

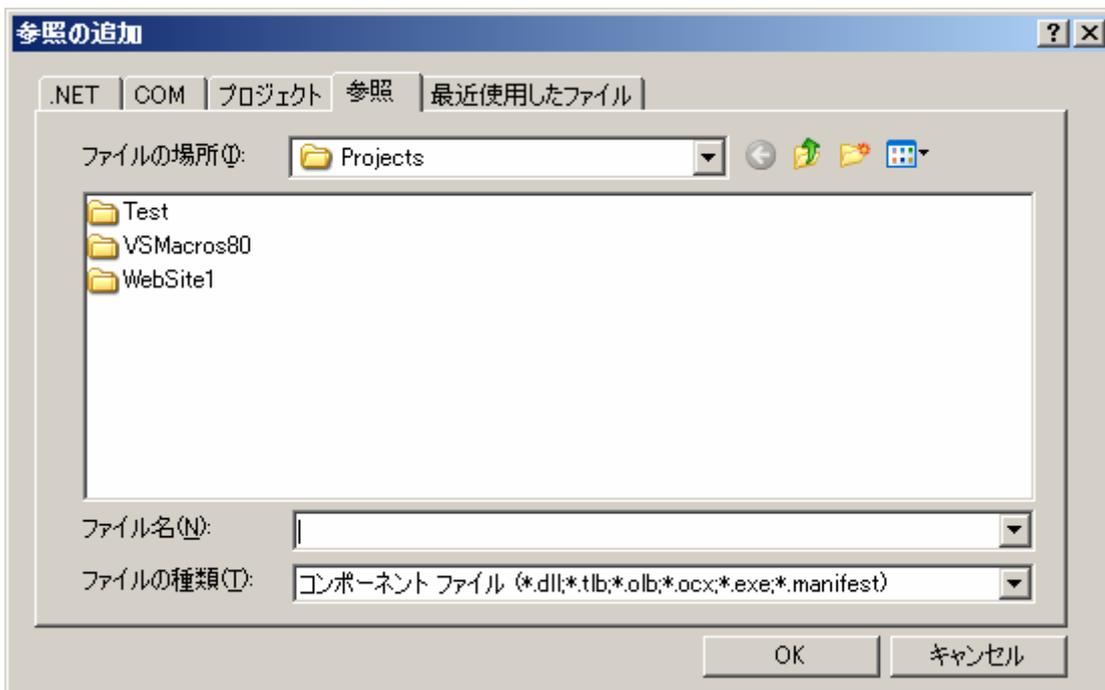
# 【DotNetBarcodeV2.4.0 クラスライブラリ(DotNetBarcode.dll)の仕様書】

## 1. DotNetBarcode クラスライブラリを使う前に

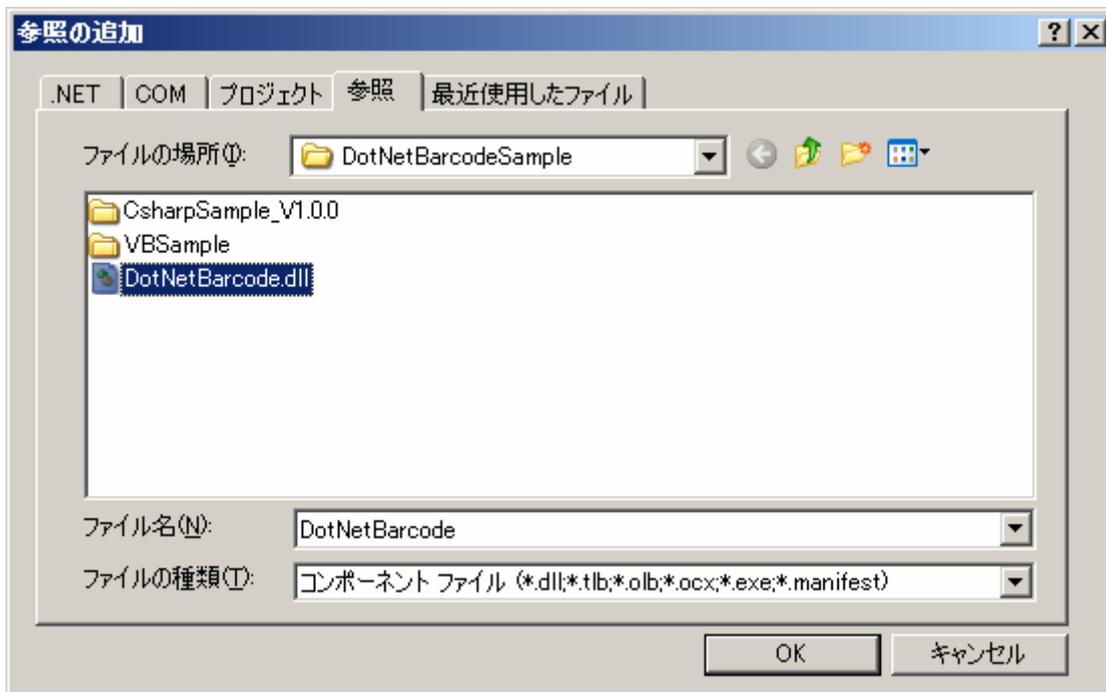
DotNetBarcode クラスライブラリを使う前に、Visual Studio のメニューより、「プロジェクト」－「参照の追加」を選択します。



次に、「参照」タブボタンを押します。



次に、DotNetBarcode クラスライブラリである、DotNetBarcode.dll を選択し、「OK」ボタンを押します。



これで、DotNetBarcode クラスライブラリを使う準備が整いました。

## 2. オブジェクトの新規作成

DotNetBarcode クラスライブラリのオブジェクトは、DotNetBarcode クラス 1 つです。DotNetBarcodeV2.3.0 はルート名前空間を System ですので、Visual Basic の場合は

```
Imports System
```

を C# の場合は

```
using System;
```

をプログラムのはじめに入れておいてください。

bc1 という名前のバーコードオブジェクトを作る場合は以下のように作成します。

### 【Visual Basic の場合】

```
Imports System
```

```
Dim bc1 As DotNetBarcode = New DotNetBarcode
```

### 【C# の場合】

```
using System;
```

```
DotNetBarcode bc1 = new DotNetBarcode ();
```

オブジェクトを新規に作成する時に、バーコードのタイプを指定することができます。(オプション)

QRCode => 二次元バーコードの標準。携帯電話などで読み取り可能

Jan13 => JAN(Japan Article Number)の 13 桁。一般商品に印字され、身近なバーコード。

Jan8 => JAN(Japan Article Number)の 8 桁。一般商品に印字され、身近なバーコード。

Code39 => 数字以外も表示できるコード。産業界で使われています。

**【例】** Dim bc1 As DotNetBarcode = New DotNetBarcode(DotNetBarcode.Types.Code39)

上記例は、Visual Basic で Code39 を使う場合です。

## 3. メソッド一覧

【メソッド名】 WriteBar(code As String, X As Single, Y As Single, X2 As Single, Y2 As Single, ev As System.Drawing.Graphics)

【機能】 バーコードを表示する。

【パラメーター】

code:表示するバーコードの文字

X:バーコードを表示する左上隅の x 座標。

Y:バーコードを表示する左上隅の y 座標。

X2:バーコードを表示する幅。

Y2:バーコードを表示する高さ。

ev:System.Drawing.Graphics

【注意点】

QR コードを使う場合は、バーコードが正方形になるため、Width と High の小さい方の大きさにバーコードが作成されます。ただし、QR コードは 1 ピクセル単位でバーコードを作成するため、Width と High の小さい方の大きさに収まる最大のピクセルを自動的に選択します。1 ピクセルでバーコードを表示しても大きさが収まらない場合は、バーコードは表示されません。

【メソッド名】 CopyToClipboard(code As String, Width As Single, High As Single)

【機能】 バーコードをクリップボードにコピーする。

【パラメーター】

code:表示するバーコードの文字

Width:バーコードをコピーする幅。

High:バーコードをコピーする高さ。

【注意点】

QR コードを使う場合は、バーコードが正方形になるため、Width と High の小さい方の大きさにバーコードが作成されます。ただし、QR コードは 1 ピクセル単位でバーコードを作成するため、Width と High の小さい方の大きさに収まる最大のピクセルを自動的に選択します。1 ピクセルでバーコードを表示しても大きさが収まらない場合は、バーコードはコピーされません。

【メソッド名】 Save(code As String, FileName As String, Width As Single, High As Single)

【機能】 バーコードを SaveFileType で指定したファイル形式に保存する。

指定可能なファイル形式は BitMap,Gif,Jpeg,Png,Tiff

BitMap => ビットマップ・イメージ形式

Gif => GIF ファイル・イメージ形式

Jpeg => JPEG ファイル・イメージ形式

Png => PNG ファイル・イメージ形式

Tiff => TIFF ファイル・イメージ形式

【V2.3.0 から V2.4.0 への重要な変更点】

.Net Framework の制約により、バーコードの保存時の WMF と EMF ファイルの指定は、PNG でしか保存されないことが判明しました。このため、マニュアル上は WMF と EMF ファイルのサポートはやめました。(実際には指定はできますが、PNG で保存されます)

【パラメーター】

code:表示するバーコードの文字

FileName:保存するファイル名。ファイル名にファイルタイプの指定がない時は、自動的に保存するファイル形式に従って補われます。

**Width:**バーコードを保存する幅。

**High:**バーコードを保存する高さ。

**【注意点】**

QR コードを使う場合は、バーコードが正方形になるため、**Width** と **High** の小さい方の大きさにバーコードが作成されます。ただし、QR コードは1ピクセル単位でバーコードを作成するため、**Width** と **High** の小さい方の大きさに収まる最大のピクセルを自動的に選択します。1ピクセルでバーコードを表示しても大きさが収まらない場合は、バーコードは保存されません。

=====

以下のメソッドはQRコード（二次元バーコード）専用のメソッドです。

(Jan13,Jan8,Code39 では使えません)

=====

**【メソッド名】 QRWriteBar(code As String, X As Single, Y As Single, pPixelSize As Integer, ev As System.Drawing.Graphics)**

**【機能】**

1モジュール（1つの点の大きさ：ピクセル）の大きさを指定してQRコードを表示する

**【パラメーター】**

**code:**表示するバーコードの文字

**X:**バーコードを表示する左上隅の x 座標。

**Y:**バーコードを表示する左上隅の y 座標。

**pPixelSize:**1モジュール（1つの点の大きさ：ピクセル）の大きさ。

**ev:**System.Drawing.Graphics

**【注意点】**

このメソッドを使うと、**【プロパティ名】 Type** が1次元バーコードの指定になっていてもQRコードを表示します。

**【メソッド名】 QRCopyToClipboard(code As String, pPixelSize As Integer)**

**【機能】**

1モジュール（1つの点の大きさ：ピクセル）の大きさを指定してQRコードをクリップボードにコピーする。

**【パラメーター】**

**code:**表示するバーコードの文字

**pPixelSize:**1モジュール（1つの点の大きさ：ピクセル）の大きさ。

**【注意点】**

このメソッドを使うと、**【プロパティ名】 Type** が1次元バーコードの指定になっていてもQRコードを表示します。

**【メソッド名】 QRSave(code As String, FileName As String, pPixelSize As Integer)**

**【機能】**

1モジュール（1つの点の大きさ：ピクセル）の大きさを指定してQRコードをSaveFileTypeで指定したファイル形式に保存する。

指定可能なファイル形式は BitMap,Gif,Jpeg,Png,Tiff

BitMap => ビットマップ・イメージ形式

Gif => GIF ファイル・イメージ形式

Jpeg => JPEG ファイル・イメージ形式

Png => PNG ファイル・イメージ形式

Tiff => TIFF ファイル・イメージ形式

#### 【V2.3.0 から V2.4.0 への重要な変更点】

.Net Framework の制約により、バーコードの保存時の WMF と EMF ファイルの指定は、PNG でしか保存されないことが判明しました。このため、マニュアル上は WMF と EMF ファイルのサポートはやめました。(実際には指定はできますが、PNG で保存されます)

#### 【パラメーター】

code:表示するバーコードの文字

FileName:保存するファイル名。ファイル名にファイルタイプの指定がない時は、自動的に保存するファイル形式に従って補われます。

pPixelSize:1 モジュール (1つの点の大きさ:ピクセル) の大きさ。

#### 【注意点】

このメソッドを使うと、【プロパティ名】Type が 1次元バーコードの指定になっていても QRコードを表示します。

## 4. プロパティ一覧

#### 【プロパティ名】 Type

バーコードのタイプを表す。(Default は Jan13)

指定可能なタイプは、QRCode、Jan13、Jan8、Code39

QRCode=>二次元バーコードの標準。携帯電話などで読み取り可能

Jan13 => JAN(Japan Article Number)の 1 3桁。一般商品に印字され、身近なバーコード。

Jan8 => JAN(Japan Article Number)の 8桁。一般商品に印字され、身近なバーコード。

Code39 => 数字以外も表示できるコード。産業界で使われています。

#### 【プロパティ名】 BarColor

バーコードの色を変える。(Default は Color.Black)

#### 【プロパティ名】 BackGroundColor

バーコードの背景色を変える。(Default は Color.White)

#### 【プロパティ名】 FontColor

文字のフォントの色を変える。(Default は Color.Black)

#### 【プロパティ名】 FontBackGroundColor

文字のフォントの背景色を変える。(Default は Color.White)

#### 【プロパティ名】 FontName

文字のフォントの名前を変える。(Default は"MS Pゴシック")

【注意】: この設定は 1次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

#### 【プロパティ名】 FontSize

文字のフォントの大きさを変える。(Default は 9)

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 AddChechDigit

チェックデジットを追加する。(Default は False)

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 PrintChar

文字を表示する。(Default は True)

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 PrintCheckDigitChar

文字を表示する時にチェックデジットを表示する。(Default は False)

Code39 のときは PrintCheckDigitChar=True の場合は\*も表示される。

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 WriteDebugInfo

バーコードのデバッグ情報を書く。(Default は False)

バーコードのデバッグ情報は 1 文字に対して、奇数目のバーか偶数目のバーか最後のバーかで色を変える

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 DebugInfoOddColor

バーコードのデバッグ情報の奇数番目のバーの色。(Default は Color.DarkBlue)

バーコードのデバッグ情報は 1 文字に対して、奇数目のバーか偶数目のバーか最後のバーかで色を変える

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 DebugInfoEvenColor

バーコードのデバッグ情報の偶数番目のバーの色。(Default は Color.LightBlue)

バーコードのデバッグ情報は 1 文字に対して、奇数目のバーか偶数目のバーか最後のバーかで色を変える

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 DebugInfoLastColor

バーコードのデバッグ情報の最後のバーの色。(Default は Color.Red)

バーコードのデバッグ情報は 1 文字に対して、奇数目のバーか偶数目のバーか最後のバーかで色を変える

【注意】：この設定は 1 次元バーコード(JAN13,JAN8,CODE39)のみ有効です。

【プロパティ名】 SaveFileType (DotNetBarcode V2 で追加)

ファイルに保存する時のファイルフォーマットを指定する(Default は BitMap)

指定可能なファイル形式は BitMap,Emf,Gif,Jpeg,Png,Tiff,Wmf

BitMap => ビットマップ・イメージ形式

Emf => メタファイル・イメージ形式

Gif => GIF ファイル・イメージ形式

Jpeg => JPEG ファイル・イメージ形式

Png => PNG ファイル・イメージ形式

Tiff => TIFF ファイル・イメージ形式



シンボルの型番(バージョン)を設定します。

(Default は Automatic)

指定可能な値は、Automatic, Ver01, Ver02... Ver39, Ver40

Automatic => 自動モードを選択(バーコードにする表示する文字が収まるような最小のバージョンが自動的に選択されます。)

Ver01 => バージョン 1 を選択

Ver02 => バージョン 2 を選択

\*

\*

Ver39 => バージョン 39 を選択

Ver40 => バージョン 40 を選択

**【注意】**

指定されたバージョンでは、バーコードにする表示する文字が収まらない場合は、自動的に最適なバージョンが選択されます。

**【プロパティ名】 QRInquireVersion**

自動に設定された場合に、選択されたバージョンを返します。

指定されたバージョンより、文字数の方が大きかった場合は、その文字数を表示する最小のバージョンになります。

文字数が、最大文字数を超えた場合は0になります。その場合は、バーコードのデータは空になります。

**【プロパティ名】 QRInquireNumericBits**

数字モードで、指定した文字でQRコードを作成した時に必要な文字のビット数を返す。

**【注意】**

バーコードにする表示する文字に数字以外の文字がある場合は0を返す。

**【プロパティ名】 QRInquireAlphaNumericBits**

英数字モードで、指定した文字でQRコードを作成した時に必要な文字のビット数を返す。

**【注意】**

バーコードにする表示する文字に英数字以外の文字がある場合は0を返す。

**【プロパティ名】 QRInquireKanjiBits**

漢字モードで、指定した文字でQRコードを作成した時に必要な文字のビット数を返す。

**【注意】**

バーコードにする表示する文字に漢字以外の文字がある場合は0を返す。

**【プロパティ名】 QRInquireBinaryBits**

8ビットバイト・モードで、指定した文字でQRコードを作成した時に必要な文字のビット数を返す。

**【プロパティ名】 QRQuitZone**

クワイエット・ゾーン(余白)のモジュール数

(Default は 4)

**【プロパティ名】 QRInquireModuleSize**

選択されたモジュールのサイズ(縦横それぞれの点の数)を返す。

(実行前の Default は0)

#### 【プロパティ名】 Rotate

回転角度（右回り）を指定する。

回転してもバーコードを表示する座標は必ず右上を指定します。

指定できる値は、`Rotates.Rotate0`,`Rotates.Rotate90`,`Rotates.Rotate180`,`Rotates.Rotate270` の4つの値  
一次元バーコードにも、QRコードにも適用される

`QRRotate` が0以外の値が指定されQRコードが表示される場合、`Rotate` の値をセットすると  
`QRRotate` は0になり無視される。

#### 【プロパティ名】 QRRotate

任意の回転角度を指定可能（QRコード専用）

回転してもバーコードを表示する座標は右上を指定します。

`QRRotate` の値をセットすると、`Rotate` の値は `Rotates.Rotate0` になり無視される。

#### 【プロパティ名】 QRBackGroundFileName

QRコードの背景に表示するビットマップ(や `Jpeg,Gif`)のファイル名

`QRBackGroundFileName` がブランクの場合は、背景のファイルは使われない。

(`BackColor` などが使われる。)

#### 【プロパティ名】 QRColorBigMark

位置検出パターン（大きいマーク）の色を指定します。

(Default は `Color.Black`)

##### 【注意】

この値を変更すると自動的に、`QRColorBigMark1`,`QRColorBigMark2`,`QRColorBigMark3` も  
同じ値で設定されます。

この値は設定のみで、値の照会はできません。値の照会する場合は、`QRColorBigMark1`,  
`QRColorBigMark2`,`QRColorBigMark3` で照会します。

この値は `BarColor` を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRBackColorBigMark

位置検出パターン（大きいマーク）の背景色を指定します。

(Default は `Color.White`)

##### 【注意】

この値を変更すると自動的に、`QRBackColorBigMark1`,`QRBackColorBigMark2`,  
`QRBackColorBigMark3` も同じ値で設定されます。

この値は設定のみで、値の照会はできません。値の照会する場合は、`QRBackColorBigMark1`,  
`QRBackColorBigMark2`,`QRBackColorBigMark3` で照会します。

この値は `BackColor` を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRColorBigMark1

右上の位置検出パターン（大きいマーク）の色を指定します。

(Default は `Color.Black`)

##### 【注意】

この値は `BarColor` を変更すると自動的にその設定した値に変更されます。

また、この値は QRColorBigMark を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRBackColorBigMark1

右上の位置検出パターン（大きいマーク）の背景色を指定します。

(Default は Color.White)

##### 【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRBackColorBigMark を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRColorBigMark2

右下の位置検出パターン（大きいマーク）の色

(Default は Color.Black)

##### 【注意】

この値は BarColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRColorBigMark を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRBackColorBigMark2

右下の位置検出パターン（大きいマーク）の背景色

(Default は Color.White)

##### 【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRBackColorBigMark を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRColorBigMark3

左上の位置検出パターン（大きいマーク）の色

(Default は Color.Black)

##### 【注意】

この値は BarColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRColorBigMark を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRBackColorBigMark3

左上の位置検出パターン（大きいマーク）の背景色

(Default は Color.White)

##### 【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRBackColorBigMark を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRColorSmallMark

位置合わせパターン（小さいマーク）の色

(Default は Color.Black)

##### 【注意】

この値は BarColor を変更すると自動的にその設定した値に変更されます。

#### 【プロパティ名】 QRBackColorSmallMark

位置合わせパターン（小さいマーク）の背景色

(Default は Color.White)

【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

【プロパティ名】 QRColorBigMarkBorder

位置検出パターンの境界線（大きいマークの境界）の色

(Default は Color.White)

【注意】

この値を変更すると自動的に、QRColorBigMarkBorder1,QRColorBigMarkBorder2,QRColorBigMarkBorder3 も同じ値で設定されます。

この値は設定のみで、値の照会はできません。値の照会する場合は、QRColorBigMarkBorder1,QRColorBigMarkBorder2,QRColorBigMarkBorder3 で照会します。

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

【プロパティ名】 QRColorBigMarkBorder1

右上の位置検出パターンの境界線（大きいマークの境界）の色

(Default は Color.White)

【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRColorBigMarkBorder を変更すると自動的にその設定した値に変更されます。

【プロパティ名】 QRColorBigMarkBorder2

右下の位置検出パターンの境界線（大きいマークの境界）の色

(Default は Color.White)

【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRColorBigMarkBorder を変更すると自動的にその設定した値に変更されます。

【プロパティ名】 QRColorBigMarkBorder3

左上の位置検出パターンの境界線（大きいマークの境界）の色

(Default は Color.White)

【注意】

この値は BackGroundColor を変更すると自動的にその設定した値に変更されます。

また、この値は QRColorBigMarkBorder を変更すると自動的にその設定した値に変更されます。

【プロパティ名】 QRColorTimingPattern

タイミング・パターン（左と上の点線）の色

(Default は Color.Black)

【注意】

この値は BarColor を変更すると自動的にその設定した値に変更されます。

【プロパティ名】 QRBackColorTimingPattern

タイミング・パターン（左と上の点線）の背景色

(Default は Color.White)

【注意】

この値は `BackColor` を変更すると自動的にその設定した値に変更されます。

**【プロパティ名】 QRColorVersionInformation**

型番情報の色

(Default は `Color.Black`)

**【注意】**

この値は `BarColor` を変更すると自動的にその設定した値に変更されます。

**【プロパティ名】 QRBackColorVersionInformation**

型番情報の背景色

(Default は `Color.White`)

**【注意】**

この値は `BackColor` を変更すると自動的にその設定した値に変更されます。

**【プロパティ名】 QRColorFormatInformation**

形式情報の色

(Default は `Color.Black`)

**【注意】**

この値は `BarColor` を変更すると自動的にその設定した値に変更されます。

**【プロパティ名】 QRBackColorFormatInformation**

形式情報の背景色

(Default は `Color.White`)

**【注意】**

この値は `BackColor` を変更すると自動的にその設定した値に変更されます。

**【プロパティ名】 QRColorQuietZone**

クワイエット・ゾーン (余白) の色

(Default は `Color.White`)

**【注意】**

この値は `BackColor` を変更すると自動的にその設定した値に変更されます。

## 5. **【重要】DotNetBarcodeV2.0.0 から V2.2.0 以降への重要な変更点**

DotNetBarcodeV2.0.0 までのルート名前空間は `DotNetBarcode` でした。しかし、オブジェクト名も同じ `DotNetBarcode` になっているため、VB の `Imports` や C# の `using` で `DotNetBarcode` とルート名前空間を指定してもオブジェクトへのアクセスは `DotNetBarcode.DotNetBarcode` とすべてに指定する必要があり、使いやすくないという意見がありました。

DotNetBarcodeV2.2.0 以降はルート名前空間を `System` に変更しました。そのため、VB の `Imports` や C# の `using` で `System` を指定すると(通常はすでに入っている場合もあります) `DotNetBarcode` だけでオブジェクトへアクセスできるようになりました。

しかし、既に既存のプログラムで使用されている場合は、プログラムを下記のように変更願います。

**【DotNetBarcode V2.0.0 以前】**

**【Visual Basic の場合】**

```
Dim bc1 As DotNetBarcode.DotNetBarcode = New DotNetBarcode.DotNetBarcode  
bc1.Type = DotNetBarcode.DotNetBarcode.Types.Jan13
```

**【C#の場合】**

```
DotNetBarcode.DotNetBarcode bc1 =new DotNetBarcode.DotNetBarcode();  
bc1.Type = DotNetBarcode.DotNetBarcode.Types.Jan13
```

**【DotNetBarcode V2.2.0 以降(V2.3.0 を含む)】**

**【Visual Basic の場合】**

```
Imports System  
Dim bc1 As DotNetBarcode = New DotNetBarcode  
bc1.Type = DotNetBarcode.Types.Jan13
```

**【C#の場合】**

```
using System;  
DotNetBarcode bc1 =new DotNetBarcode();  
bc1.Type = DotNetBarcode.Types.Jan13
```