

Microsoft .NET

Select Case / Switch

Wichtige Unterschiede

Inhaltsverzeichnis

Vorwort	3
Zusammenfassen mehrerer Bedingungen.....	4
Nicht-konstante Datentypen	4
Nahezu beliebige Auswahlbedingungen	4
Gültigkeitsbereiche	5
Zweige als Sprungziele	6

Vorwort

Wer kennt das nicht: Man braucht mal eben eine kurze Info zu einem bestimmten Thema und hofft, dass im Internet etwas zu finden sein müsste. Dann fängt man an zu surfen und surft und surft und surft. Oft braucht man buchstäblich Stunden, um alle benötigten Informationen zu einem bestimmten Thema zu finden. Ist Ihnen das nicht auch schon so gegangen?

Um Ihnen die Arbeit etwas zu erleichtern, habe ich mir die Mühe gemacht, Informationen zu einigen ausgewählten Themen in sogenannte Kurzdokumentationen zusammenzustellen. Dabei darf nicht unerwähnt bleiben, dass ich natürlich auch nicht ohne Fehler bin und ebenfalls einige Dinge nicht weiß. Falls Ihnen beim Studium einer Kurzdokumentation also irgendwelche Lücken oder Irrtümer begegnen, würde ich mich sehr über einen kurzen Hinweis freuen, damit ich die Kurzdokumentationen auf den neuesten Stand bringen kann.

Nun bleibt mir nur noch, Ihnen viel Freude beim Lesen der vorliegenden Unterlage zu wünschen.

Viele Grüße,
Michael Scholz

Zusammenfassen mehrerer Bedingungen

Bei SELECT CASE können mehrere Bedingungen in einem Case zusammengefasst werden, bei SWITCH ist dies nicht möglich. Das Zusammenfassen mehrerer Bedingungen in einem Zweig erhöht die Lesbarkeit des Codes und erspart eine Menge Tipparbeit, da Codesegmente nicht dupliziert werden müssen.

In C stand eine ähnliche Funktionalität zur Verfügung, indem mehrere CASE ohne BREAK-Anweisung spezifiziert wurden. Diese Möglichkeit des „Durchfallens“ besteht in C# jedoch ebenfalls nicht, da in jedem CASE ein BREAK erforderlich ist.

Stattdessen können entweder mehrere CASE-Zweige hintereinander notiert oder die Ausführung mittels GOTO CASE an andere Zweige übertragen werden. In diesem Fall kann das BREAK entfallen.

```
int a = 3;
switch (a)
{
    case 1: case 2: case 3:
        break;
    case 4:
        break;
}
```

Nicht-konstante Datentypen

Das Verzweigen anhand nicht-konstanter Objekte, wie sie etwa in Aufzählungen wie COLOR, PENS und BRUSHES vorkommen, wird nicht unterstützt. In diesen Fällen muss anstelle von SWITCH auf IF-Anweisungen mit mehreren ELSE-Zweigen zurückgegriffen werden. SWITCH kann lediglich für numerische Datentypen, Aufzählungstypen und den Datentyp string genutzt werden.

```
Select Case Me.BackColor
    Case Color.Red, Color.Green
        ...
    Case Color.Blue
        ...
End Select
```

Nahezu beliebige Auswahlbedingungen

Bei SELECT CASE können nahezu beliebige Auswahlbedingungen angegeben werden, wie aus folgendem Beispiel ersichtlich ist:

```
Select Case True
    Case Button Is Me.Button1
        ...
    Case Button Is Me.Button2, Button Is Me.Button3
        ...
End Select
```

```

Select Case 2
  Case Test1(1, 2, 3), a + 2
    ...
  Case...
    ...
End Select

```

```

Select Case Test2(...)
  Case Is < 10 + a, 21 To 30, 45, 46
    ...
  Case 15 To 20
    ...
End Select

```

Gültigkeitsbereiche

CASE-Zweige definieren einen eigenen Gültigkeitsbereich. Deshalb ist es möglich, Variablen gleichen Namens in verschiedenen Zweigen zu deklarieren.

```

Dim Name As String = "walter"
Select Case Name
  Case "Werner"
    Dim b As Integer
  Case "Holger"
    Dim b As Integer
  Case "Walter"
End Select

```

SWITCH erlaubt derlei Konstruktionen nicht, da die CASE-Zweige keine eigenen Gültigkeitsbereiche definieren.

```

// FEHLERHAFTER CODE !!!

string Name = "Walter";
switch (Name) {
  case "Werner":
    int b;
    break;
  case "Holger":
    int b;
    break;
  case "Walter":
    break;
}

```

Das kann durch Bildung neuer Anweisungsblöcke zwar geändert werden, allerdings verursacht diese manuelle Eingabe auch eine Menge Tipparbeit.

```
string Name = "Walter";
switch (Name) {
    case "Werner":
        {
            int b;
            break;
        }
    case "Holger":
        {
            int b;
            break;
        }
    case "Walter":
        break;
}
```

Zweige als Sprungziele

Anders als in SELECT CASE dürfen in SWITCH ja wie bereits erwähnt nur konstante Werte ermittelt werden. Das birgt allerdings auch einen Vorteil in sich. Die einzelnen CASE-Zweige der SWITCH-Anweisung werden dadurch eindeutig und können als Sprungziele aus anderen Zweigen verwendet werden, wie es folgendes Beispiel zeigt.

```
int a = 4;
switch (a)
{
    case 1: case 2: case 3:
        MessageBox.Show("Hallo");
        break;
    case 4:
        goto case 5;
        break;
    case 5:
        MessageBox.Show("Meldung von 5");
        break;
}
```

Dabei muss der Wert nicht exakt dem des CASE-Zweiges entsprechen, sondern kann auch zum Beispiel durch eine Berechnung ermittelt werden.

```
int a = 4;
switch (a)
{
    case 1: case 2: case 3:
        MessageBox.Show("Hallo");
        break;
    case 4:
        goto case 2 + 3;
        break;
    case 5:
        MessageBox.Show("Meldung von 5");
        break;
}
```

Zusammenfassung

Begriff	Erklärung
Mehrere Bedingungen	in SWITCH nicht möglich, stattdessen GOTO CASE verwenden oder mehrere CASE-Zweige hintereinander notieren
Nicht-konstante Datentypen	In SWITCH nicht möglich. SWITCH kann lediglich für numerische Datentypen, Aufzählungstypen und den Datentyp string genutzt werden.
Auswahlbedingung IS	wird in SWITCH nicht unterstützt
Auswahlbedingung TO	wird in SWITCH nicht unterstützt
[Test auf Berechnung]	wird in SWITCH nicht unterstützt
Gültigkeitsbereich	SWITCH unterhält in seinen CASE-Zweigen von Haus aus keine eigenständigen Gültigkeitsbereiche
Zweige als Sprungziele	die CASE-Zweige der SWITCH-Anweisung können wegen ihrer Eindeutigkeit mit der Anweisung GOTO CASE angesprungen werden.