bx:barcode

Mit dem Tag `barcode` können verschiedene 1D und 2D-Codes erzeugt werden (Beispiele siehe unten). Ein Check auf Fehler ist mit bx:if möglich. Die Größe ist frei wählbar. Das erzeugte Format ist PNG (ggf. mit Transparenz)

Allgemeine Syntax

```
<bx:barcode.qr width="250" height="250" [type="path"] [rotate="90"]> Inhalt </bx:barcode.qr>
```

Parameter für alle Typen

width und height

Geben die Größe des Barcodes an.

Ein weißer Rand ist teilweise automatisch enthalten und kann bei manchen Typen auch beeinflusst werden.

type

- **image**: (Standard) es wird ein img-Tag erzeugt, Parameter wie style oder class im Barcode-Tag werden in das img-Tag übernommen.
- **path**: es wird nur der Pfad zum Bild ab "/" zurückgegeben (ohne Host, falls Haupthost-Haken im Projekt nicht gesetzt wurde).
- **url**: es wird eine komplette URL zum Bild zurückgegeben (aufgerufener Host oder Projekt-Haupthost).

rotate

Positive Werte drehen das erzeugte Bild im Uhrzeigersinn, negative Werte entgegen (in Grad, z.B. 90).

Die Größe des erzeugten Bildes wird neu berechnet, um den rotierten Barcode unterzubringen (Hintergrund ggf. transparent).

Es ist generell empfehlenswert einen minimalen Kontrast-Rand (z.B. in weiß) zu setzen,damit Barcodeleser den Code einfacher identifizieren können.

Es kann weitere Einschränkungen, je nach offizieller Spezifikation des jeweiligen Barcode-Types, geben, die hier nicht extra aufgeführt sind.Die Ausgabegröße des Barcode-Bilds musst nicht exakt der angegebenen `width` / `height` entsprechen (liegt aber zumindest nicht darüber). Falls eine pixelgenaue Ausgabe benötigt wird, sollte ein Container um das Bild gepackt werden.

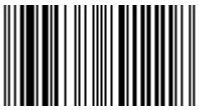
Typ-spezifische Informationen

Die gezeigten Beispiele müssen nicht unbedingt den Anforderungen des jeweiligen Standards entsprechen! Es können also auch potentiell invalide Barcodes entstehen. Es ist die Spezifikation des jeweiligen Types zu Rate zu ziehen!

1-Dimensionale Codes

Bei allen 1-dimensionalen Codes kann der parameter `margin` verwendet werden. Dieser legt die Größe des Randes links und rechts fest (**0** für keinen Rand, Standard ist **10**).

Typ	Ausgabe	Einschränkungen
`<bx:barcode.codabar>`	 "12345"	Beginnt und endet mit einem Guard-Zeichen aus: `ABCD`. ODER beginnt und endet mit einem Guard-Zeichen aus: `TN*E`. Falls keine Guards gesetzt sind, wird automatisch `A` genommen. Enthält nur Zeichen aus: `0123456789-$:/.+`
`<bx:barcode.code_39>`	 "BATIX"	maximal 80 Zeichen aus: `0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ-. *$/+%`
`<bx:barcode.code_93>`	 "BATIX"	maximal 80 Zeichen aus: `0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ-. $/+%abcd*`
`<bx:barcode.code_128>`	 "BATIX"	maximal 80 Zeichen

Typ	Ausgabe	Einschränkungen
`<bx:barcode.itf>` (Interleaved Two of Five)	 "123456"	Weniger als 80 Ziffern, gerade Anzahl Am besten 6, 10, 12, 14, 16, 24, oder 44 Ziffern
`<bx:barcode.ean_8>`	 "12345678"	genau 8 Ziffern
`<bx:barcode.ean_13>`	 "1234567891118"	genau 13 Ziffern Prüfziffer muss stimmen
`<bx:barcode.upc_a>`	 "12345678910"	genau 11 Ziffern (ohne Prüfziffer, wird automatisch berechnet) ODER genau 12 Ziffern (mit Prüfziffer)
`<bx:barcode.upc_e>`	 "12345678"	genau 8 Ziffern

2-Dimensionale Codes

`<bx:barcode.qr>`

Ausgabe



`"http://batix.de"`

Parameter

ecl
gibt das ErrorCorrectionLevel an:

- L ~7% (Standard)
- M ~15%
- Q ~25%
- H ~30%

Einschränkungen

Die Länge des verschlüsselten Strings kann die Ausgabe beeinflussen.

`<bx:barcode.datamatrix>`

Ausgabe

 "http:/batix.de" (rechts mit shape=square)

 "Lorem ipsum dolor sit amet" (rechts mit shape=rectangle)

Parameter

shape

- `none` (Standard, wird automatisch je nach Inhalt gewählt)
- `square` (Quadrat erzwingen)
- `rectangle` (Rechteck erzwingen)

`<bx:barcode.aztec>`

Ausgabe



`"http://batix.de"`

Parameter

ecl
gibt das ErrorCorrectionLevel an (in %, (0 - 100)), 25 sollte Minimum sein.

layers
gibt die Anzahl der Aztec-Layer an

- 0 (Standard): Minimum benutzen
- -1, -2, -3, -4: kompakter Modus
- 1 - 32: selbst angeben (nicht-kompakter Modus)

charset
gibt den Zeichensatz an, der zur Codierung des Inhaltes verwendet werden soll.
Standard ist utf-8.

`<bx:barcode.pdf_417>`

Ausgabe



`"BATIX"`

Parameter

margin
legt die Größe des Randes rundherum fest

ecl
gibt das ErrorCorrectionLevel an.
0 - 8, Standard ist 2.

charset
gibt den Zeichensatz an, der zur Codierung des Inhaltes verwendet werden soll.
Standard ist utf-8.

compact
kompakten Modus benutzen? true oder false.
Standard ist false.

compaction
welcher Verdichtungsmodus benutzt werden soll.

- auto (Standard)
- text
- byte
- numeric

dimensions

- minimale und maximale Spalten- (Cols) und Zeilenanzahl (Rows)
- kommagetrennte Zahlen
- Reihenfolge: minCols,maxCols,minRows,maxRows
- Standard: 2,30,2,30

Fehlerabfrage

```
<bx:if><bx:barcode.ean_8>1234</bx:barcode.ean_8><bx:if.else>Es konnte leider kein Barcode erzeugt werden!</bx:if.else></bx:if>
```

Da bx:barcode im Fehlerfall nichts ausgibt, kann ein alternativer Text mittels bx:if ausgegeben werden.

Beispiele

```
<bx:barcode.qr>Hallo</bx:barcode.qr>


<bx:barcode.datamatrix shape="rectangle">Hallo</bx:barcode.datamatrix>
```