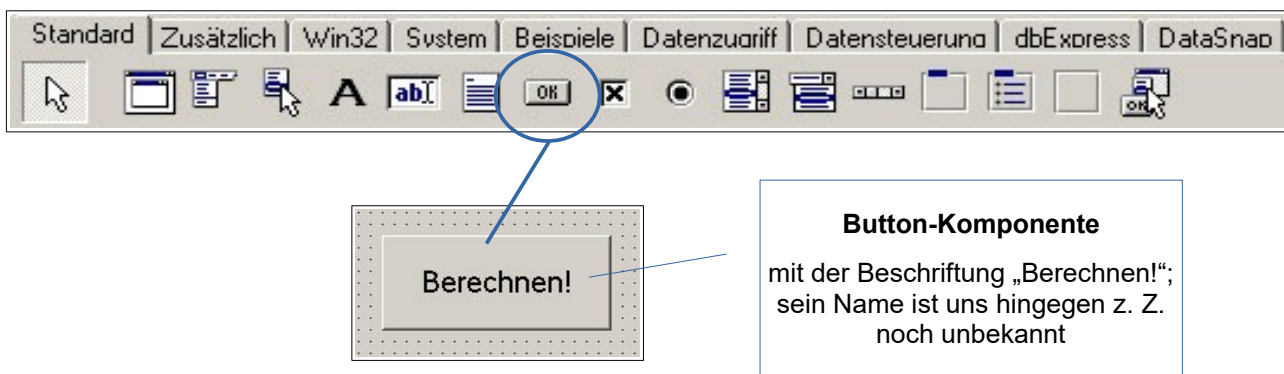


Grundlegende Begriffsklärungen in Delphi: Komponente – Eigenschaft – Ereignis

Komponente (Objekt)

In der sog. **Komponentenpalette** von Delphi befinden sich die einzelnen **Komponenten**, auch **Objekte** genannt. Wir benutzen sie, um die Oberfläche [kurz: GUI, engl. für „Graphical User Interface] unseres Delphi-Programms zu gestalten. Dies sind z. B. *Button, Label, Edit, RadioButton, Panel* etc.

Betrachten wir im Folgenden exemplarisch ein/e Objekt/Komponente, nämlich den *Button*:



Eigenschaft (Attribut)

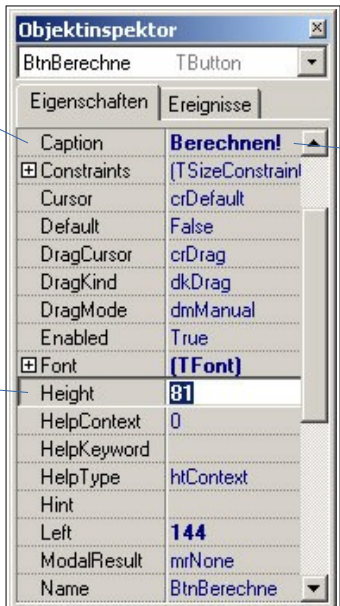
Dieser *Button* hat bestimmte **Eigenschaften**, auch **Attribute** genannt. Sie bestimmen v. a. sein Aussehen, ähnlich wie die Hautfarbe oder Frisur das Aussehen eines Menschen beschreiben. Eine Übersicht aller Eigenschaften des *Buttons* finden wir im **Objektinspektor**:

Eigenschaft „Caption“
für die Beschriftung des Buttons

 („Caption“ ist der **Bezeichner (Name)** der Eigenschaft)

Eigenschaft „Height“
für die Höhe des Buttons, angegeben in Pixel (Bildpunkte)

 („Height“ ist der **Bezeichner (Name)** der Eigenschaft)



Der **Wert** der Caption-Eigenschaft des Buttons ist „Berechnen!“, d. h. der Button ist damit **beschriftet**.

Sein Name hingegen steht unter der Eigenschaft **Name** (s. u.) und lautet/hat den Wert **BtnBerechne**.

Ereignis (Methode)

Unser *Button* verfügt nicht nur über Eigenschaften, sondern reagiert auch auf bestimmte **Ereignisse**, d. h. er besitzt **Methoden**. Im Gegensatz zu Eigenschaften muss eine Komponente keine Ereignisse haben – nur wenn wir sie definieren. Eine Listung der möglichen Ereignisse unseres Buttons finden wir wiederum im **Objektinspektor**, diesmal jedoch unter dem Tab „Ereignisse“:

Ereignis „OnClick“

wenn man auf den Button draufklickt

(„OnClick“ ist der **Bezeichner** (Name) des Ereignisses)

Ereignis „OnMouseMove“

wenn man mit dem Mauszeiger über den Button drüberfährt

(„OnMouseMove“ ist der **Bezeichner** (Name) des Ereignisses)

Objektinspektor

BtnBerechne TButton

Eigenschaften Ereignisse

Action	
OnClick	BtnBerechneClick
OnContextPopup	
OnDragDrop	
OnDragOver	
OnEndDock	
OnEndDrag	
OnEnter	
OnExit	
OnKeyDown	
OnKeyPress	
OnKeyUp	
OnMouseDown	
OnMouseMove	BtnBerechneMouseMove
OnMouseUp	
OnStartDock	
OnStartDrag	

Alles angezeigt

Das OnClick-Ereignis des Buttons ist **definiert** (vorhanden).

Was beim Klicken auf den Button nun passieren soll, ist in der Delphi-Prozedur **BtnBerechneClick** im Quellcode angegeben, deren Rumpf Delphi automatisch anlegt und deren expliziter Inhalt (Befehle) der Programmierer schreiben muss.

Das Variablenkonzept mit Hilfe des „Schachtelmodells“ verstehen

Eine Eigenschaft (Attribut) einer Komponente (Objekts) besitzt immer einen **Bezeichner** (Namen), einen **Datentyp** und einen aktuellen **Wert**. Im Objektinspektor oben ist bspw. „Caption“ der Bezeichner einer bestimmten Eigenschaft der Button-Komponente. Die Eigenschaft hat den Wert „Berechnen!“ und ist damit vom Datentyp „String“ (Zeichenkette).

Komponente	Button (Name: BtnBerechne; Typ: TButton)		
Eigenschaft mit Bezeichner	Caption	Height	...
Wert der Eigenschaft	Berechnen!	81	
Datentyp des Wertes der Eigenschaft	String (Zeichenkette)	Integer (Ganzzahl)	

Die Werte der Attribute lassen sich über **Zuweisungen** ändern. Dabei bieten sich uns in Delphi prinzipiell zwei Möglichkeiten:

- 1) Eingabe des Attributwertes direkt im Objektinspektor, z. B. mit Mausklick auf das Feld rechts neben „Caption“ und anschließender Eingabe von „Berechnen!“.

2) Eingabe des Attributwertes an passender Stelle im Programmcode, bspw. um die Beschriftung des Buttons während des Programmablaufs zu verändern:

```
BtnBerechne.Caption := 'Starten';
```

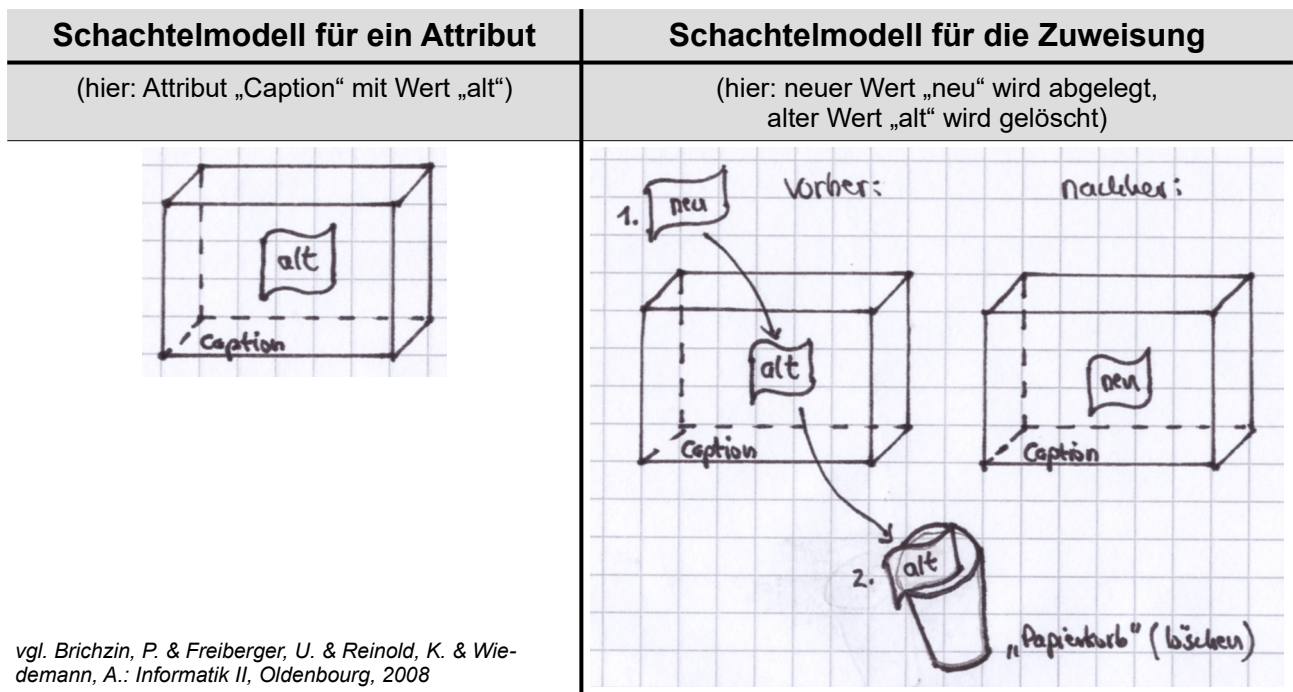
Dabei bedeuten:

- „BtnBerechne“: Wähle die Komponente mit dem Bezeichner „BtnBerechne“ aus.
- „Caption“: Greife auf die Eigenschaft „Caption“ dieser Komponente zu.
- „:=“: Weise der Eigenschaft folgenden (neuen) Wert zu.
- „'Starten'“: Der neue Wert des Komponenten-Attributs (hier in Hochkommas zu setzen, da vom Datentyp String).

Da man die Werte eines Attributs verändern kann, sind alle Attribute zugleich Variablen (ähnlich wie in der Mathematik, z. B. „x“ oder „y“).

Man kann sich Attribute auch mit Hilfe des „Schachtelmodells“ vorstellen:

Ein Attribut ist wie eine Schachtel mit einem Zettel. Die Beschriftung entspricht dem Bezeichner, der Eintrag auf dem Zettel dem Wert, die Schachtelgröße dem Datentyp. Eine Zuweisung bedeutet: Den alten Zettel in der Schachtel wegwerfen und durch einen neuen Zettel ersetzen.

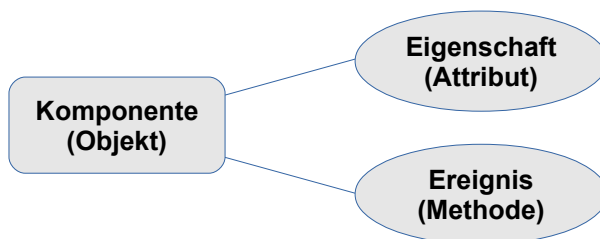


Die „Schachtel eines Attributs“ vom Datentyp Boolean ist kleiner als die eines Attributs vom Datentyp Integer, weil der Datentyp Boolean nur zwei verschiedene Werte annehmen kann (True und False), der Datentyp Integer jedoch einen wesentlich größeren Wertebereich umfasst (von -2147483648 bis 2147483647).

Die Vorstellung des „Schachtelmodells“ bezieht sich *nur* auf die Attribute und ihre Attributwerte. Methoden können nicht mit dem Schachtelmodell erklärt werden.

Zusammenfassung

- Die Objekte in Delphi heißen *Komponenten*.
- Alle Komponenten besitzen in jedem Fall *Eigenschaften (Attribute)* und ggf. auch, wenn der Programmierer diese definiert, *Ereignisse (Methoden)*. Eigenschaften und Ereignisse variieren je nach Komponente. Manche Eigenschaften und Ereignisse kommen bei verschiedenen/mehreren Komponenten vor (z. B. hat jede Komponente eine Eigenschaft „Name“), andere hingegen sind komponentenspezifisch.



- Eine Übersicht über (fast) alle Eigenschaften und Ereignisse einer Komponente findet sich – nachdem man die Komponente angeklickt (markiert) hat – im *Objektinspektor*.
- Es gilt, den Eigenschaften und Ereignissen einer Komponente die jeweils gewünschten *Werte* zuzuweisen.
- Jeder Attributwert ist von einem bestimmten *Datentyp*, z. B. Integer. Je nach der Größe des Datentyps, d. h. wie viele Werte durch ihn dargestellt werden können, wird auch mehr oder weniger Speicherplatz benötigt.
- Die Attribut-Zuweisungen erfolgen entweder vor Programmstart direkt im Objektinspektor oder – für Änderungen während des Programmablaufs – im Programmcode.
- Man kann sich Attribute (allgemein: Variablen) mit Hilfe des „Schachtelmodells“ vorstellen.