

# OUTLOOK: Feiertage mit VBA-Skript erstellen/entfernen

Dr. V. Thormählen, 28. Sept. 2019

---

## Inhalt

Abbildungen .....	I
Listings .....	I
1 Aufgabenstellung .....	1
2 Feiertage mit eingebauter OUTLOOK-Funktion in den Standardkalender eintragen .....	1
3 Benutzerformular erstellen .....	2
4 Feiertage in den Standardkalender einfügen .....	5
4.1 Duplikate verhindern .....	6
4.2 Nur Wochenfeiertage in den Standardkalender einfügen .....	7
5 Feiertage aus dem Standardkalender entfernen .....	7
6 VBA-Quellcode .....	II
6.1 Benutzerformular erstellen .....	II
6.2 Kalendereinträge erstellen .....	IV
6.3 Kalendereinträge entfernen .....	VIII
6.4 Termine am Wochenende erkennen .....	VIII
Literaturverzeichnis .....	VIII

## Abbildungen

Abbildung 1: Gesetzliche Feiertage in 16 deutschen Bundesländern (Quelle [1]) .....	1
Abbildung 2: OUTLOOK-Kalenderoption: <i>Feiertage in Kalender eintragen</i> .....	1
Abbildung 3: 276 erzeugte Kalendereinträge bei Anwendung der OUTLOOK-Standardfunktion .....	2
Abbildung 4: Boolesche Schalter zur Bestimmung best. gesetzlicher Feiertage durch den Benutzer ....	2
Abbildung 5: Benutzerformular mit Registerblatt 0 des Multiseiten-Steuerelements .....	3
Abbildung 6: Benutzerformular mit Registerblatt 1 des Multiseiten-Steuerelements .....	3
Abbildung 7: Erfolgsmeldung nach Erstellung der Feiertage 2019 in 'Nordrhein-Westfalen' .....	6
Abbildung 8: 13 Feiertage im Bundesland 'Nordrhein-Westfalen' im Jahr 2019 .....	6
Abbildung 9: Die Routine <i>AllDayEventExists</i> verhindert doppelte Kalendereinträge .....	7
Abbildung 10: Erfolgsmeldung nach dem Entfernen aller Feiertage aus dem Standardkalender .....	7

## Listings

Listing 1: Benutzerformular erstellen .....	III
Listing 2: Kalendereinträge erstellen .....	VII
Listing 3: Kalendereinträge entfernen .....	VIII
Listing 4: Termine am Wochenende erkennen .....	VIII

# 1 Aufgabenstellung

In diesem Beitrag wird demonstriert, wie mit einer VBA-Prozedur die Feiertage in einem ausgewählten deutschen Bundesland für ein vorgegebenes Kalenderjahr erzeugt und automatisch in den OUTLOOK-Standardkalender eingefügt werden können. Auch das automatische Entfernen der erstellten Kalendereinträge wird vorgeführt.

Zuvor wird dargestellt, warum das Erstellen von Kalendereinträgen mit der eingebauten OUTLOOK-Kalenderoption „*Feiertage in Kalender eintragen*“ in der föderativ gegliederten Bundesrepublik Deutschland nur bedingt geeignet ist.

Die gesetzlichen Feiertage sind in den 16 deutschen Bundesländern unterschiedlich geregelt (s. Abbildung 1). Welche Tage gesetzliche Feiertage sind, richtet sich nach den Feiertagsgesetzen des jeweiligen Bundeslandes. Sie besitzen nicht nur arbeitsrechtliche Bedeutung, sondern spielen auch für die Personaleinsatzplanung im Allgemeinen eine wichtige Rolle, beispielweise für die Terminplanung von Freiberuflern.

Datum	Gesetzliche Bezeichnungen	BW	BY	BE	BB	HB	HH	HE	MV	NI	NW	RP	SL	SN	ST	SH	TH
	Referenzen und Anmerkungen	[A-1]	[A-2]	[A-3]	[A-4]	[A-5]	[A-6]	[A-7]	[A-8]	[A-9]	[A-10]	[A-11]	[A-12]	[A-13]	[A-14]	[A-15]	[A-16]
1. Januar	Neujahr(stag)	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
6. Januar	Heilige Drei Könige (Epiphania), Erscheinungsfest	Ja	Ja												Ja		
8. März	Frauentag			Ja													
Donnerstag vor Ostersonntag	Gründonnerstag	[A-1]															
Freitag vor Ostersonntag	Karfreitag	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
	siehe Osterdatum			Ja				Ja *									
Montag nach Ostersonntag	Ostersonntag	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
1. Mai	Erster Mai, Tag der Arbeit	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
	(Christi-Himmelfahrtstag)	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
	Ostersonntag + 39 Tage <sup>[A-17]</sup>																
	Ostersonntag + 49 Tage <sup>[A-17]</sup>			Ja				Ja *									
Montag nach Pfingstsonntag	Pfingstmontag	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
	Ostersonntag + 60 Tage <sup>[A-17]</sup>	Ja	Ja					Ja			Ja	Ja	Ja	[A-18]			[A-18]
8. August	Augsburger Hohes Friedensfest			[A-19]													
15. August	Maria Himmelfahrt(stag)			[A-19]									Ja				
20. September	Weltkindertag																Ja
3. Oktober	Tag der Deutschen Einheit	Ja <sup>[A-20]</sup>	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
31. Oktober	Reformationstag-fest <sup>[A-21]</sup>	[A-1]			Ja	Ja	Ja			Ja	Ja				Ja	Ja	Ja
1. November	Allerheiligen(tag)	Ja	Ja								Ja	Ja	Ja				
	Mittwoch vor dem 23. November														Ja		
25. Dezember	1./Erster Weihnachts(fer)tag	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
26. Dezember	2./Zweiter Weihnachts(fer)tag	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
	Gesamtzahl	12	14	10	12	10	10	*	10	10	11	11	12	12	11	10	12
	Gesamtzahl im überwiegenden Teil eines Landes geltender und nicht stets auf einen Sonntag fallender Feiertage <sup>[A-22]</sup>	12	13	10	10	10	10	10	10	10	11	11	12	11	11	10	11
	davon <sup>[A-23]</sup> an einem festen Datum	7	8	6	6	6	6	5	6	6	6	6	7	6	7	6	7
	davon <sup>[A-23]</sup> an einem festen Wochentag	5	5	4	4	4	4	5	4	4	5	5	5	5	4	4	4

Abbildung 1: Gesetzliche Feiertage in 16 deutschen Bundesländern (Quelle [1])

# 2 Feiertage mit eingebauter OUTLOOK-Funktion in den Standardkalender eintragen

OUTLOOK 2010 besitzt eine eingebaute Funktion, um Feiertage automatisch in den standardmäßigen Kalender einzutragen<sup>1</sup> (s. Abbildung 2).

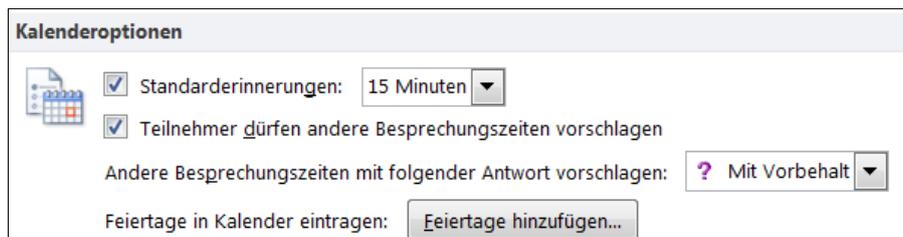


Abbildung 2: OUTLOOK-Kalenderoption: Feiertage in Kalender eintragen

<sup>1</sup> Navigation: Datei > Optionen > Kalender > Kalenderoptionen > Feiertage hinzufügen

Die Schaltfläche "Feiertage hinzufügen" (s. Abbildung 2) erzeugt insgesamt 276 Kalendereinträge, davon für 12 Jahre 8 bewegliche Feiertage (s. Abbildung 3, gelb markiert) und für 20 Jahre andere Feiertage.

Zahl d. Jahre	Feiertag
12	Himmelfahrt
12	Fronleichnam
12	Buß- und Betttag (z. T.)
12	Ostersonntag
12	Ostermontag
12	Karfreitag
12	Pfingstsonntag
12	Pfingstmontag
20	Allerheiligen
20	Maria Himmelfahrt
20	1. Weihnachtstag
20	2. Weihnachtstag
20	Tag der Deutschen Einheit
20	Heilige Drei Könige
20	Neujahrstag
20	Reformationstag
20	Tag der Arbeit
<b>276</b>	<b>Kalendereinträge</b>

Abbildung 3: 276 erzeugte Kalendereinträge bei Anwendung der OUTLOOK-Standardfunktion

Der Reformationstag ist aber nur in 9 von 16 Bundesländern ein gesetzlicher Feiertag (s. Abbildung 1). Fronleichnam ist nur in 6 Bundesländern ausnahmslos ein gesetzlicher Feiertag, in Sachsen und Thüringen aber nur teilweise (s. Abbildung 4).

Schalter	Bundesland	Name des Feiertags	Eingrenzung ...
bolKatholisch	Bayern (BY)	Mariä Himmelfahrt	bei überwiegend katholischer Bevölkerung
	Thüringen (TH)	Fronleichnam	
bolFronleichnam	Sachsen (SN)	Fronleichnam	für best. Gemeinden in den Kreisen Bautzen u. Westlausitz
bolFriedensfest	Bayern (BY)	Friedensfest	Stadtgebiet von Augsburg

Abbildung 4: Boolesche Schalter zur Bestimmung best. gesetzlicher Feiertage durch den Benutzer

Feiertage wie das Augsburger *Friedensfest* in Bayern, der *Weltkindertag* in Thüringen sowie der *Internationale Frauentag* in Berlin fehlen in der Auflistung (s. Abbildung 3). Kurz, bei Nutzung der eingebauten Funktion "Feiertage hinzufügen" durch den Benutzer muss der Standardkalender in OUTLOOK händisch angepasst werden. Das ist bei 276 automatisch erzeugten Kalendereinträgen für 12 bzw. 20 Jahre eine ziemliche Herausforderung.

### 3 Benutzerformular erstellen

Aufgrund der Unzulänglichkeiten der eingebauten Funktion "Feiertage hinzufügen" für die praktische Terminplanung bietet es sich an, eine maßgeschneiderte Lösung für das Hinzufügen von Feiertagen in den OUTLOOK-Standardkalender zu erstellen. Basis dafür ist ein benutzerdefiniertes Formular, das in OUTLOOK erstellt wird. Die Gestaltung dieses Formulars zeigen Abbildung 5 und Abbildung 6.

Abbildung 5: Benutzerformular mit Registerblatt 0 des Multiseiten-Steuerelements

Abbildung 6: Benutzerformular mit Registerblatt 1 des Multiseiten-Steuerelements

Weitere Einzelheiten des benutzerdefinierten Formulars sind in den folgenden Tabellen 1, 2 und 3 enthalten:

Steuerelement	Typ	Beschriftung
lblTitle	Bezeichnungsfeld	OUTLOOK-Kalendereinträge einfügen oder entfernen
multipage1	Multiseiten	frmMultiSeiten
cmdOK	Befehlsschaltfläche	OK
cmdCancel	Befehlsschaltfläche	Abbruch

Tabelle 1: Die vier Steuerelemente des benutzerdefinierten Formulars

Registerblatt	Index	Beschriftung
pgInsert	0	einfügen
pgRemove	1	entfernen

Tabelle 2: Die beiden Registerblätter des Multiseiten-Steuerelements

Steuerelement	Typ	Beschriftung
<i>Steuerelemente im Registerblatt 'pgInsert'</i>		
lblJahr	Bezeichnungsfeld	Feiertage im Jahr
txtJahr	Textfeld	
lblSelection	Bezeichnungsfeld	Bitte eine deutsches Bundesland auswählen
cboLand	Kombinationsfeld	
fraSpecial	Rahmen	Regelungen für die Länder [BY, SN, TH]
chkKatholisch	Kontrollkästchen	BY: Überwiegend katholische Bevölkerung?
chkAugsburg	Kontrollkästchen	BY: Augsburger Friedensfest?
chkFronleichnam	Kontrollkästchen	SN u. TH: Fronleichnam?
<i>Steuerelemente im Registerblatt 'pgRemove'</i>		
chkRemove	Kontrollkästchen	Wirklich alle Feiertage entfernen?

Tabelle 3: Die neun Steuerelemente in den Registerblättern des Multiseiten-Steuerelements

Der VBA-Quellcode des benutzerdefinierten Formulars ist in Listing 1 enthalten. Er enthält fünf Subroutinen mit folgenden Aufgaben (s. Tabelle 4).

Name	Aufgabe
UserForm_Initialize()	Setzt Anfangswerte für das Benutzerformular
cboLand_Change()	Reagiert auf eine Änderung des Kombinationsfelds 'cboLand'
UserForm_QueryClose(...)	Verhindert das Schließen des Benutzerformulars mit dem roten Kreuz
cmdOK_Click()	Ruft eine zugeordnete Prozedur auf und führt sie aus.
cmdCancel_Click()	Beendet das Benutzerformular

Tabelle 4: Subroutinen des Benutzerformulars

## 4 Feiertage in den Standardkalender einfügen

Für das ausgewählte Jahr (2019) und das ausgewählte deutsche Bundesland (NW – Nordrhein-Westfalen) (s. Abbildung 5) erstellt die Prozedur *InsertCalendarItems* die entsprechenden Einträge in den OUTLOOK-Standardkalender. Diese Prozedur benötigt 5 Argumente:

Name des Arguments	Inhalt des Arguments
intYr	(aktuelles) Kalenderjahr
strLand	Langname eines dt. Bundeslandes
bolFriedensfest	boolescher Schalter für Augsburger Friedensfest
bolFonleichnam	boolescher Schalter für Fronleichnam
bolKatholisch	boolescher Schalter für überwiegend kath. Bevölkerung

Tabelle 5: Argumente der Prozedur *InsertCalendarItems*

Die Prozedur *InsertCalendarItems* (s. Listing 2) benötigt zwei Subroutinen:

Name der Subroutine	Aufgabe der Subroutine
<i>SetRecurringHolidaysFromYear</i>	Wiederkehrende Feiertage ab Jahr <i>intYr</i> für das Land <i>strKuerzel</i> in den Standardkalender eintragen. <ul style="list-style-type: none"> <li>• Führt für jeden festen Feiertag die Routine <i>CreateRecurringAllDayEvent</i> aus.</li> <li>• Ruft dort jeweils die Funktion <i>AllDayEventExists</i> auf.</li> <li>• Erstellt dann ggf. einen <i>wiederkehrenden</i> Termin.</li> </ul>
<i>SetMoveableHolidaysForThisYear</i>	Bewegliche Feiertage des Jahres <i>intYr</i> für das Land <i>strKuerzel</i> in den Standardkalender eintragen: <ul style="list-style-type: none"> <li>• Bestimmt bewegliche Feiertage mit den beiden Funktionen <i>EasterSunday</i> und <i>BussUndBetttag</i>.</li> <li>• Führt für jeden beweglichen Feiertag die Subroutine <i>CreateSingleAllDayEvent</i> aus.</li> <li>• Ruft dort jeweils die Funktion <i>AllDayEventExists</i> auf.</li> <li>• Erstellt dann ggf. einen <i>einmaligen</i> Termin.</li> </ul>

Tabelle 6: Subroutinen der Prozedur *InsertCalendarItems*

Die Prozedur *SetMoveableHolidaysForThisYear* (s. Tabelle 6, 1. Spalte, 3. Zeile) benötigt drei Funktionen (s. Tabelle 7):

Name der Funktion	Aufgabe der Funktion
<i>EasterSunday</i>	Ostersonntag eines best. Jahres ermitteln.
<i>BussUndBetttag</i>	Das Datum des Buß- und Bettags eines best. Jahres ermitteln.
<i>AllDayEventExists</i>	Gibt 'True' zurück, wenn ein Kalendereintrag am angegebenen Datum und Betreff bereits vorhanden ist, sonst 'False'.

Tabelle 7: Benötigte Funktionen für die Prozedur *SetMoveableHolidaysForThisYear*

Bei Anwendung der Prozedur *InsertCalendarItems* mit den beiden Argumenten (vgl. Abbildung 5)

- NW (für Nordrhein-Westfalen)
- 2019 (für das Jahr 2019)

wird beispielsweise diese Erfolgsmeldung ausgegeben (s. Abbildung 7):

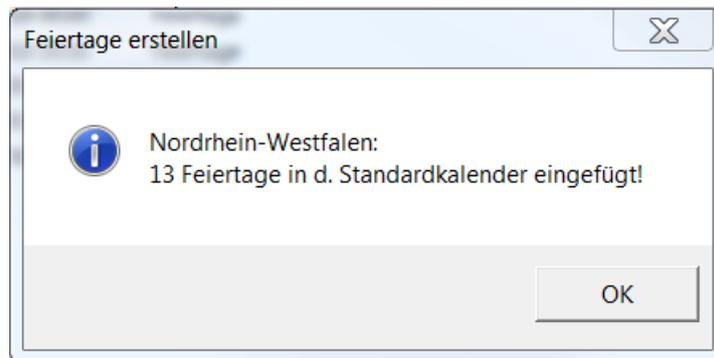


Abbildung 7: Erfolgsmeldung nach Erstellung der Feiertage 2019 in 'Nordrhein-Westfalen'

Unter der Kategorie „*Feiertage*“ sind die 13 relevanten Feiertage im OUTLOOK-Standardkalender zu erkennen (s. Abbildung 8)<sup>2</sup>. Davon entfallen 2 Feiertage (Ostersonntag u. Pfingstsonntag) auf ein Wochenende. Die anderen 11 Feiertage sind sog. Wochenfeiertage, weil sie *nicht* auf ein Wochenende (Sa/So) fallen.

Betreff	Termin-/Bespr...	Beginn	Ende	Serienmuster	Kategorien
Kategorien: Feiertage (13 Elemente)					
Neujahr		Di 01.01.2019 00:00	Mi 02.01.2019 00:00	jedes Jahr am 1. Januar	Feiertage
Karfreitag		Fr 19.04.2019 00:00	Sa 20.04.2019 00:00		Feiertage
Ostersonntag		So 21.04.2019 00:00	Mo 22.04.2019 00:...		Feiertage
Ostermontag		Mo 22.04.2019 00:00	Di 23.04.2019 00:00		Feiertage
Tag d. Arbeit		Mi 01.05.2019 00:00	Do 02.05.2019 00:00	jedes Jahr am 1. Mai	Feiertage
Chr. Himmelfahrt		Do 30.05.2019 00:00	Fr 31.05.2019 00:00		Feiertage
Pfingstsonntag		So 09.06.2019 00:00	Mo 10.06.2019 00:...		Feiertage
Pfingstmontag		Mo 10.06.2019 00:00	Di 11.06.2019 00:00		Feiertage
Fronleichnam		Do 20.06.2019 00:00	Fr 21.06.2019 00:00		Feiertage
Tag der Deutschen Einheit		Do 03.10.2019 00:00	Fr 04.10.2019 00:00	jedes Jahr am 3. Oktober	Feiertage
Allerheiligen		Fr 01.11.2019 00:00	Sa 02.11.2019 00:00	jedes Jahr am 1. November	Feiertage
1. Weihnachtstag		Mi 25.12.2019 00:00	Do 26.12.2019 00:00	jedes Jahr am 25. Dezember	Feiertage
2. Weihnachtstag		Do 26.12.2019 00:00	Fr 27.12.2019 00:00	jedes Jahr am 26. Dezember	Feiertage

Abbildung 8: 13 Feiertage im Bundesland 'Nordrhein-Westfalen' im Jahr 2019

#### 4.1 Duplikate verhindern

Die Subroutine *AllDayEventExists* (s. Listing 2) verhindert doppelte Kalendereinträge, wenn die Prozedur *InsertCalendarItems* versehentlich mit den zuvor gewählten Argumenten erneut ausgeführt wird, ohne dass vorher alle Kalendereinträge der Kategorie *Feiertage* entfernt wurden. In diesem Fall wird folgende Fehlermeldung ausgegeben:

<sup>2</sup> 11 bundesweite/deutschlandweite Feiertage sowie Fronleichnam (Do) und Allerheiligen (Fr).

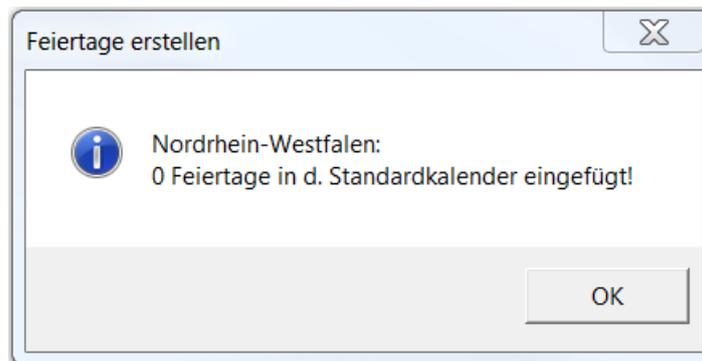


Abbildung 9: Die Routine *AllDayEventExists* verhindert doppelte Kalendereinträge

## 4.2 Nur Wochenfeiertage in den Standardkalender einfügen

Um dafür zu sorgen, dass nur sog. Wochenfeiertage in den OUTLOOK-Standardkalender eingefügt werden, kann die Funktion *IsWeekendAppointment* (s. Listing 4) mit dem Argument *dtmStart* (d. h. Startdatum) eines einmaligen oder wiederkehrenden Termins in den Ablauf der Prozedur *InsertCalendarItems* eingebaut werden<sup>3</sup>.

## 5 Feiertage aus dem Standardkalender entfernen

Das Entfernen aller Kalendereinträge der Kategorie *Feiertage* aus dem OUTLOOK-Standardkalender erfolgt via Benutzerformular (vgl. Abbildung 6), indem die Prozedur *DeleteCalendarItemsByCategory* (s. Listing 3) ausgeführt wird. Abbildung 10 beinhaltet die entsprechende Erfolgsmeldung.

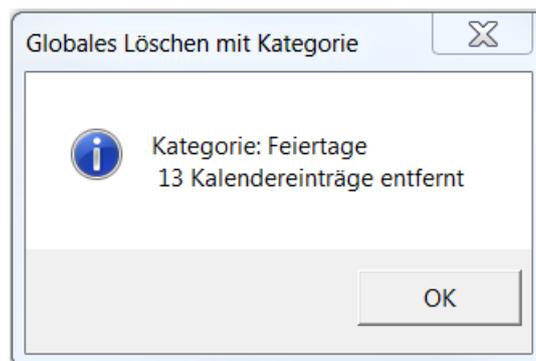


Abbildung 10: Erfolgsmeldung nach dem Entfernen aller Feiertage aus dem Standardkalender

---

<sup>3</sup> Wochenfeiertage fallen auf einen Montag bis inkl. Samstag. Die Funktion *IsWeekendAppointment* betrachtet aber Feiertage, die auf einen Samstag fallen, nicht als Wochenfeiertag. Es wird eine 5-Tage-Woche angenommen.

## 6 VBA-Quellcode

### 6.1 Benutzerformular erstellen

```
Option Explicit
' Globale Konstante und Variable definieren
Const conCategory As String = "Feiertage" ' OUTLOOK-Kategorie
Dim intCreated As Integer ' Zähler f. erstellte Kalendereinträge

Private Sub UserForm_Initialize()
    Dim arrState() As String
    ' Mit der Split-Funktion ein 0-basiertes eindimensionales Array mit 16 Elementen erstellen.
    arrState = Split("BW Baden-Württemberg|BY Bayern|BE Berlin|BB Brandenburg|" & _
        "HB Bremen|HH Hamburg|HE Hessen|MV Meckenburg-Vorpommern|" & _
        "NI Niedersachsen|NW Nordrhein-Westfalen|RP Rheinland-Pfalz|" & _
        "SL Saarland|SN Sachsen|ST Sachsen-Anhalt|SH Schleswig-Holsten|TH Thüringen","|")

    With MultiPagel ' Multiseiten-Steuererelement
        .Value = 0 ' erste Registerseite anzeigen
        .Style = fmTabStyleTabs ' Standard-Stil
        .TabOrientation = fmTabOrientationTop ' Register oben anordnen
        .TabFixedWidth = 0 ' automatische Breite
        .TabFixedHeight = 0 ' automatische Höhe
    End With

    With Me
        .cboLand.List = arrState ' Datenquelle des Kombinationsfelds bestimmen
        .cboLand.SetFocus ' Kombinationsfeld fokussieren
        .cboLand.Value = arrState(9) ' Kombinationsfeld mit Anfangswert (9. Array-Element) belegen
        .txtYear = Year(Date) ' aktuelles Kalenderjahr bestimmen
        .chkKatholisch = .chkAugsburg = .chkFronleichnam = False
        .chkKatholisch.Enabled = False
        .chkAugsburg.Enabled = False
        .chkFronleichnam.Enabled = False
    End With
    intCreated = 0
End Sub

Private Sub cboLand_Change()
    Dim intResponse, intStyle As Integer
    Dim strMsg, strTitle As String
    strTitle = "Gesetzlicher Feiertag?"
    intStyle = vbYesNo + vbQuestion + vbDefaultButton2
    With Me
        .chkKatholisch = False
        .chkAugsburg = False
        .chkFronleichnam = False
    End With
    Select Case Left(Me.cboLand.Value, 2)
        Case "BY"
            strMsg = "BY: Ist Augsburger Friedenfest ein Feiertag?"
            intResponse = MsgBox(strMsg, intStyle, strTitle)
            If intResponse = vbYes Then
                With Me.chkAugsburg
                    .Enabled = True
                    .Value = True
                    .Enabled = False
                End With
            End If
            strMsg = "BY: Mariä Himmelfahrt ein Feiertag?"
            intResponse = MsgBox(strMsg, intStyle, strTitle)
            If intResponse = vbYes Then
                With Me.chkKatholisch
                    .Enabled = True
                    .Value = True
                    .Enabled = False
                End With
            End If
    End Select
End Sub
```

```

    End If
    Case "SN", "TH"
        strMsg = "SN: Ist Fronleichnam ein Feiertag?"
        intResponse = MsgBox(strMsg, intStyle, strTitle)
        If intResponse = vbYes Then
            With Me.chkFronleichnam
                .Enabled = True
                .Value = True
                .Enabled = False
            End With
        End If
    End Select
    If Me.cboLand.ListIndex = -1 Then
        Me.cboLand.SetFocus
    End If
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        MsgBox "Bitte das Formular mit der Schaltfläche 'Abbruch' schließen!"
        Cancel = True
    End If
End Sub

Private Sub cmdOK_Click()
    On Error GoTo Exit_Point
    Me.Hide
    If Me.MultiPage1.Value = 0 Then
        With Me
            ' Kalendereinträge erstellen (mit 5 Argumenten)
            Call InsertCalendarItems(.txtYear, .cboLand, _
                .chkAugsburg, .chkKatholisch, .chkFronleichnam)
        End With
    End If
    If Me.MultiPage1.Value = 1 Then
        If Me.chkRemove = True Then
            ' Kalendereinträge entfernen (mit OUTLOOK-Kategorie)
            Call DeleteCalendarItemsByCategory(conCategory)
        End If
    End If
Exit_Point:
    Unload Me
End Sub

Private Sub cmdCancel_Click()
    Me.Hide
    Unload Me
End Sub

```

Listing 1: Benutzerformular erstellen

## 6.2 Kalendereinträge erstellen

```
Private Sub InsertCalendarItems(intYr As Integer, strLand As String,
    bolFriedensfest As Boolean, bolKatholisch As Boolean, bolFronleichnam As Boolean)
    ' Aufgabe: Dieses VBA-Skript fügt dem Outlook-Standardkalender die Feiertage eines best.
    ' Jahres hinzu.
    ' Argumente:
    '   intYr           aktuelles Kalenderjahr
    '   strLand        Langname für dt. Bundesland
    '   bolFriedensfest boolescher Schalter für Augburger Friendsfest in BY
    '   bolFonleichnam boolescher Schalter für Fronleichnam
    '   bolKatholisch  boolescher Schalter für überwiegend kath. Bevölkerung
    Dim strKuerzel As String * 2
    On Error GoTo Error_Point
    ' Deutsches Bundesland ermitteln
    strKuerzel = UCase(Left(strLand, 2)) ' Kurzname f. Bundesland
    strLand = Mid(strLand, 4)          ' Langname f. Bundesland
    intCreated = 0                     ' Anfangswert für Zähler
    ' Feste Feiertage des Jahres intYr im Land strKuerzel in den Standardkalender eintragen.
    Call SetRecurringHolidaysFromYear(intYr, strKuerzel, bolFriedensfest, bolKatholisch)
    ' Bewegliche Feiertage d. Jahres intYr im Land strKuerzel in d. Standardkalender eintragen.
    Call SetMoveableHolidaysForThisYear(intYr, strKuerzel, bolFronleichnam)
    MsgBox strLand & ": " & vbCrLf & CStr(intCreated) & " Feiertage in d. Standardkalender
eingefügt!", vbInformation, "Feiertage erstellen"
    Me.Show ' Benutzerformular anzeigen
Exit_Point:
    Exit Sub
Error_Point:
    MsgBox "Laufzeitfehler (" & Err.Number & ") " & Err.Description, vbCritical, "Feiertage"
    Resume Exit_Point
End Sub

Private Function EasterSunday(ByVal lngYear As Long) As Date
    ' Aufgabe: Ostersonntag eines best. Jahres ermitteln.
    Dim lngDiff As Long
    Dim lngDelta As Long
    ' Wenn kein Jahr übergeben wird, das aktuelle Jahr verwenden.
    If lngYear = 0 Then lngYear = Year(Now)
    lngDiff = (((255 - 11 * (lngYear Mod 19)) - 21) Mod 30) + 21
    lngDelta = lngDiff + IIf(lngDiff > 48, 1, 0) + 6 - _
        ((lngYear + Int(lngYear / 4) + lngDiff + IIf(lngDiff > 48, 1, 0) + 1) Mod 7)
    ' Osterdatum zurückgeben
    EasterSunday = DateAdd("d", lngDelta, DateSerial(lngYear, 3, 1))
End Function

Private Function BussUndBetttag(intY As Integer) As Date
    ' Aufgabe: Buß- und Betttag eines Jahres berechnen
    BussUndBetttag = DateSerial(intY, 11, 23)
    BussUndBetttag = BussUndBetttag - Weekday(BussUndBetttag, vbThursday)
End Function

Private Function AllDayEventExists(ByVal dtmDate As Date, ByVal strSubject As String) As
Boolean
    ' Aufgabe: Gibt True zurück, wenn der Kalendereintrag am angegebenen Datum und Betreff
    ' vorhanden ist.
    Dim blnExists As Boolean
    Dim objFolder As Outlook.Folder
```

```

Dim objItems As Outlook.Items
Dim strSearch As String
blnExists = False
strSearch = "[Start] = '" & FormatDateTime(dtmDate, vbShortDate) & " 0:00' And [Subject] = '" & strSubject & "'"
Set objItems =
Outlook.Application.GetNamespace("MAPI").GetDefaultFolder(olFolderCalendar).Items
objItems.Sort "[Start]"
objItems.IncludeRecurrences = True
Set objItems = objItems.Restrict(strSearch)
If Not objItems Is Nothing Then
    If objItems.Count > 0 Then
        Dim objItem As Object
        Set objItem = objItems.GetFirst()
        If Not objItem Is Nothing Then
            blnExists = True
        End If
    End If
End If
Set objItem = Nothing
AllDayEventExists = blnExists
End Function

```

```

Private Sub CreateSingleAllDayEvent(ByVal dtmDate As Date, ByVal strSubject As String)

```

```

' Aufgabe: Erstellt ein ganztägiges Ereignis am angegebenen Datum und Betreff,
' wenn dieses ganztägige Ereignis noch nicht vorhanden ist.

```

```

If Not AllDayEventExists(dtmDate, strSubject) Then
    Dim olAppt As Outlook.AppointmentItem
    Set olAppt = Outlook.Application.CreateItem(olAppointmentItem)
    With olAppt
        .Start = dtmDate
        .Subject = strSubject
        .AllDayEvent = True
        .ReminderSet = False
        .Categories = conCategory
        .Save
        If .BusyStatus <> olBusy Then
            .BusyStatus = olBusy ' ... als "gebucht" anzeigen
            .Save
        End If
    End With
    intCreated = intCreated + 1
    Set olAppt = Nothing
End If
End Sub

```

```

Private Sub CreateRecurringAllDayEvent(ByVal dtmDate As Date, ByVal strSubject As String)

```

```

' Aufgabe: Erstellt ein ganztägiges Ereignis zum angegebenen Datum (dtmDate)
' mit dem angegebenen Betreff (strSubject) und legt ein jährliche wiederkehrendes
' Serienmuster an, falls es noch nicht vorhanden ist.

```

```

If Not AllDayEventExists(dtmDate, strSubject) Then
    Dim olAppt As Outlook.AppointmentItem
    Dim olPattern As Outlook.RecurrencePattern
    Set olAppt = Outlook.Application.CreateItem(olAppointmentItem)
    With olAppt
        .Start = dtmDate
        .Subject = strSubject
        .AllDayEvent = True
        .ReminderSet = False
        .Categories = conCategory

```

```

    If .BusyStatus <> olBusy Then
        .BusyStatus = olBusy ' ... als "gebucht" anzeigen
    End If
    Set olPattern = .GetRecurrencePattern()
    With olPattern ' Serienmuster: jährlich
        .RecurrenceType = olRecurYearly
        .DayOfMonth = Day(dtmDate)
        .MonthOfYear = Month(dtmDate)
        .PatternStartDate = olAppt.Start
        .NoEndDate = True
    End With
    .Save
End With
intCreated = intCreated + 1
Set olPattern = Nothing
Set olAppt = Nothing
End If
End Sub

```

```

Private Sub SetMoveableHolidaysForThisYear(ByVal intY As Integer, strKuerzel As String,
bolFronleichnam As Boolean)

```

```

' Aufgabe: Bewegliche Feiertage eines Jahres bestimmen.
Dim dtmEaster, dtmBusstag As Date
dtmEaster = EasterSunday(intY)
Call CreateSingleAllDayEvent(DateAdd("d", -2, dtmEaster), "Karfreitag")
Call CreateSingleAllDayEvent(dtmEaster, "Ostersonntag")
Call CreateSingleAllDayEvent(DateAdd("d", 1, dtmEaster), "Ostermontag")
Call CreateSingleAllDayEvent(DateAdd("d", 39, dtmEaster), "Chr. Himmelfahrt")
Call CreateSingleAllDayEvent(DateAdd("d", 49, dtmEaster), "Pfingstsonntag")
Call CreateSingleAllDayEvent(DateAdd("d", 50, dtmEaster), "Pfingstmontag")
If InStr(1, "BW BY HE NW RP SL", strKuerzel) > 0 Then
    Call CreateSingleAllDayEvent(DateAdd("d", 60, dtmEaster), "Fronleichnam")
Else
    If InStr(1, "SN TH", strKuerzel) > 0 And bolFronleichnam Then
        Call CreateSingleAllDayEvent(DateAdd("d", 60, dtmEaster), "Fronleichnam")
    End If
End If
If strKuerzel = "SN" Then ' Sachsen
    dtmBusstag = BussUndBetttag(intY)
    Call CreateSingleAllDayEvent(dtmBusstag, "Buß- u. Betttag")
End If
End Sub

```

```

Private Sub SetRecurringHolidaysFromYear(ByVal intY As Integer, ByVal strKuerzel As String,
bolKatholisch As Boolean, bolFriedensfest As Boolean)

```

```

' Aufgabe: Feste Feiertage eines Jahres setzen.
Call CreateRecurringAllDayEvent(DateSerial(intY, 1, 1), "Neujahr")

If InStr(1, "BW BY ST", strKuerzel) > 0 Then
    Call CreateRecurringAllDayEvent(DateSerial(intY, 1, 6), "Hl. 3 Könige")
End If

If strKuerzel = "BE" Then
    Call CreateRecurringAllDayEvent(DateSerial(intY, 3, 8), "Int. Frauentag")
End If

Call CreateRecurringAllDayEvent(DateSerial(intY, 5, 1), "Tag d. Arbeit")

If strKuerzel = "BY" And bolFriedensfest Then
    Call CreateRecurringAllDayEvent(DateSerial(intY, 8, 8), "Augsb. Frieden")

```

```

End If

If strKuerzel = "SL" Or (strKuerzel = "BY" And bolKatholisch) Then
    Call CreateRecurringAllDayEvent(DateSerial(intY,8,15),"Mariä Himmelfahrt")
End If

If strKuerzel = "TH" Then
    Call CreateRecurringAllDayEvent(DateSerial(intY,9,20),"Weltkindertag")
End If

Call CreateRecurringAllDayEvent(DateSerial(intY,10,3),"Tag der Deutschen Einheit")
If InStr(1,"BB HB HH MV NI SN ST SH TH",strKuerzel) > 0 Then
    Call CreateRecurringAllDayEvent(DateSerial(intY,10,31),"Reformationstag")
End If

If InStr(1,"BW BY NW RP SL",strKuerzel) > 0 Then
    Call CreateRecurringAllDayEvent(DateSerial(intY,11,1),"Allerheiligen")
End If

Call CreateRecurringAllDayEvent(DateSerial(intY,12,25),"1. Weihnachtstag")

Call CreateRecurringAllDayEvent(DateSerial(intY,12,26),"2. Weihnachtstag")
End Sub

```

**Listing 2: Kalendereinträge erstellen**

## 6.3 Kalendereinträge entfernen

```
Private Sub DeleteCalendarItemsByCategory(conCategory As String)
    ' Aufgabe: Alle Kalendereinträge einer bestimmten Kategorie (conCategory) entfernen.
    Dim olFolder As Outlook.MAPIFolder ' Verzeichnis
    Dim olItems As Outlook.Items ' Kalendereinträge
    Dim olAppt As Outlook.AppointmentItem ' Termin
    Dim intLoop As Integer ' Schleifenzähler
    Dim intCnt As Integer ' Löschezähler
    Set olFolder = Outlook.Application.GetNamespace("MAPI").GetDefaultFolder(olFolderCalendar)
    Set olItems = olFolder.Items
    For intLoop = olItems.Count To 1 Step -1
        Set olAppt = olItems.item(intLoop)
        With olAppt
            If .Categories = conCategory Then
                .Delete
                intCnt = intCnt + 1
            End If
        End With
    Next intLoop
    MsgBox "Kategorie: " & conCategory & vbCrLf & Str(intCnt) & " Kalendereinträge entfernt",
vbOKOnly + vbInformation, "Globales Löschen mit Kategorie"
    Set olItems = Nothing
    Set olFolder = Nothing
    Me.Show ' Benutzerformular anzeigen
    ' Outlook.Application.Quit ' Anwendung beenden
End Sub
```

Listing 3: Kalendereinträge entfernen

## 6.4 Termine am Wochenende erkennen

```
Private Function IsWeekendAppointment(ByVal dtmStart As Date) As Boolean
    ' Aufgabe: Termin/Ereignis am Wochenende (Sa., So.) erkennen.
    Select Case Weekday(dtmStart, vbSunday)
        Case vbSaturday, vbSunday
            IsWeekendAppointment = True
        Case Else
            IsWeekendAppointment = False
    End Select
End Function
```

Listing 4: Termine am Wochenende erkennen

## Literaturverzeichnis

- [1] wikipedia, „Gesetzliche Feiertage in Deutschland,“ 16 09 2019. [Online]. Available: [https://de.wikipedia.org/wiki/Gesetzliche\\_Feiertage\\_in\\_Deutschland](https://de.wikipedia.org/wiki/Gesetzliche_Feiertage_in_Deutschland). [Zugriff am 29 09 2019].