

Quadsoft.ExpressionParser

.NET Klassenbibliothek zur Berechnung mathematischer Ausdrücke

Geeignet ab .NET Framework Version 2.0 SP1

Version: 1.0.1

Entwickler: Adrian Jablonski

Website: <http://www.quadsoft.org>

E-Mail: support@quadsoft.org

Inhalt

1	Lizenz
1.1.....	Haftungsausschluss
2	Funktionen
3	Dokumentation
3.1.....	Klassendiagramm
3.2.....	Eigenschaften
3.3.....	Methoden
3.4.....	Programmierbeispiele

1. Lizenz

Sie dürfen diese Bibliothek für Ihre (auch kommerziellen) Projekte unter der Bedingung der Namens und Webseitenennung des Herstellers verwenden, um die Funktionalität eines mathematischen Parsers nutzen zu können. Dabei ist es untersagt, die Datei in eine andere einzukompilieren (Merging), sie zu dekompileieren, oder sie ohne diese Lizenz zu vertreiben.

Beispiel für korrekte Namensnennung in den Credits:

Dieses Programm verwendet die Quadsoft.ExpressionParser Bibliothek von Adrian Jablonski (<http://www.quadsoft.org>)

1.1 Haftungsausschluss

Dieses Werk wird bereitgestellt "wie es ist", ohne jegliche Garantien, Haftungsansprüchen oder sonstigen Bedingungen. Die Nutzung erfolgt auf dem alleinigen Risiko des Benutzers. Ferner kann der Autor nicht die Fehlerfreiheit des Werkes, ja nicht einmal die korrekte Funktionalität gewährleisten. Jeder, der dieses Werk verwendet, muss diese Bedingungen akzeptieren.

2. Funktionen

- Korrekte Auswertung von Klammern, Verschachtelungen und hintereinander vorkommenden Operatoren

- Enthaltene Funktionen: *Sin, Cos, Tan, Asin, Acos, Atan, Sqrt, Ln, Log, Exp*

- Modi für trigonometrische Funktionen: *DEG, RAD*

- Unterstützte Operatoren: +, -, *, /, *Mod*, ^ (binär), ! (unär)

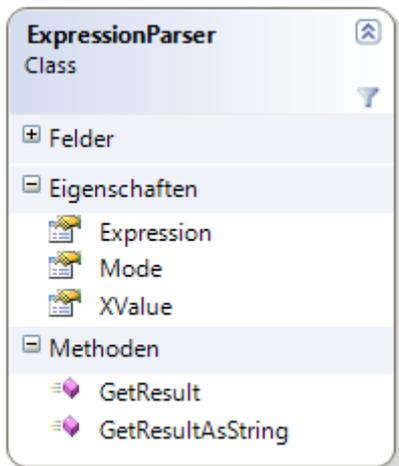
- Enthaltene Konstanten: *e, pi* (oder π)

- Unterstützung einer Laufvariablen "x" im Term (z.B. für Funktionenplotter)

- Unterstützung von "*ans*"

3. Dokumentation

3.1 Klassendiagramm



3.2 Eigenschaften

System.String Expression

Der auszuwertende mathematische Ausdruck. Eventuelle Leerzeichen müssen vorher eliminiert werden.

Quadsoft.TrigonometricMode Mode

Legt den Modus für die Rechnung mit Trigonometrischen Funktionen fest. (DEG = Gradmaß; RAD = Bogenmaß)

System.Double XValue

Wert der laufvariablen „x“, sofern sie im Ausdruck vorkommt.

3.3 Methoden

System.Double GetResult()

Gibt den berechneten Wert des mathematischen Ausdrucks zurück.

System.String GetResultAsString()

Wie GetResult(), aber als String.

3.4 Programmierbeispiele

VB.NET

```
' Neue Instanz des Parsers erstellen
Dim parser As New Quadsoft.ExpressionParser()
' Den Ausdruck festlegen
parser.Expression = "11*2(3-435+435)*3435*sin(43/34-4)"
' Modus für trigonometrische Funktionen
parser.Mode = Quadsoft.TrigonometricMode.Degrees
' Ergebnis berechnen und anzeigen
Console.WriteLine(parser.GetResultAsString())
Console.ReadLine()
```

C#

```
// Neue Instanz des Parsers erstellen
Quadsoft.ExpressionParser parser = new Quadsoft.ExpressionParser();
// Den Ausdruck festlegen
parser.Expression = "11*2(3-435+435)*3435*sin(43/34-4)";
// Modus für trigonometrische Funktionen
parser.Mode = Quadsoft.TrigonometricMode.Degrees;
// Ergebnis berechnen und anzeigen
Console.WriteLine(parser.GetResultAsString());
Console.ReadLine();
```