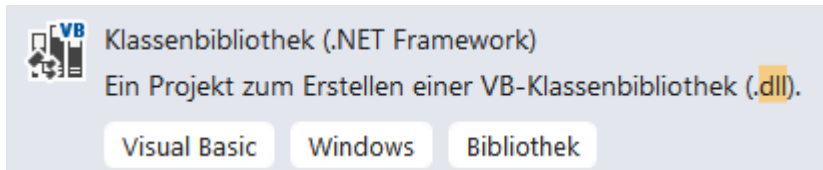




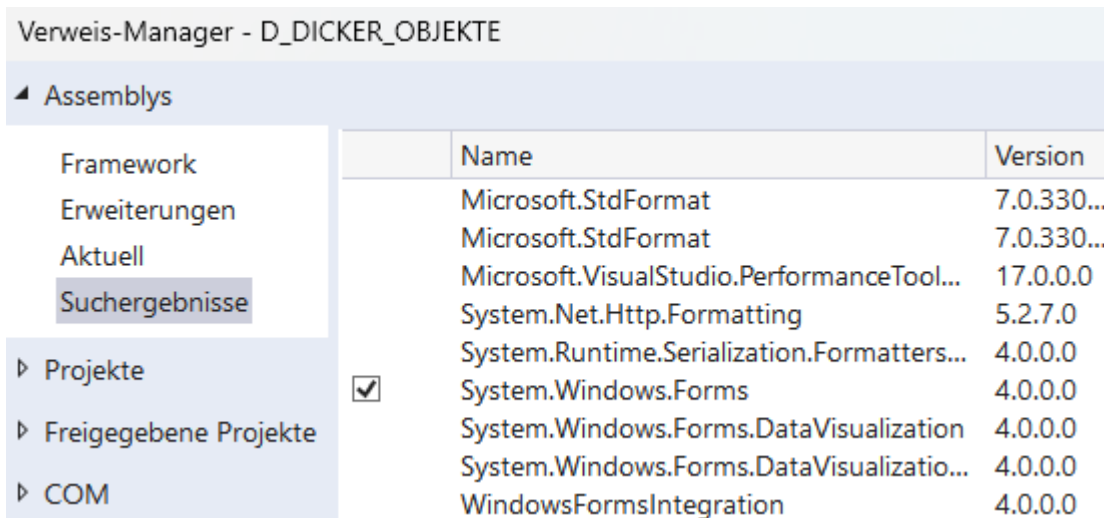
DLL-Projekt

Speicherort > P:\VISUAL STUDIO - DD_DLL__KURS_01\D_DICKER_OBJEKTE

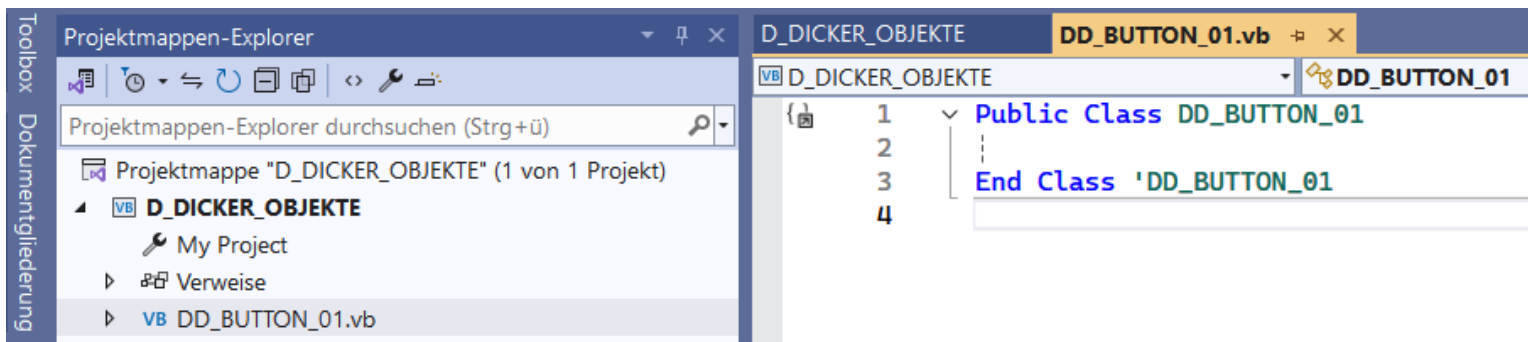
Neues **DLL-Projekt** in Visual Studio erstellen (hier: D_DICKER_OBJEKTE) mit ...



Zu den Verweisen muss **System.Windows.Forms** hinzugefügt werden ...



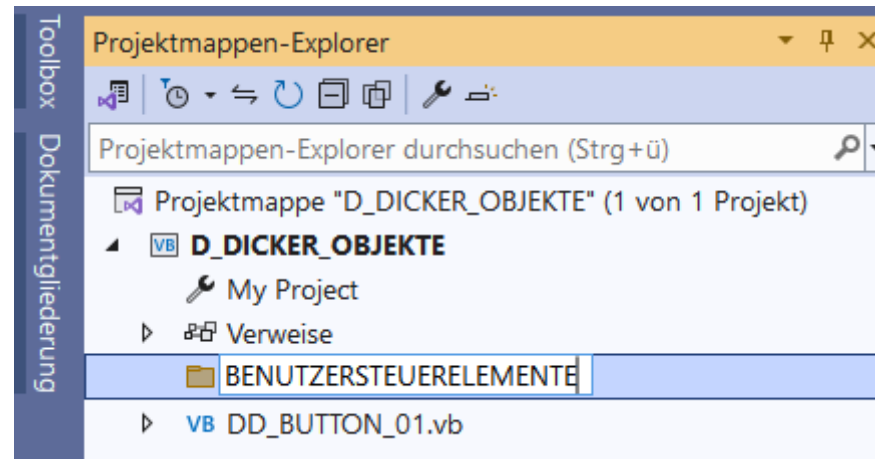
Ich möchte einen neuen graphischen Button (Schaltfläche) mit einen **eigenen Bildhintergrund** entwickeln. Die Klasse in der DLL wird in **DD_BUTTON_01.vb** umbenannt.



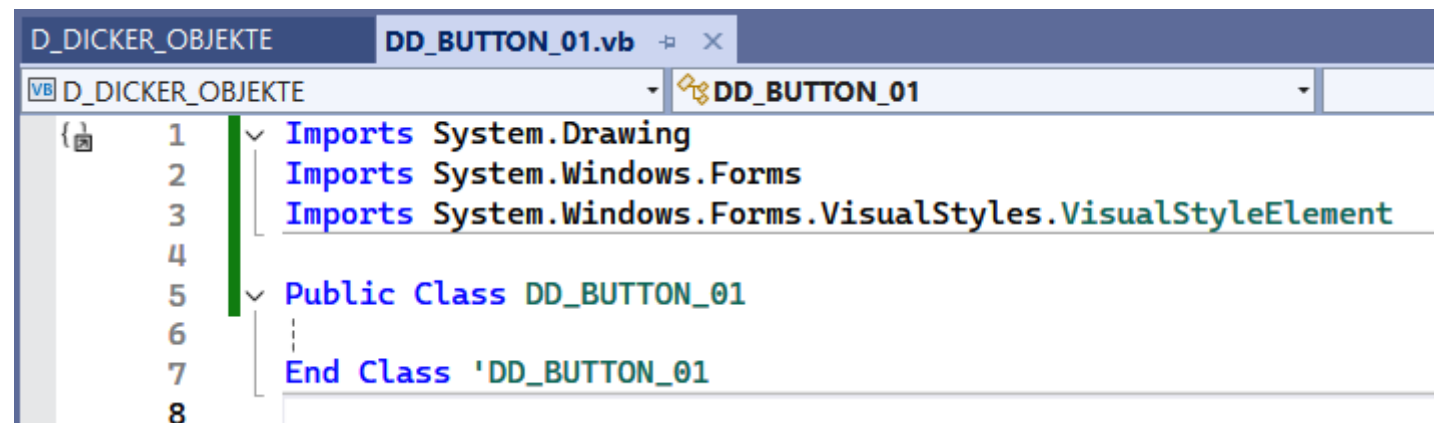


DLL-Projekt

Den Projekt wird ein Ordner **BENUTZERSTEUERELEMENTE** hinzugefügt ...



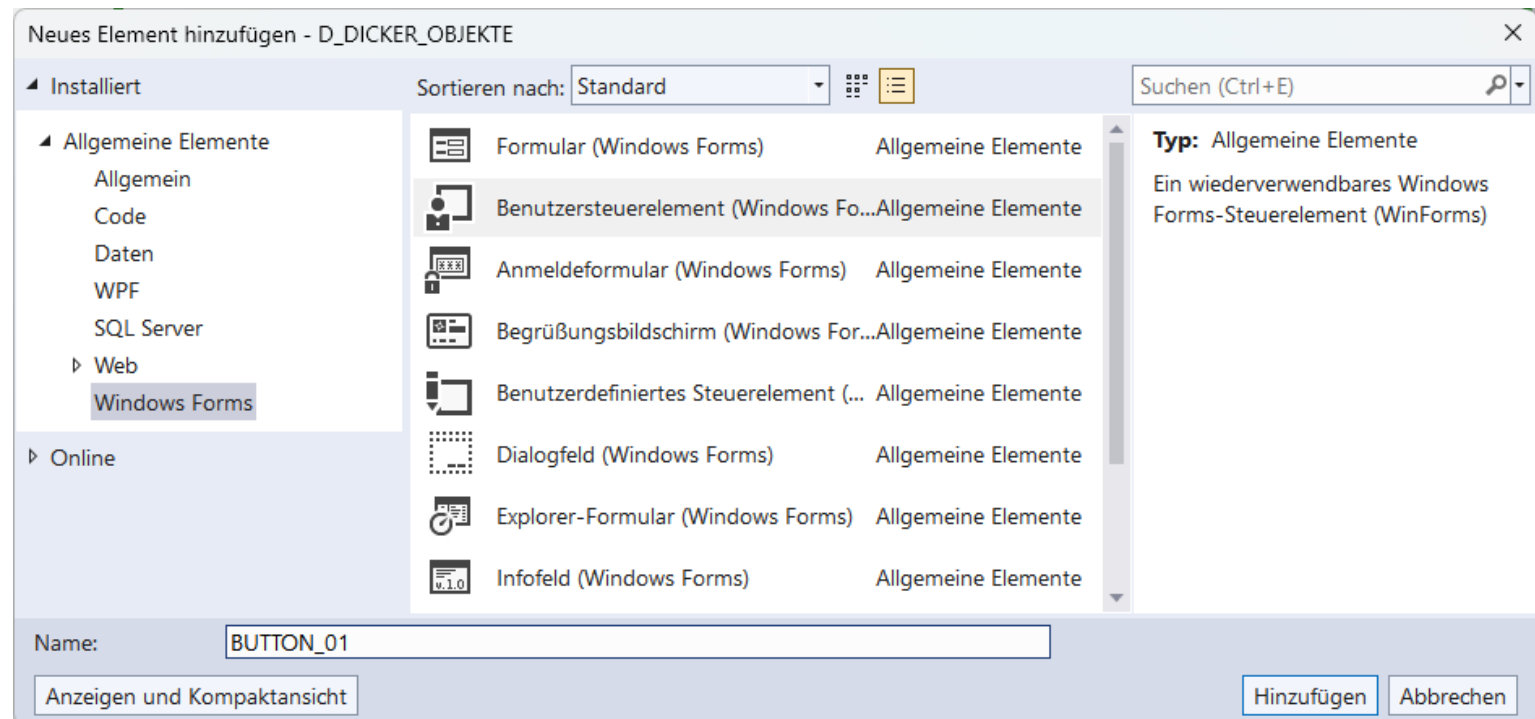
Im Codeeditor werden oben die **drei Imports** Einträge hinzugefügt ...



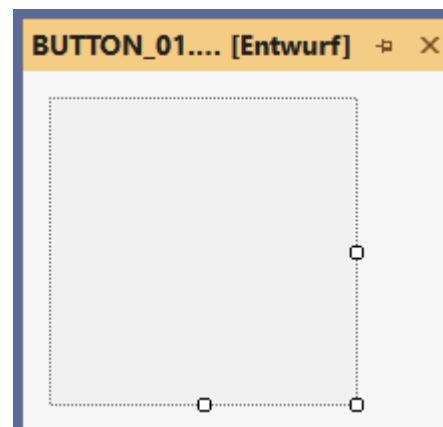


DLL-Projekt

Indem Sie auf den Ordner **BENUTZERSTEUERELEMENTE** mit der rechten Maustaste klicken, eröffnet sich in Visual Studio ein Untermenü. Wählen Sie hier **Hinzufügen > Neues Element ...** Hier klicke ich nun auf **Benutzersteuerelement (Windows Forms Allgemeine Elemente)** und benenne dieses mit den Namen **BUTTON 01 ...**



Es öffnet sich für die Bearbeitung der **Entwurf**. Hier können wir alle Einstellungen vornehmen die wir benötigen um den Button zu erstellen.

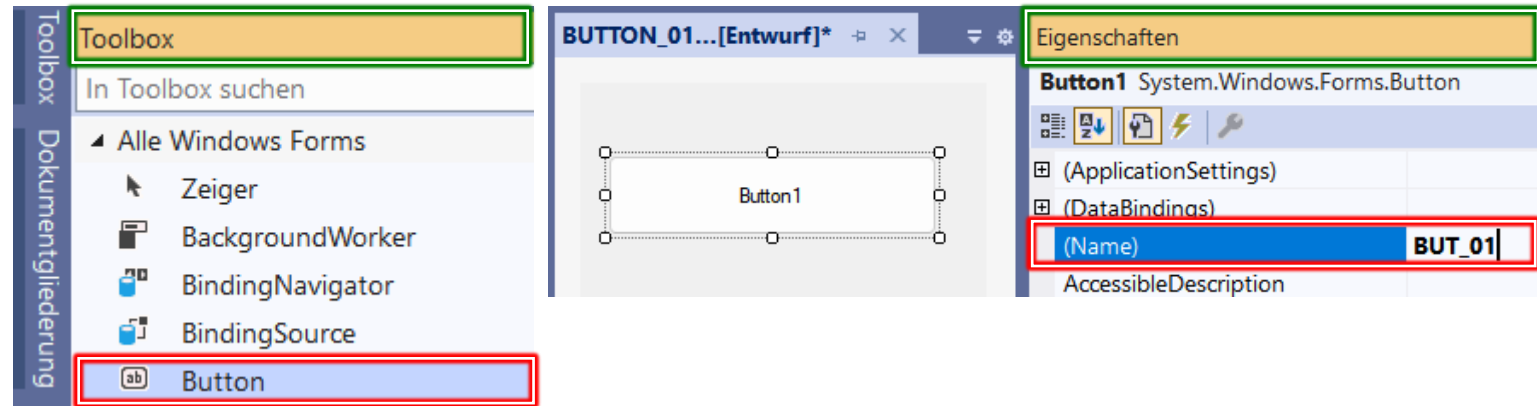




DLL-Projekt

Indem Sie auf den Ordner **BENUTZERSTEUERELEMENTE** mit der rechten Maustaste klicken, eröffnet sich in Visual Studio ein Untermenü. Wählen Sie hier **Hinzufügen > Neues Element ...** Hier klicke ich nun auf **Benutzersteuerelement (Windows Forms Allgemeine Elemente)** und benenne dieses mit den Namen **BUT_01 ...**

Über die **Toolbox** füge ich einen **Button** hinzu. Über **Eigenschaften** formatieren Sie diese Schaltfläche nach Ihren Wünschen ...



Nun benötigen wir zwei **Grafiken** für den Hintergrund der Schaltfläche. Die **erste Grafik** zeigt den Normalzustand, die **zweite Grafik** wird angezeigt, wenn man sich mit der Maus über der Schaltfläche befindet (**MouseHover**). Hierzu habe ich folgende Grafiken angefertigt:



Hintergrund_BUT_01_100x100_Mouse_Hover_Rot.png



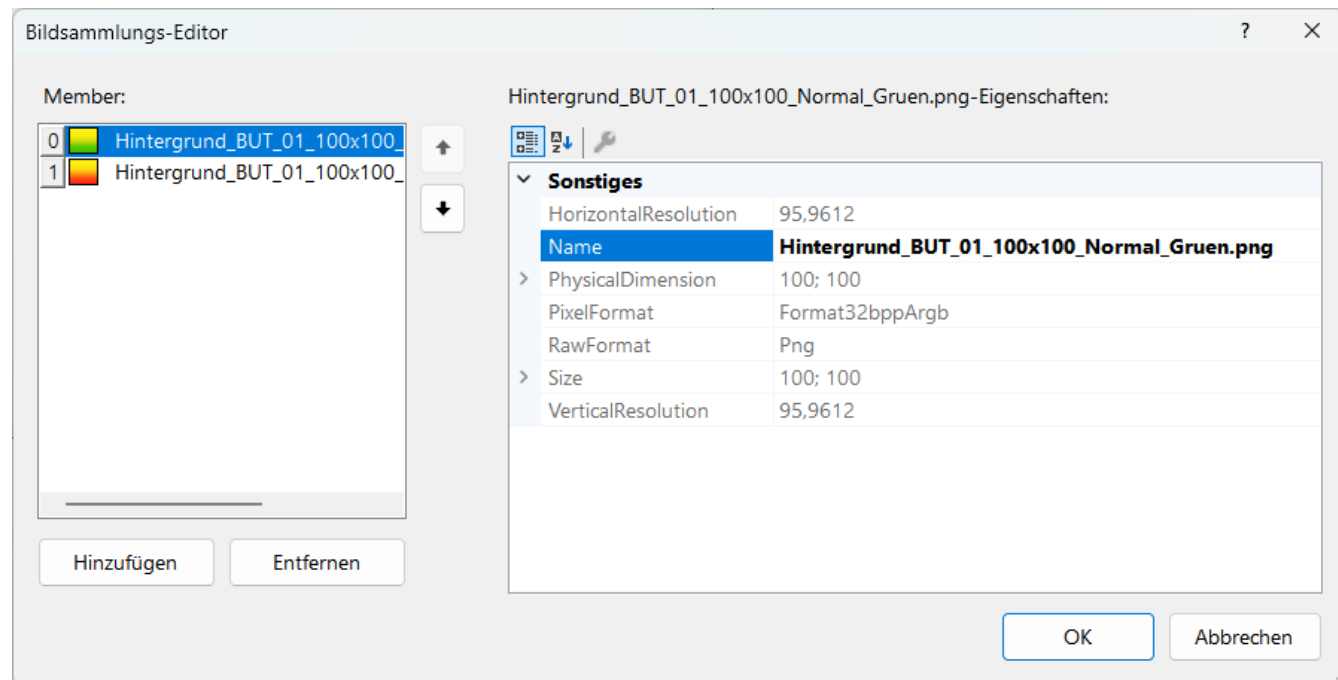
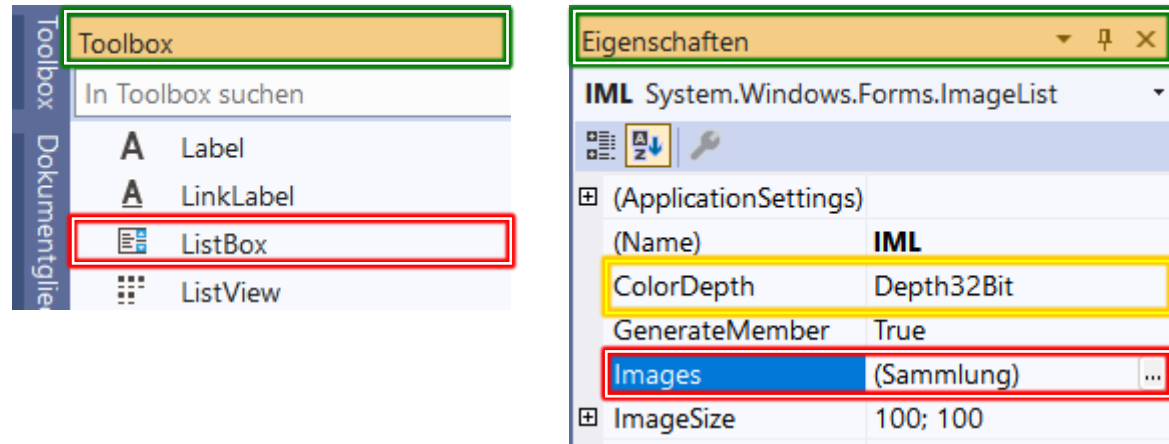
Hintergrund_BUT_01_100x100_Normal_Gruen.png



DLL-Projekt

Nun benötigen wir über die **Toolbox** das Steuerelement **ImageList**. Ziehen Sie diese auf Ihr Benutzersteuerelement. Die **ListBox** habe ich in **IML** unbenannt. Stellen Sie unter **Eigenschaften** die **ColorDepth** auf **Depth32Bit** ein.

Fügen Sie nun die zwei zuvor erstellten Grafiken zur **IML** hinzu. Unter **Eigenschaften** klicken Sie nun unter **Images** auf die Schaltfläche der drei Punkte [...] nach (Sammlung). Nun öffnet sich der **Bildsammlungs-Editor**. Über die Schaltfläche **Hinzufügen**, können Sie die zwei entsprechenden Grafiken auf Ihren Computer zur **IML** Bildersammlung (Member) einfügen.





DLL-Projekt

Wir haben nun folgende **Steuerelemente** in der Projektmappe "D_DICKER_OBJEKTE".

Steuerelemente im Projekt:	
UserControl:	BUTTON_01
Button:	BUT_01
ImageList:	IML

Sie können das Projekt jederzeit testen, indem Sie in Visual Studio in der Entwicklungsumgebung im Menü den Menüpunkt **Starten** anklicken. Das Ergebnis sehen Sie im Testcontainer der hier rechts dargestellt ist.

Achtung:

Vorher muss jedoch der Programmcode den einzelnen Steuerelementen zugeordnet werden.

Den Programmcode des Projektes D_DICKER_OBJEKTE sehen Sie auf der folgenden Seite.



DLL-Projekt

Programmcode > Teil 1

```
Imports System.Drawing
Imports System.Windows.Forms
Imports MS.Internal
```

```
Public Class BUTTON_01
```

```
    Private VARS_PRO_DD_BUT_01_TEXT As String = "Schaltfläche"
```

```
    'Bilder ...
    '~~~~~
```

```
    Friend IMG_Normal As Image
```

```
    Friend IMG_Mouse_Hover As Image
```

```
    'Events ...
    '~~~~~
```

```
    Public Event DD_BUTTON_01_CLICK() 'As EventHandler
```

```
    Sub New()
```

```
        'oooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo
```

```
        ' Dieser Aufruf ist für den Designer erforderlich.
```

```
        InitializeComponent()
```

```
        ' Fügen Sie Initialisierungen nach dem InitializeComponent()-Aufruf hinzu.
```

```
        '~~~~~
```

```
        IMG_Normal = IML.Images("Hintergrund_BUT_01_100x100_Normal_Gruen.png")
```

```
        IMG_Mouse_Hover = IML.Images("Hintergrund_BUT_01_100x100_Mouse_Hover_Rot.png")
```

```
    End Sub 'NEW ooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo
```

```
#Region " BENUTZERSTEUERELEMENT " '*****
```

```
    Private Sub BUTTON_Load(sender As Object, e As EventArgs) Handles Me.Load
```

```
        '-----
```

```
        'Ersteinstellung beim Laden der Schaltfläche ...
```

```
        '~~~~~
```

```
        With BUT_01
```

```
            .Text = PRO_DD_BUT_01_TEXT
```

```
            .BackgroundImage = IMG_Normal
```

```
            .BackgroundImageLayout = Windows.Forms.ImageLayout.Stretch
```

```
        End With
```

```
    End Sub 'BUTTON_Load -----
```

```
#End Region 'BENUTZERSTEUERELEMENT *****
```



DLL-Projekt

Programmcode > Teil 2

```
#Region " EVENTS " '*****

Private Sub SUB_Button_MouseClick(sender As Object, e As MouseEventArgs) _
Handles Me.MouseClick,
    BUT_01.Click
    '-----
    'Wichtig: Ereignis DD_BUTTON_01_CLICK auslösen. Code muss im Hautprogramm
    'geschrieben werden ...
    '-----
    RaiseEvent DD_BUTTON_01_CLICK()
End Sub 'SUB_Button_MouseClick -----

#End Region 'EVENTS *****

#Region " PROPERTY " '*****

Public Property PRO_DD_BUT_01_TEXT As String
    Get
        Return VARS_PRO_DD_BUT_01_TEXT
    End Get
    Set(ByVal value As String)
        VARS_PRO_DD_BUT_01_TEXT = value
    End Set
End Property

Public Property PRO_DD_BUT_01_TEXT_ART_und_GROESSE As Font
    Get
        Return BUT_01.Font
    End Get
    Set(value As Font)
        BUT_01.Font = value
    End Set
End Property

Public Property PRO_DD_BUT_01_TEXT_FARBE As Color
    Get
        Return BUT_01.ForeColor
    End Get
    Set(value As Color)
        BUT_01.ForeColor = value
    End Set
End Property

#End Region 'PROPERTY *****
```




DLL-Projekt

Programmcode > Teil 3

```
#Region " EREIGNISSE " '*****  
  
Private Sub BUT_01_MouseHover(  
sender As Object, e As EventArgs) Handles BUT_01.MouseHover  
    '-----  
    BUT_01.BackgroundImage =  
        IML.Images("Hintergrund_BUT_01_100x100_Mouse_Hover_Rot.png")  
End Sub 'BUT_01_MouseHover -----  
  
Private Sub BUT_01_MouseLeave(  
sender As Object, e As EventArgs) Handles BUT_01.MouseLeave  
    '-----  
    BUT_01.BackgroundImage =  
        IML.Images("Hintergrund_BUT_01_100x100_Normal_Gruen.png")  
End Sub 'BUT_01_MouseLeave -----  
  
#End Region 'EREIGNISSE *****  
  
End Class 'BUTTON_01
```

Das Projekt **D_DICKER_OBJEKT.dll** ist hiermit beendet. Den Programmcode können Sie gerne in Ihre eigenes DLL-Projekt kopieren und nach Ihren Wünschen modifizieren.

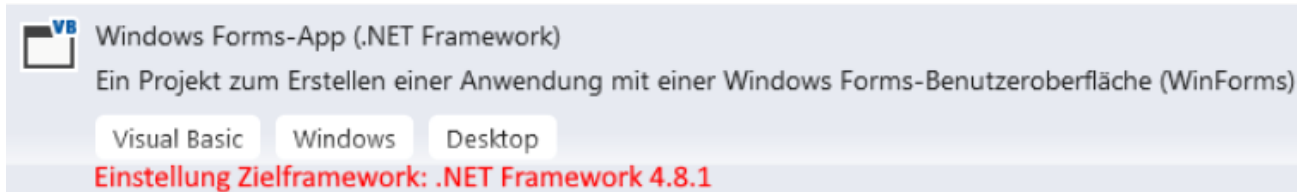
Wie die **Schaltfläche** in einen **Neuen Projekt** verwendet werden kann, sehen Sie auf den folgenden Seiten.



Test-Projekt

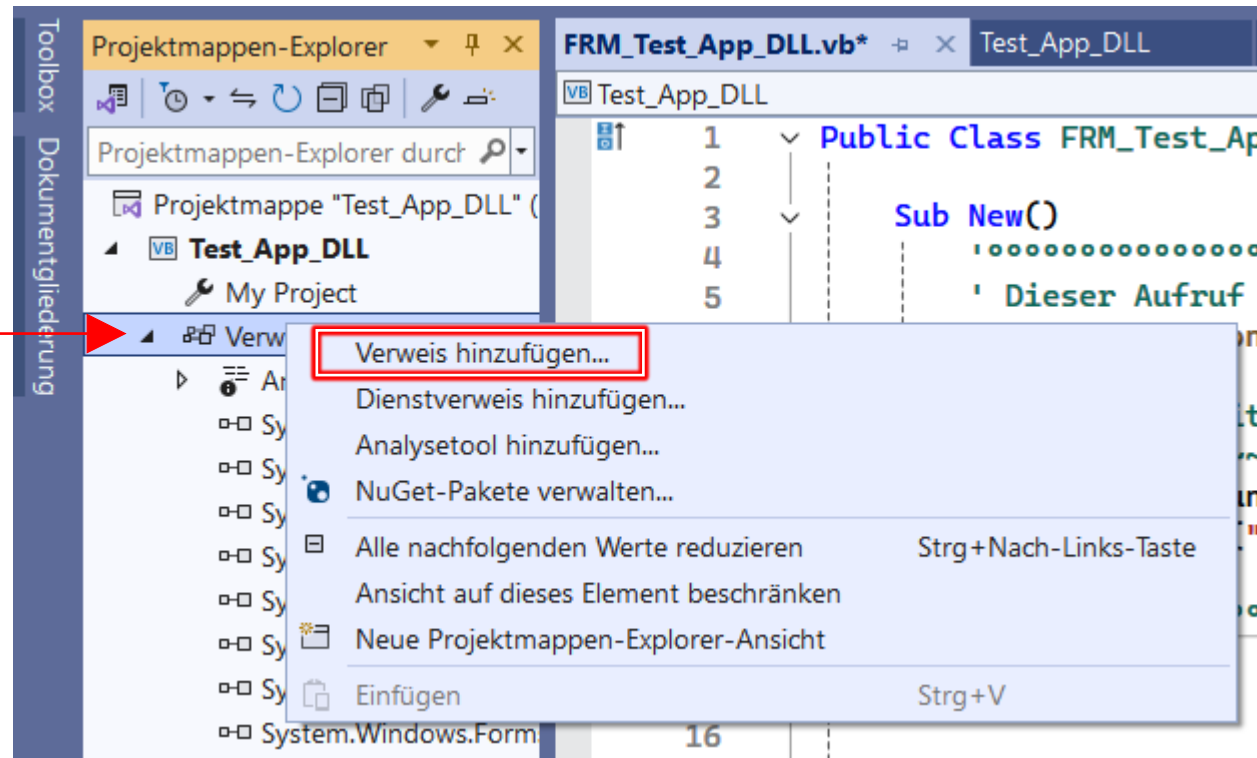
Speicherort > P:\VISUAL STUDIO - DD_DLL__KURS_01\Test_App_DLL

Neues **Test-Projekt** in Visual Studio erstellen (hier: Test_App_DLL) mit ...



Zu den Verweisen muss nun die neue DLL aus dem Projekt **D_DICKER_OBJEKTE** hinzugefügt werden. In DLL-Projekt lautet hier der Pfad **P:\VISUAL STUDIO - DD_DLL__KURS_01\D_DICKER_OBJEKTE\bin\Debug**. Die DLL-Datei die einzubinden ist nennt sich im Beispiel **D_DICKER_OBJEKTE.dll**.

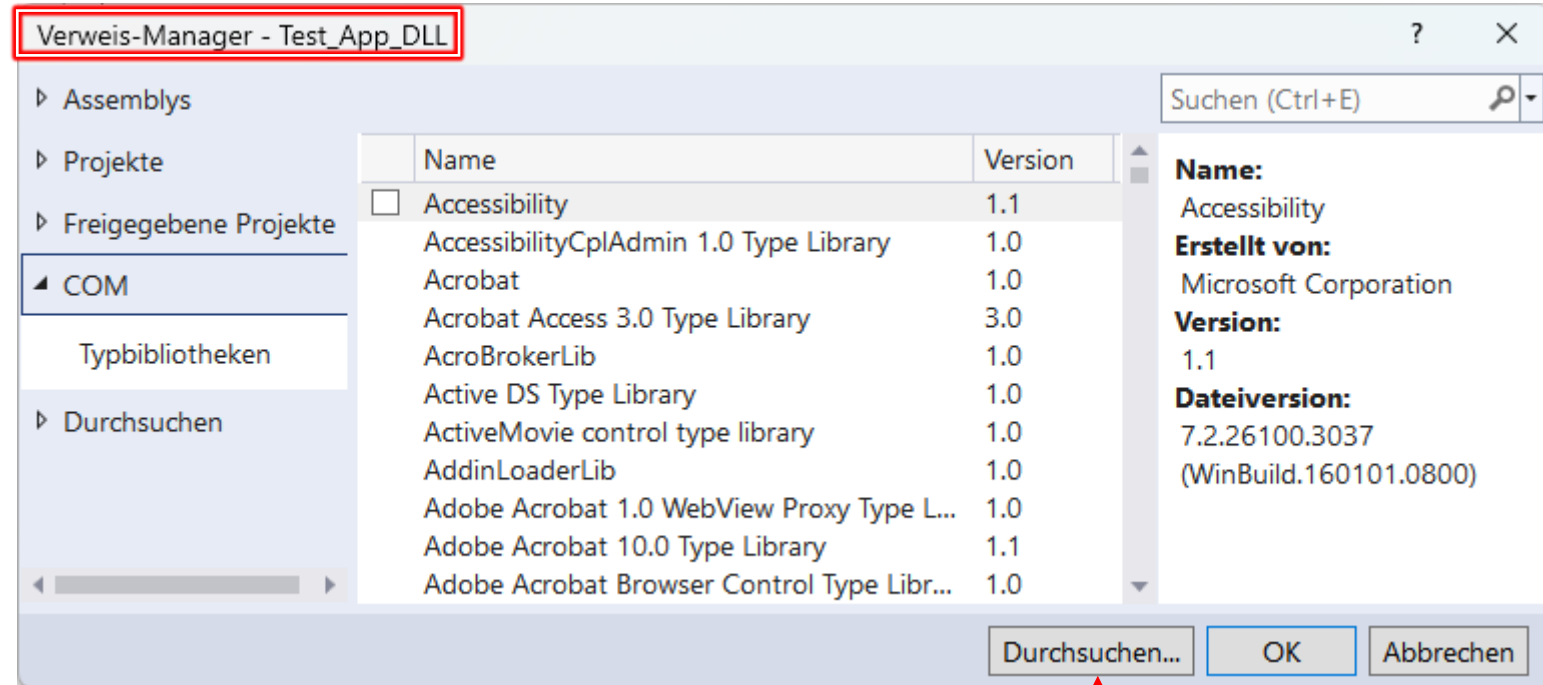
Klicken Sie hier ...





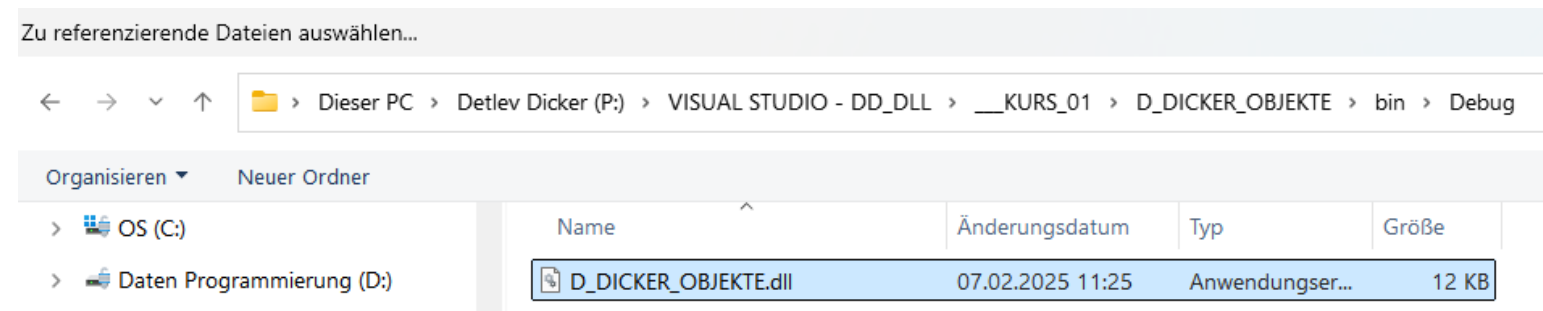
Test-Projekt

Es wird nun der **Verweis-Manager** geöffnet ...



Klicken Sie hier ...

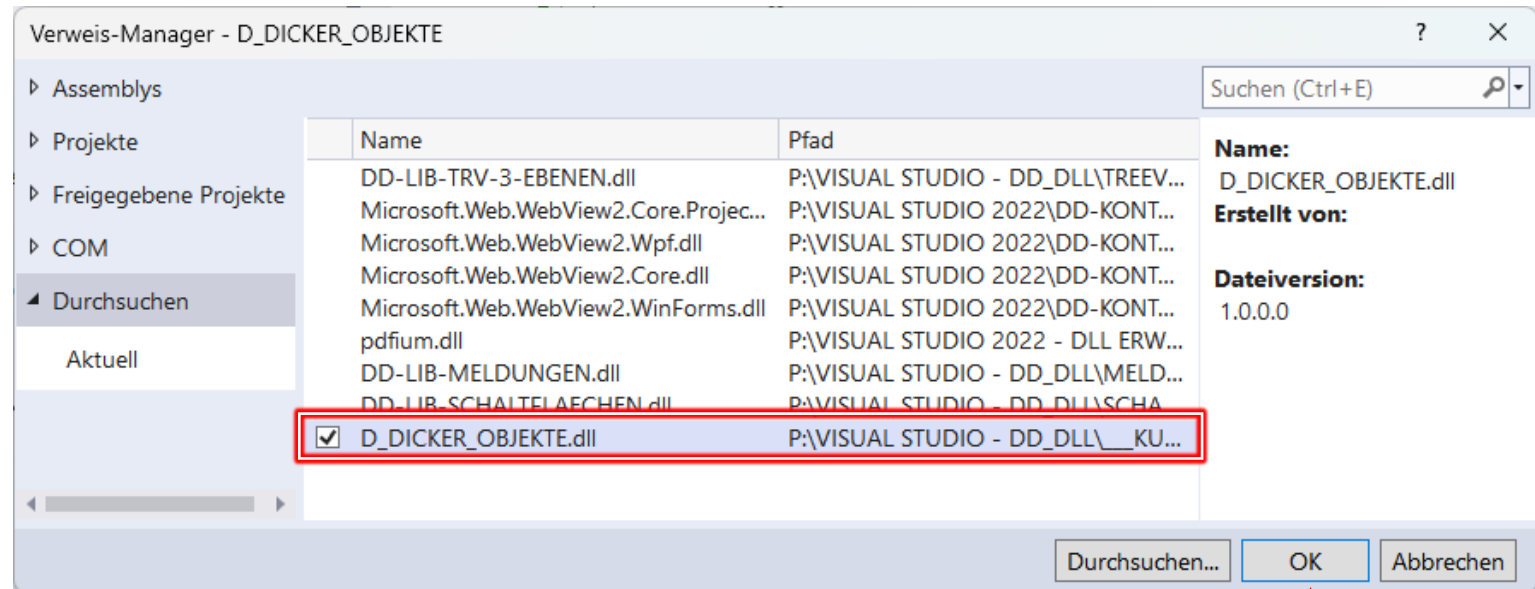
Wählen Sie nun den Pfad Ihres DLL-Projektes aus, in der sich die DLL-Datei befindet und markieren Sie diese.





Test-Projekt

Im Verweis-Manager markieren Sie nun Ihre DLL-Datei

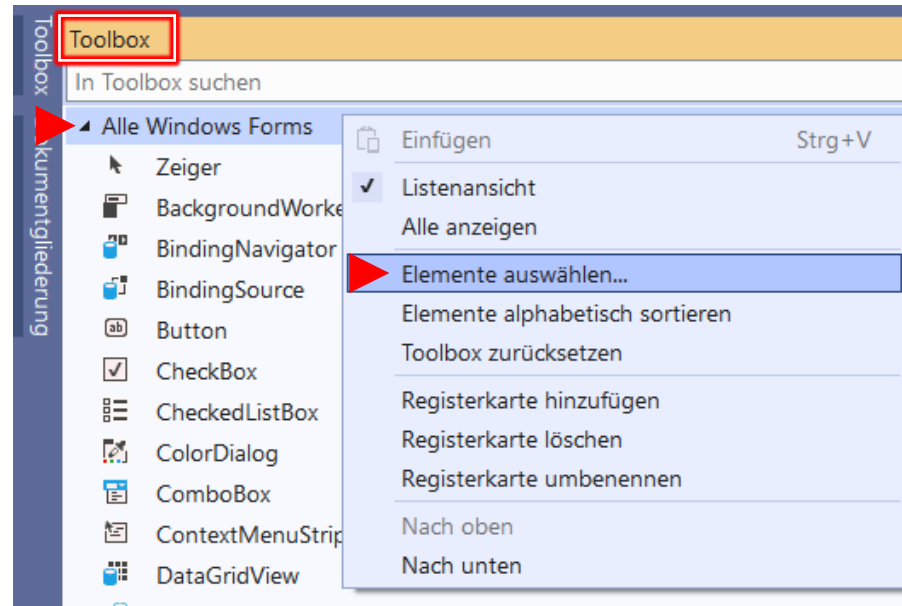


Klicken Sie dann auf OK



Test-Projekt

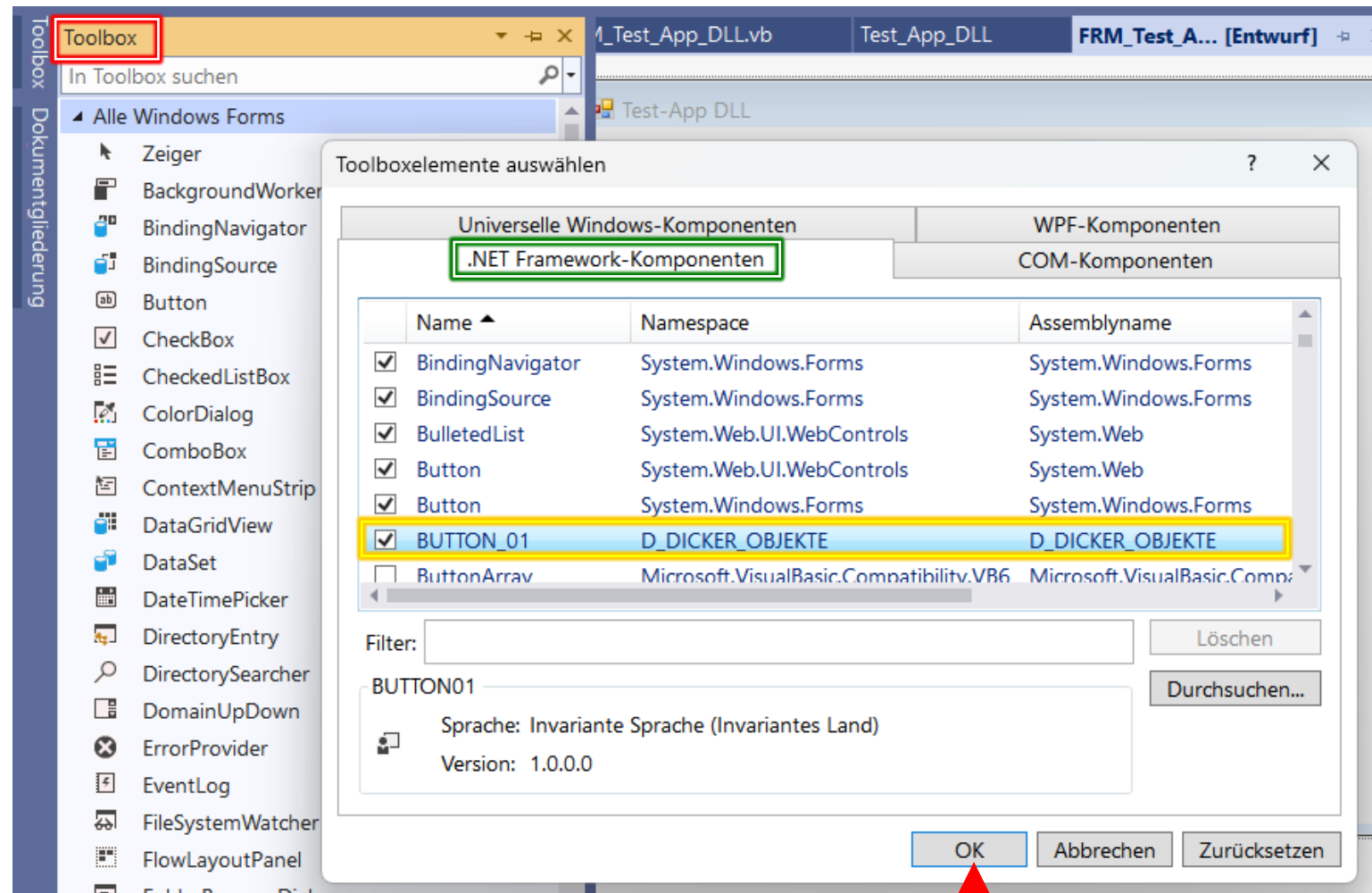
Um die neue Schaltfläche in Ihren Projekt einzubinden und in der **Toolbox** anzuzeigen gehen Sie wie folgt vor. Klicken Sie mit der **rechten Maustaste** auf **Alle Windows Forms**. Wählen Sie dann **Elemente auswählen...**, die Sie mit der linken Maustaste anklicken.





Test-Projekt

Unter .NET Framework-Komponenten müssen Sie nun den Namen **BUTTON_01** auswählen. Achten Sie darauf, dass der Haken im Kästchen gesetzt ist.

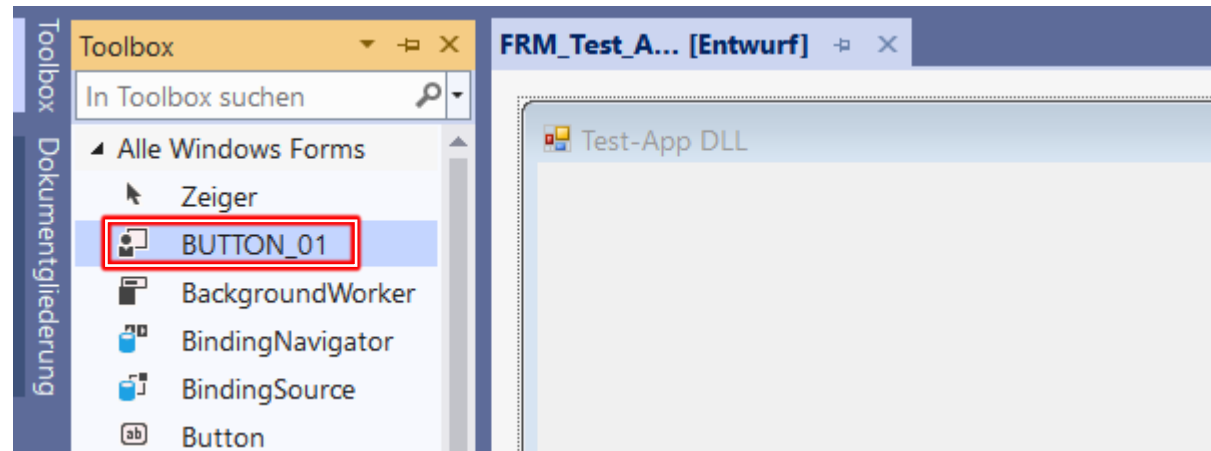


Klicken Sie dann auf OK

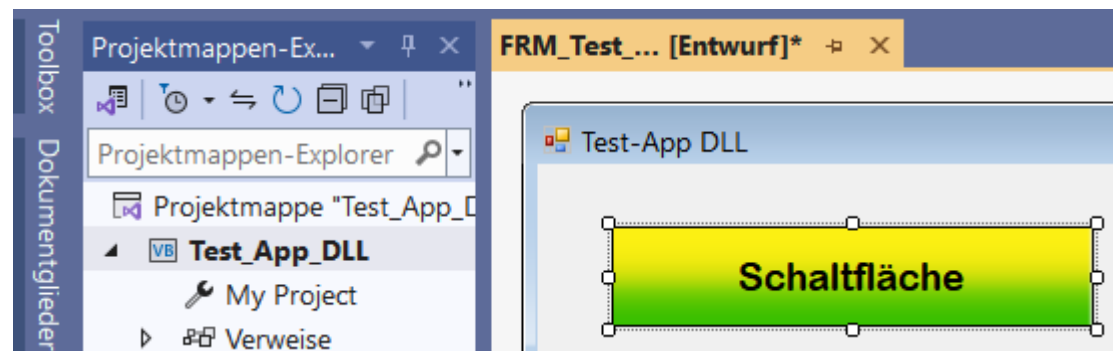


Test-Projekt

Um nun die Schaltfläche zu Ihrem Projekt hinzuzufügen, klicken Sie mit der **linken Maustaste** auf **BUTTON_01** und ziehen Sie (ohne die Maustaste loszulassen) die Schaltfläche auf die **Entwurfs-Oberfläche** Ihres **Formulars**.



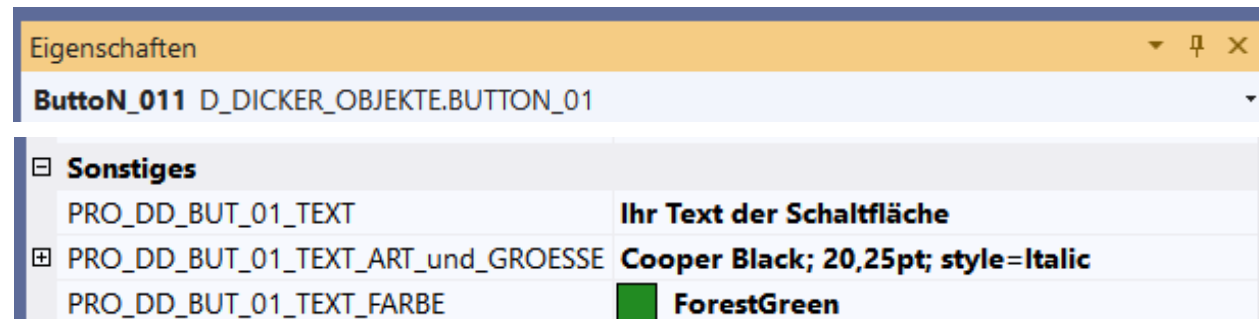
Wenn Sie die **linken Maustaste** nun **loslassen**, befindet sich die Schaltfläche auf Ihrem **Formular** und kann nun unter **Eigenschaften** nach Ihren Wünschen formatiert werden.



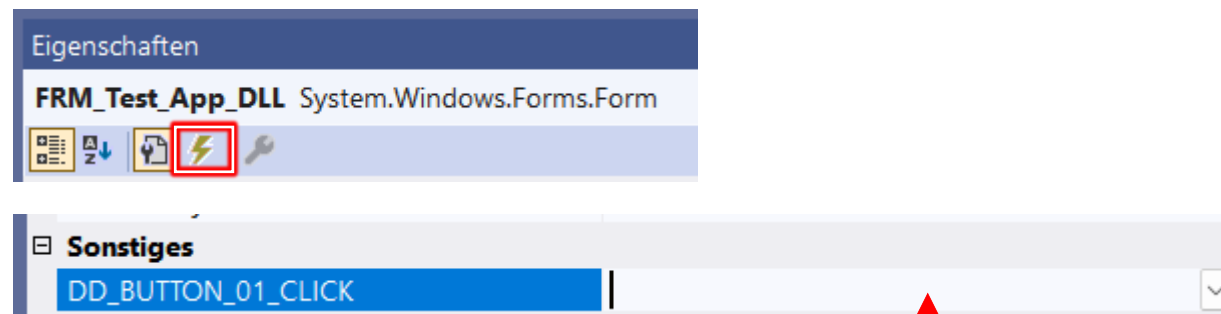


Test-Projekt

Als Beispiel weise ich unter **Eigenschaften** folgende Werte zu:



Es muss unter **Eigenschaften** noch das **Click**-Ereignis hinzugefügt und programmiert werden. Durch **Doppelklick** öffnet sich der Code-Editor und Sie können Ihren Programmcode einfügen.



Doppelklick



Test-Projekt

Der komplette Programmcode des Test-Projektes sieht nun wie folgt aus:

```
Public Class FRM_Test_App_DLL

#Region " EREIGNISSE " '*****

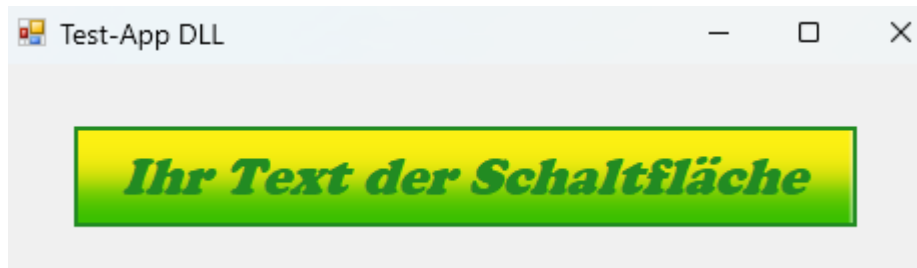
    '-----+
    ' | T E S T |
    '-----+

    Private Sub ButtonN_011_DD_BUTTON_01_CLICK() _
    Handles ButtonN_011.DD_BUTTON_01_CLICK
        '-----
        MessageBox.Show("Test Button_011 > KCLICK ...")
    End Sub 'ButtonN_011_DD_BUTTON_01_CLICK -----

#End Region 'EREIGNISSE *****

End Class 'FRM_Test_App_DLL
```

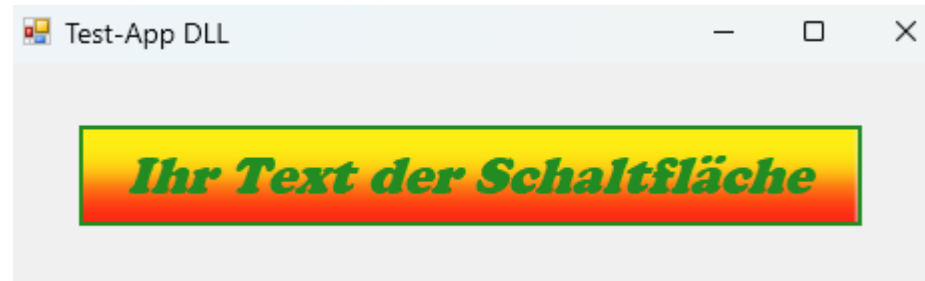
Hier sehen Sie die Programmausführung beim Starten des Test-Projektes:



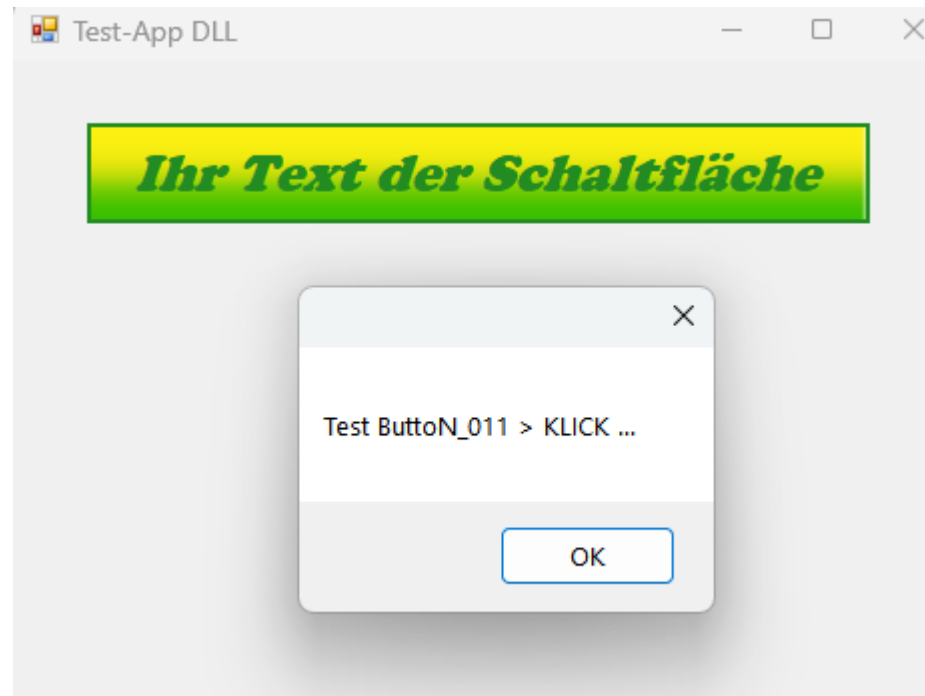


Test-Projekt

Hier, wenn sich der Mauszeiger die Schaltfläche berührt:



Hier, wenn die Schaltfläche angeklickt wurde ...





Ende Kurs 1

Dies ist ein **Beispiel**, wie ein benutzerdefiniertes Steuerelement (hier ein Button) in eine **Vielzahl Ihrer Projekte** verwendet werden kann.

Der DLL-Datei kann jede Menge von Steuerelementen (die Sie erstellen) enthalten.

Ihren Ideen sind keine Grenzen gesetzt ...