

Kapitel 3

Den Makrorekorder verwenden

In diesem Kapitel lernen Sie

- wie Sie den Makrorekorder verwenden können
- den Unterschied zwischen relativer und absoluter Aufzeichnung kennen
- den aufgezeichneten Code zu interpretieren
- wie Makros per Mausklick gestartet werden können

3.1 Ein paar Worte zu VBA

VBA ist die Abkürzung für *Visual Basic for Applications*. Das bedeutet *Programmierung für eine Applikation* wie beispielsweise Excel. VBA basiert auf der Programmiersprache Visual Basic (abgekürzt VB), wobei mittels VB eigenständige Programme entwickelt werden können (*.exe). Eine VBA-Prozedur hingegen kann nur ausgeführt werden, wenn die damit verbundene Applikation installiert und geöffnet ist. Die Programmiersprache darf keineswegs unterschätzt werden. Nicht umsonst nennt man sie »Sprache« und jede Sprache will zuerst gelernt sein.

VBA ist eine typische objektorientierte Programmiersprache. Während der gesamten Programmierung werden bestehende Objekte erzeugt, und dann auf sie zugegriffen und so verändert. Was genau Objekte sind und welche es gibt, erfahren Sie in Kapitel 6.

VBA gibt es nicht nur in Excel, sondern mittlerweile in nahezu allen Office-Programmen. Sie können VBA somit auch in Word, Access usw. verwenden. Dabei möchte ich Sie jedoch gleich der Illusion berauben, dass ein in Excel erstellter Programmcode ohne Weiteres auch in Word verwendet werden kann. Word verfügt über andere Objekte, Eigenschaften und Methoden, die wiederum zuerst gelernt sein wollen.

Damit vorerst genug an Theorie.

3.2 Den Makrorekorder verwenden

Auf den nachfolgenden Seiten lernen Sie, was ein Makrorekorder ist und wie er verwendet werden kann.

Kapitel 3 Den Makrorekorder verwenden

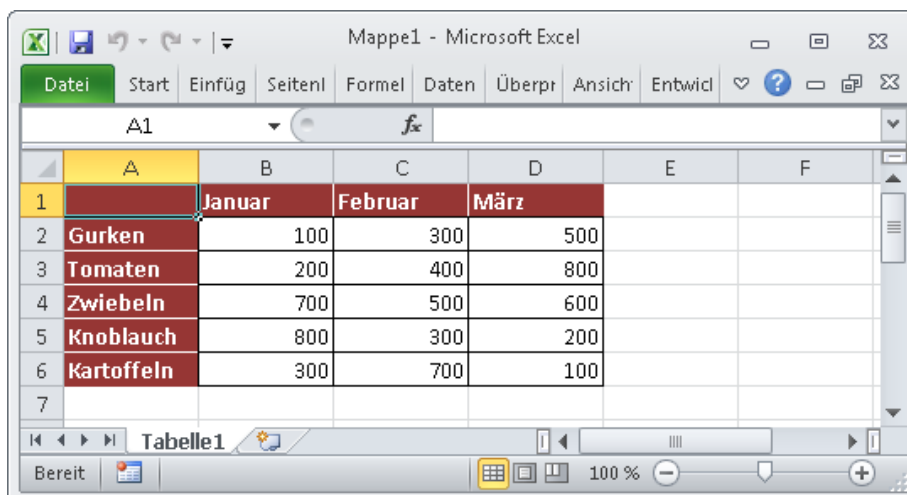
Was ist ein Makrorekorder?

Wer sich heutzutage noch an den guten alten Kassettenrekorder erinnert, kann diesen im weitesten Sinne mit dem Makrorekorder vergleichen. Mit Ersterem lassen sich hörbare Töne aufnehmen, mit dem Makrorekorder hingegen Arbeitsschritte. Der Makrorekorder ist das geeignete Hilfsmittel, um ohne vorherige Kenntnisse der Programmiersprache VBA VBA-Programmcode zu erzeugen. Während der Makrorekorder eingeschaltet ist, wird jeder Mausklick und das Drücken der Tasten bzw. die hierdurch ausgelösten Befehle aufgezeichnet. Im Hintergrund wird dabei automatisch ein VBA-Code erzeugt. Später können die aufgezeichneten Schritte jederzeit wieder ausgeführt oder bearbeitet und angepasst werden.

Ein Makro aufzeichnen

Bevor Sie mit der Aufzeichnung eines Makros beginnen, sollten Sie sich genau überlegen, was Sie tun möchten. Je weniger Schritte und Korrekturen Sie während der Aufzeichnung ausführen müssen, desto weniger Code wird erzeugt. Entsprechend schneller ist das Makro, wenn Sie es später wieder ausführen. Der Code ist zudem besser überschaubar und kann somit einfacher analysiert und nachbearbeitet werden.

In unserem Beispiel verfügen wir über eine Tabelle, deren Zahlen wir formatieren möchten.



	A	B	C	D	E	F
1		Januar	Februar	März		
2	Gurken	100	300	500		
3	Tomaten	200	400	800		
4	Zwiebeln	700	500	600		
5	Knoblauch	800	300	200		
6	Kartoffeln	300	700	100		
7						

Abbildung 3.1: Die Ausgangslage für unser Makro

Der Bereich B2:D6 soll formatiert werden. Wir überlegen uns also, welche Arbeitsschritte dazu erforderlich sind.

1. Der Zellbereich (B2:D6) muss markiert werden.
2. Ein Zahlenformat muss zugewiesen werden.
3. Die Zahlen werden fett und kursiv formatiert.

3.2 Den Makrorekorder verwenden

Nachdem Sie die genauen Arbeitsschritte kennen, die Sie mittels Makrorekorder aufzeichnen wollen, können Sie mit der Aufzeichnung des Makros beginnen. Die Arbeitsschritte setzen sich wie folgt zusammen:

1. Rufen Sie den Makrorekorder auf. Klicken Sie hierzu auf der Registerkarte *Entwicklertools* in der Gruppe *Code* auf *Makro aufzeichnen*.

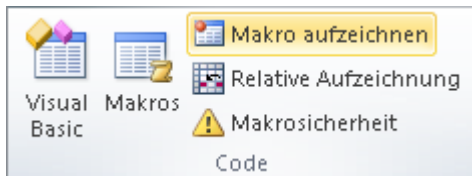


Abbildung 3.2: Die Makroaufzeichnung starten

2. Geben Sie in das Textfeld *Makroname* einen aussagekräftigen Namen ein. Beachten Sie hierbei, dass dieser mit einem Buchstaben beginnen muss und keine Leerzeichen enthalten darf. Sonderzeichen, Punkte, Kommas usw. sind ebenfalls nicht erlaubt.

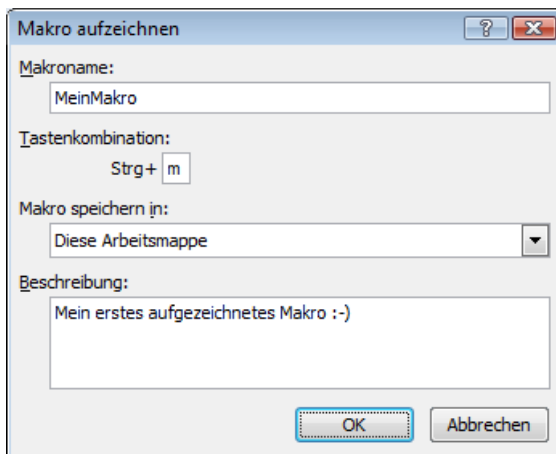


Abbildung 3.3: Das Dialogfeld *Makro aufzeichnen*

3. Legen Sie, falls gewünscht, bei *Tastenkombination* eine Taste fest, um das Makro später über diese Tastenkombination aufrufen zu können. Bedenken Sie hierbei, dass dadurch vordefinierte Excel-Tastenkombinationen überschrieben werden.
4. Im Feld *Beschreibung* können Sie, wenn gewünscht, einen Text eingeben.
5. Klicken Sie auf die Schaltfläche *OK*, um die Aufzeichnung zu starten.
6. Führen Sie nun die zuvor geplanten Arbeitsschritte aus. Sobald Sie damit fertig sind, können Sie die Aufzeichnung beenden. Klicken Sie dazu auf der Registerkarte *Entwicklertools* in der Gruppe *Code* auf die Schaltfläche *Aufzeichnung beenden*.

Kapitel 3 Den Makrorekorder verwenden

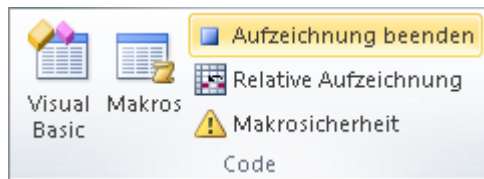


Abbildung 3.4: Die Makroaufzeichnung beenden

Ein Makro ausführen

Das zuvor aufgezeichnete Makro können Sie nun jederzeit wieder ausführen. Wenn Sie dem Makro eine Tastenkombination zugewiesen haben, können Sie es durch Drücken dieser Tastenkombination starten. Anderenfalls führen Sie folgende Schritte aus:

1. Klicken Sie auf der Registerkarte *Entwicklertools* in der Gruppe *Code* auf die Schaltfläche *Makros* oder verwenden Sie alternativ die Tastenkombination **Alt + F8**.

Das Dialogfeld *Makro* wird geöffnet. Im Listenfeld *Makroname* werden alle vorhandenen Makros angezeigt.

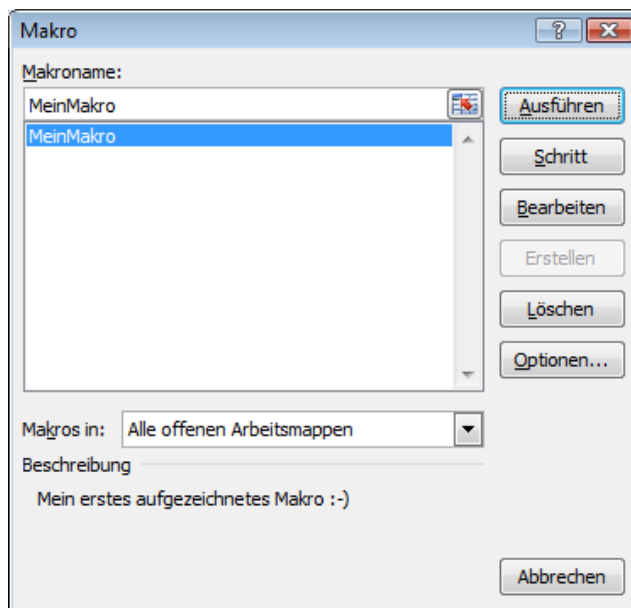


Abbildung 3.5: Das aufgezeichnete Makro auswählen

2. Wählen Sie das Makro aus, das Sie ausführen möchten.
3. Klicken Sie abschließend auf die Schaltfläche *Ausführen*, um den Vorgang zu starten. Das Makro führt nun die aufgezeichneten Schritte aus. Je nach Rechengeschwindigkeit geschieht dies blitzschnell, sodass die einzelnen Arbeitsschritte kaum wahrnehmbar sind.

Es gibt kein Zurück

Wenn Sie mit Makros arbeiten, müssen Sie unbedingt wissen, dass es für einmal ausgeführte Prozeduren kein Zurück gibt. Die Funktion *Rückgängig* wird nach dem Ausführen eines Makros automatisch zurückgestellt. Wenn Sie sicherstellen möchten, dass ein Makro keine vorhandenen Daten überschreibt, gibt es zwei Möglichkeiten:

- Speichern Sie die Mappe vor dem Ausführen eines Makros unter einem anderen Namen ab, damit Sie eine Sicherheitskopie haben.
- Speichern Sie die Mappe vor dem Ausführen des Makros ab. Wenn tatsächlich unerwünschte Veränderungen vorgenommen wurden, schließen Sie die Mappe, ohne sie zuvor noch einmal zu speichern, und öffnen sie dann erneut.

Eine Tastenkombination nachträglich anpassen

Sie können die Tastenkombination auch erst im Nachhinein zuweisen oder löschen oder ändern.

1. Rufen Sie dazu erneut das Dialogfeld *Makro* auf.
2. Wählen Sie in dem Listenfeld das Makro aus, dessen Tastenkombination Sie ändern möchten.
3. Klicken Sie auf die Schaltfläche *Optionen*, um das Dialogfeld *Makrooptionen* zu öffnen.

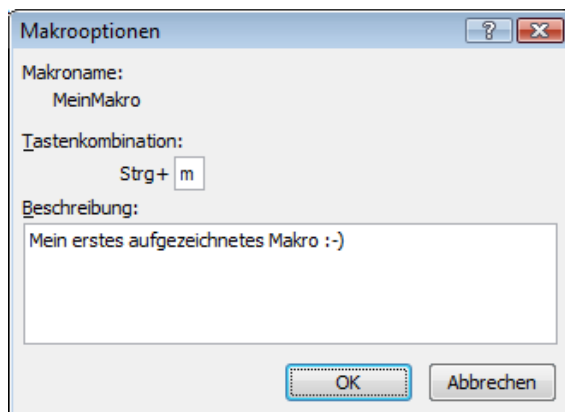


Abbildung 3.6: Die Tastenkombination anpassen

4. Geben Sie die gewünschte Tastenkombination ein und schließen Sie die beiden geöffneten Dialogfelder.

Kapitel 3 Den Makrorekorder verwenden

Ein Makro löschen

Sollten Sie ein Makro nicht mehr benötigen, können Sie es selbstverständlich löschen.

1. Klicken Sie auf der Registerkarte *Entwicklertools* in der Gruppe *Code* auf die Schaltfläche *Makros*. Das Dialogfeld *Makro* wird geöffnet.
2. Wählen Sie das zu löschende Makro aus und klicken Sie dann auf die Schaltfläche *Löschen*.
3. Bestätigen Sie die Sicherheitsabfrage, indem Sie die Schaltfläche *Ja* anklicken.

Die persönliche Makroarbeitsmappe (PERSONAL.XLSB)

Wenden wir uns nun dem Speicherort für Ihre Makros bzw. Ihren VBA-Code zu.

Wenn Sie ein Makro aufzeichnen, wird dieses automatisch in der aktuell geöffneten Arbeitsmappe gespeichert, sofern Sie vor der Aufzeichnung keinen anderen Speicherort ausgewählt haben. Im Dialogfeld *Makro aufzeichnen*, das Sie über *Entwicklertools/Code/Makro aufzeichnen* aufrufen, kann der Speicherort geändert werden. Im Dropdown-Listenfeld *Makro speichern in* stehen drei Speicherorte zur Auswahl:

- Persönliche Makroarbeitsmappe
- Neue Arbeitsmappe
- Diese Arbeitsmappe

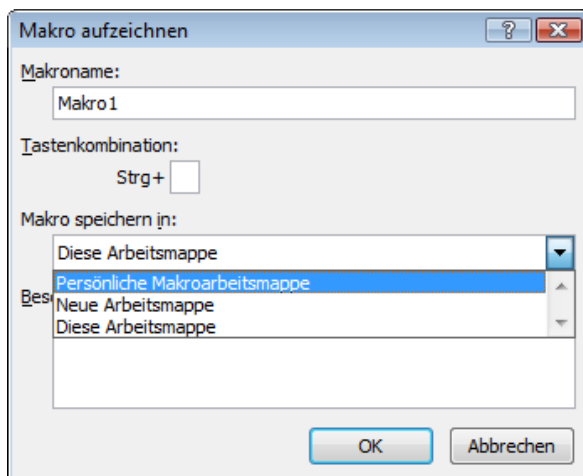


Abbildung 3.7: Ein Makro in der Arbeitsmappe *PERSONAL.XLSB* abspeichern

Wenn Sie *Diese Arbeitsmappe* wählen, steht das Makro ausschließlich in der Arbeitsmappe zur Verfügung, aus der heraus Sie die Aufzeichnung starten.

3.2 Den Makrorekorder verwenden

Sie können die aktive Arbeitsmappe auch unberührt lassen und das Makro in einer neuen Arbeitsmappe erstellen. Wählen Sie dazu den Eintrag *Neue Arbeitsmappe* aus. Die neue Arbeitsmappe wird automatisch erzeugt, sobald Sie die OK-Schaltfläche anklicken.

Sie können ein Makro auch für sämtliche Arbeitsmappen, sprich für die gesamte Excel-Anwendung, zur Verfügung stellen. Wählen Sie dazu den Speicherort *Persönliche Makroarbeitsmappe* aus. Dabei wird das Makro in einer Datei namens *PERSONAL.XLSB* gespeichert.



Wichtig: Wo befindet sich die PERSONAL.XLSB?

Die Datei *PERSONAL.XLSB* steht nach der Installation von Excel bzw. Office nicht automatisch zur Verfügung. Sie wird erst dann erstellt, wenn ein Makro mit dem Speicherort *Persönliche Makroarbeitsmappe* aufgezeichnet wurde. Die Datei *PERSONAL.XLSB* wird im Office-Unterverzeichnis *XLSTART* angelegt. In Windows 7 und Windows Vista lautet der Pfad in der Regel:

`C:\Benutzer\[Benutzername]\AppData\Roaming\Microsoft\Excel\XLSTART\PERSONAL.XLSB`

In älteren Windows-Versionen lautete der Pfad:

`C:\Dokumente und Einstellungen\[Benutzername]\Anwendungsdaten\Microsoft\Excel\XLSTART\PERSONAL.XLSB`

Anstelle von *Benutzer* steht oft auch *Users* und anstelle von *[Benutzername]* steht Ihr Windows-Anmeldename.

Wenn Sie die *PERSONAL.XLSB* löschen möchten, öffnen Sie im Windows-Explorer den oben genannten Pfad und entfernen die Datei.

Bitte beachten Sie: Der Ordner *AppData* bzw. *Anwendungsdaten* ist standardmäßig versteckt und erst verfügbar, wenn er sichtbar gemacht wird.

In Windows 7 und Windows Vista finden Sie die Einstellung, um Systemordner bzw. Dateien einzublenden, im Windows-Explorer unter *Organisieren/Ordner- und Suchoptionen* auf der Registerkarte *Ansicht* unter *Versteckte Dateien und Ordner*, Option *Alle Dateien und Ordner anzeigen*.

In älteren Windows-Versionen finden Sie die Einstellung im Windows-Explorer unter *Extras/Ordneroptionen*, Registerkarte *Ansicht*, Option *Alle Dateien und Ordner anzeigen*.

Wenn Sie die Datei *PERSONAL.XLSB* einmal erzeugt haben, ist sie im Projekt-Explorer des VBA-Editors verfügbar. Die darin enthaltenen Makros können eingesehen und bearbeitet werden.

Kapitel 3 Den Makrorekorder verwenden

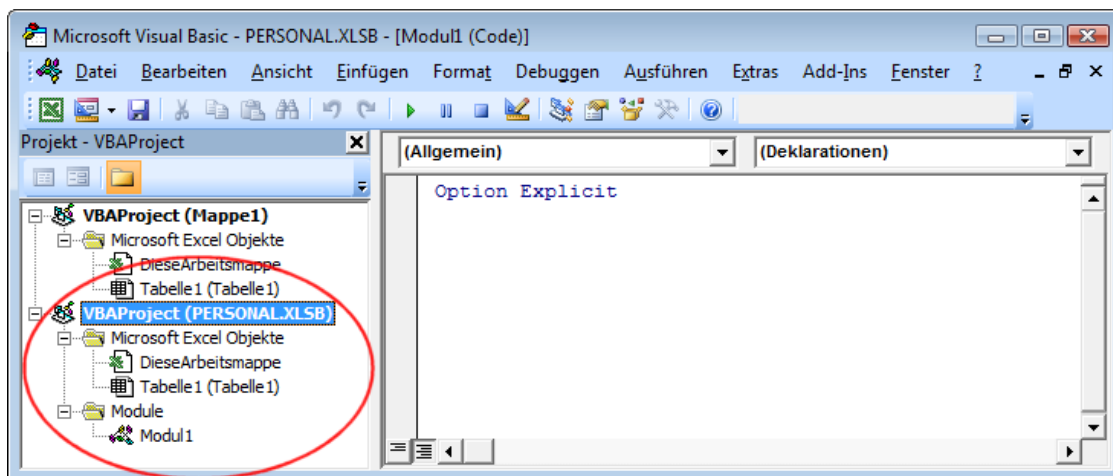


Abbildung 3.8: Das VBA-Projekt *PERSONAL.XLSB*

Die Makroarbeitsmappe *PERSONAL.XLSB* wird beim Start von Excel automatisch im Hintergrund geladen. Die Arbeitsmappe ist, zum Schutz vor versehentlichen Änderungen, nicht sichtbar.

Wenn Sie ein Makro in der persönlichen Makroarbeitsmappe erstellt haben, erscheint beim Schließen der Applikation eine Meldung mit der Abfrage, ob Sie die Änderungen in der persönlichen Makroarbeitsmappe speichern möchten. Klicken Sie auf *Ja*, wenn Sie die Makros speichern möchten. Nur so stehen sie Ihnen später, beim erneuten Start von Excel, wieder zur Verfügung.

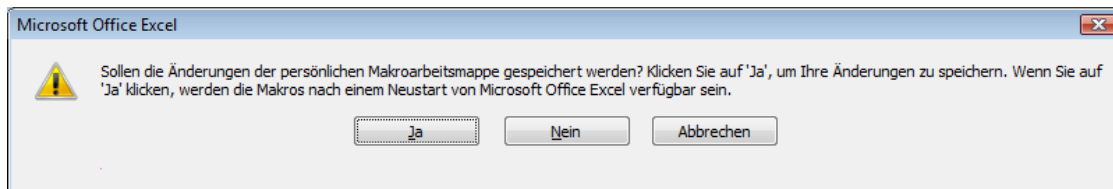


Abbildung 3.9: Speicherabfrage bezüglich *PERSONAL.XLSB*

3.3 Relative und absolute Aufzeichnung

Bei dem Makro, das wir weiter oben aufgezeichnet haben, wurden alle Zellbezüge als absolute Angaben im Makro gespeichert. Woran ist das zu erkennen? Beim Ausführen des Makros wird immer auf die festgelegten Zellen Bezug genommen. Wenn Sie bei der Aufzeichnung die Zellen *B2:D6* markiert haben, werden die Formatierungen immer diesem Bereich zugewiesen.

Nun könnte es jedoch auch vorkommen, dass sich die zu formatierende Tabelle an einer anderen Stelle im Tabellenblatt befindet. Dann würde uns das Makro mit den absoluten Zellbezügen nichts mehr nutzen. Microsoft hat das bedacht und ermöglicht auch eine relative Aufzeichnung. Wie müssen Sie dabei vorgehen?

3.4 Den aufgezeichneten Makrocode analysieren

1. Starten Sie die Makroaufzeichnung, indem Sie *Entwicklertools/Code/Makro aufzeichnen* wählen.
2. Bevor Sie Ihre Arbeitsschritte ausführen, klicken Sie die Schaltfläche *Relative Aufzeichnung* an; sie wird dann hervorgehoben dargestellt.

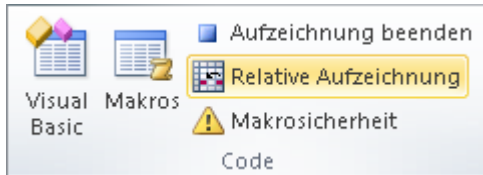


Abbildung 3.10: Die relative Aufzeichnung aktivieren

3. Führen Sie Ihre Arbeitsschritte wie gewohnt aus und beenden Sie danach die Aufzeichnung.

Wenn Sie das Makro später ausführen, werden die Formatierungen der Zellen relativ von der Zelle aus vorgenommen, aus der heraus das Makro gestartet wird.



Tipp: Aufzeichnung in nur einer Zelle

Sie können sowohl bei der absoluten als auch bei der relativen Aufzeichnung vor dem Aufzeichnen nur eine Zelle aktivieren und nur dieser die gewünschten Formate zuweisen.

Wenn Sie die Formatierungen später auf mehrere Zellen anwenden möchten, markieren Sie diese und führen das Makro aus. Damit werden alle Zellen innerhalb der Markierung entsprechend formatiert.

3.4 Den aufgezeichneten Makrocode analysieren

Da Sie VBA lernen möchten, interessiert es Sie bestimmt, was während der Aufzeichnung eines Makros geschehen ist bzw. welcher Code im Hintergrund mitgeschrieben wurde. Wenn Sie sich den Code anschauen möchten, müssen Sie zur Entwicklungsumgebung wechseln. Es gibt verschiedene Wege, um zum *Visual Basic-Editor* zu gelangen.

- Der schnellste Weg führt über die Tastenkombination **Alt** + **F11**.
- Sie können auch den Weg über *Entwicklertools/Code/Visual Basic* wählen.
- Eine weitere Alternative stellt das Kontextmenü der Blattregisterkarte dar. Sie finden darin den Befehl *Code anzeigen*.

Egal welchen Weg Sie gewählt haben, Sie sollten sich nun im VBA-Editor befinden. Die folgenden Schritte zeigen, wie Sie das aufgezeichnete Makro finden.

1. Um den Makrocode zu finden, muss der Projekt-Explorer eingeblendet sein. Sollte dies nicht der Fall sein, blenden Sie ihn über die Tastenkombination **Strg** + **R** ein.

Wenn die VBA-Umgebung seit der Installation von Office nicht verändert wurde, finden Sie den Projekt-Explorer an der linken Seite des Fensters (weitere Erläuterungen zur Entwicklungsumgebung finden Sie in Kapitel 4).

Kapitel 3 Den Makrorekorder verwenden

- Öffnen Sie den Ordner *Module* und doppelklicken Sie dort auf *Modul1*. Im Editor wird nun der gesuchte Code angezeigt.

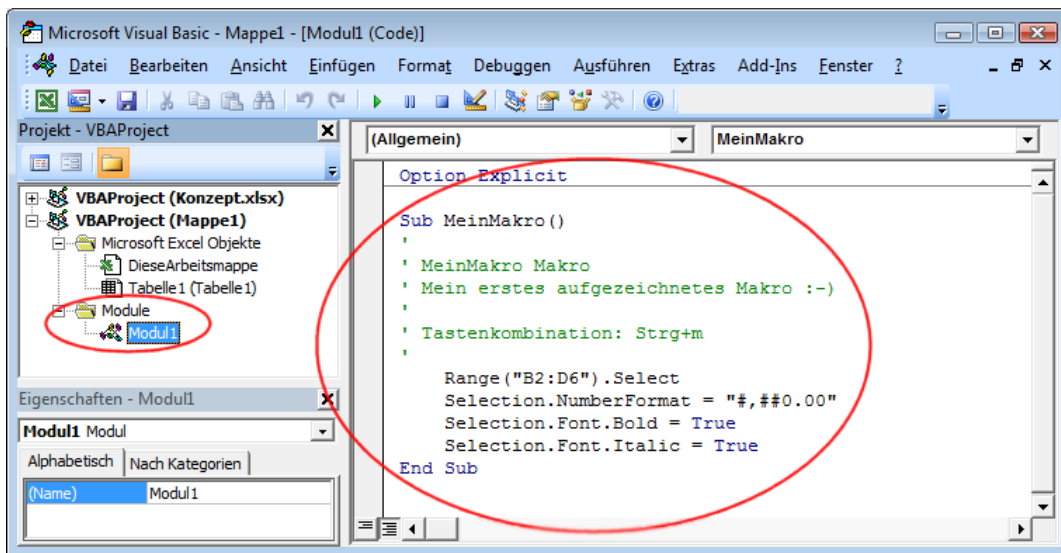


Abbildung 3.11: Der aufgezeichnete VBA-Code

Wie setzt sich der Code zusammen?

Wenn Sie sich bei der Aufzeichnung des absoluten Makros an die Anweisungen gehalten haben, dürfte sich nun ungefähr folgender Code im VBA-Editor präsentieren:

```
Sub MeinMakro()  
'  
' MeinMakro Makro  
' Mein erstes aufgezeichnetes Makro :-)  
' Tastenkombination: Strg+m  
'  
Range("B2:D6").Select  
Selection.NumberFormat = "#,##0.00"  
Selection.Font.Bold = True  
Selection.Font.Italic = True  
End Sub
```

Der Code besteht aus folgenden Elementen:

- Sub ... () ist die Einleitung, sozusagen der Titel des Codes.
- Danach folgen einige Zeilen, denen ein einfaches Anführungszeichen (') vorangestellt ist. Dabei handelt es sich um reine Informationen. Man nennt solche Zeilen Kommentarzeilen. Diese Zeilen haben keinen Einfluss auf den eigentlichen Code. Es sind Informationen, die Sie im Dialogfeld *Makro aufzeichnen* mitgeliefert haben: der *Makroname*, die *Beschreibung* und eventuell die *Tastenkombination*.

3.4 Den aufgezeichneten Makrocode analysieren

- Etwas eingerückt finden Sie den eigentlichen Code. Dies sind Ihre ausgeführten Arbeitsschritte und somit der Kern des Geschehens.
- Der Code wird mit einem `End Sub` abgeschlossen. Diese Zeile kennzeichnet das Ende der Prozedur.

Den Makrocode bereinigen

Der Makrorekorder ist zwar eine wunderbare Sache und wird Sie sicherlich noch lange begleiten, auch wenn Sie bereits etwas mehr Erfahrung mit VBA haben. Dennoch hat er seine Tücken. Viele der Codezeilen sind überflüssig und lassen sich vermeiden. Das Ganze wirkt noch recht unprofessionell.

In unserem kleinen Beispiel wird dies noch nicht so deutlich. Nichtsdestotrotz ist dies ein guter Anfang, um einen einfachen Code zu optimieren. Wir wollen dies in zwei Schritten tun. Um das Ganze hier etwas übersichtlicher zu gestalten, lassen wir die Kommentarzeilen weg.

1. Wenn ein Befehl zu Beginn einer Zeile mehrmals auftritt, hier das `Select`, kann dies mittels einer `With`-Anweisung zusammengefasst werden. Der Code sieht folgendermaßen aus:

```
Sub MeinMakro()  
  Range("B2:D6").Select  
  With Selection  
    .NumberFormat = "#,##0.00"  
    .Font.Bold = True  
    .Font.Italic = True  
  End With  
End Sub
```

2. Wenn Sie genau hinschauen, fällt Ihnen auf, dass immer noch `Select` und `Selection` verwendet wird. Beides ist überflüssig und lässt sich wie folgt reduzieren (nähere Informationen dazu erhalten Sie später in diesem Buch):

```
Sub MeinMakro()  
  With Range("B2:D6")  
    .NumberFormat = "#,##0.00"  
    .Font.Bold = True  
    .Font.Italic = True  
  End With  
End Sub
```

Ein VBA-Programmierer versucht immer, seinen Code möglichst kurz und übersichtlich zu halten, da hierdurch einiges an Tipparbeit gespart werden kann. Dies dient nicht nur der Bequemlichkeit, sondern vor allem auch der besseren Übersicht in komplexeren Projekten und, in erster Linie, dem Erreichen einer möglichst kurzen Ausführungszeit.



Tip: Lernen mit dem Makrorekorder

Benutzen Sie den Makrorekorder zu Beginn sehr häufig und versuchen Sie, den Code, der aufgezeichnet wird, zu verstehen. Gute Lerneffekte erzielen Sie in der Regel, wenn Sie versuchen, den Code zu optimieren und zu verkürzen. Verwenden Sie zu Beginn einfache Beispiele wie das Formatieren von Zellen.

Kapitel 3 Den Makrorekorder verwenden

3.5 Schnellstart für Makros

Einen Weg zum schnellen Starten eines Makros kennen Sie ja bereits. Er führt über die Zuweisung einer Tastenkombination zu einem Makro. Wenn Sie das Makro jedoch vielen Benutzern zur Verfügung stellen möchten, kennen diese unter Umständen die Tastenkombination nicht. Um den Benutzern dennoch einen komfortablen Start des Makros zu ermöglichen, gibt es verschiedene Möglichkeiten, wie Sie gleich erfahren werden.

Eine Formulschaltfläche einfügen

Bei einer Formulschaltfläche handelt es sich um ein rechteckiges Element, das auf dem Tabellenblatt platziert werden kann. Mit diesem Rechteck wird ein Makro verknüpft, das später durch Anklicken der Formulschaltfläche ausgeführt werden kann.

1. Die Formulschaltfläche finden Sie im Menü der Schaltfläche *Einfügen* (Registerkarte *Entwickler-Tools*, Gruppe *Steuerelemente*). Klicken Sie im Bereich *Formularsteuerelemente* auf *Schaltfläche* (*Formularsteuerelement*).

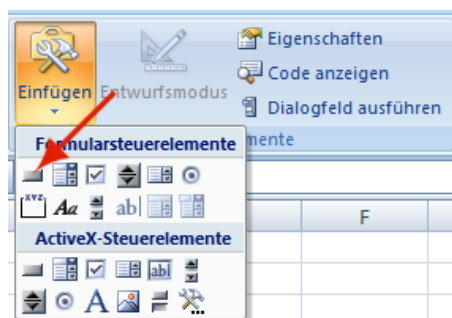


Abbildung 3.12: Eine Formulschaltfläche einfügen

2. Der Mauszeiger sollte nun die Form eines Kreuzes aufweisen. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen auf Ihrem Tabellenblatt auf.
Sobald Sie die Maustaste loslassen, öffnet sich das Dialogfeld *Makro zuweisen*.

3.5 Schnellstart für Makros

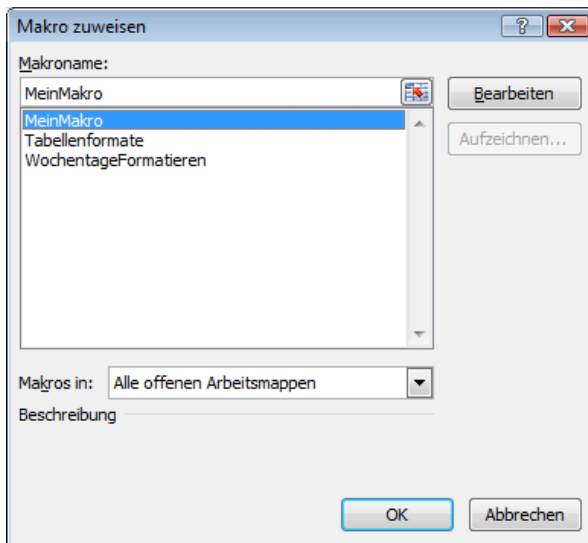


Abbildung 3.13: Das gewünschte Makro mit der Formulschaltfläche verknüpfen

3. Wählen Sie im Listenfeld *Makroname* das Makro aus, das Sie der Formulschaltfläche zuweisen möchten, und beenden Sie den Vorgang, indem Sie die Schaltfläche *OK* anklicken.
4. Die Schaltfläche befindet sich nun im Entwurfsmodus. Um den Entwurfsmodus zu verlassen, klicken Sie auf eine beliebige Zelle auf dem Tabellenblatt oder drücken die **[Esc]**-Taste.
5. Das Makro lässt sich nun per Klick auf die Formulschaltfläche auf bequeme Weise ausführen.

Eine Formulschaltfläche umbenennen

Die Schaltfläche erhält standardmäßig die Beschriftung *Schaltfläche 1*. Wenn sich mehrere Schaltflächen auf dem Tabellenblatt befinden, sind diese fortlaufend nummeriert. Um die gesamte Beschriftung inklusive der Nummer zu sehen, muss die Schaltfläche eine entsprechende Größe haben.

Auf Wunsch können Sie den Text, der auf der Schaltfläche angezeigt wird, ändern:

1. Klicken Sie mit der rechten Maustaste auf die Schaltfläche.
2. Wählen Sie im Kontextmenü den Befehl *Text bearbeiten*.
3. Der darauf angezeigte Text kann jetzt überschrieben werden.

Viel mehr Formatierungsmöglichkeiten gibt es bei Formulschaltflächen leider nicht. Sie suchen vergeblich, wenn Sie die Hintergrundfarbe oder die Schrift verändern möchten. Wenn Ihnen das Formatieren der Schaltfläche wichtig ist, müssen Sie mit einer ActiveX-Schaltfläche arbeiten. Wie das funktioniert, wird gleich erläutert.

Kapitel 3 Den Makrorekorder verwenden

ActiveX-Befehlsschaltflächen verwenden

ActiveX-Befehlsschaltflächen verhalten sich insofern anders, als dass sie einerseits bis zu einem gewissen Grad formatiert werden können und andererseits das Anbinden des Makros durch einen VBA-Code erfolgt. Sehen wir uns zuerst an, wo Sie die ActiveX-Befehlsschaltfläche finden und wie Sie ihr ein Makro zuordnen können.

1. Die ActiveX-Befehlsschaltfläche finden Sie im Menü der Schaltfläche *Einfügen* (Registerkarte *Entwicklertools*, Gruppe *Steuerelemente*). Klicken Sie im Bereich *ActiveX-Steuerelemente* auf *Befehlsschaltfläche (ActiveX-Steuerelement)*.

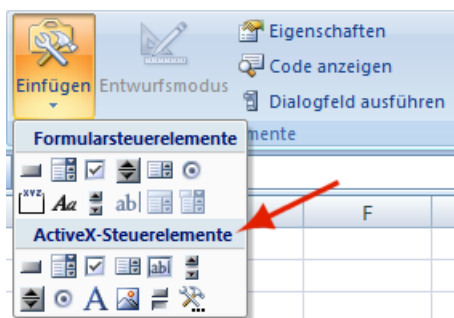


Abbildung 3.14: Eine ActiveX-Befehlsschaltfläche einfügen

2. Der Mauszeiger sollte nun die Form eines Kreuzes aufweisen. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen auf Ihrem Tabellenblatt auf. Die Befehlsschaltfläche trägt den Namen *CommandButton1*.
3. Beachten Sie, dass hierdurch auf der Registerkarte *Entwicklertools* die Schaltfläche *Entwurfsmodus* aktiviert wurde. Dieser Modus ist notwendig, wenn Sie die Schaltfläche formatieren oder ihr einen Befehl zuweisen möchten. Letzteres tun wir nun, um unser Makro mit der Befehlsschaltfläche zu verknüpfen. Doppelklicken Sie auf die Befehlsschaltfläche. Sie werden direkt in den VBA-Editor geleitet.
4. Das Codefragment für die Befehlsschaltfläche steht bereits zur Verfügung. Es handelt sich um ein *Click-Ereignis*. Mehr zu Ereignisprozeduren erfahren Sie in Kapitel 11. Schreiben Sie zwischen die beiden bereits vorhandenen Zeilen den Befehl `Call`, gefolgt vom Namen des Makros.

```
Private Sub CommandButton1_Click()  
    Call MeinMakro  
End Sub
```

5. Verlassen Sie nun den VBA-Editor. Um das Makro ausführen zu können, müssen Sie zudem den Entwurfsmodus für Steuerelemente deaktivieren. Klicken Sie hierzu auf der Registerkarte *Entwicklertools* in der Gruppe *Steuerelemente* auf die Schaltfläche *Entwurfsmodus*.

Wie bereits erwähnt, lassen sich ActiveX-Befehlsschaltflächen gestalterisch verändern. Sie können der Schaltfläche beispielsweise eine rote Hintergrundfarbe zuweisen und die Schriftart verändern.

1. Am einfachsten ist das, wenn Sie das Eigenschaftsfenster einblenden. Hierzu klicken Sie auf der Registerkarte *Entwicklertools* in der Gruppe *Steuerelemente* die Schaltfläche *Eigenschaften* an.

3.5 Schnellstart für Makros

2. Damit die Eigenschaften der Befehlsschaltfläche bearbeitet werden können, müssen Sie den Entwurfsmodus mit der Schaltfläche *Entwurfsmodus* (Registerkarte *Entwicklertools*, Gruppe *Steuerelemente*) aktivieren.
3. Wenn Sie nun die Befehlsschaltfläche anklicken, verändert sich der Inhalt des Eigenschaftensfensters. Es werden die Eigenschaften der Schaltfläche angezeigt.
4. Die Hintergrundfarbe kann mittels *BackColor* verändert werden. Klicken Sie auf den Pfeil neben dem Eintrag *BackColor*. Eine Farbauswahl mit zwei Registerkarten wird angezeigt.

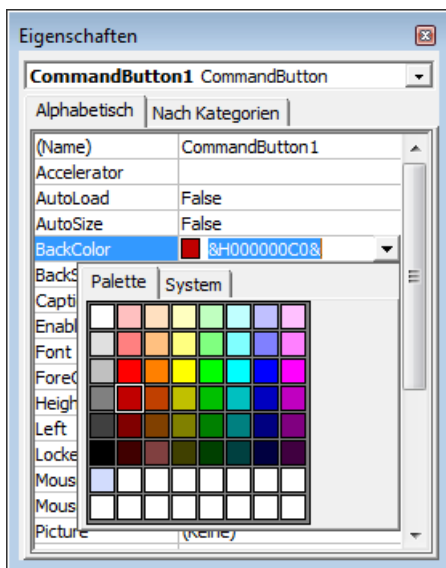


Abbildung 3.15: Die Farbpalette im Eigenschaftensfenster

5. Aktivieren Sie die Registerkarte *Palette*, wenn Sie die Schaltfläche farbig gestalten möchten. Wählen Sie per Mausklick die gewünschte Farbe aus.
6. Die Farbpalette enthält im unteren Bereich 16 leere Felder. Es handelt sich dabei um frei belegbare Felder. Klicken Sie mit der rechten Maustaste auf eines der leeren Felder. Das Dialogfeld *Farbe definieren* wird geöffnet.
7. Wählen Sie die gewünschte Farbe aus. Die Farbe wird nun auf das ehemals leere Feld übertragen und kann jederzeit wieder verwendet werden.

Kapitel 3 Den Makrorekorder verwenden

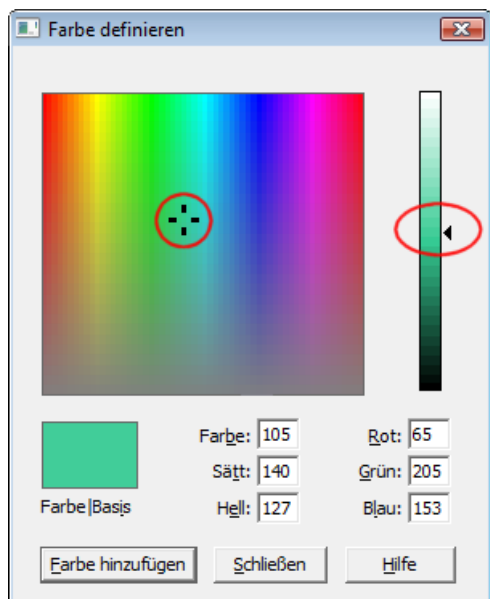


Abbildung 3.16: Eigene Farben in die Farbpalette aufnehmen

Mit dem Eigenschaftsfenster können verschiedenste Einstellungen vorgenommen werden. In Bezug auf ActiveX-Befehlsschaltflächen tritt oft der Wunsch auf, die Schriftart oder deren Farbe anzupassen. Die Eigenschaft für die Schriftart nennt sich *Font*. Für die Schriftfarbe verwenden Sie *FontColor*.

Ein Makro in die Symbolleiste für den Schnellzugriff aufnehmen

Leider können seit der Excel-Version 2007 Makros nicht mehr so flexibel außerhalb des Tabellenblatts angesiedelt werden, wie das noch zu Zeiten der Symbolleisten der Fall war. Ein kleines Hintertürchen hat Microsoft allerdings mit der sogenannten *Symbolleiste für den Schnellzugriff* offen gelassen.

Sie können nach Belieben interne Befehle oder eigene Makros in der unscheinbaren Symbolleiste für den Schnellzugriff zur Verfügung stellen.

Um Ihr Makro in die Symbolleiste für den Schnellzugriff zu integrieren, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf den Bereich des Menübands. Wählen Sie im Kontextmenü den Befehl *Symbolleiste für den Schnellzugriff anpassen*. Das Dialogfeld *Excel-Optionen* wird geöffnet.
2. Wählen Sie im linken Dropdown-Listefeld *Befehle auswählen* den Eintrag *Makros* aus. Die verfügbaren Makros werden nun im Listefeld darunter angezeigt.

3.5 Schnellstart für Makros

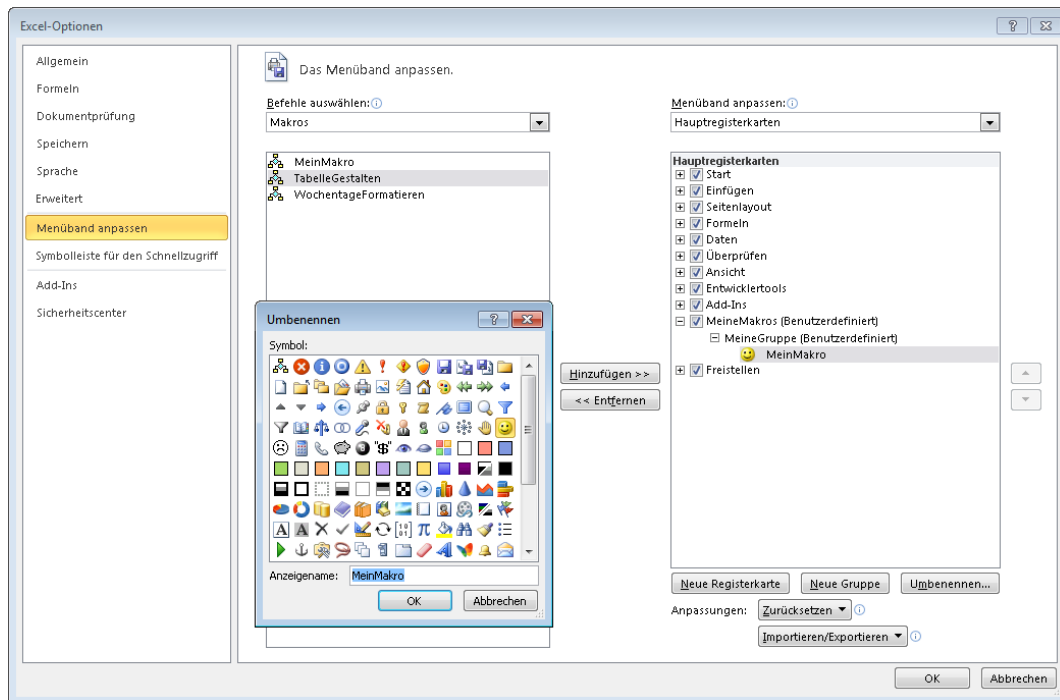


Abbildung 3.17: Das Makro auswählen und sein Symbol anpassen

3. Doppelklicken Sie auf das Makro, das in die Symbolleiste für den Schnellzugriff aufgenommen werden soll. Es wird ins rechte Listenfeld übertragen.
4. Links neben dem ausgewählten Makro wird das Standardmakrosymbol angezeigt. Dieses wird nach Abschluss der Aktion in der Symbolleiste für den Schnellzugriff sichtbar. Auf Wunsch können Sie das Symbol ersetzen. Klicken Sie dazu auf die Schaltfläche *Ändern*. Das Dialogfeld *Schaltfläche 'Ändern'* wird angezeigt. Wählen Sie hier das gewünschte Symbol aus.
5. Schließen Sie nun alle noch offenen Dialogfelder. Das Makro steht nun mit dem gewählten Symbol in der Symbolleiste für den Schnellzugriff zur Verfügung.



Hinweis: Schaltflächen nur in bestimmter Mappe verfügbar machen

Sie haben die Möglichkeit, die Schaltfläche in der Symbolleiste für den Schnellzugriff nur in einer bestimmten Arbeitsmappe einzublenden.

Wählen Sie dazu im Dialogfeld *Excel-Optionen* im rechten Dropdown-Listenfeld den Eintrag *Für 'Mappe x'* aus. Die Makros, die Sie nun vom linken in das rechte Listenfeld kopieren, werden nur für die entsprechende Arbeitsmappe in der Symbolleiste für den Schnellzugriff angezeigt.

Kapitel 3 Den Makrorekorder verwenden

3.6 Übungen zu diesem Kapitel

In diesem Abschnitt finden Sie einige Übungen zu diesem Kapitel. Die Antworten und Lösungen finden Sie auf der Website www.richtig-einsteigen.de.

Übung 3.1

Wie müssen Sie vorgehen, um die Registerkarte *Entwicklertools* einzublenden?

Übung 3.2

Sie haben ein Makro aufgezeichnet, das folgendermaßen aussieht:

```
Sub Makro1()  
  Range("A1:C12").Select  
  Selection.Font.Bold = True  
  Selection.NumberFormat = "0.0"  
  Selection.NumberFormat = "0.00"  
  With Selection.Font  
    .Color = -16776961  
    .TintAndShade = 0  
  End With  
End Sub
```

Versuchen Sie, den Code so gut wie möglich zu optimieren.

Übung 3.3

Welche Schaltfläche müssen Sie aktivieren, bevor Sie einen Code aufzeichnen, damit dieser später auf beliebige Bereiche im Tabellenblatt angewendet werden kann?

Übung 3.4

Welche Möglichkeiten zum schnellen Starten eines Makros haben Sie in diesem Kapitel kennengelernt?

3.7 Zusammenfassung

In diesem Kapitel haben Sie Ihre ersten Schritte in Richtung VBA-Programmierung unternommen.

- Sie haben erfahren, wofür der Makrorekorder da ist und wissen nun, wie dieser bedient werden kann.

3.7 Zusammenfassung

- Sie sind nun in der Lage, ein aufgezeichnetes Makro im VBA-Editor zu finden und den aufgezeichneten Code zu optimieren.
- Sie können Makros schneller starten, indem Sie ihnen eine Tastenkombination zuweisen, eine Schaltfläche auf dem Tabellenblatt einfügen oder eine Schaltfläche für das Makro in die Symbolleiste für den Schnellzugriff aufnehmen.