jednego przecinka.Pozwala wprowadzać cyfry do wyświetlenia.

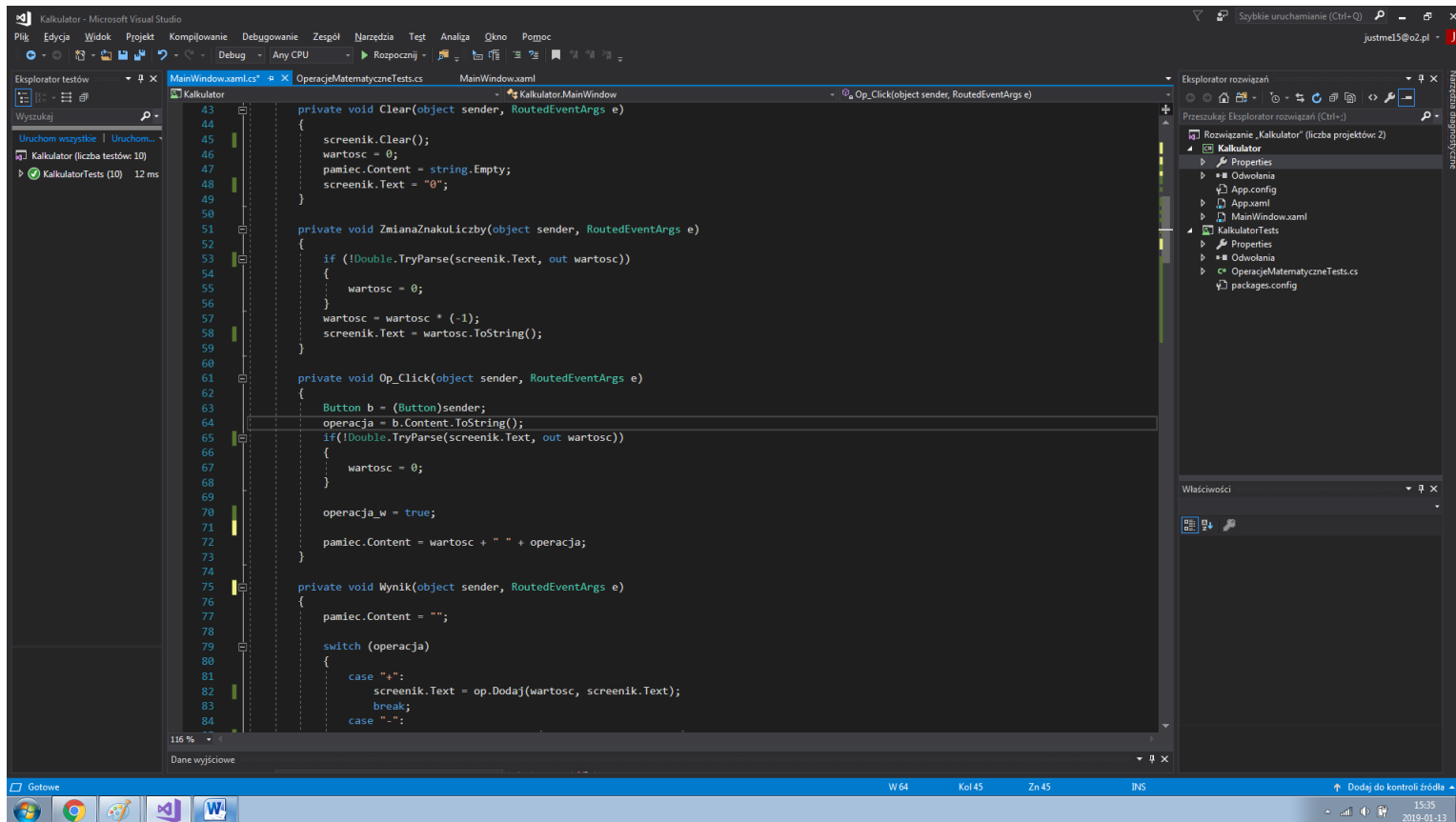
///„screenik” to textbox na którym są wyświetlane liczby.



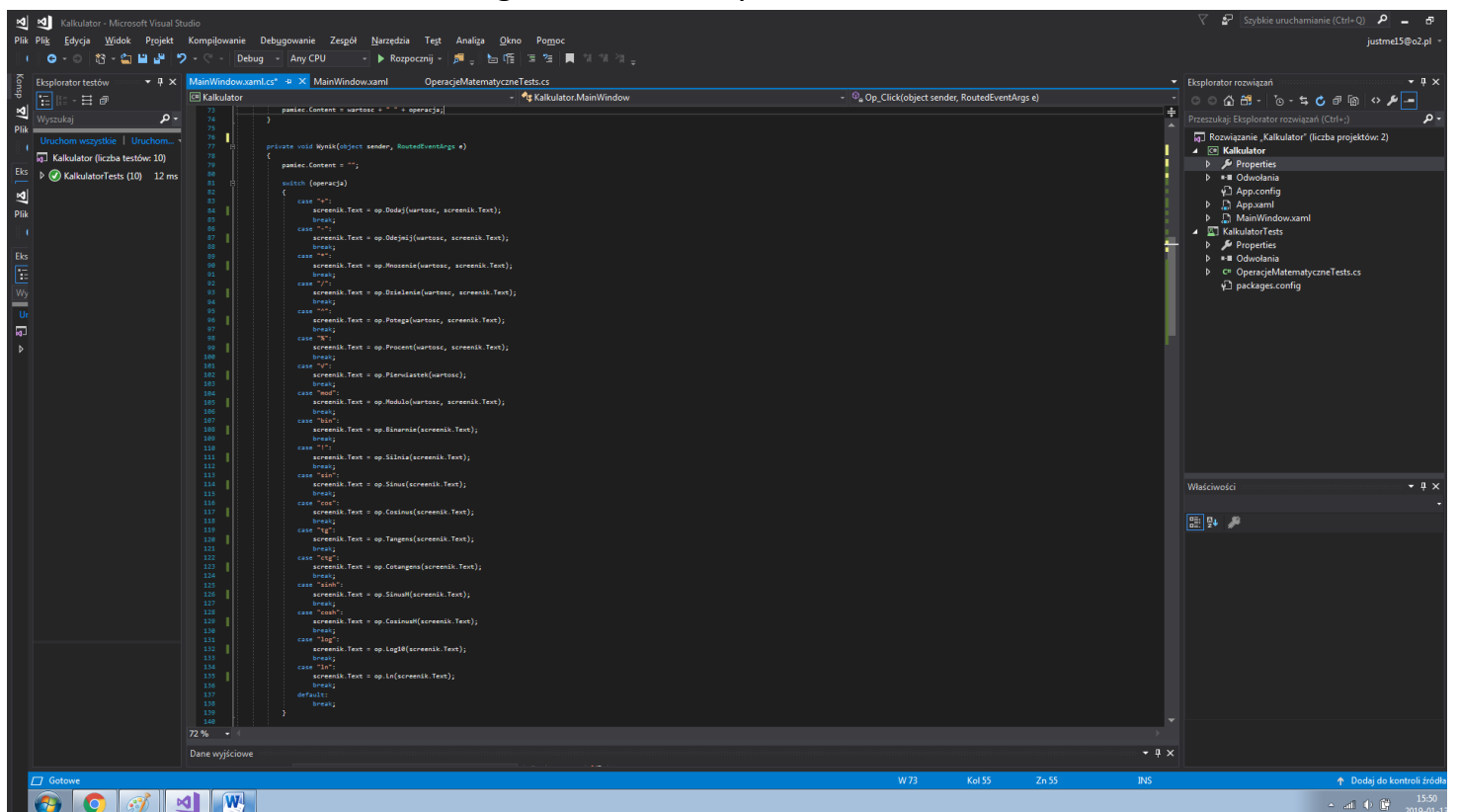
//Funkcja „Clear” czyści wartość labela zostawiając cyfrę „0”

//Op_Click obsługuje kliknięcia na Operatory matematyczne

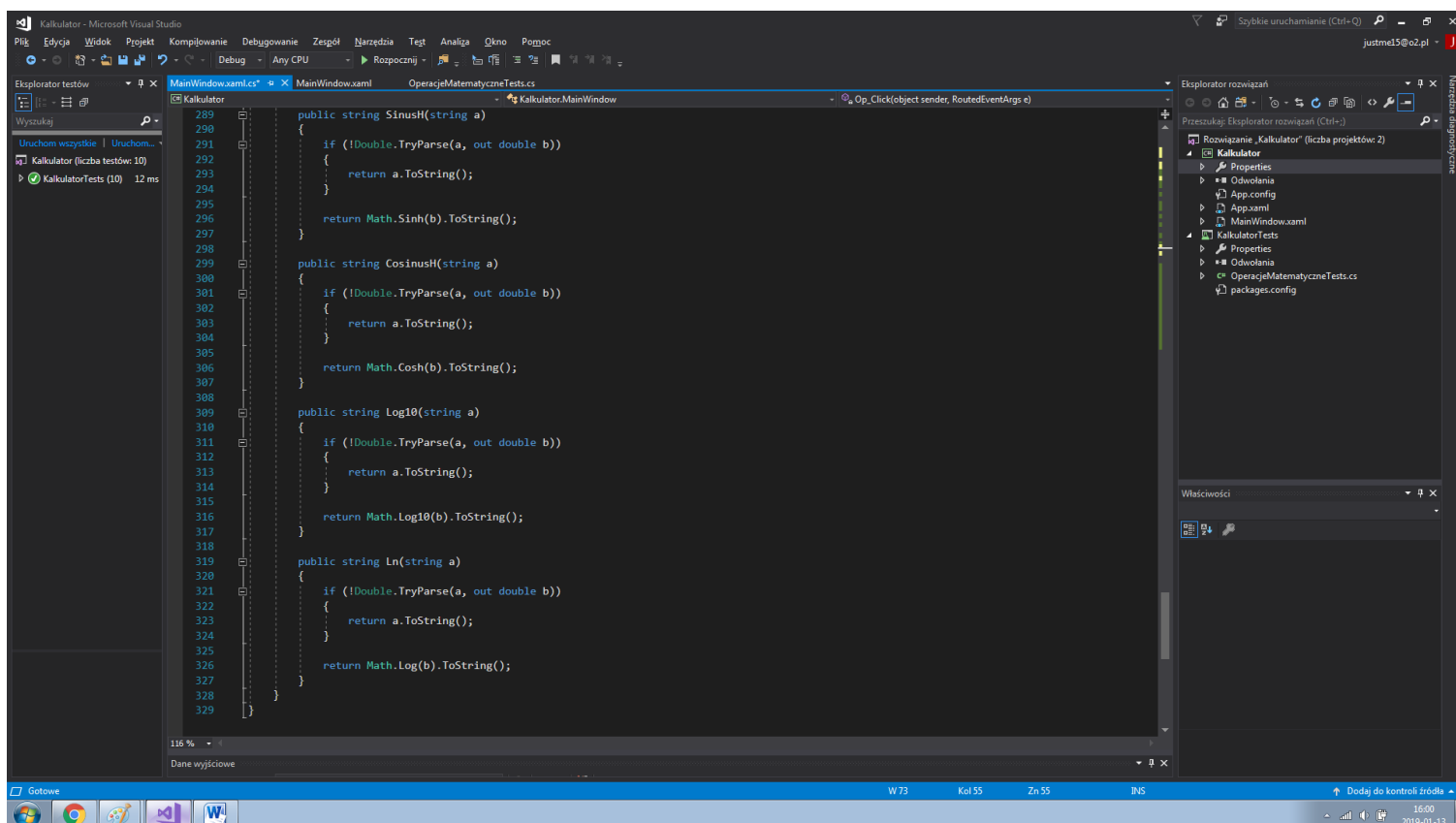
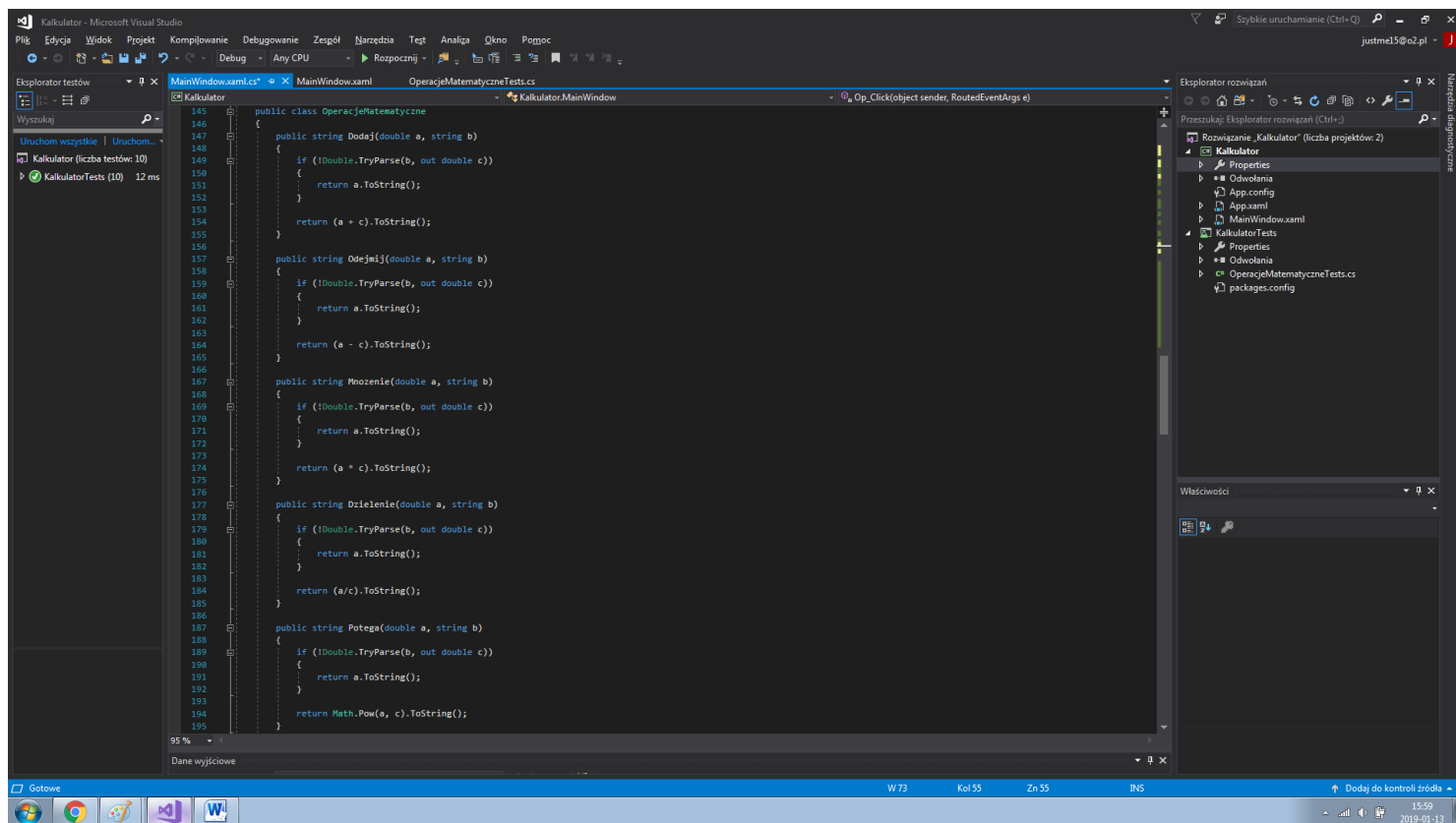
//„pamiec” to zmienna od wyświetlania wartości w labelu



//Funkcja „Wynik” zajmuje się przeprowadzeniem danej Operacji matematycznej w zależności od tego co naciśniemy na kalkulatorze



Dalej jest napisana klasa Operacje matematyczne która zajmuję się przeprowadzeniem żądanej operacji matematycznej, korzystając tutaj z konwertowania i sprawdzania skonwertowania stringa na inta metodą TryParse().

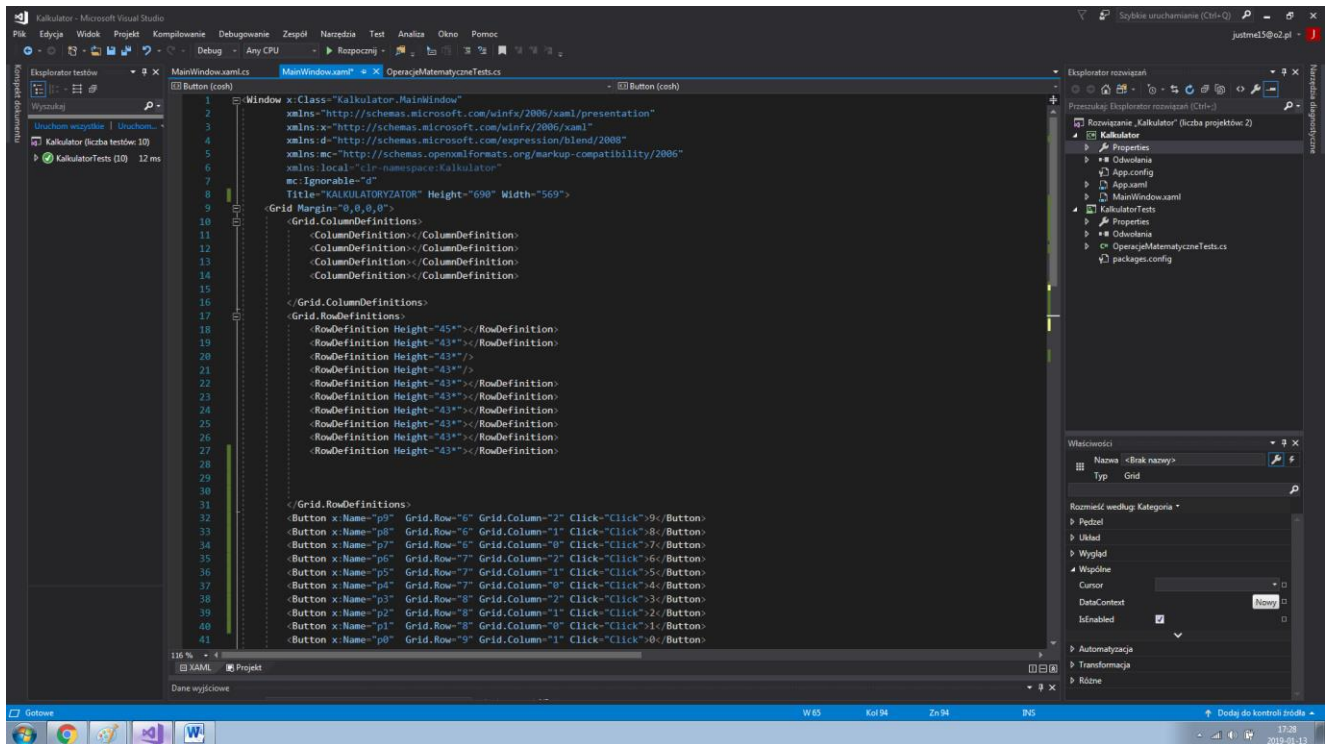


Lista dostępnych operacji:

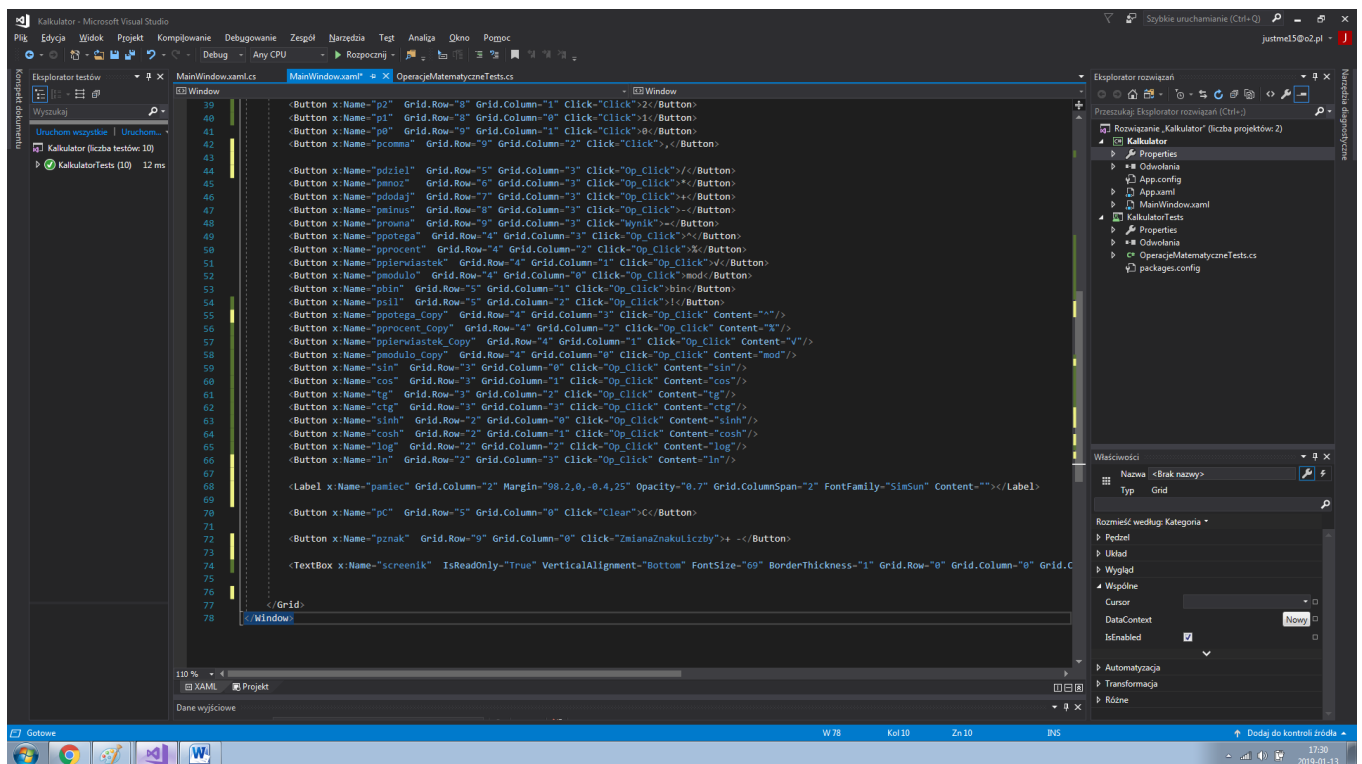
op.Dodaj ("+") → sumuje liczby
op.Odejmij ("-") → odejmuje liczby
op.Mnozenie ("*") → mnoży liczby
op.Dzielenie ("/") → dzieli liczby
op.Potega ("^") → podnosi liczbę do danej potęgi
op.Procent ("%") → wpisujemy liczbę, potem „%” drugą liczbę i uzyskujemy jakim procentem pierwszej liczby jest liczba druga
op.Pierwiastek ("√") → pierwiastkuje liczbę
op.Modulo ("mod") → daje resztę z dzielenia
op.Binarnie ("bin") → podaje daną liczbę w systemie binarnym
op.Silnia ("!") → liczy silnie liczby
op.Sinus ("sin") → liczy sinus liczby (Przyjmuje wartość w radianach)
op.Cosinus ("cos") → liczy cosinus liczby (Przyjmuje wartość w radianach)
op.Tangens ("tg") → liczy tangens liczby (Przyjmuje wartość w radianach)
op.Cotangens ("ctg") → liczy cotangens liczby (Przyjmuje wartość w radianach)
op.SinusH ("sinh") → liczy sinus hiperboliczny
op.Log10 ("log") → liczy logarytm dziesiętny liczby
op.Ln ("ln") → liczy logarytm naturalny liczby (o podstawie „e”)

Tutaj mamy okienko z kodem określającym wygląd naszej aplikacji: tytuł, rozmiar okienka, margines, ilość i wielkość rzędów i kolumn, położenie każdego buttonu, labela i textbxa, opis tego co się dzieje po wciśnięciu buttonu.

Dalej jest wprowadzanie i wyświetlanie cyfr za pomocą funkcji „Click”



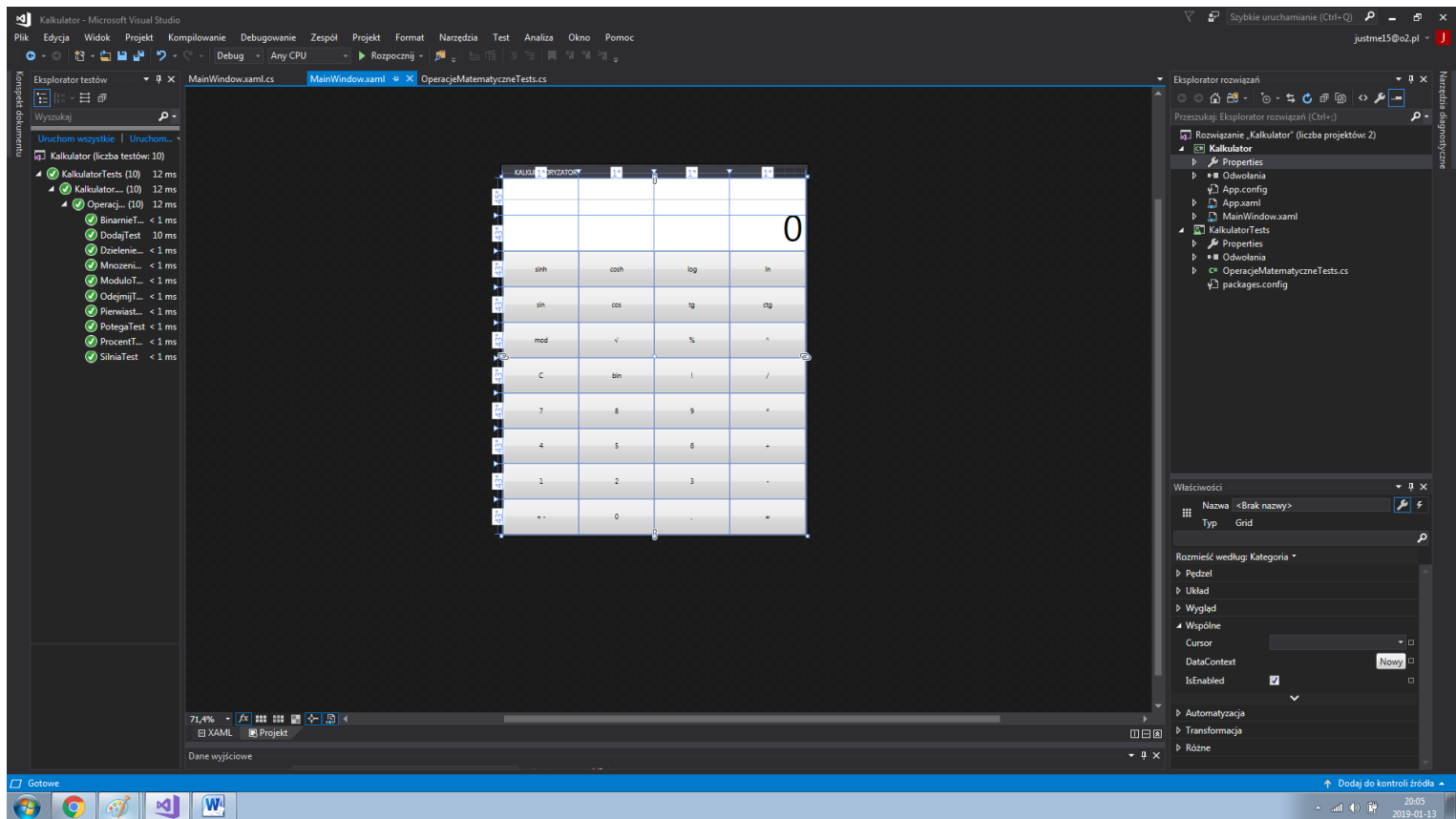
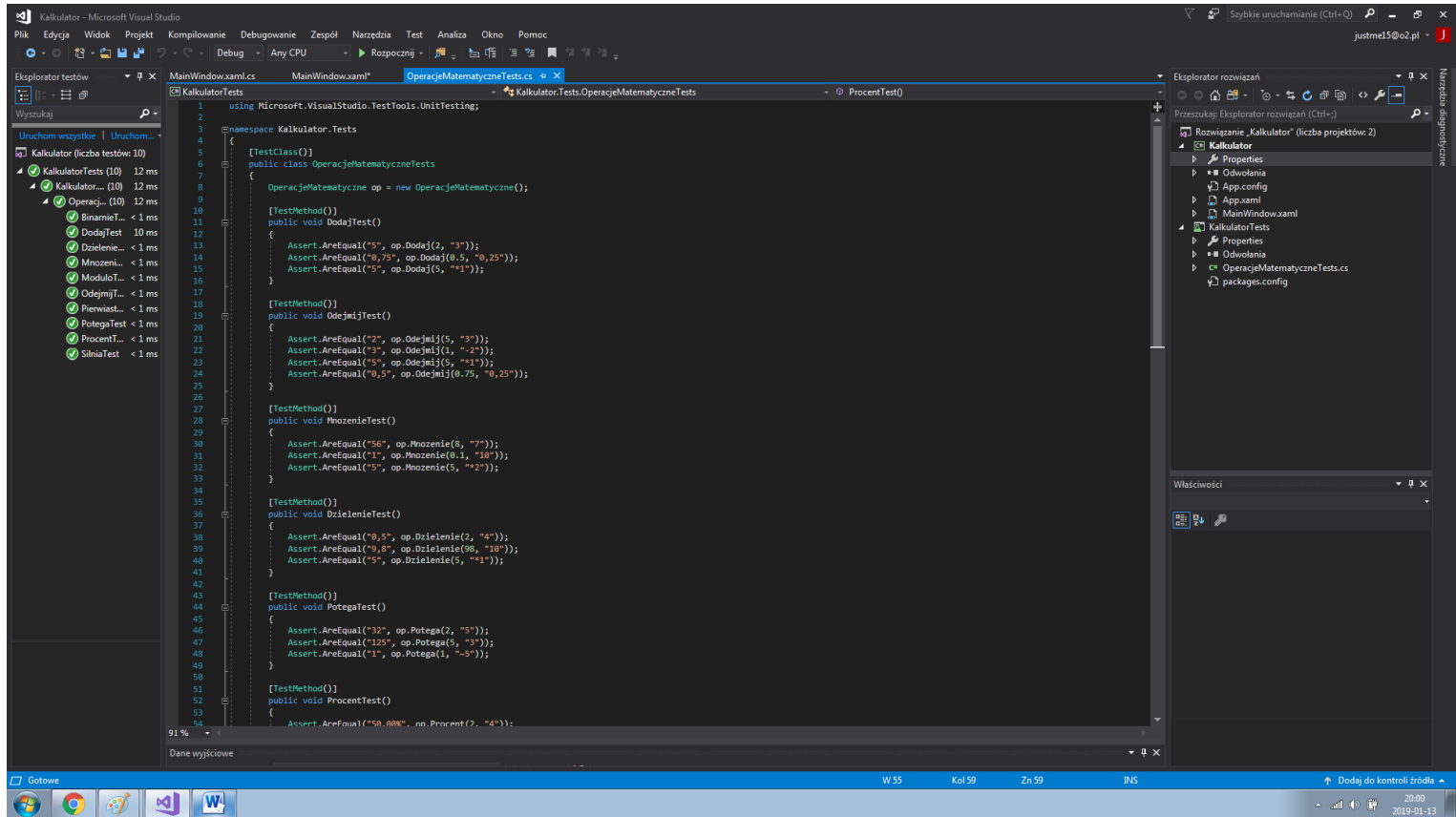
Następnie wszystkie funkcje matematyczne, które się wykonują za pośrednictwem funkcji „Op_Click”.



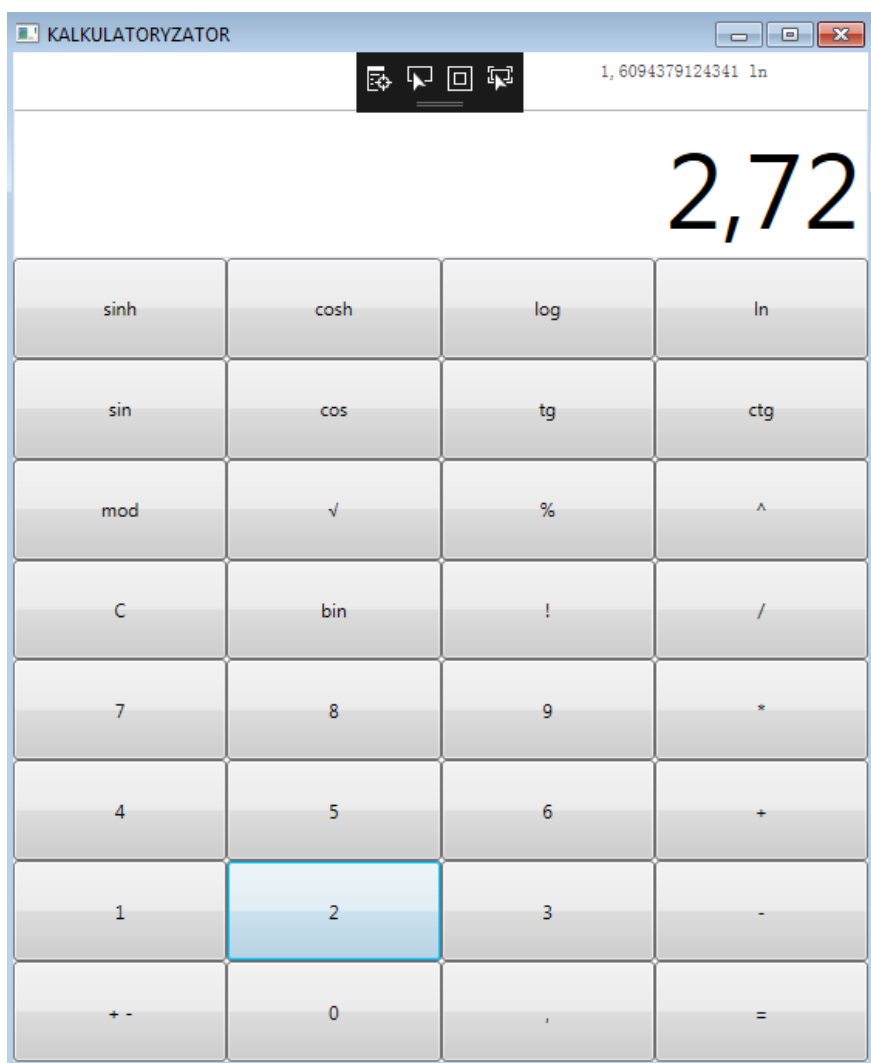
Na końcu są ustawione właściwości dla labela i textboxa czyli mniejszego i większego wyświetlacza

W kolejnej zakładce zostały przeprowadzone testy naszych operacji matematycznych, które potwierdzają działanie funkcji i dają gwarancję, że wszystko dalej działa jak chcemy, po

dopisywaniu kolejnych linii kodu



Wygląd po uruchomieniu aplikacji



//Kalkulator podaje wartości z rozwinięciem dziesiętnym standardowym dla double-a

//Rozmiary kalkulatora można modyfikować, możliwa jest też maksymalizacja okna

//Najpierw należy wprowadzić liczbę potem nacisnąć działanie a żeby uzyskać wynik”=”

//Nazwy funkcji są samoopisujące