

# U1 Erster Kontakt mit Borland C++ Builder

## Umfeld

Borland C++ Builder bietet eine komplette Programmierumgebung für die Programmierung in C/C++:

- Editor zur Eingabe des Quelltextes
- Compiler und Linker zum Erzeugen des Maschinencodes
- Debugger zum Austesten des Codes
- Projektverwaltung zum Verwalten der Files bei grösseren Programmierprojekten
- Diverse weitere Hilfsmittel, vor allem für die Windows- und OO-Programmierung

In einem ersten Schritt wollen wir ein Minimalprogramm codieren, ohne uns weiter um Details zu kümmern. Alle Schritte werden später eingehend erklärt.

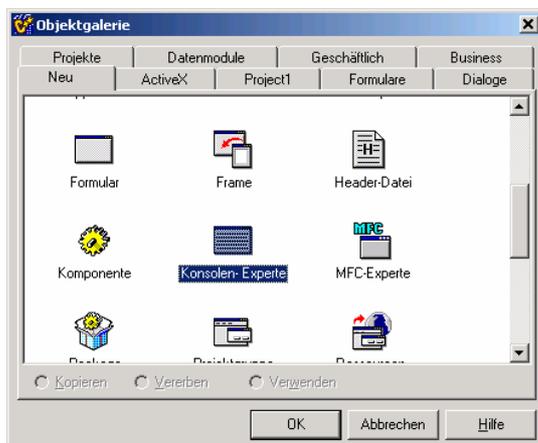
## Vorgehen

Vorbereitung:

Windows 2000 booten, einloggen und Borland C++ Builder (BCB) aufstarten. Es erscheint eine Arbeitsoberfläche mit vier verschiebbaren Fenstern.

Programm erstellen:

1. Eröffnen eines neuen, leeren Projektes über *Datei/Neu*. Den Konsolen-Experten starten.

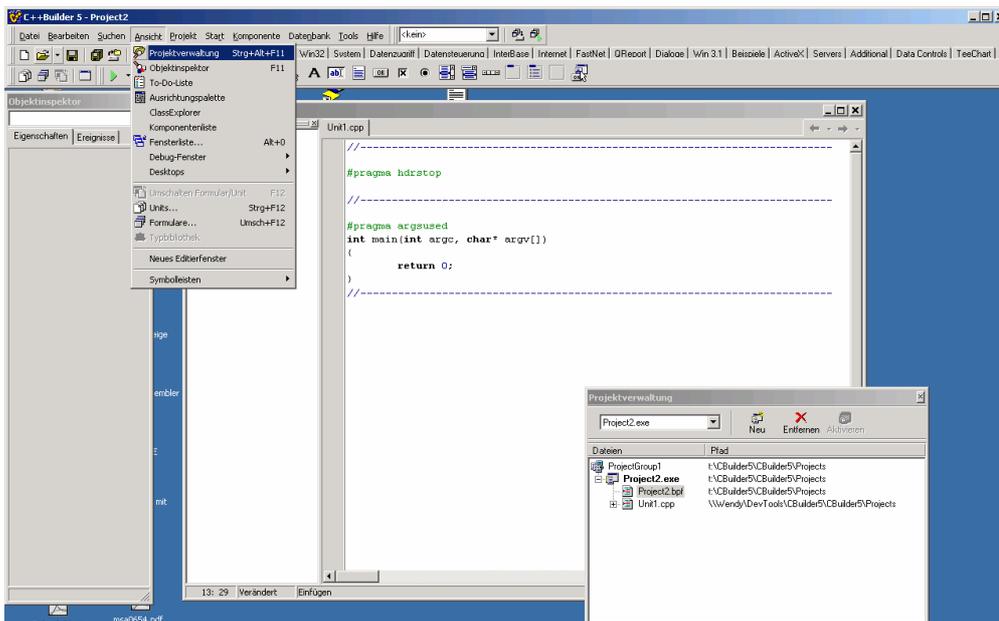


Nun C++- Konsolenanwendung einstellen. Die Projektquelle bleibt leer, weil kein Projekt übernommen wird.



Am Schluss wird gefragt, ob das standardmässig geöffnete Projekt mit dem Namen `Project1` gespeichert werden soll. **Diese Frage mit NEIN beantworten.**

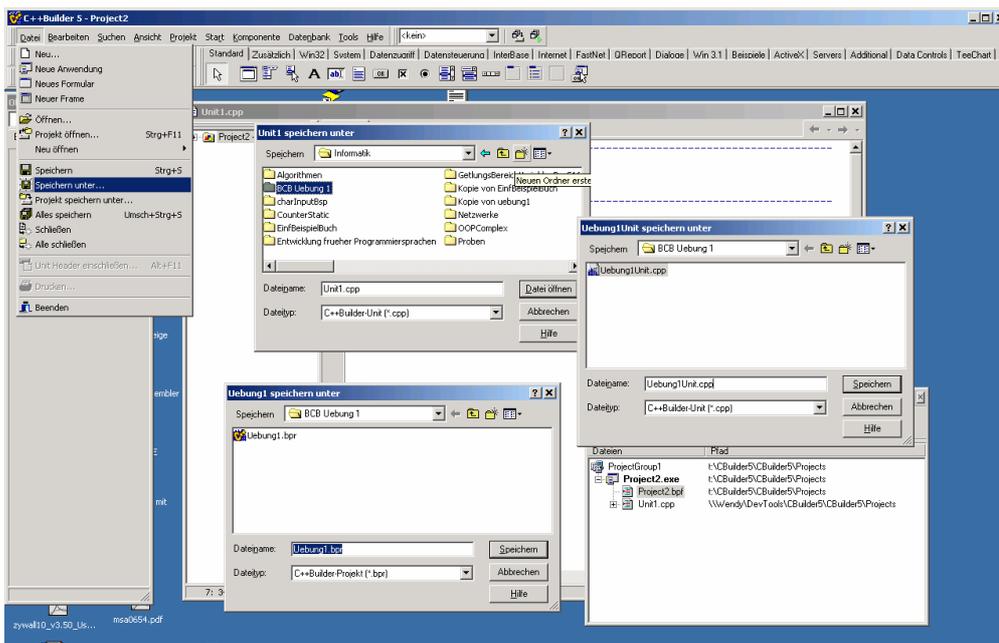
Es erscheint das noch leere Konsolenprojekt. Ein Grundprogramm mit dem Namen `UNIT1.CPP` ohne weitere Funktionalität wurde bereits automatisch erzeugt.



Über *Ansicht/Projektverwaltung* kann das Projektmanagerfenster eingeblendet werden. Es zeigt alle zum Programmierprojekt gehörenden Files an.

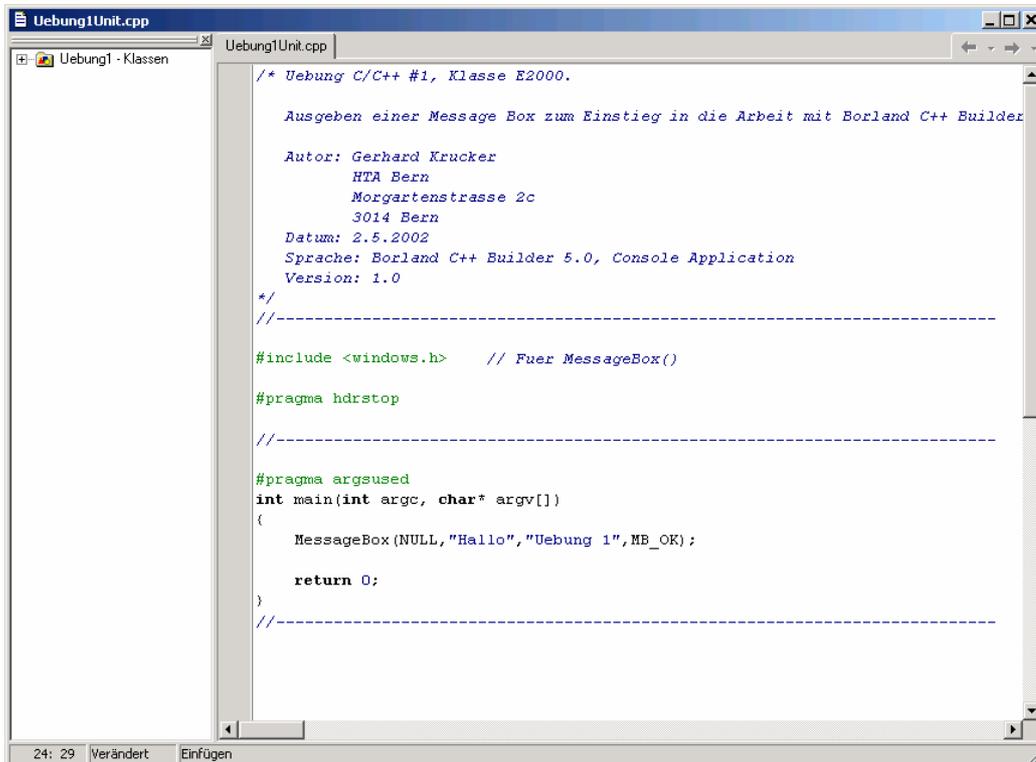
Bevor codiert wird, wird das Projekt und das Hauptprogramm (`UNIT1.CPP`) mit aussagekräftigen Namen gespeichert:

- Namen des Projektes (= Namen des entstehenden EXE-Files) eingeben: `Uebung1.bpr`
- Namen des Sourcecodes: `Uebung1Unit.cpp`  
Sicherstellen, dass auf dem persönlichen Laufwerk gespeichert wird.
- Passenden Pfad wählen: Im Speicherdialog kann ein neues Directory mit dem Projektnamen erzeugt werden:



Dieses leere Programm könnte bereits kompiliert und gestartet werden.

2. Eingeben des C-Quelltextes gemäss Beispiel:



```
Uebung1Unit.cpp
Uebung1 - Klassen
Uebung1Unit.cpp

/* Uebung C/C++ #1, Klasse E2000.

Ausgeben einer Message Box zum Einstieg in die Arbeit mit Borland C++ Builder

Autor: Gerhard Krucker
HTA Bern
Morgartenstrasse 2c
3014 Bern

Datum: 2.5.2002
Sprache: Borland C++ Builder 5.0, Console Application
Version: 1.0
*/
//-----

#include <windows.h> // Fuer MessageBox()

#pragma hdrstop

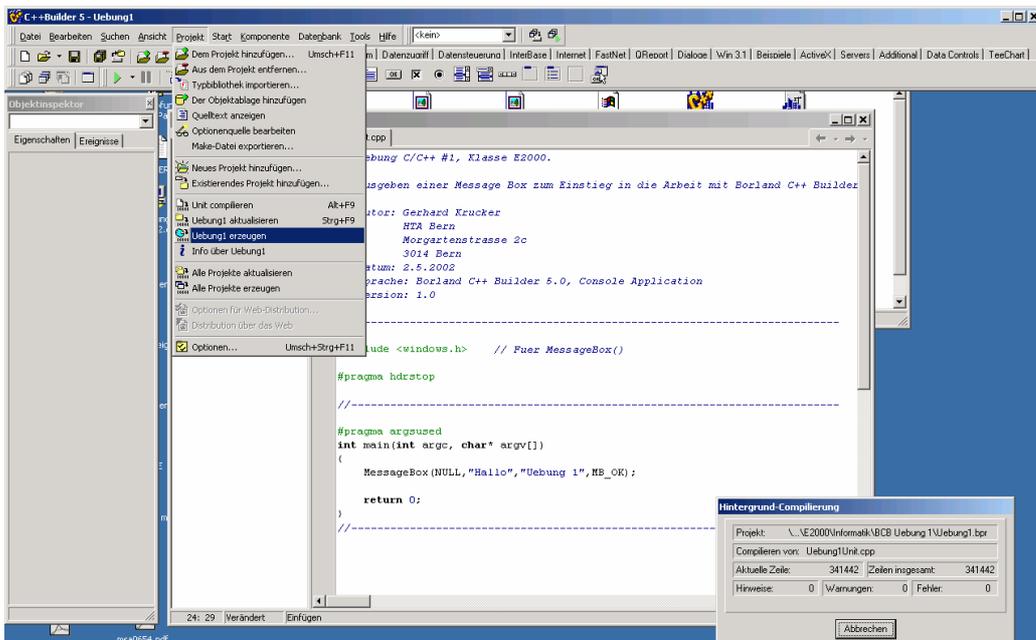
//-----

#pragma argsused
int main(int argc, char* argv[])
{
    MessageBox(NULL, "Hallo", "Uebung 1", MB_OK);

    return 0;
}
//-----

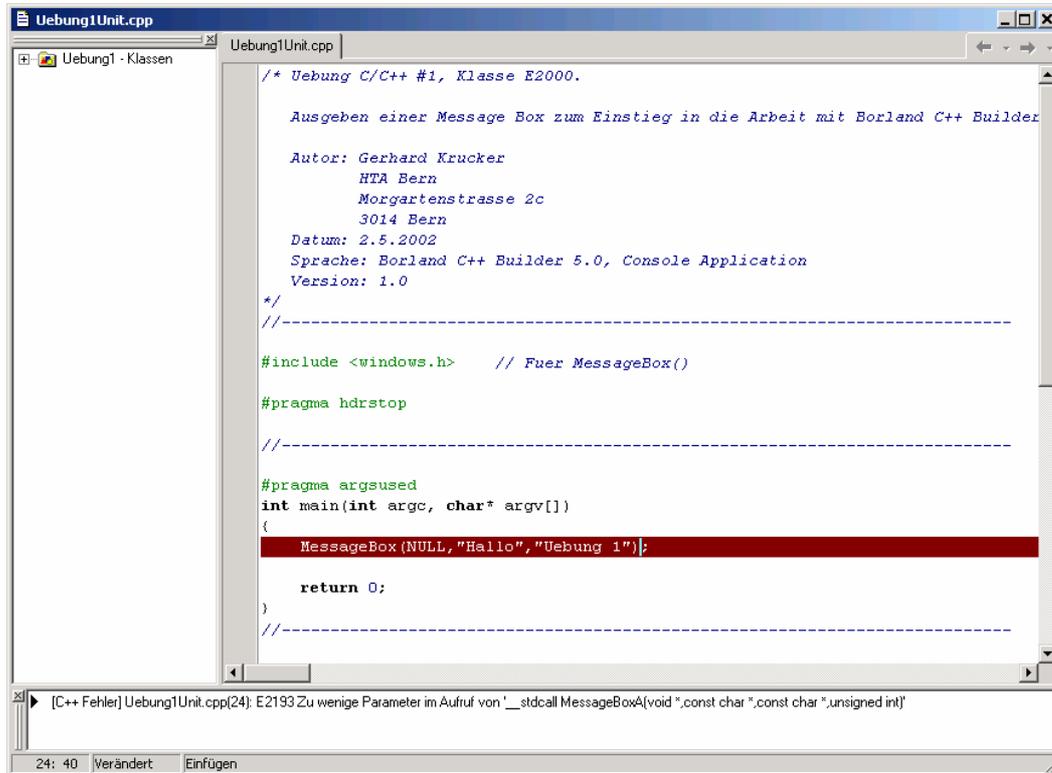
24: 29 | Verändert | Einfügen
```

3. Erzeugen des Maschinencodes über Projekt/Uebung1 erzeugen:



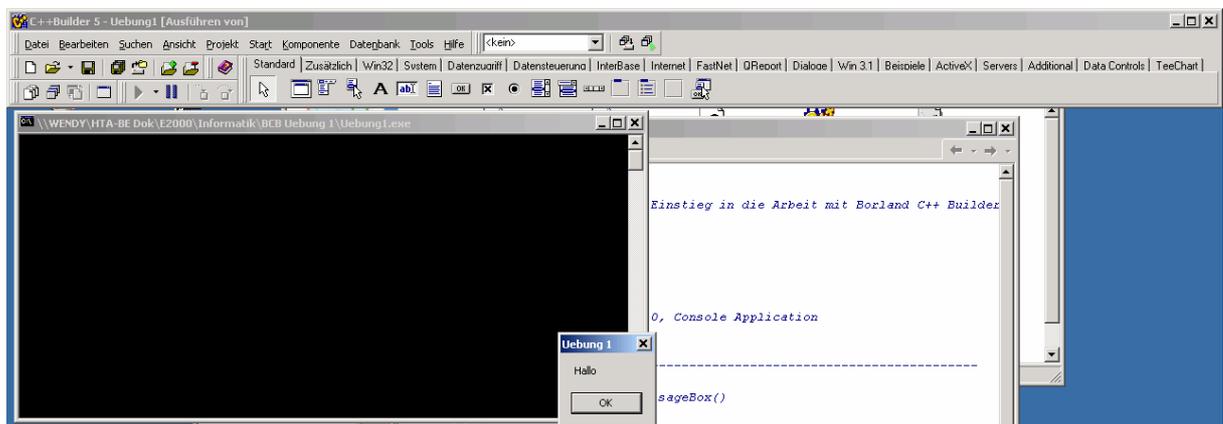
Der Kompilationsfortschritt wird in einem eigenen Statusfenster angezeigt. Bei erfolgreicher Kompilation wird automatisch der Linker gestartet und ein ablauffähiges EXE erzeugt.

Werden Fehler im Quellcode erkannt, wird der Erstellungsgang mit einer entsprechenden Fehlermeldung abgebrochen:



Hinweis: Nur aus Durchläufen ohne Fehler können lauffähige Programme (EXE) erstellt werden.

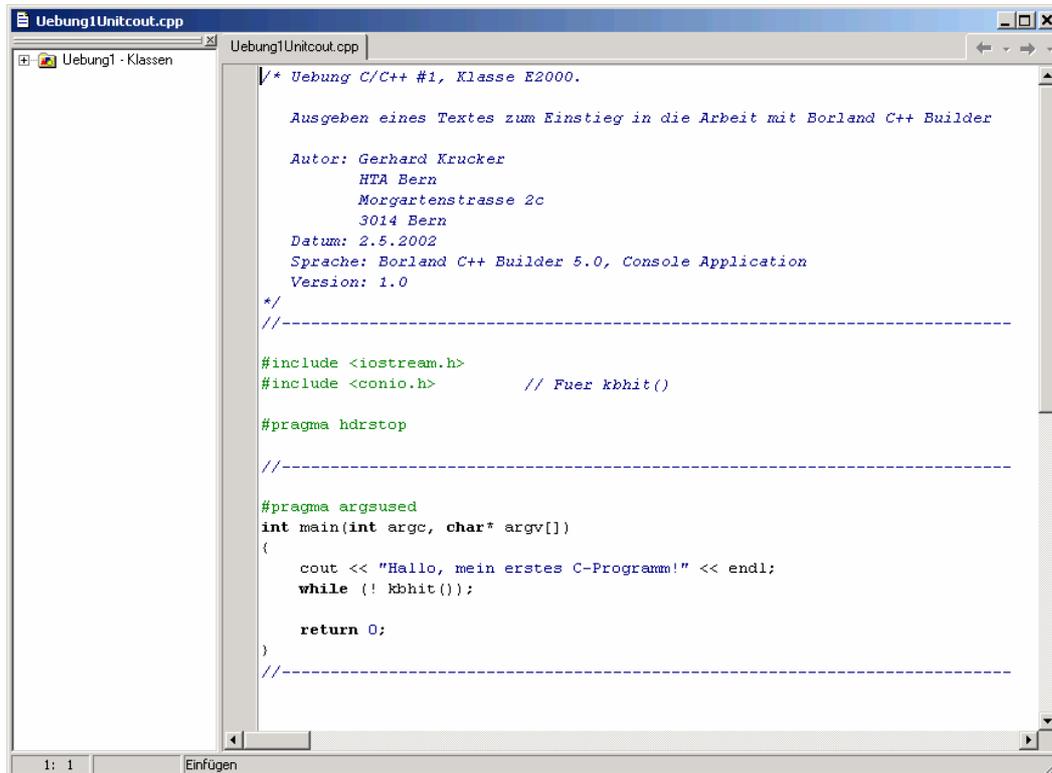
4. Das Programm kann direkt aus der Entwicklungsumgebung mit dem Knopf  gestartet und durch Druck auf den OK-Knopf in der Message Box beendet werden.



Viel Erfolg!

### Bemerkung

Eine reine Konsolenausgabe können Sie mit folgendem Programmcode erreichen:



```
Uebung1Unitcout.cpp
Uebung1Unitcout.cpp
Uebung1 - Klassen

/* Uebung C/C++ #1, Klasse E2000.

Ausgeben eines Textes zum Einstieg in die Arbeit mit Borland C++ Builder

Autor: Gerhard Krucker
HTA Bern
Morgartenstrasse 2c
3014 Bern
Datum: 2.5.2002
Sprache: Borland C++ Builder 5.0, Console Application
Version: 1.0
*/
//-----

#include <iostream.h>
#include <conio.h> // Fuer kbhit()

#pragma hdrstop

//-----

#pragma argsused
int main(int argc, char* argv[])
{
    cout << "Hallo, mein erstes C-Programm!" << endl;
    while (! kbhit());

    return 0;
}
//-----

1: 1 Einfügen
```