

Programmieren mit einer Entwicklungsumgebung

Bsp: Visual C++

Die von Microsoft zur Verfügung gestellte Software Visual C++ dient zur graphischen Programmierung mit der Programmiersprache C++. Der Unterschied zwischen Visual C++ und Dev-Cpp, mit dem wir bisher gearbeitet haben, liegt darin, dass man in ersterem von Anfang an in einer so genannten „Form“ arbeitet (Das Hauptfenster des Programms, dass man im Laufe der Programmierung editiert).

Allgemeines

Bevor mit dem Programmieren begonnen wird, empfiehlt es sich ein neues Projekt zu erstellen, da dieses viele Grunddateien eines Programmes selbst erstellt. Sobald dies geschehen ist sehen wir auch schon ein Fenster mit dem Namen „Form.h“. Jetzt können wir dieses Fenster beliebig verändern z.Bsp einen Button einfügen, eine Textbox einfügen, Größe und Ausrichtung ändern, Farbe ändern, usw.

Grundlagen

Visual C++ beinhaltet die „Toolbox“, die alle Steuerelemente enthält und sich als recht nützlich erweist, da man ansonsten viele Codezeilen selbst programmieren müsste, um z.Bsp einen Button hinzuzufügen. Die Befehle in Visual C++ sind identisch mit jenen, die auch in Dev-Cpp benutzt werden. Nur die Syntax ist manchmal etwas verändert. Es gibt allerdings auch einige neue Befehle, wie z.Bsp „this→“.

Onlinetutorials C++:

- [C++ Tutorial](#) - umfangreiches Onlinetutorial zu den Grundlagen von C++, sehr gut und sehr genau erklärt. Eignet sich hervorragend zur Vertiefung!
- [Visual C++ - Einführung](#) - Auszüge aus einer Literatur bezüglich Visual C++ Einführung.

Bsp: Visual C#

Visual C# gehört zu den objektorientierten Programmiersprachen. Wie auch in Visual C++ kann man hier in der so genannten „Form“ arbeiten, dem Standardfenster mit bereits vorgefertigten Grundlagen beim Öffnen eines neuen Projekts. Dieses Formular bietet beim Start eine Windowsfenster mit den

üblichen Minimierungs-, Maximierungs- und Schließenbuttons im rechten oberen Eck.

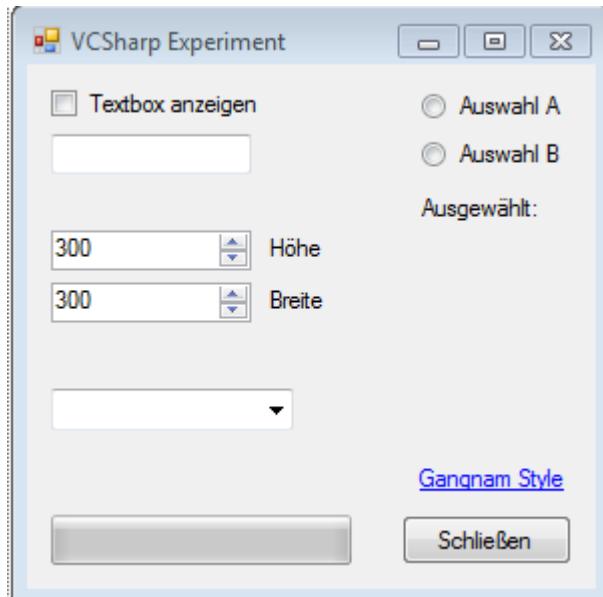
In der 2010er-Version des MS Visual Studio fehlt nur bei Visual C++ die sogenannte 'Intellisense'-Unterstützung, daher dieser Exkurs zu C#. Diese ist eine sehr gute Hilfe beim Programmieren mit einer Entwicklungsebene. Wenn man einen Buchstaben eingibt erscheinen sofort alle Optionen, also Befehle, Methoden, Eigenschaften usw., die zu diesem Zeitpunkt der Programmierung möglich sind. Ein weiterer Unterschied in der Syntax ist, dass der Zugriff auf Eigenschaften und Methoden eines Objektes nicht mehr durch „→“ ausgeführt werden sondern durch „..“

Bsp:

```
Visual C++ this->Close();
Visual C#  this.Close();
```

Grundlagen

Die Grundlagen sind identisch wie bei Visual C++ eng mit der Toolbox verknüpft. Die Schwierigkeit besteht darin die durch die Toolbox in die Form eingefügten Objekte richtig Funktionen zuzuweisen. Um da zu veranschaulichen nehmen wir einen Button aus der Toolbox und ziehen es in das Form Fenster.



Durch einen Doppelklick auf den Button „Schließen“ wird die Methode (früher auch bekannt als Funktion) button1_Click() ausgeführt. Diese Methode muss zuvor ausprogrammiert werden, d.h. alles was darin steht wird bei einem Click auf den Button „Schließen“ ausgeführt. In diesem Fall wird das Formularfenster geschlossen. Mit dem „..“-Operator erhält man Zugriff auf die Methoden und Eigenschaften eines Objektes (vgl. → in C++)

```
private void button1_Click(object sender, EventArgs e)
{
    this.Close();
```

}

verwendete Steuerelemente (aus der Toolbox)

Im oben gezeigten Beispiel kann man sehen, wie wir schon mit einigen Objekten der Toolbox experimentiert haben im Folgenden werde ich nun die wichtigsten und interessantesten hier genauer beschreiben.

- Button:
- Label:
- Checkbox: (Name = Textbox anzeigen) Gibt an falls die Checkbox „gechecked“ ist wird die Textbox angezeigt.
- Combobox: In der Combobox wird hier über das Eigenschaftsfenster → Items (Auflistung) ein Index zugewiesen, welcher in einem Label ausgegeben wird.
- Radiobutton:
- Numeric Up/Down: Ist eine Box mit einem minimal und maximal Wert zwischen den man wählen kann. Der ausgewählte Wert ergibt jeweils das Value von der Numeric Up/Down-Box, mit 2 kleinen Buttons innerhalb kann man den Wert erhöhen oder senken bis zum minimal oder maximal Wert... Value inspricht jeweils den derzeitigen Wert in der Numeric Up/Down-Box

Eigenschaften

Ein wichtiger Aspekt beim objektorientierten Programmieren sind die Eigenschaften der Objekte. Im Quellcode erhält man Zugriff über den '`.`-Operator. Durch ein einfaches Anklicken eines Objektes zeigt uns das **Eigenschaftsfenster** in der Entwicklungsebene alle Eigenschaften des ausgewählten Objektes: z.B. Name, Größe, Farbe, Sichtbarkeit (= Visible), die beliebig geändert werden können oder auch Funktionen von Objekten ausgeschaltet werden können. Von Eigenschaften sind die jeweiligen Objekte abhängig, wie sie aussehen, sich verhalten, etc.

Allgemein gültige Eigenschaften (haben die meisten Steuerelemente gemein):

- Text:
- Name:
- Font:
- TabIndex:
- Size:
- TextAlign:
- Visable:

Ereignisse / Methoden

Ereignisse können eintreten, wenn vom Anwender Steuerelemente verwendet werden. Z.B. ein Button

wird gedrückt, dann löst man damit das Ereignis „Click“ dieses Buttons aus. Auf der Quellcodeebene wird dabei eine Methode (Unterprogramm/Funktion) erzeugt, die dann ausprogrammiert werden muss. Im Quellcode erhält man Zugriff wie bei den Eigenschaften mit dem „.“-Operator (in C#, vgl. → in C++).

Verschiedene Ereignisse, die bei Objekten/Steuerelementen ausgelösten werden können:

- Click:
- ValueChanged:
- CheckedChanged:
- MouseEnter:
- MouseLeave:
- MouseHover:

C Sharp Online Tutorial

Habe hier eine sehr gute Hilfe gefunden .

http://elearning.tutorials.de/pdf/erikbartmann/csharp/4_softwareentwicklung.pdf

From:
<http://elearn.bgamstetten.ac.at/eduwiki/> - **EduDokuWiki**

Permanent link:
http://elearn.bgamstetten.ac.at/eduwiki/doku.php?id=inf:inf6bi_201213:algodatprog:objprog:visprog

Last update: **2012/11/28 07:47**