

XImage.Barcode Reader Developer Guide

Table of Contents

XImage.Barcode Reader Developer Guide.....	1
Scan barcodes from a file.....	2
Scan from a JPEG file.....	2
Scan from a TIFF file.....	3
Scan from a PDF file	4
Scan with options.....	5
Read the specified page in a document.....	5
Scan a specified region in a page	6
Scan different type of barcodes.....	7

Scan barcodes from a file

Scan from a JPEG file

```
C#
String inputFilePath = Program.ROOT_PATH + @"\1.jpg";

Bitmap bitmap = new Bitmap(inputFilePath);

// prepare reader setting
ReaderSettings settings = new ReaderSettings();
// add expected barcode type to search
settings.AddTypesToRead(BarcodeType.Code93);

// read barcodes from a Bitmap object
Barcode[] result = BarcodeReader.ReadBarcodes(settings, bitmap);
if (result.Length > 0)
{
    foreach (Barcode barcode in result)
    { // show data (in String) for each barcode
        Console.WriteLine("Message: " + barcode.DataString);
    }
}
else
{
    Console.WriteLine("No barcode has been found!");
}
```

Scan from a TIFF file

C#

```
String inputFilePath = Program.ROOT_PATH + @"\1.tif";

TIFFDocument doc = new TIFFDocument(inputFilePath);

// prepare reader setting
ReaderSettings settings = new ReaderSettings();
// add expected barcode type to search
settings.AddTypesToRead(BarcodeType.Code39);

// read barcodes from a document
Barcode[] result = BarcodeReader.ReadBarcodes(settings, doc);
if (result.Length > 0)
{
    foreach (Barcode barcode in result)
    { // show data (in String) for each barcode
        Console.WriteLine("Message: " + barcode.DataString);
    }
}
else
{
    Console.WriteLine("No barcode has been found!");
}
```

Remark

Must to add reference “RasterEdge.XDoc.TIFF.dll” for this feature.

Scan from a PDF file

C#

```
String inputFilePath = Program.ROOT_PATH + @"\1.pdf";

PDFDocument doc = new PDFDocument(inputFilePath);

// prepare reader setting
ReaderSettings settings = new ReaderSettings();
// add expected barcode type to search
settings.AddTypesToRead(BarcodeType.Code128);

// read barcodes from a document
Barcode[] result = BarcodeReader.ReadBarcodes(settings, doc);
if (result.Length > 0)
{
    foreach (Barcode barcode in result)
    { // show data (in String) for each barcode
        Console.WriteLine("Message: " + barcode.DataString);
    }
}
else
{
    Console.WriteLine("No barcode has been found!");
}
```

Remark

Must to add reference “RasterEdge.XDoc.PDF.dll” for this feature.

Scan with options

Read the specified page in a document

```
C#
String inputFilePath = Program.ROOT_PATH + @"\1.pdf";

// get the 2nd page in the document
PDFDocument doc = new PDFDocument(inputFilePath);
BasePage page = doc.GetPage(1);

// prepare reader setting
ReaderSettings settings = new ReaderSettings();
// add expected barcode type to search
settings.AddTypesToRead(BarcodeType.Code128);

// read barcodes from a page
Barcode[] result = BarcodeReader.ReadBarcodes(settings, page);
if (result.Length > 0)
{
    foreach (Barcode barcode in result)
    { // show data (in String) for each barcode
        Console.WriteLine("Message: " + barcode.DataString);
    }
}
else
{
    Console.WriteLine("No barcode has been found!");
}
```

Scan a specified region in a page

C#

```
String inputFilePath = Program.ROOT_PATH + @"\1.pdf";

// get the 2nd page in the document
PDFDocument doc = new PDFDocument(inputFilePath);
BasePage page = doc.GetPage(1);

// prepare reader setting
ReaderSettings settings = new ReaderSettings();
// add expected barcode type to search
settings.AddTypesToRead(BarcodeType.Code128);

// set scan region in the page
// 10% to 50% of page width in horizontal
// and 5% to 20% of page height in vertical
settings.RegionOfInterest = new Rectangle(10, 5, 50, 20);

// read barcodes from a page
Barcode[] result = BarcodeReader.ReadBarcodes(settings, page);
```

Scan different type of barcodes

C#

```
String inputFilePath = Program.ROOT_PATH + @"\1.pdf";

// get the 2nd page in the document
PDFDocument doc = new PDFDocument(inputFilePath);

// prepare reader setting
ReaderSettings settings = new ReaderSettings();
// add expected barcode type to search
settings.AddTypesToRead(BarcodeType.Code128);
settings.AddTypesToRead(BarcodeType.PDF417);

// read barcodes from a document
Barcode[] result = BarcodeReader.ReadBarcodes(settings, doc);
if (result.Length > 0)
{
    foreach (Barcode barcode in result)
    {
        Console.WriteLine("Type: " + barcode.Type.ToString());
        Console.WriteLine("Message: " + barcode.DataString);
    }
}
else
{
    Console.WriteLine("No barcode has been found!");
}
```