

GroupBox

Eine `GroupBox` ist ein Container zur visuellen Gruppierung von Controls.

Sie dient hauptsächlich dazu, zusammengehörige Eingabefelder, Optionen oder Steuerelemente optisch voneinander abzugrenzen.

Der Text der `GroupBox` wird als Überschrift im Rahmen dargestellt

Basics

Das `GroupBox` selbst enthält keine besondere Logik.

- `GroupBox` → Container mit Beschriftung
- enthaltene Controls → eigentlicher Inhalt

GroupBox erstellen

```
# Klassisch
$groupBox = New-Object System.Windows.Forms.GroupBox

# .NET-Style
$groupBox = [System.Windows.Forms.GroupBox]::new()
```

Controls hinzufügen

Controls werden über die `Controls`-Collection hinzugefügt.

```
$groupBox.Controls.Add($textBox)

$groupBox.Controls.AddRange @(
    $label,
    $button
)
```

Controls entfernen

```
$groupBox.Controls.Remove($textBox)
```

```
$groupBox.Controls.Clear()
```

Eigenschaften

Eigenschaften / Property's

- **Text** - Überschrift der GroupBox
- **Controls** - Enthaltene Controls
- **Anchor** - Verankerung an den Rändern des Parent-Containers
- **Dock** - Automatische Ausrichtung im Parent-Container
- **AutoSize** - Größe automatisch an Inhalt anpassen
- **Enabled** - Aktiviert oder deaktiviert enthaltene Controls
- **Visible** - Sichtbarkeit der GroupBox
- **Font** - Schriftart der Überschrift
- **ForeColor** - Farbe der Überschrift
- **Padding** - Innenabstand für enthaltene Controls

Text [System.String]

Der Wert von `Text` bestimmt die Beschriftung der GroupBox.

Standardmäßig ist der Wert leer.

```
$groupBox.Text = "Office Installation"
```

Controls

[System.Windows.Forms.Control.ControlCollection]

Enthält alle Controls, die sich innerhalb der GroupBox befinden.

```
$groupBox.Controls.Add($button)
```

Enabled [System.Boolean]

Legt fest, ob die GroupBox aktiviert ist.

Wird `Enabled` auf `false` gesetzt, werden auch alle enthaltenen Controls deaktiviert.

```
$groupBox.Enabled = $false
```

Padding [System.Windows.Forms.Padding]

Legt den Innenabstand fest, der zwischen Rahmen und enthaltenen Controls eingehalten wird.

```
$groupBox.Padding = 10
```

Methoden

GroupBox.Controls

Methoden - Controls

- **Add** - Fügt ein Control hinzu
- **AddRange** - Fügt mehrere Controls hinzu
- **Remove** - Entfernt ein Control
- **Clear** - Entfernt alle Controls

Add()

```
$_ .Controls.Add($control)
```

Die Methode `Add()` fügt ein Control zur `Controls`-Collection der GroupBox hinzu.

AddRange()

```
$_ .Controls.AddRange(@(  
    $label,  
    $textbox,  
    $button  
))
```

Die Methode `AddRange()` fügt mehrere Controls gleichzeitig zur `Controls`-Collection hinzu.

Remove()

```
$.Controls.Remove($control)
```

Die Methode `Remove()` entfernt ein bestimmtes Control aus der GroupBox.

Clear()

```
$.Controls.Clear()
```

Die Methode `Clear()` entfernt alle enthaltenen Controls.

Events

Events

- **Click** - Mausklick auf die GroupBox
- **DoubleClick** - Doppelklick auf die GroupBox
- **MouseDown** - Maustaste wurde gedrückt
- **MouseUp** - Maustaste wurde losgelassen
- **MouseMove** - Maus wurde bewegt
- **MouseEnter** - Mauszeiger betritt die GroupBox
- **MouseLeave** - Mauszeiger verlässt die GroupBox
- **Enter** - Fokus betritt die GroupBox
- **Leave** - Fokus verlässt die GroupBox
- **ControlAdded** - Ein Control wurde hinzugefügt
- **ControlRemoved** - Ein Control wurde entfernt
- **Resize** - Größe wurde geändert
- **Paint** - GroupBox wird neu gezeichnet

```
$groupBox.Add_*({  
    param($sender, $e)  
})
```

- `$sender` → Die GroupBox selbst (`$this`)
- `$e` → EventArgs des jeweiligen Events

ControlAdded / ControlRemoved

Werden ausgelöst, wenn Controls zur `Controls`-Collection hinzugefügt oder daraus entfernt werden.

```
$groupBox.Add_ControlAdded({
    param($sender, $e)

    Write-Host "$($e.Control.Name) wurde hinzugefügt"
})
```

Tipps & Tricks

Typische Stolperfallen

- **Controls erscheinen nicht**
 - Position liegt außerhalb der GroupBox
- **Alle Controls werden deaktiviert**
 - `GroupBox.Enabled = $false`
- **Padding erzeugt kein automatisches Layout**
 - Controls müssen weiterhin selbst positioniert werden
- **GroupBox für Layout verwendet**
 - Für komplexe Layouts meist besser: `Panel`, `FlowLayoutPanel` oder `TableLayoutPanel`

Mentales Modell

Die `GroupBox` ist ein **Container mit Beschriftung**.

Sie gruppiert Controls optisch und logisch, besitzt jedoch keine eigene Inhaltslogik.

Wann sinnvoll?

- Einstellungen gruppieren
- Formulare strukturieren
- Optionen zusammenfassen
- RadioButtons logisch gruppieren

Wann vermeiden?

- Komplexe Layouts
 - Scrollbare Bereiche
 - Dynamische Containerlogik
 - Wenn lediglich ein Rahmen benötigt wird
-

Revision #5

Created 2026-06-06 05:29:50 UTC by John-Andreas Borinas

Updated 2026-06-11 11:08:21 UTC by John-Andreas Borinas