

Excel-Optionen über Steuerelemente setzen

Aufgabe:

Es sollen verschiedene Excel-Optionen wie z.B.

- Gitternetzlinien ein / ausblenden
- Ganzer Bildschirm ein / ausblenden
- Statusleiste ein / ausblenden

über Steuerelemente wie z.B.

- Schaltfläche
- Umschaltfläche
- Kontrollkästchen
- rechte Maustaste in der ersten Zeile

gesetzt werden.

Welches Wissen benötigen wir:

- Sind Makros ausführbar (Sicherheitseinstellung)
- Arbeiten mit dem Editor.
- Erstellen eines Steuerelementes.
- Starten einer Prozedur durch eine Schaltfläche.
- Befehle für Excel-Optionen.

Beispiel 1: Schalten der Gitternetzlinien über eine Schaltfläche.

Zeichnen Sie eine Schaltfläche nach gleichem Muster, wie im 1. Beispiel „Fisch schwimmt gegen eine radioaktive Tonne“.

Klicken Sie mit der rechten Maustaste auf die Schaltfläche und wählen Sie aus dem Kontextmenü den Punkt Eigenschaften. Dort setzen Sie die beiden Eigenschaften Name und Caption wie unten gezeigt.



Führen Sie einen Doppelklick auf die Schaltfläche aus und Sie gelangen in den Editor:



Erweitern Sie den vorhandenen Code um folgende Programmzeile:

```
Option Explicit

1 Private Sub CB_Gitternetzlinien_Click()
    ActiveWindow.DisplayGridlines = False
End Sub
```

Zeile	Beschreibung	Siehe Wissensblock
1	Schaltet die Gitternetzlinien aus (False) (Der True-Wert würde die Gitternetzlinien wieder einschalten)	Excel-Optionen

Um die Gitternetzlinien wieder einzuschalten verwenden Sie den True-Wert (Wahr)

Nun können wir mit der Schaltfläche die Gitternetzlinien aber nur so schalten, wie es im Code steht, d.h. ein, wenn der Wert „True“, bzw. aus, wenn der Wert auf „False“ steht.

Da der Befehl `ActiveWindow.DisplayGridlines` selbst einen Wahrheitswert beinhaltet, kann dieser mit dem Not-Befehl gedreht werden. D.h. aus „False“ wird „True“ und aus „True“ wird „False“.

Unsere Befehlszeilen sehen dann wie folgt aus:

```
Option Explicit

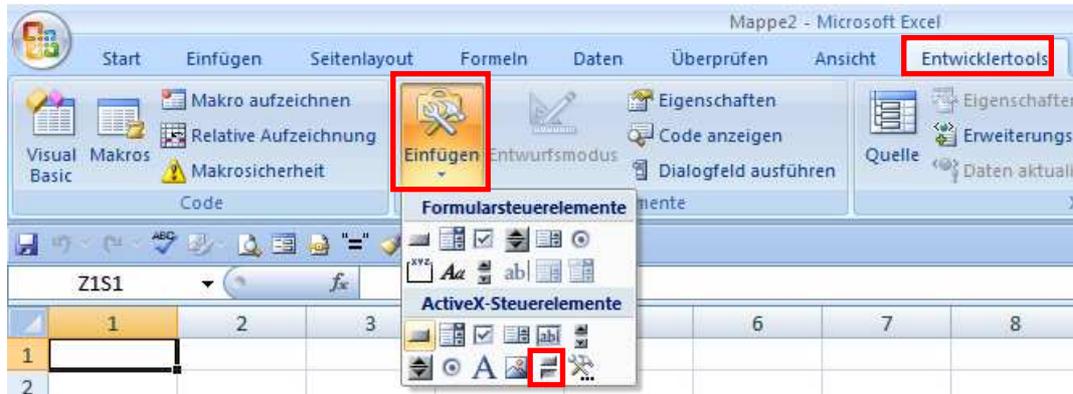
Private Sub CB_Gitternetzlinien_Click()
    ActiveWindow.DisplayGridlines = Not ActiveWindow.DisplayGridlines
End Sub
```

Excel bietet als Schaltfläche nicht nur die gewöhnliche Schaltfläche an, sondern auch eine sogenannte Umschaltfläche. Der Unterschied zwischen diesen beiden Schaltflächen ist, dass die Umschaltfläche 2 verschiedene Zustände aufweisen kann:

- eingedrückt
- nicht eingedrückt.

Diesen Umstand wollen wir nun für unsere Excel-Optionen nehmen. Bleiben wir bei unserem Beispiel Gitternetzlinien, welche wir nun über die Umschaltfläche ein und aus-blenden wollen.

Bei Excel 2007 finden wir die Umschaltfläche unter folgender Position.

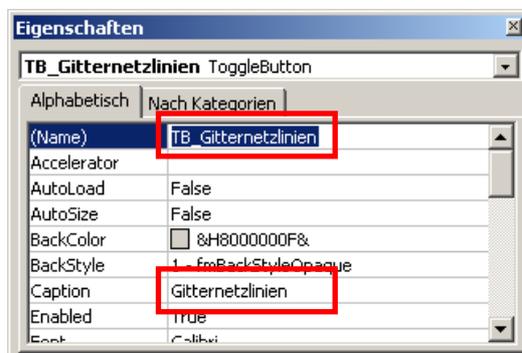


Unter Excel 2003 ist die Umschaltfläche in der Symbolleiste „Steuerelement-Toolbox“ zu finden



Zeichnen Sie nun die Umschaltfläche an einer beliebigen Position im Tabellenblatt und aktivieren Sie das Eigenschaftsfenster (Sie wissen schon: rechte Maustaste auf die Umschaltfläche und die Option „Eigenschaften“ auswählen.)

Dor setzen wir folgende Eigenschaften:



Ein Doppelklick auf die Umschaltfläche und wir erhalten wieder den Editor mit folgendem Aussehen:



Da wir den Schaltzustand der Umschaltfläche (übrigs auch ToggleButton genannt) für das Setzen der Gitternetzlinien nutzen wollen, müssen wir den Code um folgende Zeile ergänzen:

```
Option Explicit

Private Sub TB_Gitternetzlinien_Click()
1   ActiveWindow.DisplayGridlines = TB_Gitternetzlinien.Value
End Sub
```

Wenn wir noch die Beschriftung der Umschaltfläche ändern wollen, setzen wir folgenden Programmcode ein:

```
Private Sub TB_Gitternetzlinien_Click()
1   ActiveWindow.DisplayGridlines = TB_Gitternetzlinien.Value
2   If TB_Gitternetzlinien = True Then
3       TB_Gitternetzlinien.Caption = "Gitternetzlinien aus"
4   Else
5       TB_Gitternetzlinien.Caption = "Gitternetzlinien ein"
6   End If
End Sub
```

Zeile	Beschreibung	Siehe Wissensblock
1	Wenn die Gitternetzlinien eingeschaltet sind	If-Bedingung
2	dann wird in die Umschaltfläche der angegebene Text geschrieben.	Excel-Optionen
3	Sonst	
4	wird ein anderer Text geschrieben	Excel-Optionen
5	Ende der If-Bedingung	

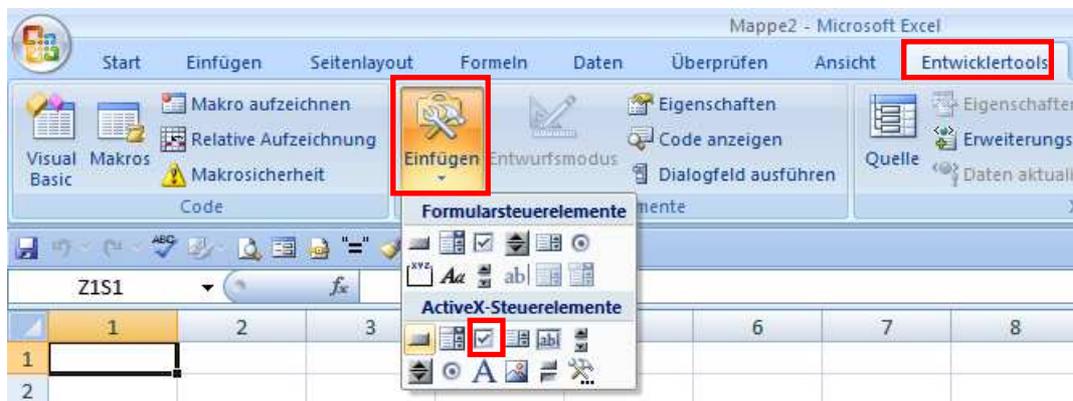
Alternativ können wir auch schreiben:

```
Private Sub TB_Gitternetzlinien_Click()
1   ActiveWindow.DisplayGridlines = TB_Gitternetzlinien.Value
   TB_Gitternetzlinien.Caption = IIf(TB_Gitternetzlinien.Value = True, _
   "Gitternetzlinien aus", _
   "Gitternetzlinien ein")
End Sub
```

Zeile	Beschreibung	Siehe Wissensblock
1	Über die If-Bedingung kann aufgrund einer Bedingung, entweder der eine Wert (wenn die Bedingung zutrifft) oder ein anderer Wert (wenn die Bedingung nicht zutrifft) zugewiesen werden. Dieser Befehl ist eigentlich ein Einzeilen –Befehl. Wenn Sie im Programm-Code 3 Zeilen sehen, liegt dies daran, das die Programmzeile mit den Zeichen „Leerzeichen“ + „ Unterstrich“ umgebrochen worden ist. Man achte strengstens auch das Leerzeichen vor dem Unterstrich.	If-Bedingung

Als nächstes wollen wir uns mal ansehen wie wir die Option „Statusleiste ein / aus“ über ein Kontrollkästchen schalten können.

Das Kontrollkästchen ist unter Excel 2007 hier zu finden:

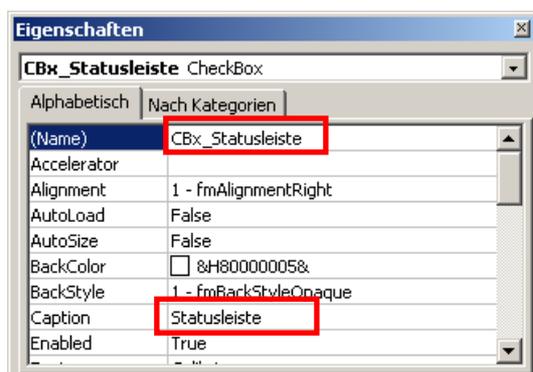


und unter Excel 2003 dort:



Das Zeichen des Kontrollkästchen geschieht in der gleichen Weise wie bei den beiden Schaltflächen-Typen.

Wir setzen dann noch die Eigenschaften (Genau: die Sache mit der rechten Maustaste auf das Kontrollkästchen):



Nun wieder ein Doppelklick auf die Kontroll-Schaltfläche und wir sind im Editor:



Wir fügen nun folgende Programmzeile hinzu:

```

Option Explicit

Private Sub CBx_Statusleiste_Click()
1   Application.DisplayStatusBar = CBx_Statusleiste.Value
End Sub

```

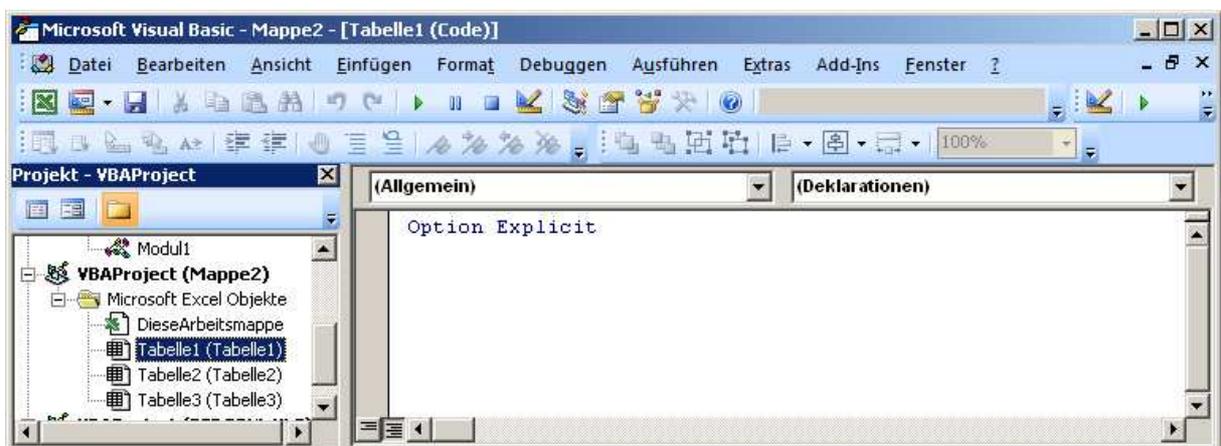
Zeile	Beschreibung	Siehe Wissensblock
1	Schaltet die Statuszeile über den Zustand des Kontrollkästchen	Excel-Optionen

Als nächstes wollen wir die Option „Ganzer Bildschirm“ über einen Doppelklick in einer Zelle der 1. Zeile eines Tabellenblattes schalten.

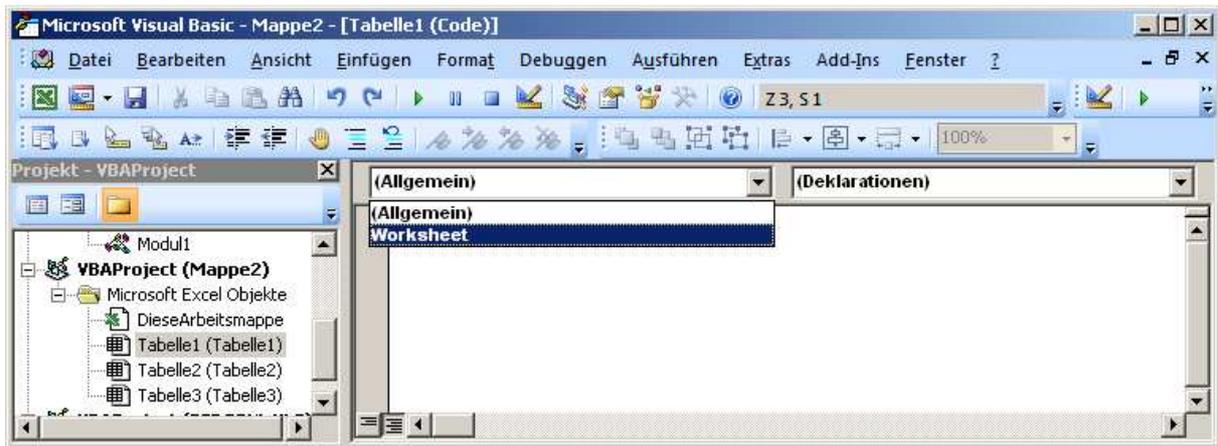
Hierzu gehen wir wie folgt vor:

Ich gehe davon aus, dass wir uns in einer Excel-Tabelle befinden und drücken die Tastenkombination **Alt** + **F11**.

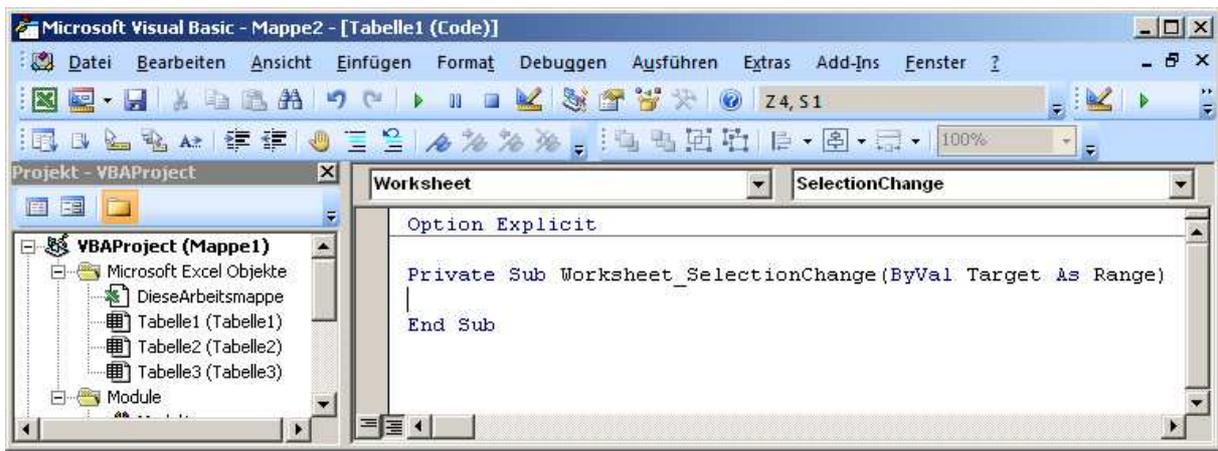
Nun befinden wir uns wieder im Editor mit wahrscheinlich folgendem Bild:



Sicherheitshalber klicken Sie auf das Objekt „Tabelle1 (Tabelle1)“ doppelt (siehe oben) und wählen aus der 1. Auswahlliste die Option „Worksheet“, siehe unten.

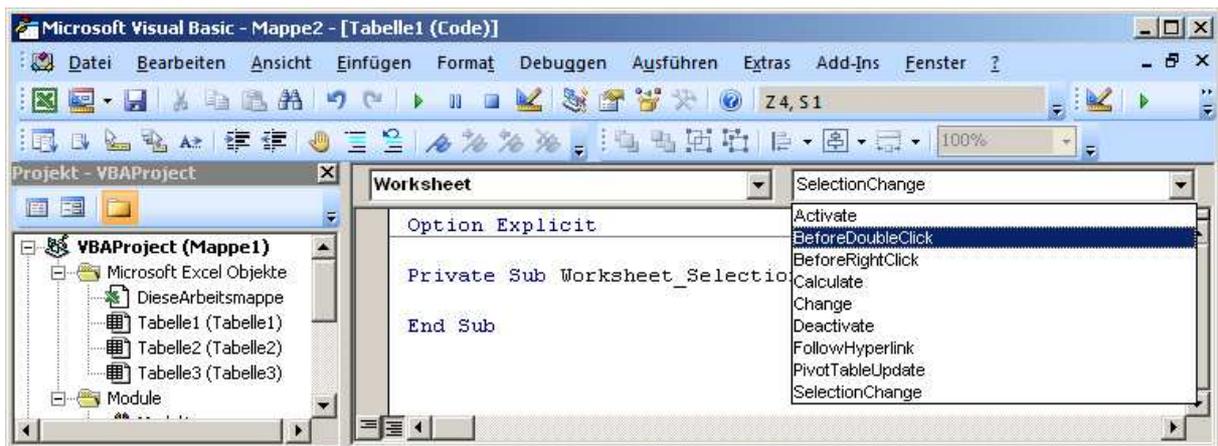


Daraufhin erhalten wir folgendes Bild:

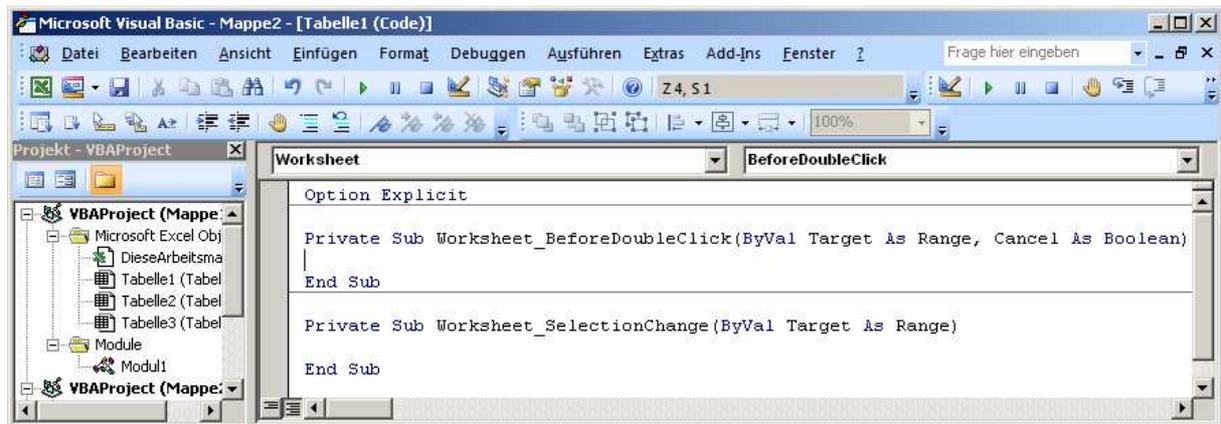


Die erschiene Prozedur wird dann gestartet wenn eine Auswähländerung in der Tabelle erfolgt.

Dies ist ein Ereignis, welches wir nicht benötigen und wählen deshalb aus dem zweiten Auswahlfeld das Ereignis „BeforeDoubleClick“



Unsere Prozedur sieht nun wie folgt aus.



Die untere Prozedur „Sub Worksheet_SelectionChange“ wird gelöscht und unsere neue Prozedur wird mit folgenden Programmzeilen ergänzt:

```
Option Explicit

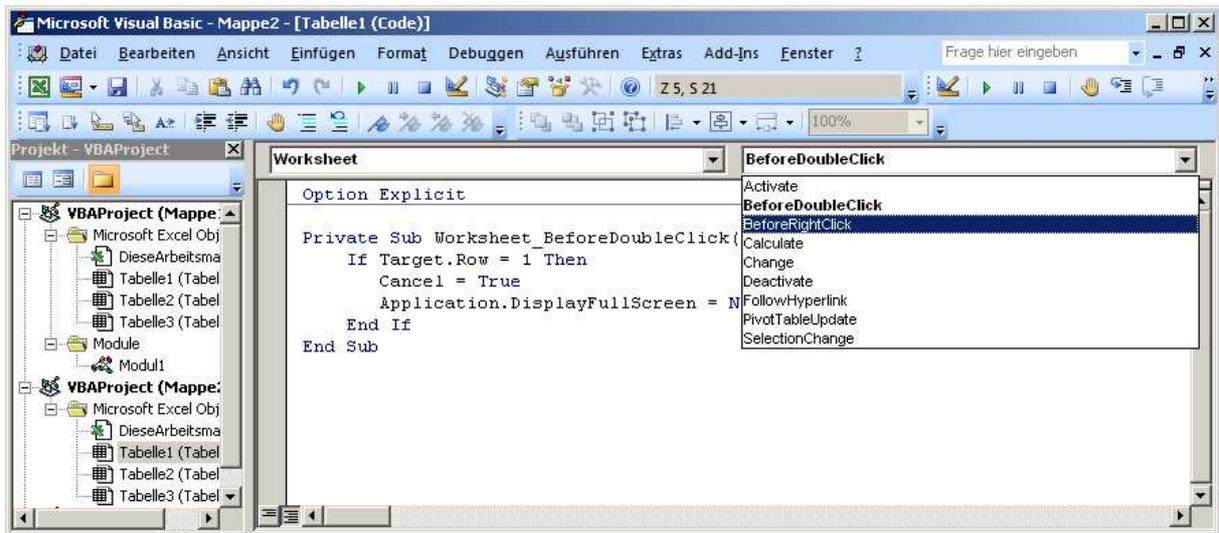
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
1   If Target.Row = 1 Then
2       Cancel = True
3       Application.DisplayFullScreen = Not Application.DisplayFullScreen
4   End If
End Sub
```

Zeile	Beschreibung	Siehe Wissensblock
1	Target ist ein Objekt von Typ Range (also ein Bereich) und hat mehrere Methoden und Eigenschaften. Die Eigenschaft Target.Row gibt uns die Zeile wieder, in der ein Doppelklick stattgefunden hat. Die Bedingung ist also dann erfüllt wenn in Zeile 1 ein Doppelklick stattgefunden hat.	
2	Der Befehl Cancel=True verhindert, dass beim Doppelklick in den Änderungsmodus der Zelle gewechselt wird.	
3	Die Option „Ganzer Bildschirm“ wird in einen Zustand gebracht, wie er vorher nicht war (Umdreh- oder Schalter-Funktion)	
4	Ende der If-Bedingung	

Beim wiederholten Doppelklick in Zeile 1, sollte nun die Option „Ganzer Bildschirm“ ein / aus – geschaltet werden können.

Als Abschluss gehen wir noch der Frage nach, ob es möglich ist, durch einen Rechtsklick mit der Maus in Zeile 1 den Explorer zu starten und das Verzeichnis zu listen, in welchem sich unsere aktuelle Mappe befindet. **Achten Sie bitte darauf, dass Sie Ihre Mappe auch wirklich abgespeichert haben !!**

Wir wechseln wieder in den Explorer (Alt + F11) und wählen in der rechten Auswahlliste den Eintrag „BeforeRightClick“ aus, vorausgesetzt, dass in der linken Liste noch „Worksheet“ steht:



Die neue Prozedur erweitern wir mit folgenden Zeilen:

```

Private Sub Worksheet_BeforeRightClick(ByVal Target As Range, Cancel As Boolean)
1   Dim Programm      As String
2   Dim Verzeichnis  As String

3   Programm = "Explorer.exe /e, "
4   Verzeichnis = Application.ThisWorkbook.Path
5   If Target.Row = 1 Then
6       Cancel = True
7       Shell Programm & Verzeichnis, vbMaximizedFocus
8   End If
End Sub

```

Zeile	Beschreibung	Siehe Wissensblock
1	Variable für das Programm, welches gestartet werden soll	Variablen
2	Variable für das Verzeichnis, welches gelistet werden soll.	
3	Zuweisung der Variable Programm. Der Zusatz „/e, “ bedeutet, dass der Verzeichnisbaum im Explorer erscheint.	
4	Zuweisung der Variable Verzeichnis	
5	Wenn die Zeile 1 einen Rechtsklick erfahren hat ... dann	
6	wird das normale Verhalten eines Rechtsklick in einer Zelle (erscheinen des Kontextmenüs) deaktiviert.	
7	starten des Programms über die Funktion Shell vbMaximizedFocus bedeutet, dass der Explorer bildschirmfüllend erscheint.	
8	Ende der If-Anweisung	

Liste der Befehle für das Ansprechen einiger Excel-Optionen

Excel-Objekte	Befehl
Menüzeile	Application.CommandBars(1).Enable = True / False
Symbol-Leisten	Application.CommandBars(<Nr>).Visible = True / False
Ganzer Bildschirm	Application.DisplayFullScreen = True / False
Gitternetzlinien ein / aus	ActiveWindow.DisplayGridlines = True / False
Blattregisterkarten ein / aus	ActiveWindow.DisplayWorkbookTabs = True / False
Horizontale Bildlaufleiste ein / aus	ActiveWindow.DisplayHorizontalScrollBar = True / False
Vertikale Bildlaufleiste ein / aus	ActiveWindow.DisplayVerticalScrollBar = True / False
Formelansicht ein / aus	ActiveWindow.DisplayFormulas = True / False
Seitenumbruch ein / aus	ActiveSheet.DisplayAutomaticPageBreaks = True / False
Zeilen und Spaltenüberschriften ein / aus	ActiveWindow.DisplayHeadings = True / False
Gliederungssymbole ein / aus	ActiveWindow.DisplayOutline = True / False
Nullwerte anzeigen ein / aus	ActiveWindow.DisplayZeros = True / False
Bearbeitungsleiste ein / aus	Application.DisplayFormulaBar = True / False
Statusleiste ein / aus	Application.DisplayStatusBar = True / False
Fenster in Tabs ein / aus	Application.ShowWindowsInTaskbar = True / False
Kommentare	Application.DisplayCommentIndicator = 0 (keine Kommentare) -1 (nur rotes Dreieck anzeigen) 1 (Kommentare anzeigen)