

1

Einführung

Was ist Scripting?

Scripting bezeichnet die Möglichkeit, einzelne Befehle bzw. Programmcode in EPLAN auszuführen. Dies geschieht über die sogenannte API (Application Programming Interface, dt. Programmierschnittstelle). Hinter der EPLAN-API verbergen sich alle Funktionen, die in der Plattform (Electric P8, Fluid, Pro-Panel usw.) vorhanden sind. Diese Programme bauen alle auf dem gleichen Programmcode auf und sind dadurch untereinander kompatibel. In den verschiedenen Applikationen sind ähnliche bzw. gleiche Funktionen enthalten, z.B. kann man Beschriftungen sowohl in Fluid als auch in Electric P8 erzeugen. Einziger Unterschied ist der Inhalt.

Diese Abläufe werden in EPLAN *Actions* (Aktionen) genannt. Ihnen ist Abschnitt 2.2, „Actions ausführen“, gewidmet, da es mehrere Wege gibt, solche Actions auszuführen. Das Wort Scripting bezieht sich meistens nur auf Scripte, die ab der Lizenz-Ausbaustufe EPLAN Electric P8 Compact genutzt werden können. Um weitere Befehle oder Funktionen ausführen zu können, benötigt man das API-Modul von EPLAN. In diesem Buch wird ausschließlich auf den Standardumfang ab der Compact-Version eingegangen.

Was sind Scripte?

Scripte sind kleine Programmcodes. In EPLAN können diese in zwei Programmiersprachen erstellt werden:

- Microsoft C# (C-Sharp)
- Microsoft VB.NET (Visual Basic.NET)

In den folgenden Kapiteln werden nur Beispiele in C# bereitgestellt, da EPLAN mit dieser Sprache fertigen Code generiert und dadurch eine optimale Vorlage liefert. Ein Script ist nicht allein ausführbar. Es muss in Verbindung mit EPLAN gestartet werden.

Was können Scripte?

Scripte können vieles, aber nicht alles. EPLAN stellt eine Reihe von Befehlen bereit, schränkt diese aber auf einen überschaubaren Bereich ein. Dadurch wird Anwender:innen der Einstieg enorm erleichtert. Auf diese Weise wird auch sichergestellt, dass keine ungewollten Aktionen, z. B. auf das Projekt, ausgeführt werden.

Wir kennen alle die wiederkehrende Aufgabe, Beschriftungen auszugeben. Je Projekt sind mehrere Exporte nötig. Jedes Mal muss das Beschriftungsschema neu ausgewählt, zusätzlich der Ordner benannt und ein Dateiname vergeben werden. Mit einem Script können wir all diese Arbeitsschritte zusammenfassen und z. B. auf einen Button im Menüband legen. Über diese Funktion können wir auch mehrere Beschriftungen nacheinander erzeugen. Auch der PDF-Export kann automatisiert werden. Du möchtest z. B. beim Schließen des Projekts automatisch ein PDF zur Änderungsverfolgung erzeugen? Mit einem Script lässt sich dies problemlos realisieren. Es werden Schnittstellen im Unternehmen benötigt, um Informationen außerhalb von EPLAN, z. B. im ERP-System, zu nutzen? Gar kein Problem! Über die Möglichkeiten im Scripting geht das auf Knopfdruck. Oft muss zwischen verschiedenen Einstellungen hin und her gewechselt werden. Das Suchen in den unzähligen Einstellungen in EPLAN ist mühselig. Dafür schreiben wir uns stattdessen ein Script für die Konfigurationen und erledigen dies unter der Projektierung.

Dies ist eine kleine Auflistung der Möglichkeiten, die mit Scripten realisiert werden können:

- Beschriftungen automatisieren
- PDF-Export
- Backup
- eigene Registerkarten im Menüband erstellen
- grafische Formulare, z. B. mit Buttons, Checkboxes und Auswahldialogen, erstellen
- Eigenschaften verändern
 - Projekteigenschaften
 - Seiteneigenschaften
- Einstellungen
 - Lesen
 - Schreiben
- Artikeldaten verändern

Das sind noch längst nicht alle Funktionen. Durch das Erweitern des Programm-codes können noch mehr Funktionen hinzugefügt werden.

Was kann das API-Modul im Vergleich zum Scripting?

Um den Unterschied etwas deutlicher zu machen, findest du im Folgenden eine kleine Auflistung der wichtigsten Merkmale des API-Moduls in EPLAN:

- Zugriff auf das komplette EPLAN-Datenmodell
- einfacheres Lesen von Objekten
- Zugriff auf mehr Objekte
- direkter Zugriff auf Projekteigenschaften/Projekteinstellungen
- mehr verfügbare Actions

■ 1.1 Menüband – der erste Schritt zum Script

1.1.1 Menüband anpassen

In EPLAN gibt es die Möglichkeit, eigene Registerkarten im Menüband, auch Tabs im Ribbon genannt, zu erstellen. Doch was hat das mit Scripting zu tun? Ein Script ist eigentlich eine Erweiterung der Funktionalität eines Buttons in einem Menüband. In einer benutzerdefinierten Registerkarte im Ribbon können vordefinierte Befehle ausgeführt werden. Dies sind alle von EPLAN offiziell unterstützten Actions, welche zudem in der Hilfe dokumentiert sind. Einen Verweis zur Hilfe findest du in Kapitel 5. Diese Befehle werden auch in einem Script verwendet. Ein Vorteil des Scripts gegenüber dem Button im Menüband ist, dass mehrere Actions ausgeführt werden können. In einer Registerkarte müsste man mehrere Schaltflächen erstellen, um zum gleichen Ergebnis zu kommen. Bei der Menge an Möglichkeiten wird der Arbeitsbereich schnell unübersichtlich.

Im Folgenden wollen wir eine neue Registerkarte im Menüband erstellen. Dazu führen wir einen Rechtsklick auf die grafische Oberfläche aus und wählen den Punkt MENÜBAND ANPASSEN ... im Kontextmenü aus (Bild 1.1).

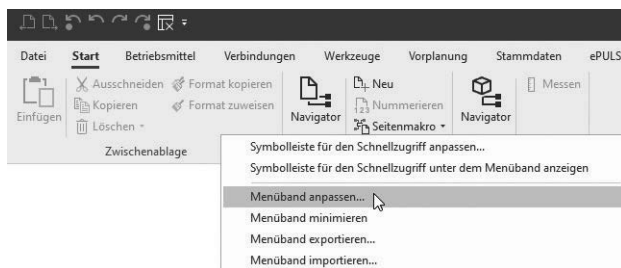


Bild 1.1
Menüband anpassen

Nun wird uns der Dialog zum Anpassen des Menübands angezeigt. Mit der Checkbox wird dargestellt, ob die Registerkarte sichtbar ist oder nicht (Bild 1.2).

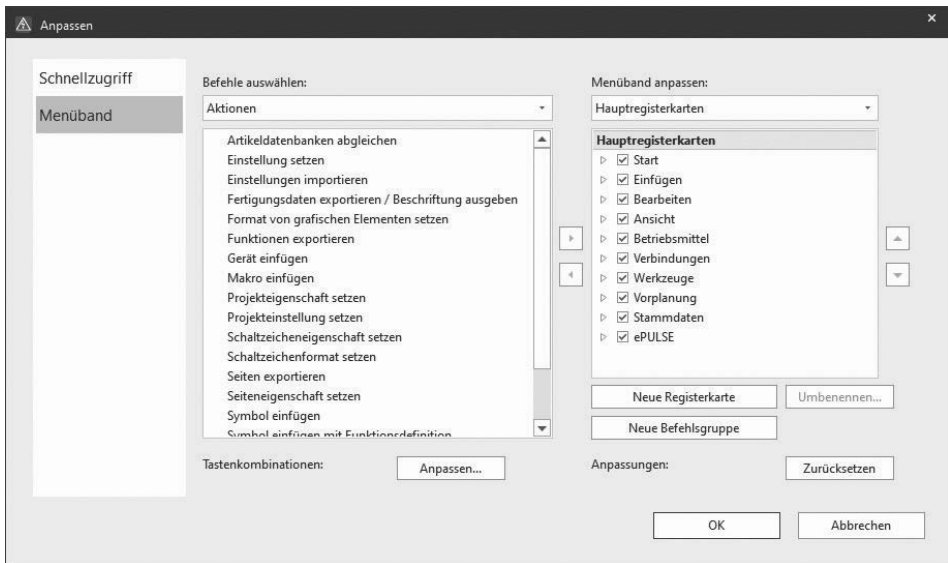


Bild 1.2 Dialog *Menüband: Anpassen*

In Bild 1.3 sehen wir die unterschiedlichen Steuerelemente, welche wir erstellen oder anpassen können:

1. Schnellzugriff: Diese Aktionen sind immer sichtbar.
2. Registerkarten: Zwischen diesen Tabs kann gewechselt werden.
3. Befehlsgruppe: Einzelne Aktionen können dadurch gruppiert werden.
4. Befehl: Der Button wird mit Anzeigename (erforderlich) und mit einem Bild (optional) angezeigt.
5. QuickInfo: Kurze Beschreibung der Aktion (optional)
6. Beschreibung: Lange Beschreibung der Aktion (optional)

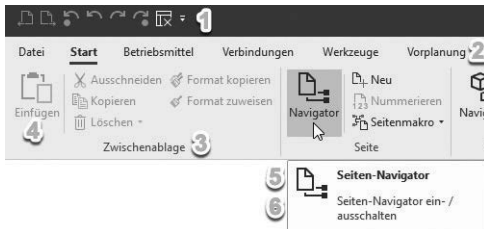


Bild 1.3
Steuerelemente im Menüband

Zusätzlich wird im Menüband das Suchfeld „Was möchten Sie tun?“ angezeigt. In der Suche sind alle Befehle im Menüband durchsuchbar (Bild 1.4).

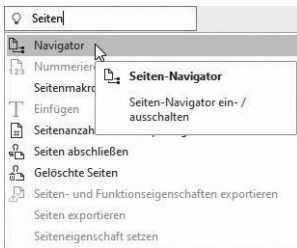


Bild 1.4
Suchfeld „Was möchten Sie tun?“

Wir erstellen über den Button NEUE REGISTERKARTE unsere eigene Registerkarte am Ende der bestehenden Registerkarten (Bild 1.5). Automatisch wurde auch eine Befehlsgruppe hinzugefügt, denn Befehle können nur in Befehlsgruppen eingefügt werden, nicht direkt in eine Registerkarte.

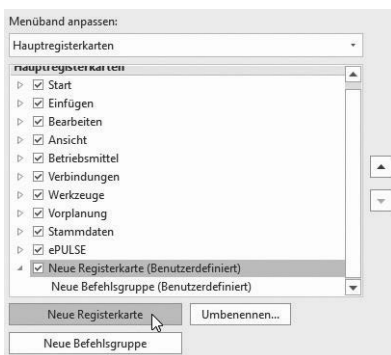


Bild 1.5
Neue Registerkarte erstellen

Über den Button UMBENENNEN ... vergeben wir den Anzeigenamen *EEP8A* (Abkürzung des Buchtitels) für die Registerkarte. Die gleiche Aktion führen wir für die Befehlsgruppe aus und vergeben den Anzeigenamen *Scripte* (Bild 1.6).

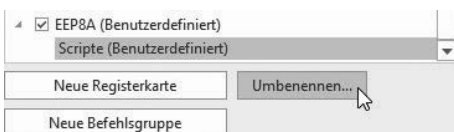


Bild 1.6
Steuerelemente Umbenennen

Alle Fensterpositionen und das Menüband werden über den sogenannten Arbeitsbereich gespeichert. Hier können wir verschiedene Schemata anlegen (Bild 1.8). Man findet die Einstellung am rechten oberen Rand des EPLAN-Fensters ARBEITSBEREICH > ARBEITSBEREICH BEARBEITEN (Bild 1.7).

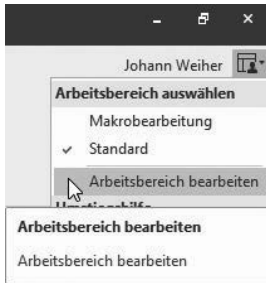


Bild 1.7
Arbeitsbereich auswählen

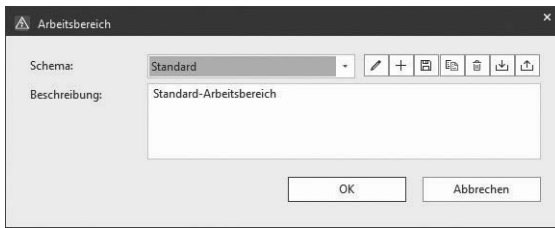


Bild 1.8
Dialog *Arbeitsbereich bearbeiten*

1.1.2 Befehle hinzufügen

Durch die verschiedenen Arbeitsbereiche können wir je nach Aufgabe das Menüband unseren Bedürfnissen anpassen.

In unserer neuen Befehlsgruppe wollen wir nun drei Befehle hinzufügen:

1. Script ausführen
2. Script laden
3. Script entladen

Diese Befehle finden wir in der Kategorie *Weitere Befehle (Datei)*. Mit dem Button HINZUFÜGEN können wir die Befehle in die Befehlsgruppe einfügen (Bild 1.9). Diese Befehle findet man über die EPLAN-Oberfläche unter DATEI > EXTRAS > SCHNITTSTELLEN.

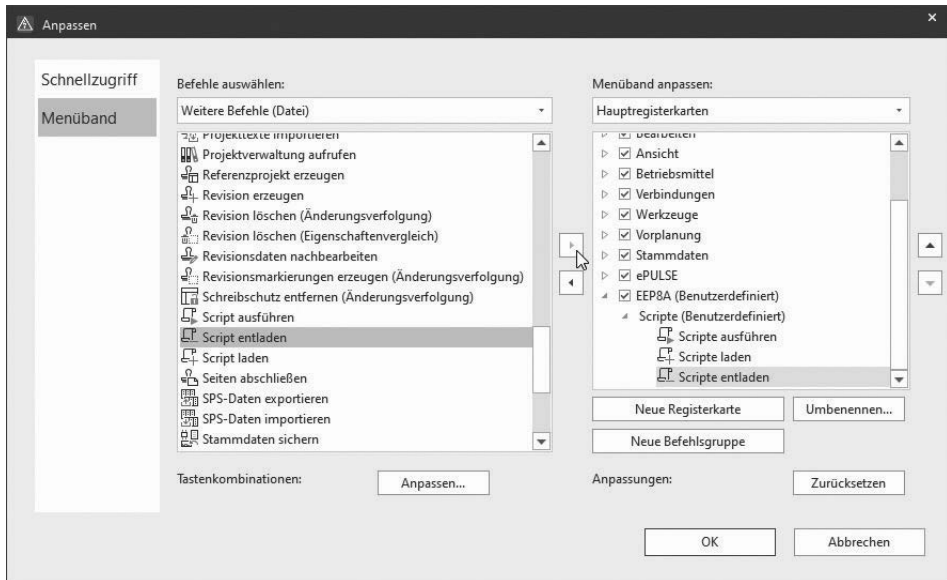


Bild 1.9 Befehle hinzufügen



HINWEIS: Befehle sind abhängig vom Lizenz- und Modulumfang. Ist man z. B. nicht im Besitz des Moduls *Revisionsverwaltung*, werden die enthaltenen Befehle nicht angezeigt.

1.1.3 Befehle mit Parametern

Nun wollen wir uns ein bisschen steigern und fügen einen weiteren Befehl hinzu – und zwar **AKTIONEN > GERÄT EINFÜGEN** (Bild 1.10) in der Befehlsgruppe *Test*.

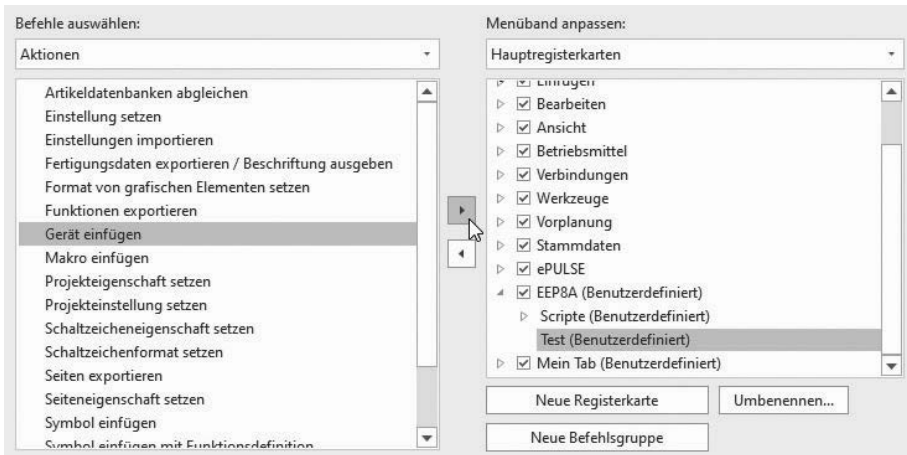


Bild 1.10 Befehl *Gerät einfügen* hinzufügen

Als Anzeigenamen vergeben wir *Motorschutzschalter einfügen*. Nun schauen wir uns das Textfeld *Befehlszeile* etwas genauer an. Auf den ersten Blick steht hier ein sehr kryptischer Text mit vielen Zeichen:

```
XDLInsertDeviceAction /PartNr:? /PartVariant:1
```

Aufgeteilt wird die Befehlszeile in zwei Bereiche:

- *Actionname*: Der Actionname steht immer am Anfang ohne Sonderzeichen.
- *Parameter*: Für eine Aktion kann es einen oder mehrere Parameter geben. Ein Parameter gibt eine Eigenschaft an, die beschreibt, wie die Action ausgeführt werden soll.

Aufbau der Befehlszeile:

```
Actionname /Parameter-1:Wert1 /Parameter-2:Wert2 /Parameter-n:Wert-n
```



HINWEIS: Es muss auf die genaue Schreibweise geachtet werden. Bei Parameterwerten mit Leerzeichen muss der Parameterwert in Anführungszeichen (" ") geschrieben werden. Auch die Leerzeichen zwischen den Parametern müssen genau eingehalten werden.

- *Action*: XDLInsertDeviceAction
- Hinter diesem Text verbirgt sich die Funktion, Geräte in EPLAN einzufügen.

- *Parameter-1: PartNr*

Dieser Parameter ist für EPLAN erforderlich, da bekannt sein muss, welche Artikelnummer eingefügt werden soll.

- *Parameter-2: PartVariant*

Der zweite optionale Parameter gibt an, in welcher Variante der Artikel eingefügt wird (standardmäßig wird dieser Wert auf „1“ gesetzt).

Wir suchen uns ein Beispielgerät, in diesem Falle einen Motorschutzschalter, aus der Artikeldatenbank aus (Bild 1.11). In der EPLAN-Beispieldatenbank finden wir die Artikelnummer *SIE.3RV2011-1EA25-0BA0*.

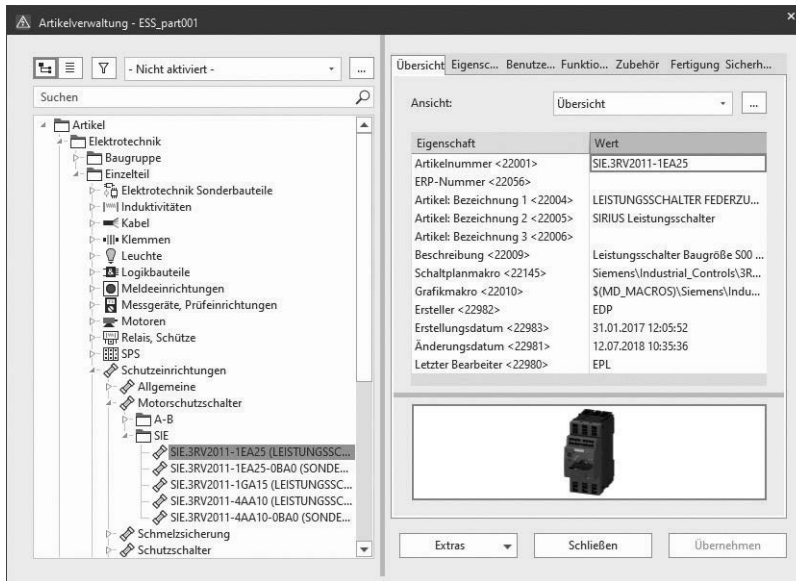


Bild 1.11 Motorschutzschalter in der Artikeldatenbank

Beim Bild wählen wir das *M* aus. Als QuickInfo hinterlegen wir *Motorschutzschalter*, und für die Beschreibung wählen wir den zulässigen Strombereich (Bild 1.12).

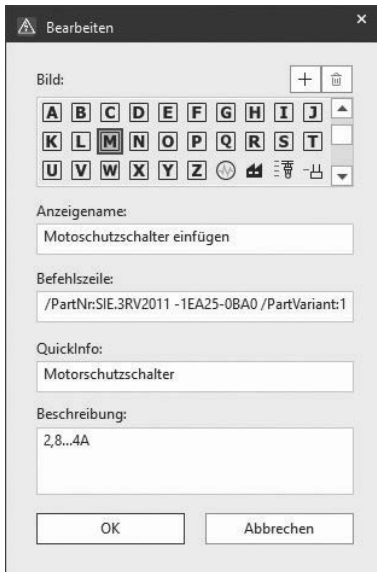


Bild 1.12
Befehl bearbeiten

Nach Betätigen des Befehls im Menüband ist unser eingestellter Motorschutzschalter im grafischen Editor am Cursor angeheftet und kann nun platziert werden (Bild 1.13).

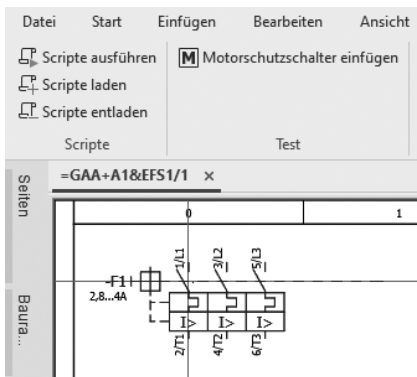


Bild 1.13
Motorschutzschalter einfügen

Wurde eine Artikelnummer angegeben, die nicht existiert, erscheint eine entsprechende Hinweismeldung (Bild 1.14).

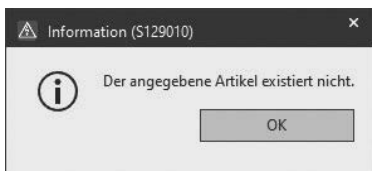


Bild 1.14
Artikelnummer nicht gefunden

Index

Symbole

.NET 13, 14

A

Action 1, 23
ActionCallingContext 44
Addition 56
API 1, 3
Arbeitsbereich 6
Argumente 11
Attribut 35
Ausführen 23
Automatisiert bearbeiten 184

B

Befehl 4, 42
Befehlsgruppe 4
Befehlszeile 37, 224
Benutzereinstellungen 99
Beschriftung 183
Betriebsmittel 238
Button 134
Bytecode 15

C

C# 1, 14, 21, 25
Case 80
Catch 61
Checkbox 136
CommandLineInterpreter 37

Compiler 15
Console 228
CSV 199
Cursor 141

D

Dateiauswahl 180
Datum 172
Debugging 155
Decider 90
DeclareAction 23, 33
DeclareEventHandler 23
DeclareMenu 23, 113
DeclareRegister 23, 36
DeclareUnregister 23, 36
Diagnose-Dialog 35
Division 56

E

Ebenen 237
Einstellungen 99
EnhancedProgress 126
Entwicklungsumgebung 16
EnumDecisionReturn 91
EplanRemoteClient 227
Ereignis 194
Escapezeichen 165
Event 194

F

Fehler 29, 157
FileSelectDecisionContext 180
Float 59
Forms 128
Formular 129

G

GetBoolSetting() 105
GetNumericSetting() 106

H

Haltepunkt 156

I

Icon 93
Integer 55
IntelliSense 25, 90

K

Klassen 72
Kommentare 27
Konsolen-App 227
Konstruktor 73
Kontextmenü 119

L

Label 138
Laden 23
Lambda 228
LINQ 228
ListView 143

M

Mauszeiger 142
Meldungen 29
Menüband 3
Menüpunkt 113

MessageBox 26, 69, 90
Multiplikation 56

O

Objekte 47
Objektorientierte Programmiersprache 72
Objektorientierte Programmierung 47
OpenFileDialog 176, 182
Operator 76
out 96

P

Parameter 8, 42, 224
PathMap 139
PDF 161, 194
Pfadvariable 55
Programmierschnittstelle 1
Progressbar 140
ProjectAction 241
projectmanagement 246
Projekteigenschaften 211, 241
Prozess 155, 161

R

ReadSettings 107
ref 96
region 130
Registerkarte 3, 42
Reguläre Ausdrücke 179
Ribbon 3
Rückgabewert 96

S

SaveFileDialog 173, 180
Schnellaktion 72, 83
Script 23
selectionset 249
Semikolon 25
SetBoolSetting() 102
SetNumericSetting() 103

SetStringSetting() 100
Settings 99
SimpleProgress 123
Sonderzeichen 206
Start 23
Steuerzeichen 49
String 48
Subtraktion 56
Switch 80
Syntax 15

T

TabIndex 139
TabStop 139
Text 48
Textdatei 199
Tooltip 4
Try 61

U

Umlaute 206
Unicode 206
Unterdrückte Dialoge 93
Using-Direktive 29

V

VB.NET 1
Verknüpfung 224
Verweise 24, 25
Visual Studio 17

W

W3C 201
Warnungen 29

X

XML 201

Z

Zeilenumbruch 49, 51