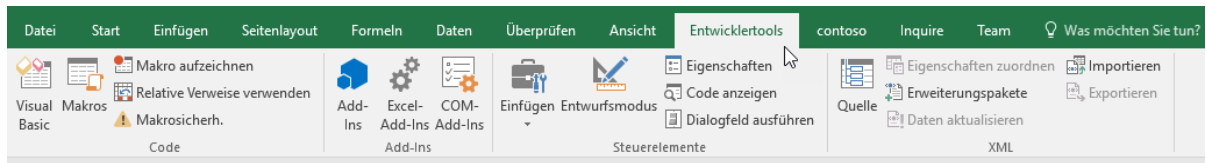


VBA

Voraussetzungen: Registerkarte „Entwicklertools“ einschalten:



Nun kann ein Makro aufgezeichnet werden.

Beispiel 1: Zahlenformat „km“.

Der Makrorekorder liefert:

```
Sub km( )  
    Selection.NumberFormat = "#,##0.00 "km" "  
End Sub
```

Beispiel 2: Ein Format (Arial, 14 Punkt, fett, gelbe Hintergrund, rote Schriftfarbe, zentriert).

Der Makrorekorder liefert:

```
Sub Format()  
'  
' Format Makro  
'  
'  
  
    With Selection.Font  
        .Name = "Arial"  
        .Size = 11  
        .Strikethrough = False  
        .Superscript = False  
        .Subscript = False  
        .OutlineFont = False  
        .Shadow = False  
        .Underline = xlUnderlineStyleNone  
        .ThemeColor = xlThemeColorLight1  
        .TintAndShade = 0  
        .ThemeFont = xlThemeFontNone  
    End With  
  
    With Selection.Font  
        .Name = "Arial"
```

```

        .Size = 14
        .Strikethrough = False
        .Superscript = False
        .Subscript = False
        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ThemeColor = xlThemeColorLight1
        .TintAndShade = 0
        .ThemeFont = xlThemeFontNone
    End With
    Selection.Font.Bold = True
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .Color = 65535
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With
    With Selection.Font
        .Color = -16776961
        .TintAndShade = 0
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
End Sub

```

Das kann gekürzt werden in:

```

Sub Format()
    With Selection.Font

```

```

        .Name = "Arial"
        .Size = 14
        .Bold = True
        .Color = -16776961
End With

```

```

Selection.Interior.Color = 65535

```

```

Selection.HorizontalAlignment = xlCenter

```

```

End Sub

```

Beispiel 3: Kopieren / Inhalte einfügen als Werte / „Ameisenstraße“ ausschalten.

Natürlich kann man für eine Zelle mit einer Formel [F2], [F9] und [Enter] drücken.

Natürlich kann man [Strg] + [C], [Alt] + [R] [V] [W], [esc] drücken.

Oder man zeichnet ein Makro auf, das folgenden Code liefert:

```

Sub InhalteInWerte()
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone,
SkipBlanks _
        :=False, Transpose:=False
    Application.CutCopyMode = False
End Sub

```

Das Leerzeichen hinter „PasteSpecial“ verweist darauf, dass es sich um eine Methode handelt – danach folgen die Parameter. „CutCopyMode“ ist eine Eigenschaft. Sie wird konsequenterweise mit einem Gleichheitszeichen geschrieben.

Beispiel 4: „Textbaustein“

Um Texte in drei Zeilen untereinander aufzuzeichnen, muss vorher die Option „Relative Verweise verwenden“ aktiviert sein. Dann liefert der Makrorekorder:

```

ActiveCell.FormulaR1C1 = "Sachlich und fachlich geprüft"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "Rene Martin"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "Oberprüfer"
ActiveCell.Offset(1, 0).Range("A1").Select

```

Schöner wäre:

```

ActiveCell.Value = "Sachlich und fachlich geprüft"
ActiveCell.Offset(1, 0).Select
ActiveCell.Value = "Rene Martin"
ActiveCell.Offset(1, 0).Select
ActiveCell.Value = "Oberprüfer"

```

Meine Kritik am Makrorekorder:

- Der Makrorekorder zeichnet meistens zu viel Code auf
- Der Makrorekorder zeichnet die Bewegungen am Bildschirm auf, also das Selektieren der Tabellenblätter oder das Markieren von Zellen.
- Der Makrorekorder zeichnet nicht alles auf, beispielsweise die Dateieigenschaften, die Bildkomprimierung und anderes
- Der Makrorekorder zeichnet an einigen (wenigen) Stellen fehlerhaften Code auf, beispielsweise bei der Datenüberprüfung, wenn benutzerdefiniert mit einer Formel gearbeitet wird
- Der Makrorekorder zeichnet nicht immer den besten Code auf.

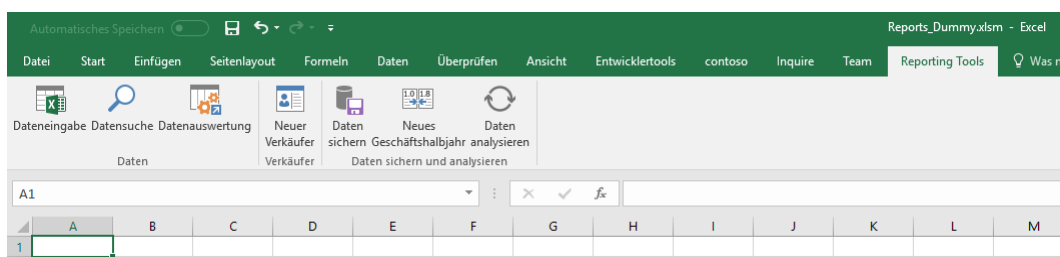
Fazit: Der Makrorekorder ist ein praktisches Werkzeug mit dem man schnell an Code – oder genauer – an die eigentlichen Objekte, Eigenschaften und Methoden gelangt. Für einen Anfänger sicherlich völlig ausreichend; ein Profi sollte jedoch immer den Code überarbeiten, das heißt:

- Voraussetzungen überprüfen
- Fehler abfangen
- unnötigen Ballast löschen
- unnötige Kommentare entfernen
- Objekte „sauber“ adressieren

Schritt 2: Die Makros können in die Datei „Personal.xlsb“ geschoben werden – dort stehen sie immer zur Verfügung, können an ein Symbol oder an eine Tastenkombination gebunden werden.

Schritt 3: Eleganter ist sicherlich ein Add-In

Am besten nimmt man hierzu eine Vorlage:



Die Datei kann umbenannt werden in Datei.zip. Wird anschließend entzippt:

Name	Änderungsdatum	Typ	Größe
_rels	09.01.2018 19:34	Dateiordner	
docProps	09.01.2018 19:34	Dateiordner	
ribbon	09.01.2018 19:34	Dateiordner	
xl	09.01.2018 19:34	Dateiordner	
[Content_Types].xml		XML Document	2 KB

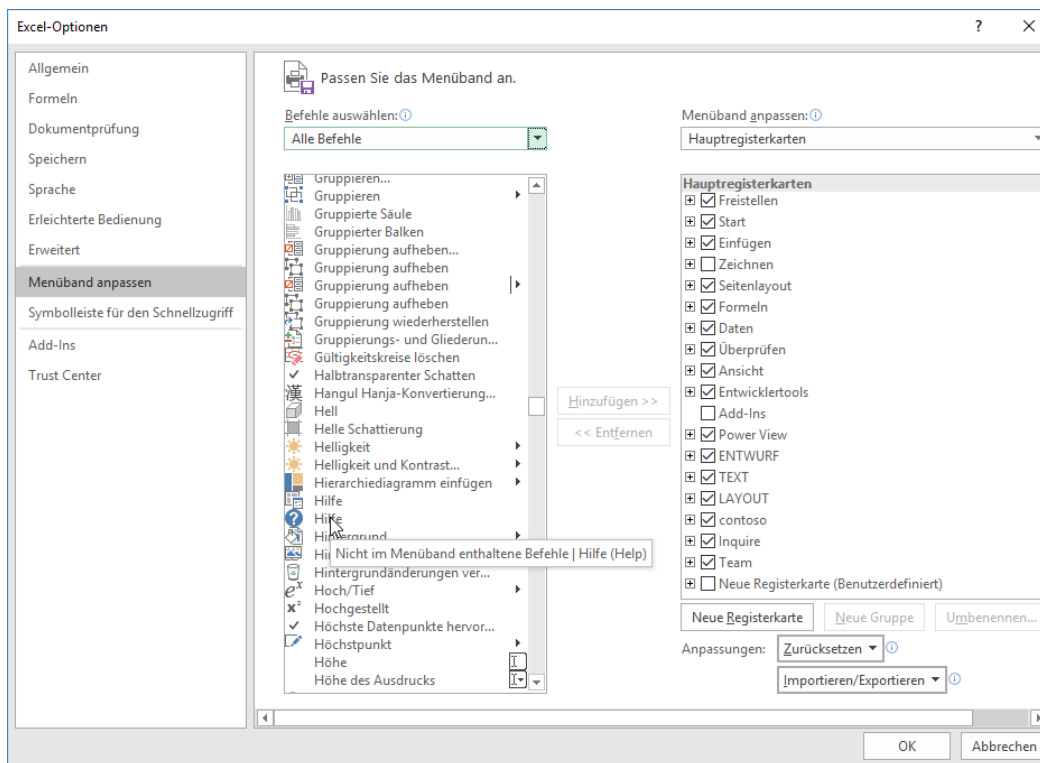
Im bereits angelegten Ordner ribbon befindet sich eine Datei ribbon.xml. Sie wird mit einem guten XML-Editor geöffnet. Dort werden die entsprechenden Attribute für Tab, Group und Button vergeben:

```

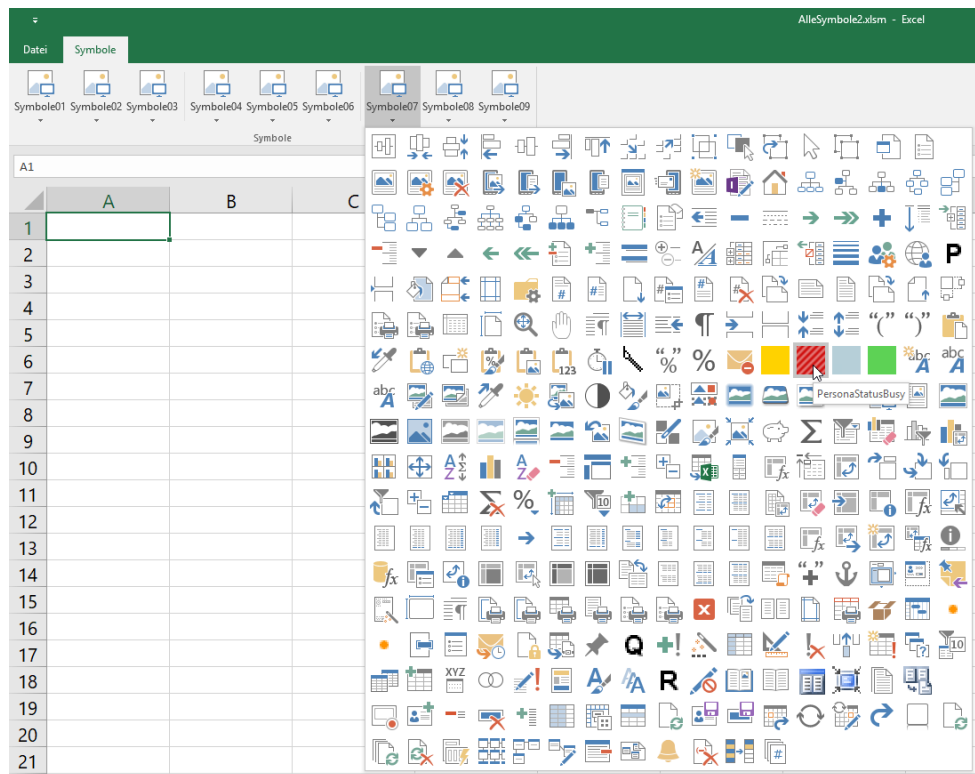
ribbon.xml
<?xml version="1.0" encoding="utf-8"?>
<customUI xmlns="http://schemas.microsoft.com/office/2006/ui/customUI">
  <ribbon startFromScratch="false">
    <tabs>
      <tab id="tabDaten" label="Excelstammtisch">
        <group id="grpReport" label="RDA">
          <button id="cmdkne" imageSrc="3DperspectiveIncrease" label="Zahlenformat: kn" onAction="cmdkne" size="large" supertip="Formatiert eine Zahl benutzerdefiniert als &quot;kn&quot;;" screentip="Format"></button>
          <button id="cmdZellformat" imageSrc="PersonasStatusBusy" label="Formatiert die Zelle" onAction="cmdFormat" size="large" supertip="Formatiert eine Zelle mit Schriftfarbe, Hintergrundfarbe, ..." screentip="Format"></button>
          <button id="cmdWerte" imageSrc="Numbering" label="Wandelt eine Formel in Werte um" onAction="cmdWerte" size="large" supertip="Wandelt eine Formel in Werte um" screentip="Werte"></button>
          <button id="cmdTextbaustein" imageSrc="Text" label="Schreibt einen Text in eine Zelle" onAction="cmdTextbaustein" size="large" supertip="Schlich und fachlich geprüft // Rene Martin // Oberprüfer" screentip="Text"></button>
        </group>
        <group id="grpSichern" label="Daten sichern und analysieren">
          <button id="cmdDokumentation" imageSrc="FunctionsLookupReferenceInsertGallery" label="Dokumentation" onAction="cmdDokumentation" size="large" supertip="Dokumentation" screentip="Dokumentation"></button>
          <button id="cmdInfo" imageSrc="Help" label="Info" onAction="cmdInfo" size="large" supertip="Informationen über das Add-In" screentip="Info"></button>
        </group>
      </tab>
    </tabs>
  </ribbon>
</customUI>

```

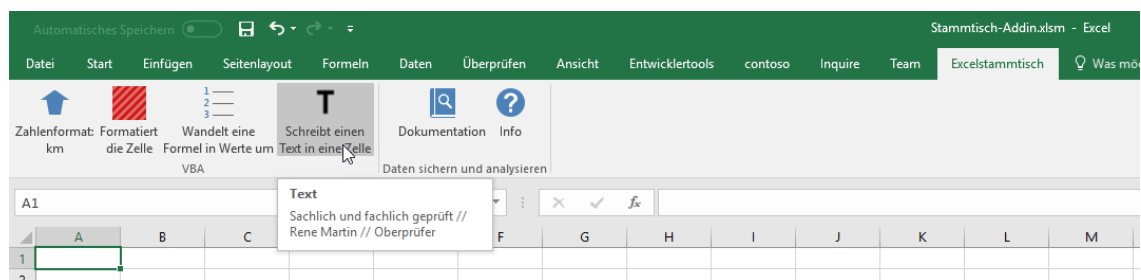
Für die Bilder verwende ich die Namen, die Microsoft für das Menüband anzeigt:



Für die Icons gibt es Hilfen die die Bilder zur Verfügung stellen:



Die Dateien werden gezippt, umbenannt und geöffnet:



An die Symbole werden die Makros gehängt:

```
Option Explicit

Public Sub cmdkm(control As IRibbonControl)
    Call km
End Sub

Public Sub cmdFormat(control As IRibbonControl)
    Call Format
End Sub

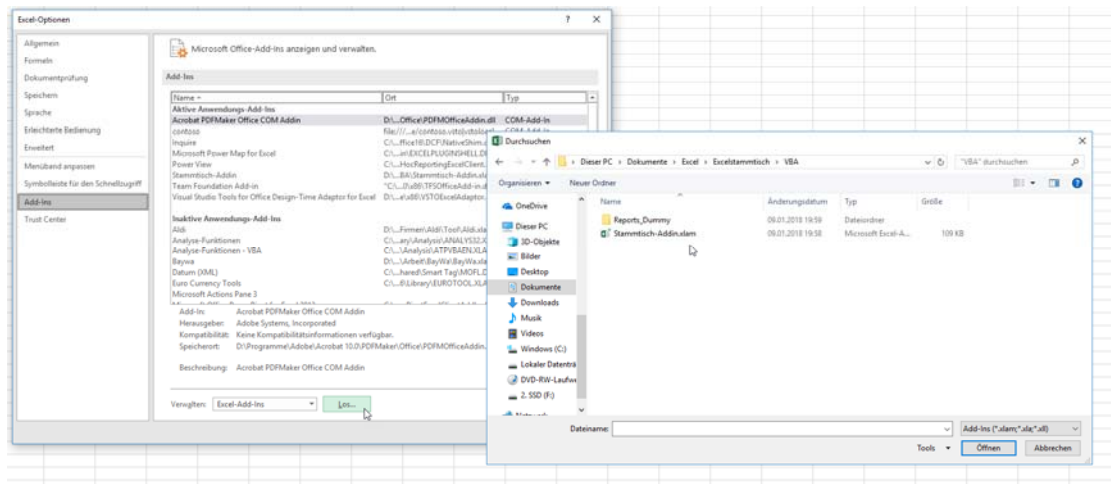
Public Sub cmdWerte(control As IRibbonControl)
    Call InhalteInWerte
End Sub

Public Sub cmdTextbaustein(control As IRibbonControl)
    Call Textbaustein
End Sub

Public Sub cmdDokumentation(control As IRibbonControl)
    Dim webshell As Object
    Set webshell = CreateObject("WScript.Shell")
    webshell.Run "http://www.compucon.de/VBA.pdf"
End Sub

Public Sub cmdInfo(control As IRibbonControl)
    MsgBox "Erstellt von Rene" & vbCrLf & "Für den Excelstammstich" & vbCrLf & "vom: 08.01.2018", vbInformation, "Excel-Stammstich"
End Sub
```

Diese Datei wird nun als Add-In (xlam) gespeichert und kann über Datei / Optionen / Add-Ins / GeheZu/Los eingebunden werden:



Damit stehen die Add-Ins immer in Excel zur Verfügung. Diese Datei kann firmenweit verteilt werden.