

Formen modifizieren

Table of contents

1 Rechtecke modifizieren.....	2
1.1 Variation Kreis.....	2
1.2 Ecken abrunden.....	3
1.3 Umrandung verändern.....	3
1.4 Form mit Textinhalt.....	4

1. Rechtecke modifizieren

Die folgenden Variationen zeigen Ihnen wie Sie mit geringen Veränderungen im Code das Beispiel-Makro mit den Rechtecken selbst modifizieren können. Die zunächst beschriebene Kreis-Variation erzeugen Sie, indem Sie den Code des ersten [Beispiel-Makro](#) selbst verändern. Auf dieselbe Weise können Sie auch alle weiteren Veränderungen vornehmen. Zusätzlich steht aber auch ein [weiteres Makro](#) mit Beschriftungen, abgerundeten Ecken und roter Umrisslinie für Sie zum Download bereit. Es enthält Kommentare zur Erläuterung, die sich jeweils auf den folgenden Code beziehen.

1.1. Variation Kreis

Natürlich können Sie auf die im Beispiel beschriebene Art nicht nur Rechtecke in Ihr Dokument einfügen, sondern auch andere Formen wie z.B. Ellipsen.

1. Diese Veränderung führen Sie durch, indem Sie im Code `RectangleShape` durch `EllipseShape` ersetzen.

```
oRechteck =
oDokument.createInstance( "com.sun.star.drawing.EllipseShape" )
```

Note:

Wie Sie bei dieser Vorgehensweise noch einmal sehen können, spielt die Bezeichnung Ihrer selbstdefinierten Variablen keine Rolle für die Codeausführung, denn `oRechteck` bezeichnet jetzt faktisch einen Kreis. Bei einer dauerhaften Änderung sollten Sie allerdings per „Suchen und Ersetzen“ aus `oRechteck` überall `oKreis` zu machen.

2. Führen Sie jetzt das Makro wieder aus, werden über die bereits vorhandenen Rechtecke noch Kreise gezeichnet, allerdings in der gleichen Farbe. Entsprechend sind nur die Umrisse erkennbar.

Dabei fällt Ihnen vielleicht auf, dass die Kreise exakt in das jeweilige Rechteck eingepasst sind. Dies kommt dadurch zustande, dass bei gleicher Größeneigenschaft die Kreise als Durchmesser den Höhen- bzw. Breitenwert des Rechtecks und somit die gleiche Größe haben. Nachdem die Werte für Höhe und Breite des Rechtecks hier identisch sind, wird ein Quadrat bzw. und jetzt im Fall der Ellipse ein Kreis erzeugt.

Dass die gleiche Positionsangabe dazu führt, dass der Kreis zentriert im Rechteck ausgerichtet wird, hat folgenden Grund:

Die Positionsangabe bei einem Kreis kann sich logischerweise nicht auf seine linke obere Ecke beziehen, da sie nicht existiert. Folglich bezieht sie sich auf die Ecke eines den Kreis umschreibenden Rechtecks.

Durch die gleiche Größe und Positionsangabe ist dieses gedachte Positionierungsrechteck mit dem gezeichneten Rechteck identisch.

3. Möchten Sie einmal nur die Kreise sehen oder einen zweiten Durchlauf, markieren Sie einfach alle Formen im Dokument und löschen sie.

Die durch das Makro generierten Kreis-Formen

1.2. Ecken abrunden

Auch das Rechteck selbst können Sie modifizieren, z.B. durch abgerundete Ecken.

Dazu genügt es, eine einzige Codezeile zu ergänzen. Dies tun Sie logischerweise nach dem Erzeugen des Objekts `RectangleShape` und vor dem Ende der Schleife. Abgesehen davon kann die Erweiterung frei platziert werden. Aus Gründen der Übersichtlichkeit bietet sich allerdings der Bereich an, in dem auch die anderen Rechteck-Eigenschaften wie Größe, Position und Füllfarben festgelegt werden.

Zugegriffen wird in diesem Fall auf die Eigenschaft `CornerRadius` des `RectangleShape`-Objekts.

```
oRechteck.CornerRadius = 500
```

Angegeben wird der Radius des Kreises, der in die Ecken des Rechtecks einbeschrieben wird, hier mit dem Wert 500. Wie bei den Abmessungen der Form verstehen sich die Werte in Hundertstelmmillimetern.

Ist die Ecke - wie bei der Standard-Form - nicht abgerundet, beträgt der Radius einfach 0. Zu kleine Kreisradien sind bei den relativ kleinen Formen des Beispiels schwer oder gar nicht sichtbar, bei einem Wert von 1000 erhalten Sie im Beispiel wieder eine Kreisform. In diesem speziellen Fall würden die vier Kreise übereinander liegen.

1.3. Umrandung verändern

Bei der Standardform ist die Umrandung schwarz und durchgehend, mit einer Linienbreite von 0. Auch diese Eigenschaften Ihrer Form verändern Sie problemlos, im Code wiederum am Besten bei den Rechteck-Eigenschaften. Die Linienbreite geben Sie wieder in Hundertstelmmillimetern an. Beim Linienstil steht 0 für keinen Stil, 1 für eine durchgezogene und 2 für eine gestrichelte Linie.

Durchgezogene Linie:

```
oRechteck.LineStyle = 1
```

Gestrichelte Linie:

```
oRechteck.LineStyle = 2
```

Die gewünschte Linienbreite geben Sie gesondert in Hundertstelmillimetern an, wie hier z.B. 200.

```
oRechteck.LineWidth = 200
```

Die Farbe Ihrer Umrandung legen Sie mit Hilfe einer weiteren Eigenschaft, nämlich LineColor fest, die konkreten Farbwerte geben Sie wieder mit Hilfe der bereits bekannten RGB-Funktion an.

```
oRechteck.LineColor = RGB(255, 0, 0)
```

In diesem Fall wird durch die angegebenen RGB-Werte eine rote Umrandung erzeugt.

Formen mit Umrandung: durchgezogene rote Linie

1.4. Form mit Textinhalt

In Draw werden Formen auch gerne mit Beschriftungen versehen. Dies lässt sich ebenfalls automatisch durchführen, indem Sie Ihr Makro entsprechend ergänzen.

Als Beschriftung wurde im Beispiel „Hallo!“ eingefügt

Die Beschriftung kann stets erst erfolgen, wenn Ihre Formen dem Dokument bereits hinzugefügt wurden. Im Beispiel geschieht dies also nach `oSeite.add(oRechteck)` und bevor Sie die innere Schleife verlassen. Auf diese Weise werden alle Formen beschriftet, die Sie innerhalb der Schleifen erzeugen.

```
oRechteck.setString("Hallo!")'iSpalte & " : " & iZeile)
oRechteck.CharColor = RGB(255,255,255)
oRechteck.CharShadowed = true
```

Der Beschriftungstext für das Objekt `oRechteck` wird durch die Methode `setString()`

erzeugt. Innerhalb der Klammern wird die gewünschte Zeichenfolge als String angegeben, im Beispiel einfach "Hallo!".

Nach `setString()` werden automatisch die Standard-Schrifteigenschaften wie z.B. in Draw die Schriftgröße 18 verwendet. Alle Abweichungen davon legen Sie selbst fest wie im Beispiel die Zeichenfarbe über die Eigenschaft `CharColor`.

Für die Auswahl der Zeichenfarbe verwenden Sie wie bei der Füllfarbe für die Form die RGB-Funktion. In diesem Fall wird als Schriftfarbe Weiß festgelegt und zusätzlich die Eigenschaft `CharShadowed` auf `true` gesetzt, um einen Schriftschatten zu ergänzen. Beides sorgt für eine bessere Lesbarkeit auf dem verschiedenfarbigen Hintergrund.

Note:

Der hier praktizierte Zugriff auf Methoden wie `setString` oder Eigenschaften wie `CharColor` und `CharShadowed` stellt eigentlich eine verkürzte Form dar. Streng genommen handelt es sich nicht um Methoden bzw. Eigenschaften des `RectangleShape`-Objekts. Vielmehr werden diese erst über das Interface `Text` zur Verfügung gestellt. Der BASIC-Interpreter ermöglicht allerdings den direkten Zugriff, genauso könnten Sie aber im Beispiel `oRechteck.Text.setString("Hallo!")` schreiben.

1.4.1. Schrift variieren

Auch Schriftgröße, -art oder -stil - wie fett oder kursiv - gestalten Sie nach Wunsch. In diesem Fall ergänzen Sie den Code des Beispiels wahlweise vor oder nach den bereits definierten Eigenschaften `CharColor` und `CharShadowed` mit weiteren Zugriffen auf Schrifteigenschaften wie z.B. `CharHeight` für die Schriftgröße.

```
oRechteck.CharHeight = 20
```

Für die Festlegung der Schriftgröße genügt es, den gewünschten Wert einzugeben. Er bezieht sich auf die übliche Angabe der Schriftgröße in pt.

Mit der Eigenschaft `CharWeight` bestimmen Sie das Schriftgewicht, also die Dicke und Stärke einer Schrift. Dies ist in dieser Form in der OpenOffice.org-Anwendung nicht möglich, in Ihrem Makro allerdings durchaus.

```
oRechteck.CharWeight = 200
```

Beim Schriftgewicht beziehen sich die Werte auf Angaben in %, d.h. 100 entspräche dem Normalwert, 200 bedeutet eine doppelt so dicke Schrift. Der Wert für normalen Fettdruck liegt bei 150. Hier ist allerdings nur eine Einteilung in 9 Stufen zwischen 50 und 200 möglich (50, 60, 75, 90, 100, 110, 150, 175, 200). Geben Sie einen Wert dazwischen ein, wird automatisch auf den nächsthöheren Wert aufgerundet. Bei Werten über 200 wird

Ihre Schrift auf den Standardwert (100%) zurückgesetzt.

```
oRechteck.CharFontName = "Zapf Chancery"
```

Natürlich muss der angegebene Font auf Ihrem System vorhanden sein. Den Schriftnamen geben Sie als String an, d.h. setzen ihn in Anführungszeichen. Andernfalls erhalten Sie beim Ausführen des Makros eine Fehlermeldung, dass eine Variable nicht gefunden wurde.