

C# 2000 Expression Beispielcodes für Konsolen- und Formularanwendung

1. "Hai!" [Konsolenanwendung]

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hai, wie geht's?");
        }
    }
}
```

2. Multiplikation mit sich selbst [Konsolenanwendung]

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            // Ausgabe von "Bitte nennen Sie eine Zahl" in der Konsole
            Console.WriteLine("Bitte nennen Sie eine Zahl");

            // Auslesen der Usereingabe aus der Konsole und
            // Speichern der Inhalts in der String-Variable "ersteZahl"
            string ersteZahl = Console.ReadLine();

            // Umrechnen der Textinformation in der Variable "ersteZahl" in
            // einen Zahlenwert
            // und Speichern in die int-Variable "dieZahl"
            int dieZahl = int.Parse(ersteZahl);

            // Multiplikation von "dieZahl" mit sich selbst
            // und Ausgabe in der Konsole

            Console.WriteLine("Das Ergebnis lautet: " + dieZahl * dieZahl);
        }
    }
}
```

3. Name und Addition zweier Zahlen [Konsolenanwendung]

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hallo, wie ist Ihr Name?");

            string Name = Console.ReadLine();

            Console.WriteLine ("Hallo, " + Name + "bitte eine Zahl?");

            string Zahl1 = Console.ReadLine();

            Console.WriteLine("Bitte noch eine Zahl eingeben");
            string Zahl2 = Console.ReadLine();

            Console.WriteLine("Ergebnis: " + (int.Parse(Zahl1) + int.Parse (Zahl2)));
        }
    }
}
```

4. Name und Zahl größer oder kleiner gleich als 10 [Konsolenanwendung]

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)           // string, da Konsolenanwendung
        {
            Console.WriteLine("Hallo, wie ist Ihr Name?");
            string Name = Console.ReadLine();

            Console.WriteLine("Hallo, " + Name + ", bitte nenne mir eine Zahl?");
            string Zahl1 = Console.ReadLine();

            Console.WriteLine("Bitte noch eine Zahl eingeben");
            string Zahl2 = Console.ReadLine();

            int summe = int.Parse(Zahl1) + int.Parse(Zahl2); // Umrechnen:
                                                            // string in int

            if(summe >= 10)
                Console.WriteLine("Die Zahl ist größer als 10");
            else {
                Console.WriteLine("Die Zahl ist kleiner oder gleich 10");
            }
            Console.WriteLine("Ergebnis: " + (int.Parse(Zahl1) + int.Parse(Zahl2)));
        }
    }
}
```

5. Name und Zahl größer oder kleiner als definierte Zahl [Konsolenanwendung]

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)           // string, da Konsolenanwendung
        {
            Console.WriteLine("Definiere eine beliebige Zahl");
            string defZahl = Console.ReadLine();

            Console.WriteLine("Bitte nenne eine Zahl");
            string Zahl = Console.ReadLine();

            if(int.Parse(Zahl) < int.Parse (defZahl))
                Console.WriteLine("Die Zahl ist kleiner als die definierte Zahl");
            else
            {
                if (int.Parse(Zahl) > int.Parse(defZahl))
                    Console.WriteLine("Die Zahl ist größer als die definierte Zahl");
                Console.WriteLine("Richtig, Sie haben die Zahl richtig geschätzt!");
            }
        }
    }
}
```

6. Flächenberechnung [Konsolenanwendung]

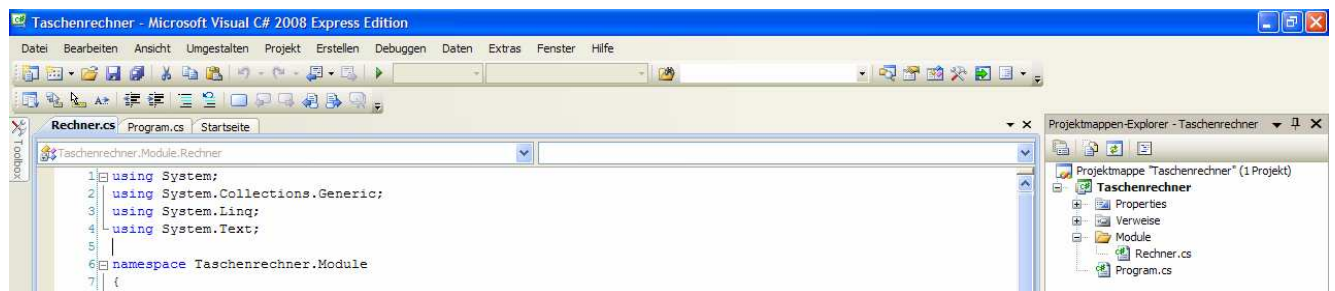
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Gib eine Zahl für die Breite ein");
            string Breite = Console.ReadLine();

            Console.WriteLine("Bitte noch eine Zahl für die Höhe");
            string Hoehe = Console.ReadLine();

            Console.WriteLine("Die Fläche:" + Flaechen(int.Parse(Breite), int.Parse(Hoehe)));
        }
        static int Flaechen(int Breite, int Hoehe)
        {
            return Breite * Hoehe;
        }
    }
}
```

7. Addition (mit Modul) [Konsolenanwendung]



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Taschenrechner.Module;

namespace Taschenrechner
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hallo Nutzer, bitte gib eine Zahl an");
            String Zahl1 = Console.ReadLine();

            Console.WriteLine("Bitte eine zweite Zahl");
            String Zahl2 = Console.ReadLine();

            Console.WriteLine("Bitte wähle die gewünschte Rechenoperation: * für
Multiplikation, / für Division, + für Addition und - für ");
            string Zeichen = Console.ReadLine();

            Rechner meinRechner;           //Deklaration "Speicherbereich schaffen"
            meinRechner = new Rechner();   //Instanzierung "Kopieren" der Klasse
                                           //und Einfüttern in den Speicherbereich

            if (Zeichen == "+")
            {
                Console.WriteLine("Die Summe aus " + Zahl1 + " und " + Zahl2 + " lautet: "
+ meinRechner.Addiere(int.Parse(Zahl1), int.Parse(Zahl2)));
            }

            else if (Zeichen == "-")
            {
                Console.WriteLine("Die Differenz aus " + Zahl1 + " und " + Zahl2 + " lautet: "
+ meinRechner.Subtrahiere(int.Parse(Zahl1), int.Parse(Zahl2)));
            }

            else if (Zeichen == "*")
            {
                Console.WriteLine("Die Multiplikation von " + Zahl1 + " mit " + Zahl2 +
" lautet: " + meinRechner.Multipliziere(int.Parse(Zahl1), int.Parse(Zahl2)));
            }

            else if (Zeichen == "/")
            {
                Console.WriteLine("Der Quotient aus " + Zahl1 + " und " + Zahl2 + " lautet: "
+ meinRechner.Dividiere(int.Parse(Zahl1), int.Parse(Zahl2)));
            }
        }
    }
}
//-----
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Taschenrechner.Module
{
    class Rechner //Klasse, die alle nötigen Rechenoperationen mitführt
    {
        public Rechner //Addiert die beiden übergebenen Rechner
        ( )
        { }

        public int Addiere(int eineZahl, int andereZahl)
        {
            return eineZahl + andereZahl;
        }

        public int Subtrahiere(int eineZahl, int andereZahl)
        {
            return eineZahl - andereZahl;
        }

        public int Multipliziere(int eineZahl, int andereZahl)
        {
            return eineZahl * andereZahl;
        }

        public float Dividiere(float eineZahl, float andereZahl)
        {
            return eineZahl / andereZahl;
        }
    }
}

```

8. Vererbung "Elefant" (mit Modul) [Konsolenanwendung]

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            Elefant meinElefant; //Deklaration (Specherbereich schaffen)
            meinElefant = new Elefant(); //Instanzierung (Schaffen einer
            //Kopie der Klasse Elefant und
            //Einlegen in die Variable
            //meinElefant": "Füttern des Speichers"
            meinElefant.Color = "rot"; //Wertzuweisung, dem Attribut
            //Color wird innerhalb der Kopie
            //des Elefanten ein Wert zugewiesen
            Console.WriteLine("Unser Elefant hat heute die Farbe "
            + meinElefant.Color);
        }
    }
}

//-----

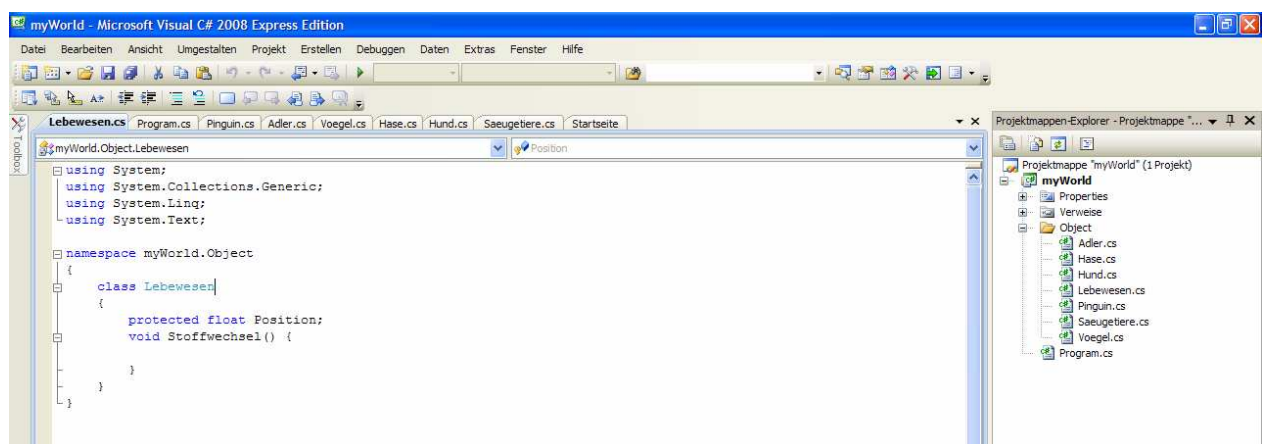
```

```

class Elefant {
    public string Color;
    string Ohren;
    public Elefant()
    {
        //Konstruktor wird automatisch bei Instanzierung ausgeführt
    }
    Elefant fangMich() {
        return this;
    }
}
}

```

9. Vererbung "myWorld" (mit mehreren Objekten und Attributen) [Konsolenanwendung]



```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using myWorld.Object;

namespace myWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            List<Lebewesen> meineTiere = new List<Lebewesen>();

            for (int i = 0; i < 1000; i++)
            {
                meineTiere.Add(new Hund());
                meineTiere.Add(new Hase());
                meineTiere.Add(new Adler());
                meineTiere.Add(new Pinguin());
            }
        }
    }
}

//-----

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Adler: Voegel
    {
        bool Wappentier = true;
    }
}

//-----

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Hase: Saeugetiere
    {
        void hoppeln()
        {
            Console.WriteLine("Hopp hopp");
        }
    }
}

//-----

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Hund: Saeugetiere
    {
        void bellen() {
            Console.WriteLine("Wau Wau");
        }
    }
}

//-----

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Lebewesen
    {
        protected float Position;
        void Stoffwechsel() {
        }
    }
}

//-----

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Pinguin: Voegel
    {
        public override void fliegen()
        {
            Console.WriteLine("Ich kann fliegen");
        }
    }
}
//-----

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Saeugetiere: Lebewesen
    {
        public string Fellfarbe;
        public Saeugetiere()
        {
            this.Fellfarbe = "rot";
        }
    }
}
//-----

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace myWorld.Object
{
    class Voegel: Lebewesen
    {
        public virtual void fliegen() {
            Console.WriteLine("Flutter Flutter");
        }
    }
}

```

10. ggT (mit Ausschluss G < 1) [Konsolenanwendung]

```
using System;
using System.Collections.Generic;
using System.Text;
namespace ggT
{
    class Program
    {
//-----

        static void Main(string[] args)
        {
            double iZahl1 = 0, iZahl2 = 0, iggT = 0; //Definitionen
            ueberschrift(); //Methoden und
            iZahl1 = EingabeZahlGroesserAlsNull(1); //Funktionen
            iZahl2 = EingabeZahlGroesserAlsNull(2);
            iggT = ggTBerechnen(iZahl1, iZahl2);
            AusgabeErgebnis(iZahl1, iZahl2, iggT);
        }
//-----

        static void ueberschrift()
        {
            Console.WriteLine("Berechnung des ggT"); //Methode Titel
            Console.WriteLine(""); //Leerzeile
        }
//-----

        static void AusgabeErgebnis(double iZahl1, double iZahl2, double iggT) //Methode Ergebnisausgabe
        {
            Console.WriteLine("Der ggT von {0} und {1}: {2} ", iZahl1, iZahl2, iggT);
        }
//-----

        static double EingabeZahlGroesserAlsNull(double izahl) //Methode Eingabe Zahl: > 0
        {
            string sZahl1 = ""; //Deklaration
            double iZahl1 = 0;

            do //solange iZahl1 < 1
            {
                if (izahl == 1) //Bedingung
                {
                    Console.Write("Bitte die erste Zahl eingeben: ");
                }
                else
                {
                    Console.Write("Und nun bitte die zweite Zahl eingeben: ");
                }

                sZahl1 = Console.ReadLine();
                iZahl1 = Convert.ToInt32(sZahl1);

                if (iZahl1 < 1) //Bedingung Ausgabe
                {
                    Console.WriteLine("Die Zahl muss größer als Null sein");
                }
            } while (iZahl1 < 1);
            return iZahl1; //Rückgabe Zahl1
        }
//-----
    }
}
```

```

static double ggTBerechnen(double iZahl1, double iZahl2) //Methode ggT-
{
    double iR = 0;    //deklariert
                        //Berechnung

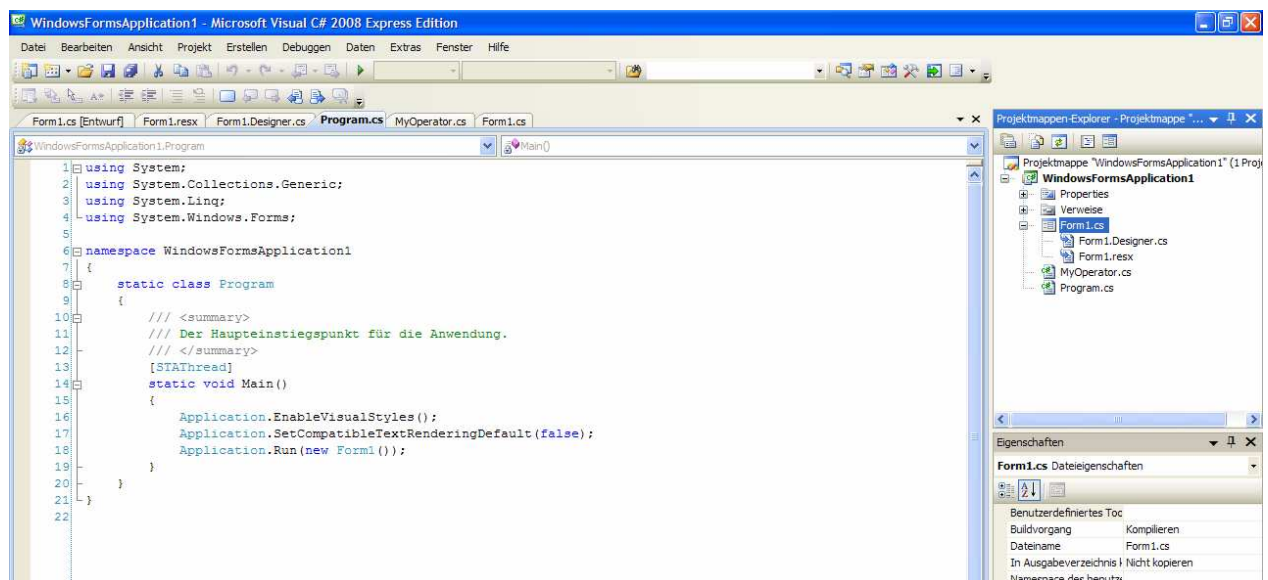
    do                //Schleife
    {
        iR = iZahl1 % iZahl2; //Modulo-Restoperator

        if (iR != 0)
        {
            iZahl1 = iZahl2; //Wertzuweisung
            iZahl2 = iR;
        }
    } while (iR != 0);

    return iZahl2; //Rückgabe
}
}
}

```

11. Taschenrechner [Formularanwendung]



```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace WindowsFormsApplication1
{
    enum MyOperator
    {
        Plus,
        Minus,
        Mal,
        Geteilt
    }
}

//-----

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        bool operatorPressed = false;
        MyOperator aktiveOperator;
        string zwischenSpeicher = "0";
        public Form1()
        {
            InitializeComponent();
        }
        // Fügt Zahl an bestehenden Wert im Eingabefeld an
        // oder, wenn Operator gedrückt, löscht Eingabefeld und
        // fügt sodann neuen Wert ein
        private void addValue(string Wert)
        {
            if (operatorPressed)
            {
                zwischenSpeicher = textBox1.Text;
                textBox1.Text = Wert;
                operatorPressed = false;
            }

            else
            {
                textBox1.Text += Wert;
            }
        }

        #region Numerische Tasten
        private void button1_Click(object sender, EventArgs e)
        {
            addValue("1");
        }

        private void button2_Click(object sender, EventArgs e)
        {
            addValue("2");
        }

        private void button3_Click(object sender, EventArgs e)
        {
            addValue("3");
        }

        private void button5_Click(object sender, EventArgs e)
        {
            addValue("4");
        }

        private void button6_Click(object sender, EventArgs e)
        {

```

```

        addValue("5");
    }
private void button7_Click(object sender, EventArgs e)
{
    addValue("6");
}

private void button9_Click(object sender, EventArgs e)
{
    addValue("7");
}

private void button10_Click(object sender, EventArgs e)
{
    addValue("8");
}

private void button11_Click(object sender, EventArgs e)
{
    addValue("9");
}
private void button14_Click(object sender, EventArgs e)
{
    addValue("0");
}
#endregion

private void button13_Click(object sender, EventArgs e)
{
    addValue(",");
}

//-----

//Taste den +Operator
private void button4_Click(object sender, EventArgs e)
{
    aktiveOperator = MyOperator.Plus;
    operatorPressed = true;
}

//Taste den -Operator
private void button8_Click(object sender, EventArgs e)
{
    aktiveOperator = MyOperator.Minus;
    operatorPressed = true;
}

//Taste den xOperator
private void button12_Click(object sender, EventArgs e)
{
    aktiveOperator = MyOperator.Mal;
    operatorPressed = true;
}

//Taste den :Operator
private void button16_Click(object sender, EventArgs e)
{
    aktiveOperator = MyOperator.Geteilt;
    operatorPressed = true;
}

```

```
//-----

//Taste für = (Ergebnis berechnen)
private void button15_Click(object sender, EventArgs e)
{
    berechneErgebnis();
}

public void berechneErgebnis()
{
    float zwischenSp = float.Parse(zwischenSpeicher);
    float eingabeFeld = float.Parse(textBox1.Text);
    float ergebnis;
    switch (aktiveOperator)
    {
        case MyOperator.Plus:
            ergebnis = zwischenSp + eingabeFeld;
            textBox1.Text = ergebnis.ToString();
            break; //bricht switch ab
        case MyOperator.Minus:
            ergebnis = zwischenSp - eingabeFeld;
            textBox1.Text = ergebnis.ToString();
            break; //bricht switch ab
        case MyOperator.Mal:
            ergebnis = zwischenSp * eingabeFeld;
            textBox1.Text = ergebnis.ToString();
            break; //bricht switch ab
        case MyOperator.Geteilt:
            if (eingabeFeld == 0)
            {
                textBox1.Text = "Division durch 0!";
            }
            else {
                ergebnis = zwischenSp / eingabeFeld;
                textBox1.Text = ergebnis.ToString();
            }
            break; //bricht switch ab
    }
    operatorPressed = false;
}
}
}
```

