

# Einführung in die komponentenorientierte Programmierung mit Borland-Delphi

## Übung 1

### Standardkomponenten und Programmerstellung

Dr. Henry Herper

Otto-von-Guericke-Universität Magdeburg

Institut für Simulation und Graphik

# Imperative und objektorientierte Programmiersprachen – Delphi

Anwendungsgebiet: Erstellung von Windows-Applikationen

Kennzeichnend für diese Programmiersprachen sind die Eigenschaften

- objektorientiert,
- ereignisgesteuert und
- visuell.

Da Delphi für Windows-Anwendungen entwickelt wurde, unterstützt es besonders den Entwurf von Nutzeroberflächen. Der Zugriff zu Windows erfolgt über die Windows-API.

# Ereignisorientierte Softwareentwicklung

Moderne Anwendungen auf graphischen Oberflächen arbeiten ereignisorientiert.

**Anwendungen reagieren auf Ereignisse.** Diese Ereignisse können vom Betriebssystem, von anderen Anwendungen oder von der Anwendung selbst erzeugt werden.

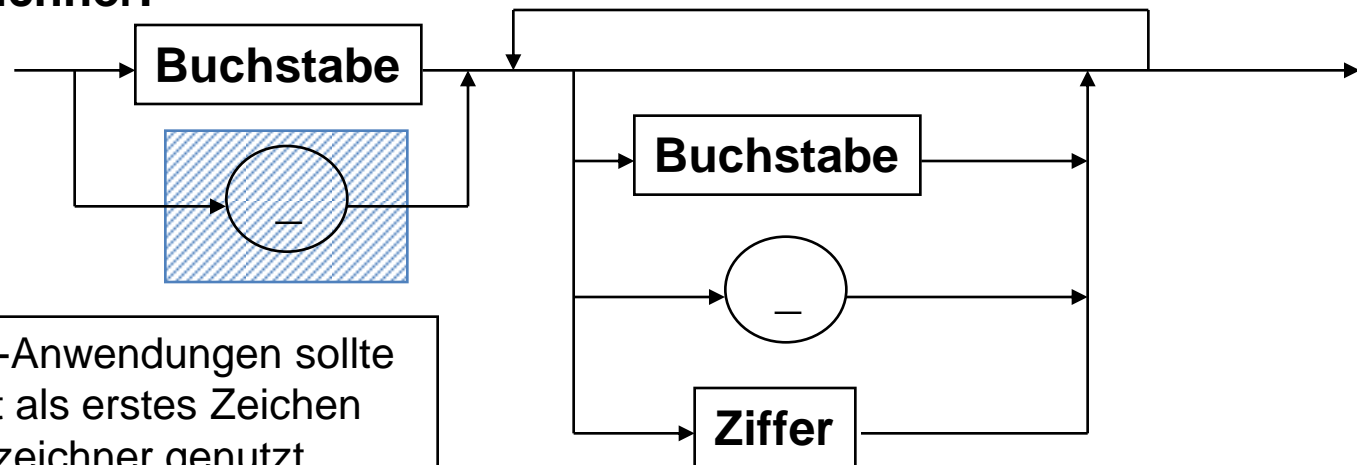
# Object-Pascal – Grundlagen – Editor

Die **maximale Zeilenlänge**, die vom Editor verarbeitet werden kann, beträgt **256 Zeichen**. Im Interesse der Lesbarkeit sollten aber nur Zeilen mit einer Länge von 80 Zeichen verwendet werden. Ein Zeilenumbruch ist an beliebigen Positionen des Quelltextes (außer in Zeichenketten) möglich.

# Object-Pascal – Grundlagen – Bezeichner

**Bezeichner** (Namen) dürfen beliebig lang sein kann, es sind nur die ersten 255 Zeichen (ältere Versionen 63 Zeichen) signifikant. Ein Bezeichner muss mit einem Buchstaben oder einem Unterstrich ( \_ ) beginnen und **darf keine Leerzeichen enthalten**. **Sonderzeichen, Umlaute und ß sind nicht als Zeichen zugelassen**. Auf das erste Zeichen können weitere Buchstaben, Ziffern und Unterstriche folgen. Reservierte Wörter dürfen nicht als Bezeichner verwendet werden.

**Bezeichner:**



In .net-Anwendungen sollte \_ nicht als erstes Zeichen für Bezeichner genutzt werden.

# Object-Pascal – Grundlagen – Bezeichner

In Object-Pascal **wird** bei dem Bezeichnern und reservierten Wörtern **nicht zwischen Groß- und Kleinschreibung unterschieden**. Delphi eigene Bezeichner der Komponenten beginnen in der Regel mit einem Großbuchstaben (ausgenommen reservierte Wörter). Zur Verbesserung der Lesbarkeit können auch innerhalb von Bezeichnern Großbuchstaben verwendet werden.

Beispiel: TForm1.Button1Click

# Object-Pascal – Projektstruktur

Delphi-Applikationen bestehen aus mehreren Quelltextdateien und vom System erzeugte Dateien.

Zu einem Projekt gehören folgende Dateien:

- **Project1.dpr – Projektdatei** (Project1.bdsproj)
- Project1.exe – compilierte Windows-Applikation
- **Unit1.pas ... Unitn.pas - Quelltext der Units**
- Unit1.dcu ... Unitn.dcu - compilierte Units
- Project1.res - Delphi-Ressourcendatei
- Project1.dof - Delphi-Projekt-Optionen
- Unit1.dfm ... Unitn.dfm – Formulardateien

...

**Ein Projekt sollte jeweils in einem separaten Verzeichnis verwaltet werden!**

# Object-Pascal – Hauptprogramm

Das **Hauptprogramm wird als Projekt bezeichnet** und automatisch generiert. Diese Datei braucht vom Nutzer im Allgemeinen nicht verändert zu werden. Die Dateiendung ist .dpr und der Dateiname darf mit keinem anderen Dateinamen des Projektes identisch sein. Diese Datei enthält auch die Informationen darüber, welche Module in Form von Units in das Projekt eingebunden sind.

```
program Beispiel_1;
uses
  Forms,
  oberflaeche1 in 'oberflaeche1.pas' {Form1};
{$R *.res}
begin
  Application.Initialize;
  Application.CreateForm(TForm1, Form1);
  Application.Run;
end.
```



# Vorgehensweise bei ereignisorientierter Softwareentwicklung

Für die Entwicklung einer Delphi-Applikation ist folgende Vorgehensweise zweckmäßig:

- Visueller Entwurf der Bedienoberfläche,
- Zuweisen der Objekteigenschaften und
- Verknüpfen der Objekte mit Ereignissen.

Die Entwicklung wird von der IDE (IntegratedDevelopment Environment) unterstützt.

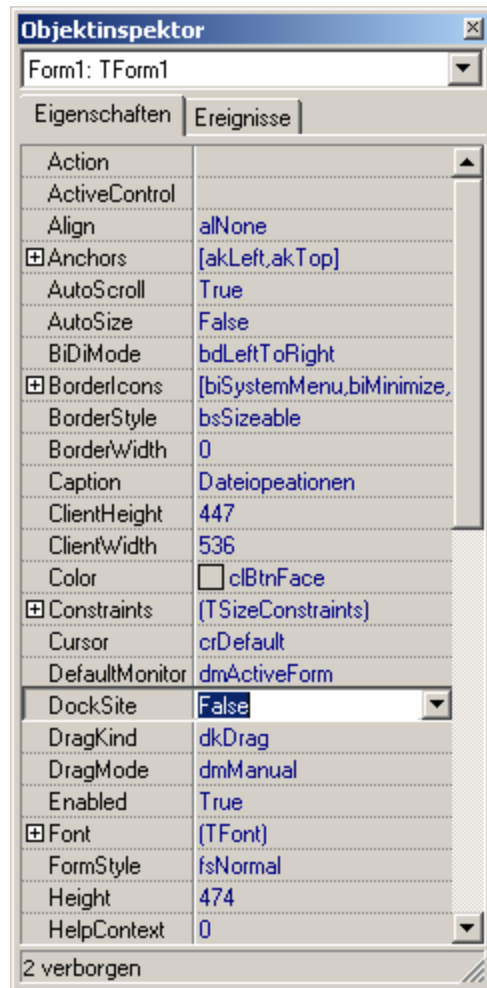
# Ereignisorientierte Softwareentwicklung mit Borland

## Delphi Ereignistypen

- Mausereignisse
- Tastaturereignisse
- Timer-Ereignisse
- Befehlsereignisse
- Eingabeereignisse
- u.v.m.

Die Ereignisse werden in Borland-Delphi als Events bezeichnet.

# Delphi – Objektinspektor - Eigenschaften

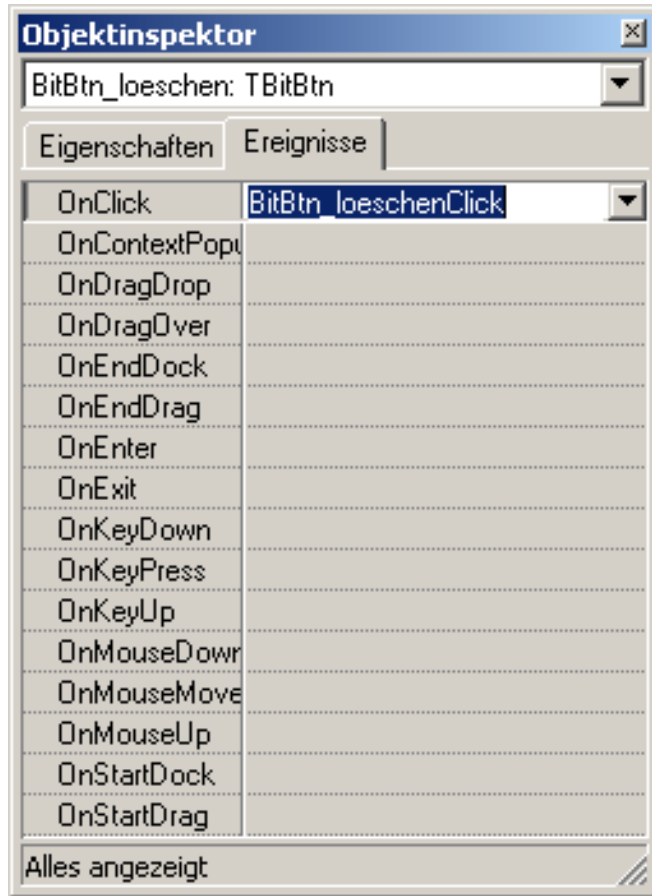


Mit dem Objektinspektor werden die **Eigenschaften des Objektes definiert**. Diese Eigenschaften beschreiben die Anfangseigenschaften und können zur Laufzeit verändert werden.

Jede Komponente verfügt über ihre speziellen Eigenschaften.

Die Eigenschaften werden bei der Erzeugung auf Standardwerte gesetzt, die der Entwickler überschreiben kann.

# Delphi – Objektinspektor - Ereignisse



Mit dem Objektinspektor werden **die für das Objekt auswertbaren Ereignisse festgelegt**. Diese Seite wird als Ereignisseite bezeichnet.

Über diese Ereignisse werden Methoden aufgerufen, die die Funktionalität des Programms realisieren.

# Ereignisbearbeitung in Delphi

Delphi **erzeugt** zur Bearbeitung von Ereignissen eine **Ereignisbehandlungsmethode**. Diese wird vom Entwickler mit dem für das Ereignis notwendigen Quelltext versehen. Es besteht die Möglichkeit, den Absender des Ereignisses auszuwerten.

Beispiel: On-Click-Methode für Button 1

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  
end;
```

# Kontrollfragen

1. Erstellen Sie eine Delphi-Applikation, die einen Button enthält. Beim Überstreichen des Buttons mit der Maus soll sich die Hintergrundfarbe des Formulars verändern, beim Drücken des Buttons der Text auf dem Button.
2. Erstellen Sie eine Delphi-Applikation, die zwei Buttons enthält. Beim Betätigen des ersten Buttons wird dieser unsichtbar. Mit dem Anlicken des zweiten Buttons wird der erste wieder sichtbar.
3. Schreiben Sie eine Delphi Applikation, die dem Nutzer in einem Optionsfeld verschiedenen Hintergrundfarben für das Formular anbietet. Die Auswahl einer Farbe soll sofort die Farbveränderung auslösen.
4. Schreiben Sie eine Delphi Applikation, die Text aus einem Editfenster in eine Label-Komponente überträgt.
5. Erarbeiten Sie unter Verwendung der Delphi-Hilfe die Eigenschaften und Einsatzmöglichkeiten von Buttons, Bit-Buttons und Speed-Buttons.
6. Erläutern Sie den grundlegenden Aufbau von Delphi-Applikationen an Hand der angelegten Dateien. Gehen Sie auf die Begriffe Formular und Unit ein.

# Kontrollfragen

7. Erarbeiten Sie unter Verwendung der Delphi-Hilfe die Eigenschaften und Einsatzmöglichkeiten von Edit- und Memo-Komponenten.
8. Erarbeiten Sie unter Verwendung der Delphi-Hilfe die Eigenschaften und Einsatzmöglichkeiten von Kontrollkästchen und Optionsfeldern.
9. Beschreiben Sie das Grundkonzept der komponentenorientierten Programmentwicklung am Beispiel von Borland-Delphi. Gehen Sie dabei auf Komponenten zur Dateneingabe, Datenausgabe und Programmsteuerung ein.
10. Erläutern Sie die Verwendung einer Delphi-Komponente am Beispiel eines Edit-Fensters. Gehen Sie dabei auf Eigenschaften und Ereignisse ein. Welche Möglichkeiten bestehen, um Eigenschaften zu verändern? Wie werden die durch Ereignisse ausgelösten Methoden implementiert?

# Literatur

Thomas Binzinger

jetzt lerne ich Delphi

Markt+Technik Verlag, 2006, ISBN-13: 978-3-8272-4108-5

Walter Doberenz, Thomas Gewinnus

Borland Delphi 2005

Carl Hanser Verlag, München Wien, ISBN-13: 978-3-446-40202-7

Dieter Orlamünder, Rüdiger Liskowsky, Heinrich Hußmann

Softwareentwicklung mit Delphi

Fachbuchverlag Leipzig, 2002, ISBN: 3-446-22088-7