

Table of Contents

Foreword	0
Part I Introduction	3
Part II How To Use Encoders	3
1 Win32 DLL Encoder.....	3
2 .NET Class Library Encoder.....	4
3 Reference Guide.....	4
Win32 DLL Encoder Functions	4
PDF417Encode Function.....	4
PDF417GetCharAt Function.....	5
PDF417GetCols Function.....	5
PDF417GetRows Function.....	5
.NET Class Library Encoder Methods	6
Encode Method.....	6
GetCols Method.....	7
GetRows Method.....	7
GetRowStringAt Method.....	7
Part III Crystal Reports	8
1 How To Use It.....	8
2 UFL Functions.....	13
PDF417FontEncode Function	13
PDF417FontGetBlock Function	14
3 Legacy UFL.....	14
How To Use It	14
How To Distribute It	19
UFL Functions	19
PDFUFLMW6Encoder Function.....	19
PDFUFLMW6GetBlock Function.....	20
Part IV Office 2007 & 2010	20
1 Word.....	20
Install Template File	20
Create Single Barcode	22
Create Multiple Barcodes	23
Mail Merge	24
2 Access.....	28
Part V Office 2000 & 2003	32
1 Word Demo.....	32
Install Template File	32
Create Single Barcode	33
Create Multiple Barcodes	34
Mail Merge	36

2	Access Demo.....	38
Part VI	License	40
	Index	0

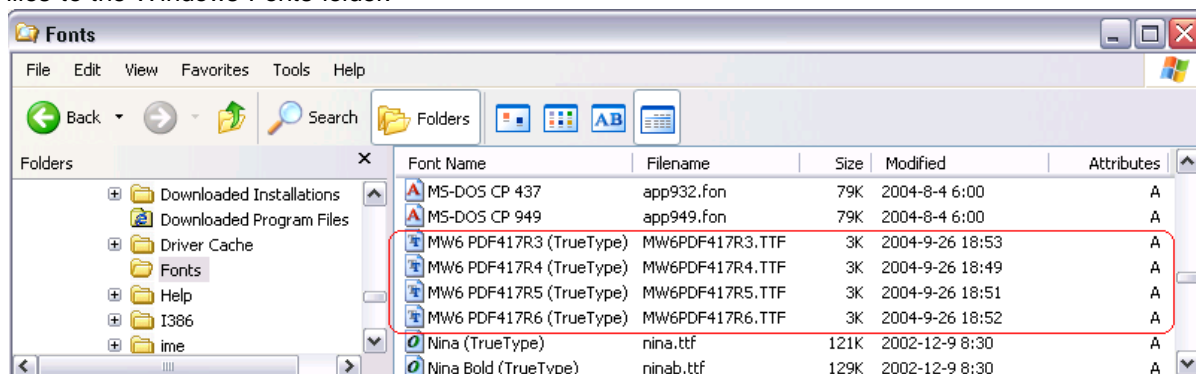
1 Introduction

MW6 PDF417 font package can print PDF417 barcode as a font, Win32 DLL encoder and .NET class library encoder are provided, the encoders are different for the trial version package and the full version package, the trial version encoders append "MW6 Demo" to the string encoded with PDF417.

There are 4 different TrueType fonts:

Font name	X to Y ratio	Height of each row at 12 points
MW6 PDF417R3	1:3	1/10 inch
MW6 PDF417R4	1:4	2/15 inch
MW6 PDF417R5	1:5	1/6 inch
MW6 PDF417R6	1:6	1/5 inch

The above font files are same for the trial version package and the full version package, copy these 4 .tff files to the Windows Fonts folder.



2 How To Use Encoders

2.1 Win32 DLL Encoder

If you build PDF417 font applications using Word, Access, Crystal Reports, VB, VC++, Delphi, Borland C++, FoxPro and PowerBuilder, Win32 DLL encoder is required to convert a regular string to PDF417 font format string.

1. For 32-bit OS such as Windows XP and NT, copy "PDF417Font.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
2. For 32-bit version of Windows Vista and above, copy "PDF417Font.dll" to the folder "C:\windows\system32".
3. For 64-bit version of Windows Vista and above, copy "PDF417Font.dll" to the folder "C:\windows\SysWow64".
4. If you want to generate PDF417 barcodes inside 64-bit Office Word, Excel, or Access, copy "PDF417Font_x64.dll" to the folder "c:\windows\system32".

2.2 .NET Class Library Encoder

If you build PDF417 font applications using VB.NET or C#, .NET class library encoder is required to convert a regular string to PDF417 font format string, copy "PDF417FontNet.dll" to your application folder.

2.3 Reference Guide

2.3.1 Win32 DLL Encoder Functions

2.3.1.1 PDF417Encode Function

Encodes a string using PDF417 format.

```
void PDF417Encode(
    LPCTSTR Message,
    WORD Mode,
    WORD ErrorCorrectionLevel,
    WORD Rows,
    WORD Columns,
    BOOL TruncateSymbol,
    BOOL HandleTilde);
```

Parameters

Message

String to be encoded using PDF417 format.

Mode

Indicates which encoding mode is used, this parameter can be one of the following values.

Value	Comment
0	Binary encoding mode
1	Text encoding mode
2	Auto mode for optimized encoding

ErrorCorrectionLevel

Indicates which error correction level is used, the valid value should be between 0 and 8, the value of 2 is recommended.

Rows

Indicates how many rows are used to encode string, the valid value should be between 3 and 90.

Columns

Indicates how many columns are used to encode string, the valid value should be between 3 and 30.

TruncateSymbol

Indicates whether to truncate symbol or not, the recommended value is FALSE.

HandleTilde

Indicates whether to process the tilde character "~" or not, if this parameter is set to TRUE, non-printable characters can be passed to the encoder by using the tilde character, "~dNNN" represents the ASCII character encoded by the 3 digits NNN, for example, "~d010" represents the character LF (line feed).

2.3.1.2 PDF417GetCharAt Function

Retrieves the ASCII value for a character element in PDF417 font data matrix.

```
WORD PDF417GetCharAt(WORD RowIndex, WORD ColIndex);
```

Parameters

RowIndex

This parameter is a 0-based index and a valid value must be between 0 and total number of rows - 1.

ColIndex

This parameter is a 0-based index and a valid value must be between 0 and total number of columns - 1.

Return Value

The return value is the ASCII value of a character element in PDF417 font data matrix.

2.3.1.3 PDF417GetCols Function

Returns the number of columns in PDF417 font data matrix.

```
WORD PDF417GetCols();
```

Return Value

The return value is the number of columns in PDF417 font data matrix.

2.3.1.4 PDF417GetRows Function

Returns the number of rows in PDF417 font data matrix.

```
WORD PDF417GetRows();
```

Return Value

The return value is the number of rows in PDF417 font data matrix.

2.3.2 .NET Class Library Encoder Methods

2.3.2.1 Encode Method

Encodes a string using PDF417 format.

[Visual Basic .NET]

```
Public Sub Encode(ByVal Message As String, _
                 ByVal Mode As Integer, _
                 ByVal ErrorCorrectionLevel As Integer, _
                 ByVal Rows As Integer, _
                 ByVal Columns As Integer, _
                 ByVal TruncateSymbol As Boolean, _
                 ByVal HandleTilde As Boolean)
```

[C#]

```
public void Encode(string Message,
                  int Mode,
                  int ErrorCorrectionLevel,
                  int Rows,
                  int Columns,
                  bool TruncateSymbol,
                  bool HandleTilde);
```

Parameters

Message

String to be encoded using PDF417 format.

Mode

Indicates which encoding mode is used, this parameter can be one of the following values.

Value	Comment
0	Binary encoding mode
1	Text encoding mode
2	Auto mode for optimized encoding

ErrorCorrectionLevel

Indicates which error correction level is used, the valid value should be between 0 and 8, the value of 2 is recommended.

Rows

Indicates how many rows are used to encode string, the valid value should be between 3 and 90.

Columns

Indicates how many columns are used to encode string, the valid value should be between 3 and 30.

TruncateSymbol

Indicates whether to truncate symbol or not, the recommended value is FALSE.

HandleTilde

Indicates whether to process the tilde character "~" or not, if this parameter is set to TRUE, non-printable characters can be passed to the encoder by using the tilde character, "~dNNN" represents the ASCII character encoded by the 3 digits NNN, for example, "~d010" represents the character LF (line feed).

2.3.2.2 GetCols Method

Returns the number of columns in PDF417 font data matrix.

```
[Visual Basic .NET]
```

```
Public Function GetCols() As Integer
```

```
[C#]
```

```
public int GetCols();
```

Return Value

The return value is the number of columns in PDF417 font data matrix.

2.3.2.3 GetRows Method

Returns the number of rows in PDF417 font data matrix.

```
[Visual Basic .NET]
```

```
Public Function GetRows() As Integer
```

```
[C#]
```

```
public int GetRows();
```

Return Value

The return value is the number of rows in PDF417 font data matrix.

2.3.2.4 GetRowStringAt Method

Concatenates characters for a row in PDF417 font data matrix to create a string and return it.

```
[Visual Basic .NET]
```

```
Public Function GetRowStringAt(ByVal RowIndex As Integer) As String
```

```
[C#]
```

```
public string GetRowStringAt(int RowIndex);
```

Parameters

RowIndex

This parameter is a 0-based index and a valid value must be between 0 and total number of rows - 1.

Return Value

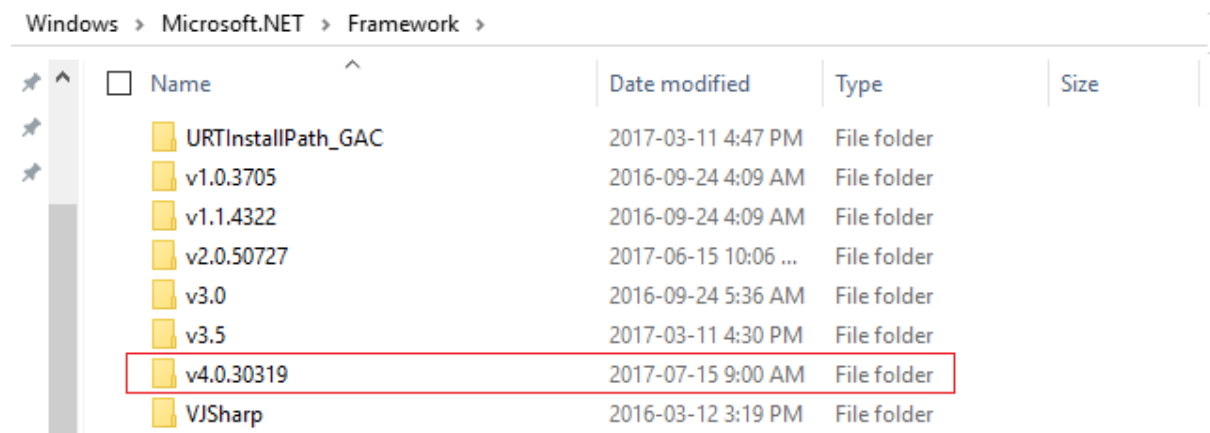
The return value is a row string for PDF417 font data matrix.

3 Crystal Reports

3.1 How To Use It

1. For 32-bit version of Crystal Reports, unzip UFL_32.ZIP and put CRUFLPDF417.dll in SysWow64 folder (64-bit version of Windows OS) or system32 folder (32-bit version of Windows OS). Run the following command as "**Run as administrator**", and the **ver_num** is the version number for 32-bit version of .NET framework.

```
C:\Windows\Microsoft.NET\Framework\ver_num\RegAsm.exe /codebase C:\Windows  
\SysWow64\CRUFLPDF417.dll
```



2. For 64-bit version of Crystal Reports, unzip UFL_64.ZIP and put CRUFLPDF417.dll in system32 folder. Run the following command as "**Run as administrator**", and the **ver_num** is the version number for 64-bit version of .NET framework.

```
C:\Windows\Microsoft.NET\Framework64\ver_num\RegAsm.exe /codebase C:\Windows  
\System32\CRUFLPDF417.dll
```

Windows > Microsoft.NET > Framework64 >

Name	Date modified	Type	Size
v2.0.50727	2017-06-15 10:06 ...	File folder	
v3.0	2016-09-24 5:36 AM	File folder	
v3.5	2016-09-24 5:36 AM	File folder	
v4.0.30319	2017-07-15 9:00 AM	File folder	
sbscmp10.dll	2016-07-16 5:43 AM	Application extens...	9 KB
sbscmp20_mscorwks.dll	2016-07-16 5:43 AM	Application extens...	9 KB
sbscmp20_perfcounter.dll	2016-07-16 5:43 AM	Application extens...	9 KB
SharedReg12.dll	2016-07-16 5:43 AM	Application extens...	9 KB

3. Open up Crystal Reports, go to "**Field Explorer**", right click on "**Formula Fields**", click on "**New**", enter "**PDF417 Barcode**", copy the following code into the Formula Editor area. Please check the sub-node "COM and .NET UFLs (u212com.dll)" or "Visual Basic (u21com.dll)" under the node "Additional Functions" if you have difficulty locating PDF417 font related functions.

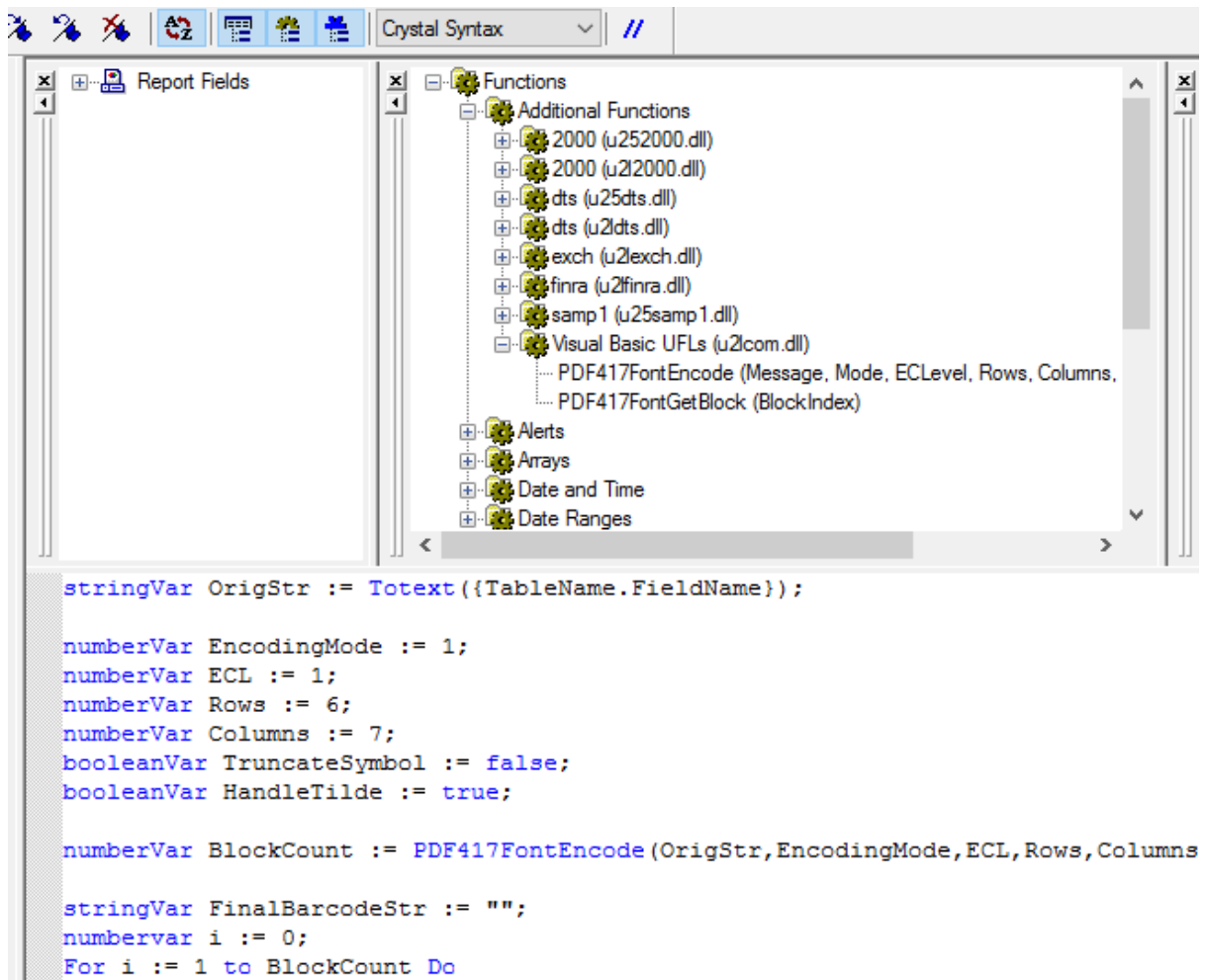
```
stringVar OrigStr := Totext({TableName.FieldName});
numberVar EncodingMode := 1;
numberVar ECL := 1;
numberVar Rows := 6;
numberVar Columns := 7;
booleanVar TruncateSymbol := false;
booleanVar HandleTilde := true;

numberVar BlockCount := PDF417FontEncode
(OrigStr,EncodingMode,ECL,Rows,Columns,TruncateSymbol,HandleTilde);

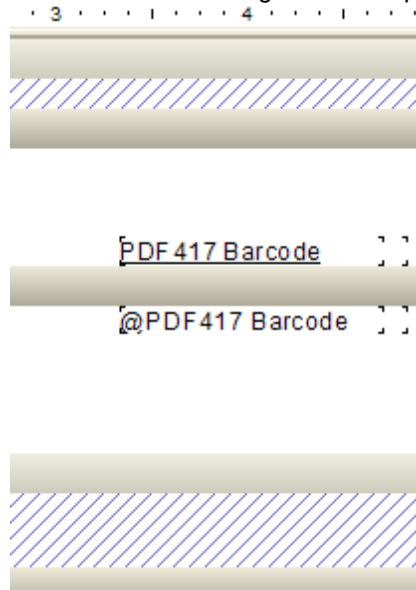
stringVar FinalBarcodeStr := "";
numberVar i := 0;
For i := 1 to BlockCount Do
(
    FinalBarcodeStr := FinalBarcodeStr + PDF417FontGetBlock (i - 1);
);

FinalBarcodeStr
```

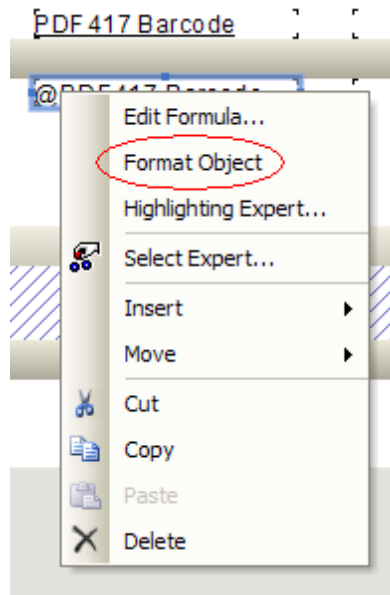
4. Change a few values to meet your application requirements, click "**Save**" and close this window.



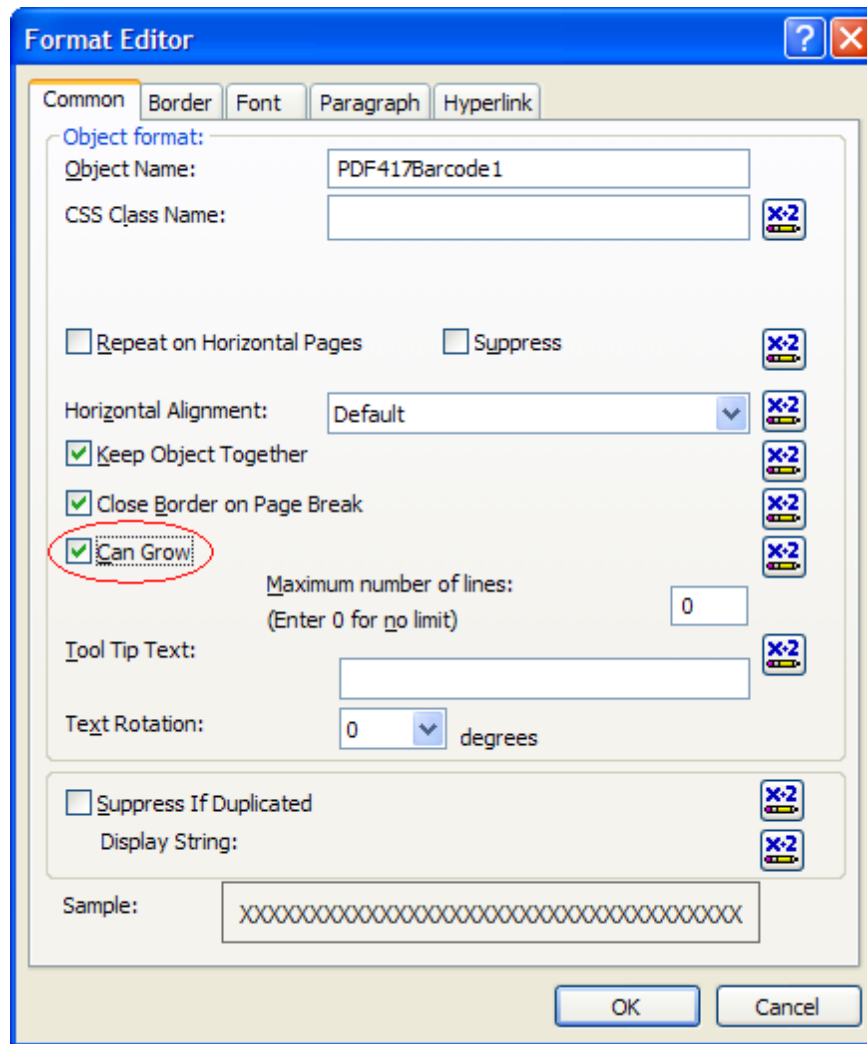
5. Click on the formula field "PDF417 Barcode" and drag it on the report.



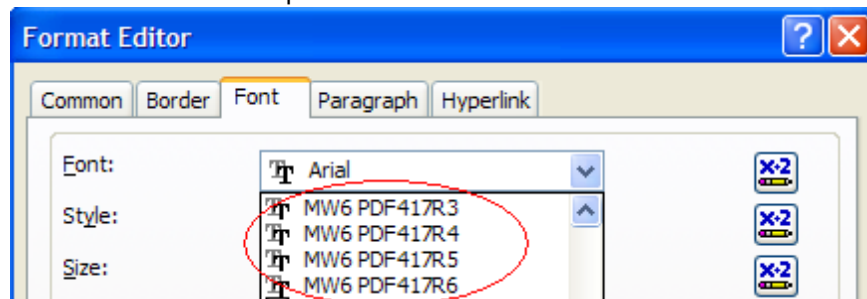
6. Right-click "**@PDF417 Barcode**" and choose "**Format Object**".



7. Toggle on "**Can Grow**" check box under "**Common**" tab.

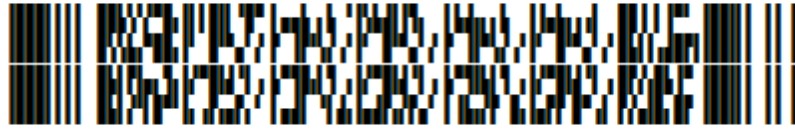


8. Choose one PDF417 font out of 4 options as the font name under "Font" tab.



9. Run the report

PDF 417 Barcode



3.2 UFL Functions

3.2.1 PDF417FontEncode Function

Encodes a string with PDF417 format.

```
Public Function PDF417FontEncode(ByVal Message As String, _  
    ByVal Mode As Integer, _  
    ByVal ErrorCorrectionLevel As Integer, _  
    ByVal Rows As Integer, _  
    ByVal Columns As Integer, _  
    ByVal TruncateSymbol As Boolean, _  
    ByVal HandleTilde As Boolean) As Integer
```

Parameters

Message

String to be encoded using PDF417 format.

Mode

Indicates which encoding mode is used, this parameter can be one of the following values.

Value	Comment
0	Binary encoding mode
1	Text encoding mode
2	Auto mode for optimized encoding

ErrorCorrectionLevel

Indicates which error correction level is used, the valid value should be between 0 and 8, the value of 2 is recommended.

Rows

Indicates how many rows are used to encode string, the valid value should be between 3 and 90.

Columns

Indicates how many columns are used to encode string, the valid value should be between 3 and 30.

TruncateSymbol

Indicates whether to truncate symbol or not, the recommended value is FALSE.

HandleTilde

Indicates whether to process the tilde character "~" or not, if this parameter is set to TRUE, non-printable characters can be passed to the encoder by using the tilde character, "~dNNN" represents the ASCII character encoded by the 3 digits NNN, for example, "~d010" represents the character LF (line feed).

Return Value

Number of PDF417 format string blocks. Each block has 254 characters, the only exception is that last block might contain <254 characters. Since Crystal Reports UFL function only allows the returned string with maximum 254 characters, we have to build entire PDF417 format string by concatenating all blocks together.

3.2.2 PDF417FontGetBlock Function

Retrieves a block data of PDF417 format string.

```
Public Function PDF417FontGetBlock(ByVal BlockIndex As Integer) As String
```

Parameters

BlockIndex

This parameter is a 0-based index and a valid value must be between 0 and total number of blocks minus 1.

Return Value

A block data of PDF417 format string.

3.3 Legacy UFL

Since our Crystal Reports' UFL DLLs are based on the latest .NET technologies, they might not work properly for some earlier versions of Windows and/or Crystal Reports (e.g. XP and Crystal Reports 9.0), please use this **Legacy UFL** instead.

3.3.1 How To Use It

1. The old versions (prior to V9) of Crystal Reports have the limitation for the string length (< **256 characters**), the MW6 PDF417 UFL encoder function can easily produce a string with more than 255 characters, so please upgrade your Crystal Reports to version 9 in order to add powerful PDF417 barcode into your reports.
 2. Go to the folder where u2lcom.dll is located and copy CRUFLPDF.dll there, and this folder varies depending on your version of Crystal Reports. If you are running a 64 bit version of Windows OS such
-

as Windows Vista 64 bit or Windows 7 64 bit, you may need to look in "*C:\Program Files (x86)*" rather than "*C:\Program Files*" folder.

Version	Folder
Crystal Reports 14 (CR2011)	C:\Program Files\Common Files\Business Objects\3.0\bin
Crystal Reports 12 (CR2008)	C:\Program Files\Common Files\Business Objects\3.0\bin or C:\Program Files\Business Objects\BusinessObjects Enterprise 12.0 \win32_x86
Crystal Reports 11 R2 (XI R2)	C:\Program Files\Business Objects\common\3.5\bin
Crystal Reports 11 (XI)	C:\Program Files\Common Files\Business Objects\3.0\bin
Crystal Reports.Net 10.2	C:\Program Files\Common Files\Business Objects\2.7\bin
Crystal Reports 10	C:\Program Files\Common Files\Crystal Decisions\2.5\bin
Crystal Reports 9	C:\Program Files\Common Files\Crystal Decisions\2.0\bin
Crystal Reports for Visual Studio 2003	C:\Program Files\Common Files\Crystal Decisions\1.1\bin
Crystal Reports.Net 1.0	C:\Program Files\Common Files\Crystal Decisions\1.0\bin

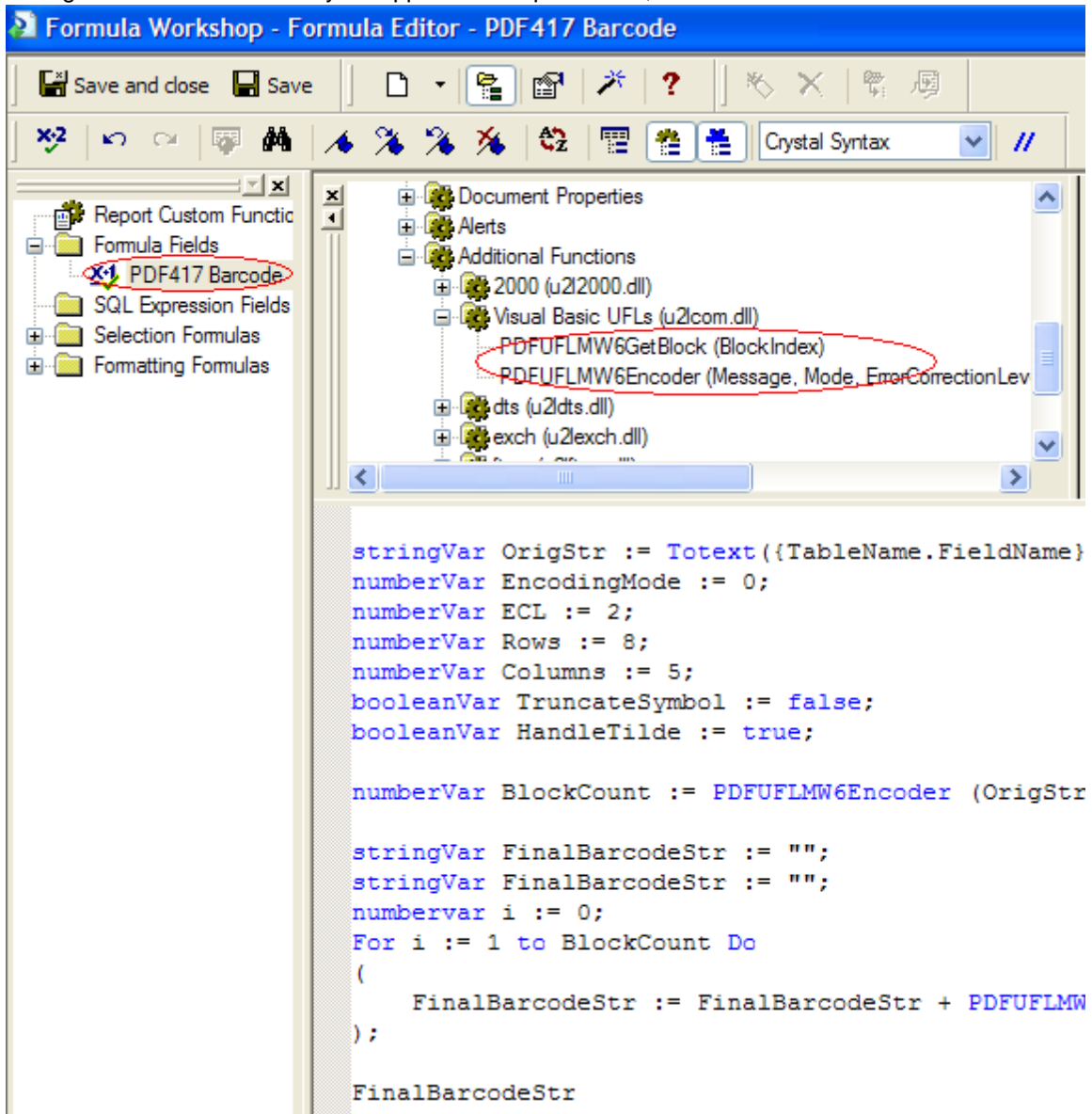
- For Windows Vista or above, you need to use an elevated Command Prompt to run *regsvr32.exe* command, click **"Start"** > **"All Programs"** > **"Accessories"**, right-click **"Command Prompt"**, and then click **"Run"** as administrator.
- For 32-bit version Windows OS, run "regsvr32 CRUFLPDF.dll" to register it. Copy "PDF417Font.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32") and move to the step 6.
- For 64-bit version Windows OS, run "C:\windows\SysWOW64\regsvr32 CRUFLPDF.dll" to register it. Copy "PDF417.dll" to the windows 32-bit system folder, which is "C:\windows\SysWOW64".
- Open up Crystal Reports, go to **"Field Explorer"**, right click on **"Formula Fields"**, click on **"New"**, enter **"PDF417 Barcode"**, copy the following code into the Formula Editor area.

```
stringVar OrigStr := Totext({TableName.FieldName});
numberVar EncodingMode := 1;
numberVar ECL := 1;
numberVar Rows := 6;
numberVar Columns := 7;
booleanVar TruncateSymbol := false;
booleanVar HandleTilde := true;

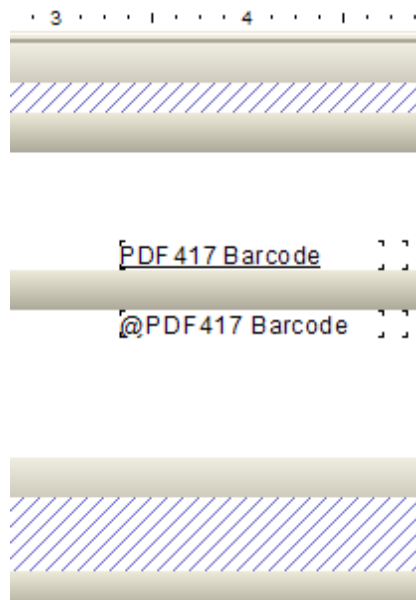
numberVar BlockCount := PDFUFLMW6Encoder
(OrigStr,EncodingMode,ECL,Rows,Columns,TruncateSymbol,HandleTilde);

stringVar FinalBarcodeStr := "";
stringVar FinalBarcodeStr := "";
numberVar i := 0;
For i := 1 to BlockCount Do
(
  FinalBarcodeStr := FinalBarcodeStr + PDFUFLMW6GetBlock (i - 1);
);
FinalBarcodeStr
```

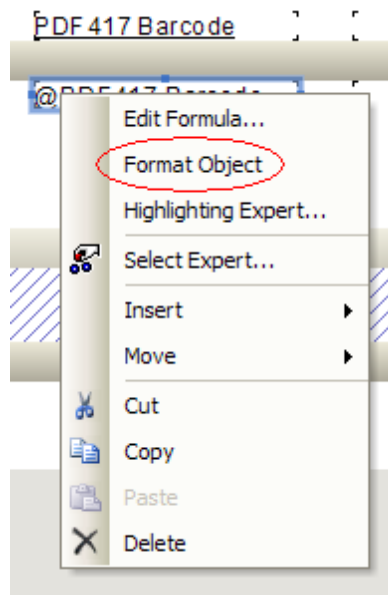
7. Change a few values to meet your application requirements, click "Save" and close this window.



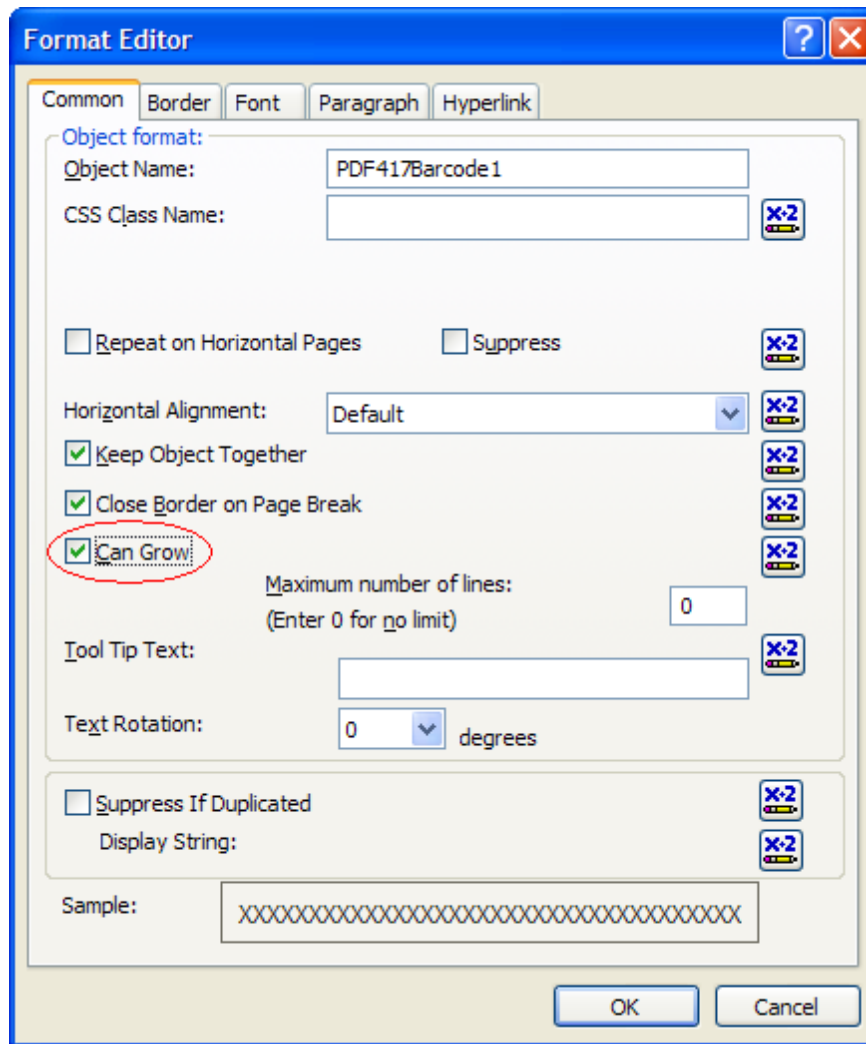
8. Click on the formula field "PDF417 Barcode" and drag it on the report.



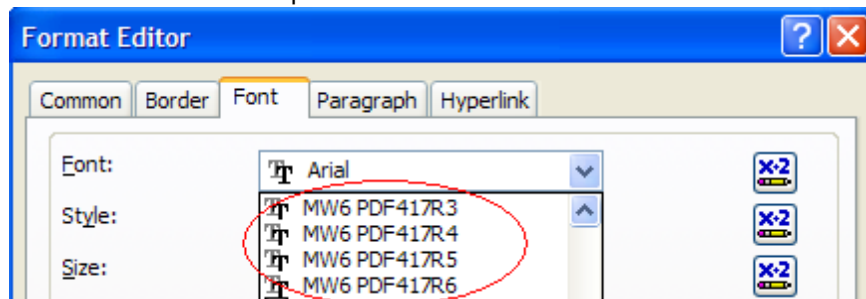
9. Right-click "**@PDF417 Barcode**" and choose "**Format Object**".



10. Toggle on "**Can Grow**" check box under "**Common**" tab.



11. Choose one PDF417 font out of 4 options as the font name under "Font" tab.



12. Run the report

PDF 417 Barcode



3.3.2 How To Distribute It

For the distribution purpose, you need to distribute appropriate MW6 PDF417 font ttf file(s), Win32 DLL Encoder (PDF417Font.dll), Crystal Reports UFL (CRUFLPDF.dll), Crystal Reports Runtime (u2lcom.dll) and VB Runtime DLL (msvbvm60. dll), VB Runtime DLL already exists on most PCs and it can be found in the system folder.

3.3.3 UFL Functions

3.3.3.1 PDFUFLMW6Encoder Function

Encodes a string using PDF417 format.

```
Public Function PDFUFLMW6Encoder(ByVal Message As String, _  
    ByVal Mode As Integer, _  
    ByVal ErrorCorrectionLevel As Integer, _  
    ByVal Rows As Integer, _  
    ByVal Columns As Integer, _  
    ByVal TruncateSymbol As Boolean, _  
    ByVal HandleTilde As Boolean) As Integer
```

Parameters

Message

String to be encoded using PDF417 format.

Mode

Indicates which encoding mode is used, this parameter can be one of the following values.

Value	Comment
0	Binary encoding mode
1	Text encoding mode
2	Auto mode for optimized encoding

ErrorCorrectionLevel

Indicates which error correction level is used, the valid value should be between 0 and 8, the value of 2 is recommended.

Rows

Indicates how many rows are used to encode string, the valid value should be between 3 and 90.

Columns

Indicates how many columns are used to encode string, the valid value should be between 3 and 30.

TruncateSymbol

Indicates whether to truncate symbol or not, the recommended value is FALSE.

HandleTilde

Indicates whether to process the tilde character "~" or not, if this parameter is set to TRUE, non-printable characters can be passed to the encoder by using the tilde character, "~dNNN" represents the ASCII character encoded by the 3 digits NNN, for example, "~d010" represents the character LF (line feed).

Return Value

Number of PDF417 format string blocks. Each block has 254 characters, the only exception is that last block might contain <254 characters. Since Crystal Reports UFL function only allows the returned string with maximum 254 characters, we have to build entire PDF417 format string by concatenating all blocks together.

3.3.3.2 PDFUFLMW6GetBlock Function

Retrieves a block data of PDF417 format string.

```
Public Function PDFUFLMW6GetBlock(ByVal BlockIndex As Integer) As String
```

Parameters

BlockIndex

This parameter is a 0-based index and a valid value must be between 0 and total number of blocks - 1.

Return Value

A block data of PDF417 format string.

4 Office 2007 & 2010

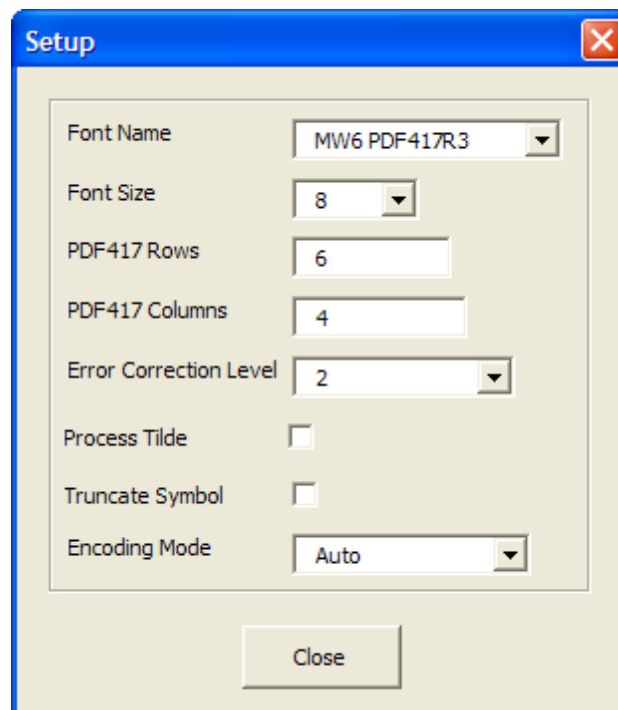
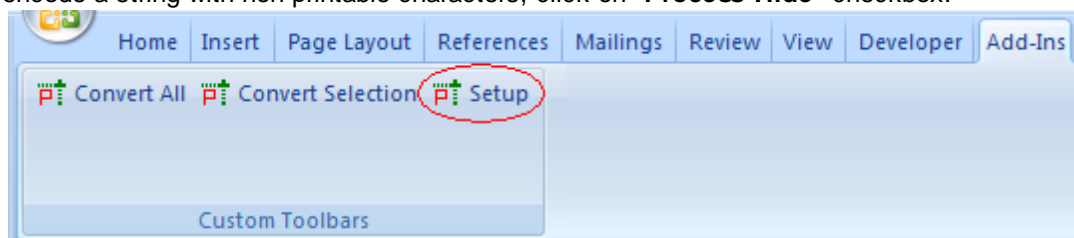
4.1 Word

4.1.1 Install Template File

1. Locate Microsoft Word Startup folder, which usually is "C:\Documents and Settings\\Application Data\Microsoft\Word\STARTUP" for Windows XP or "C:\Users\\AppData
-

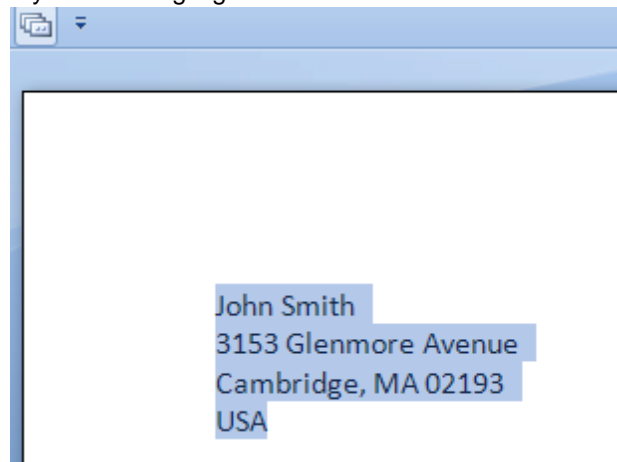
\Roaming\Microsoft\Word\STARTUP" for Windows Vista and above.

2. Copy MW6_PDF417_Font.dotm for 32-bit Office or MW6_PDF417_Font_x64.dotm for 64-bit Office to this folder.
3. For 32-bit Office, copy "PDF417Font.dll" to the windows 32-bit system folder (e.g., "C:\winnt\system32" or "C:\windows\system32") of 32-bit OS or the windows SysWow64 folder of 64-bit OS (e.g., "C:\windows\SysWow64").
4. For 64-bit Office, copy "PDF417Font_x64.dll" to the windows 32-bit system folder (e.g., "C:\windows\system32").
5. Click on "**Add-Ins**", then click on "**Setup**". Change the configurations for PDF417 format, if you want to encode a string with non-printable characters, click on "**Process Tilde**" checkbox.

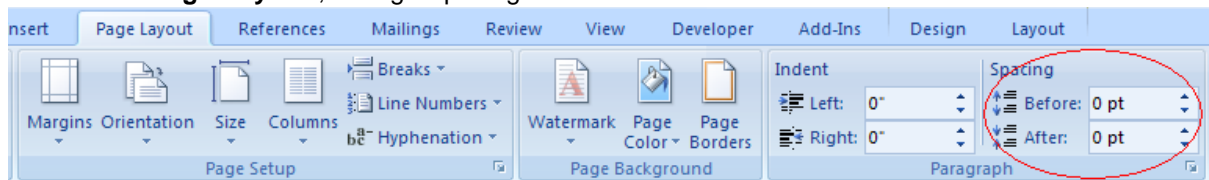


4.1.2 Create Single Barcode

1. Enter a few strings line by line and highlight them.



2. Click on "**Page Layout**", change Spacing "**Before**" and "**After**" to 0.

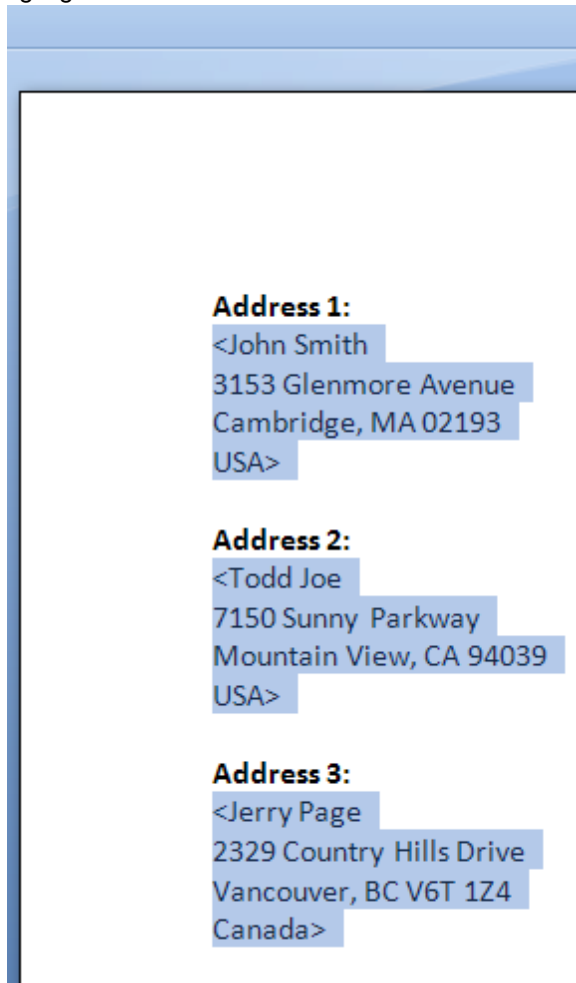


3. Click on "**Add-Ins**", then click on "**Convert Selection**" to create a PDF417 barcode.

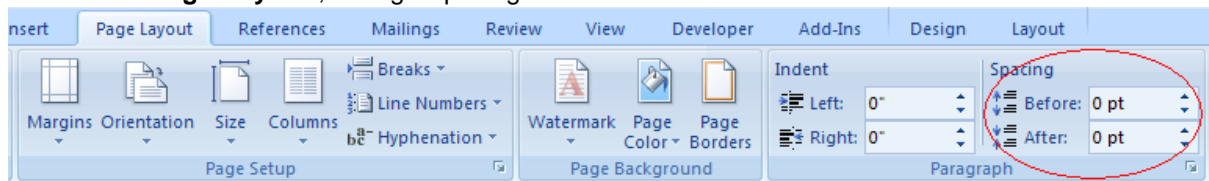


4.1.3 Create Multiple Barcodes

1. Enter a few string sections, surround those sections which will be converted to the barcodes with the "<" and ">" characters, highlight those sections.



2. Click on "**Page Layout**", change Spacing "**Before**" and "**After**" to 0.

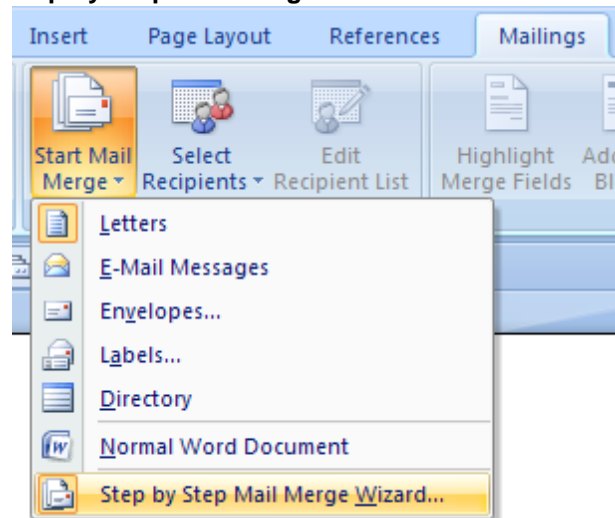


3. Click on "**Add-Ins**", then click on "**Convert All**" to create PDF417 barcodes for the string sections surrounded with the "<" and ">" characters.

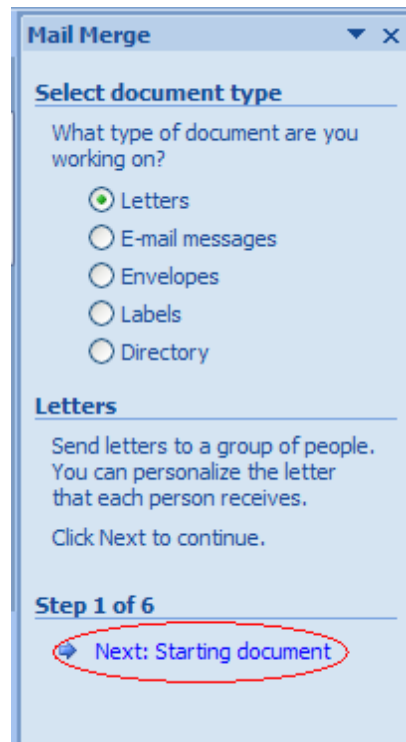


4.1.4 Mail Merge

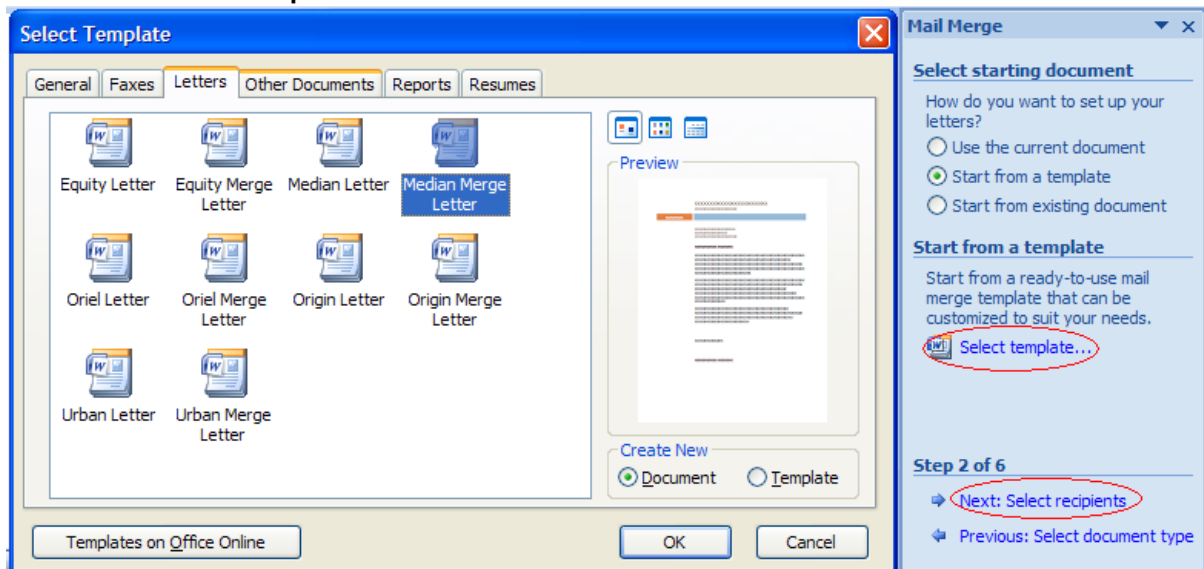
1. Click on "**Mailings**", then click on "**Start Mail Merge**". A drop-down list appears as shown below, select the last option "**Step by Step Mail Merge Wizard**".



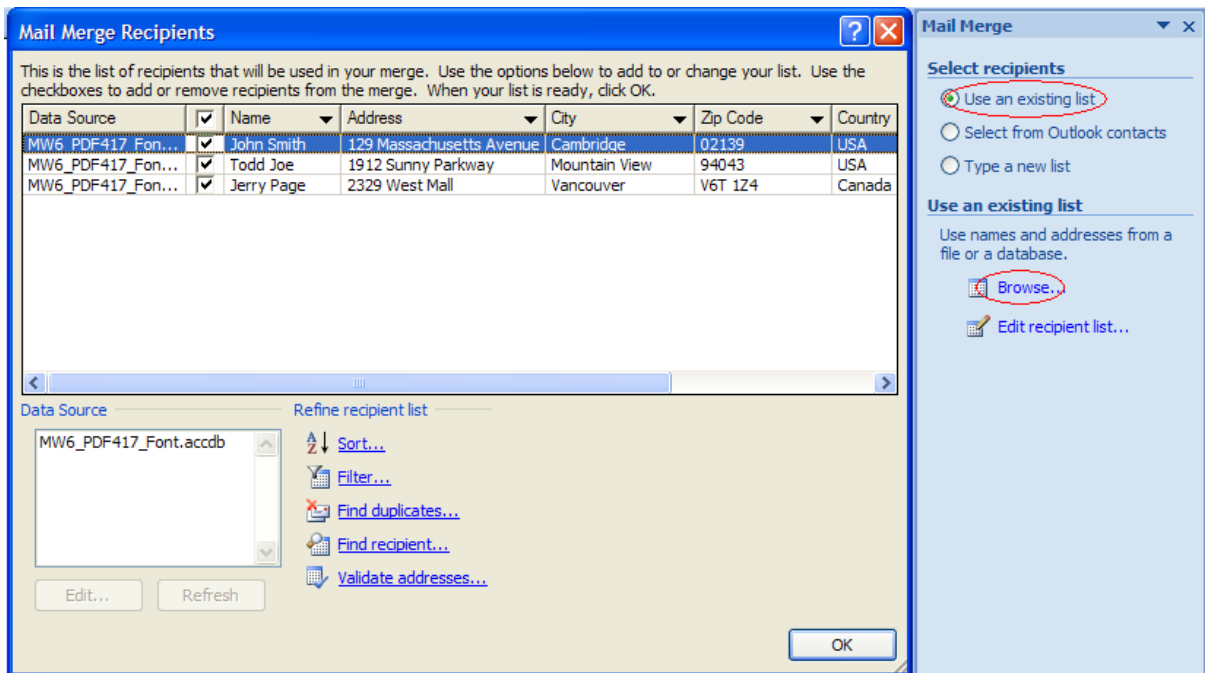
2. Select a document type and click on "**Next: Starting document**".



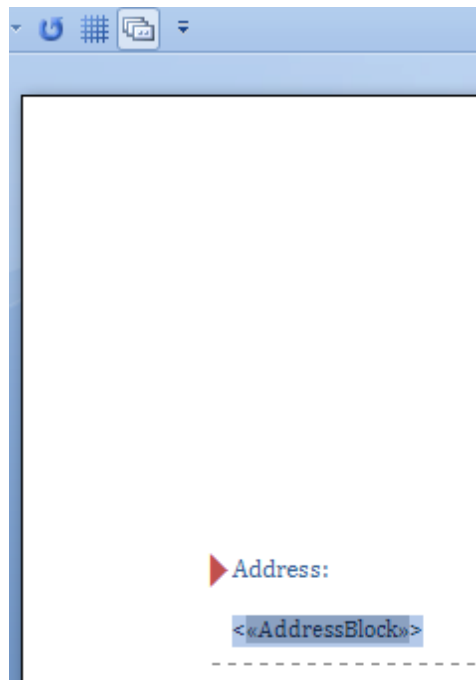
3. Click on "**Start from a template**", then click on the link "**Select template**", choose a template, click on "**Next: Select recipients**".



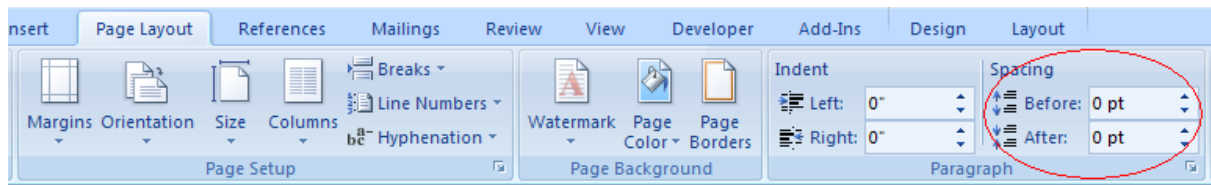
4. Select "**Use an existing list**" and click on "**Browser**" link, choose "MW6_PDF417_Font.accdb" database as an existing list, click "**Next: Write your letter**".



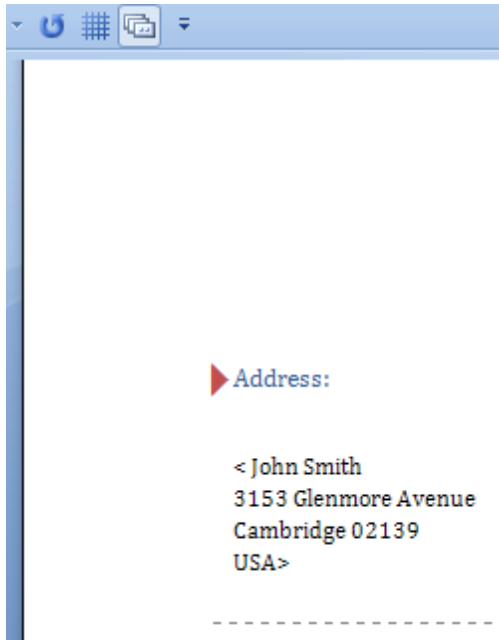
5. Surround the section which will be converted to PDF417 barcode with "<" and ">" characters and highlight it.



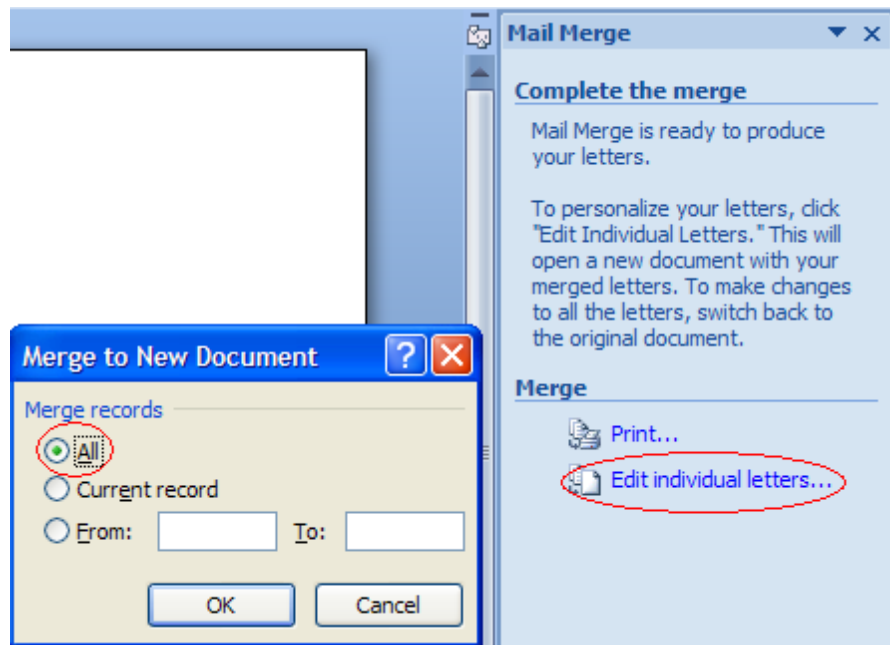
6. Click on "Page Layout", change Spacing "Before" and "After" to 0.



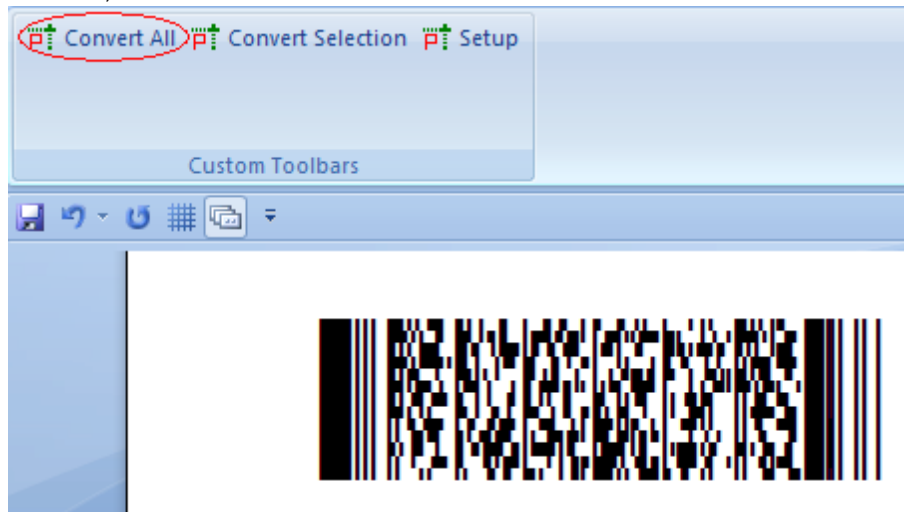
7. Click on "**Next: Preview your letters**", then click on "**Next: Complete the merge**".



8. Click on "**Edit individual letters**", this opens "**Merge to New Document**" dialog, click on "**All**" and then click on "**OK**" button.

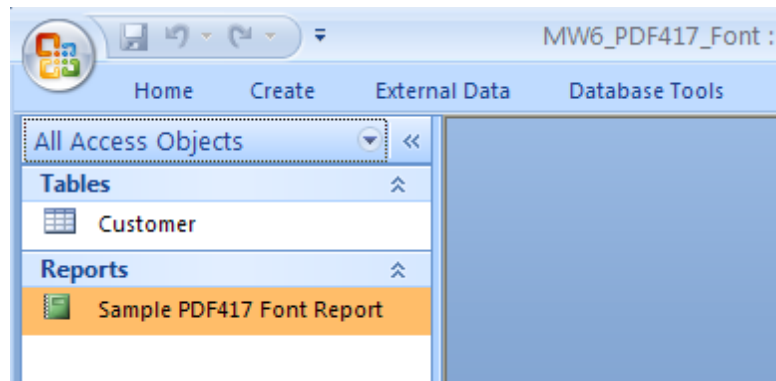


9. Click on "**Add-Ins**", then click on "**Convert All**" to create PDF417 barcodes.

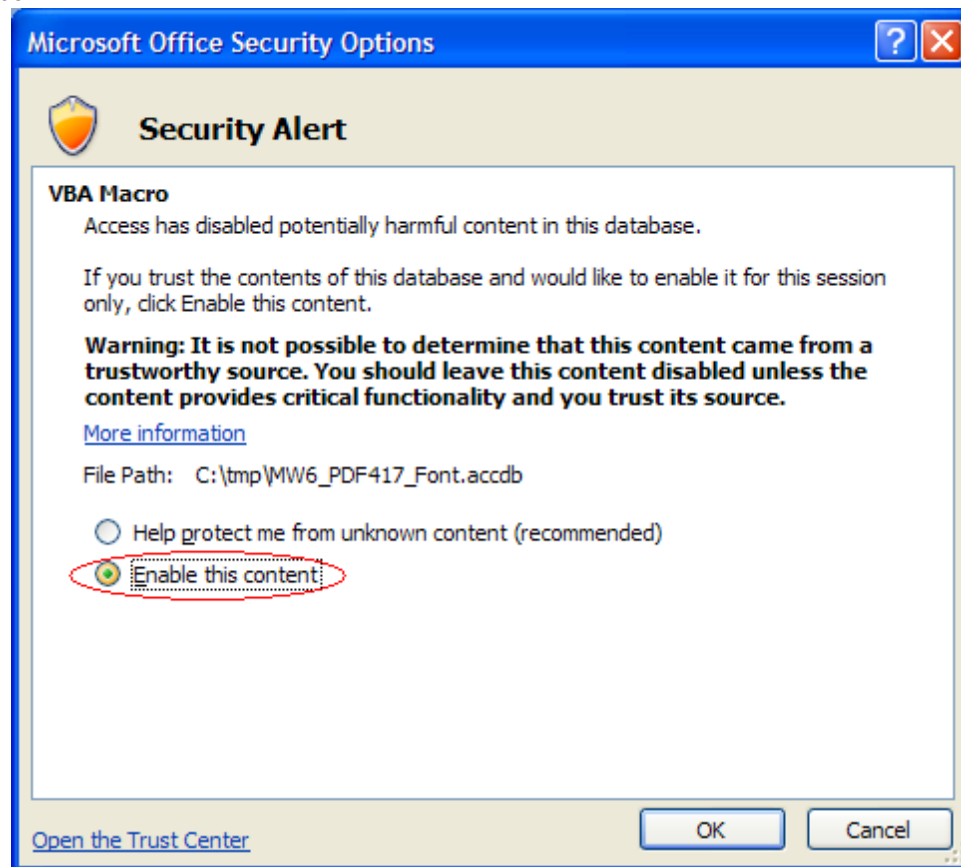


4.2 Access

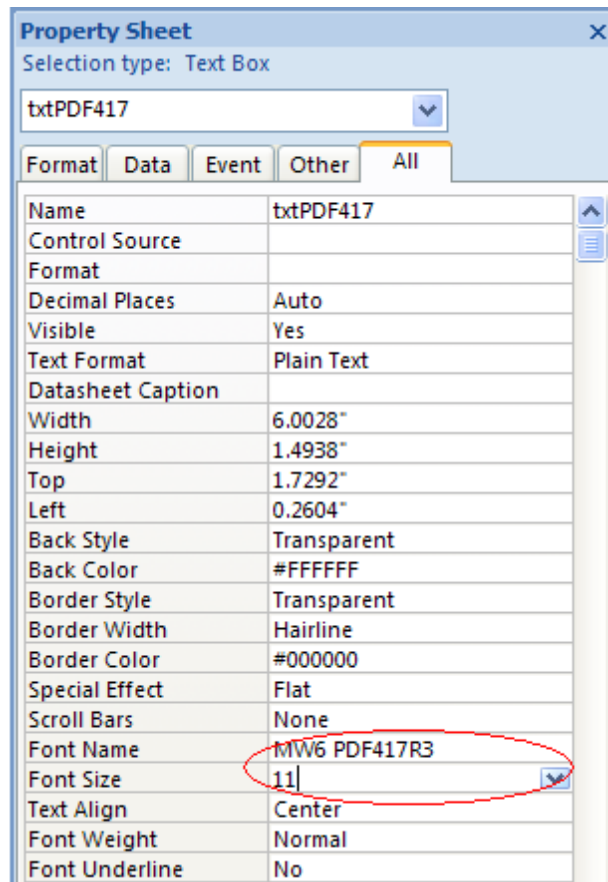
1. Copy "PDF417Font.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
2. Open MW6_PDF417_Font.accdb, select "**Sample PDF417 Font Report**".



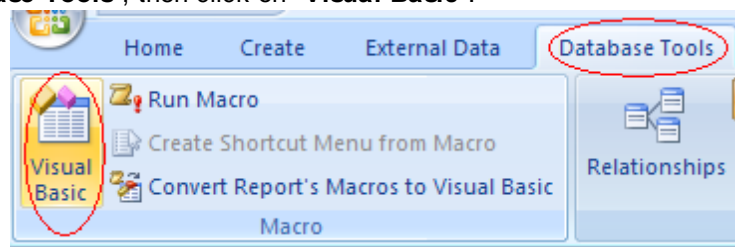
3. If you see "**Security Warning, Certain content in the database has been disabled**", click on "**Options**" to open "**Microsoft Office Security Options**" dialog, toggle on "**Enable this content**" check box.



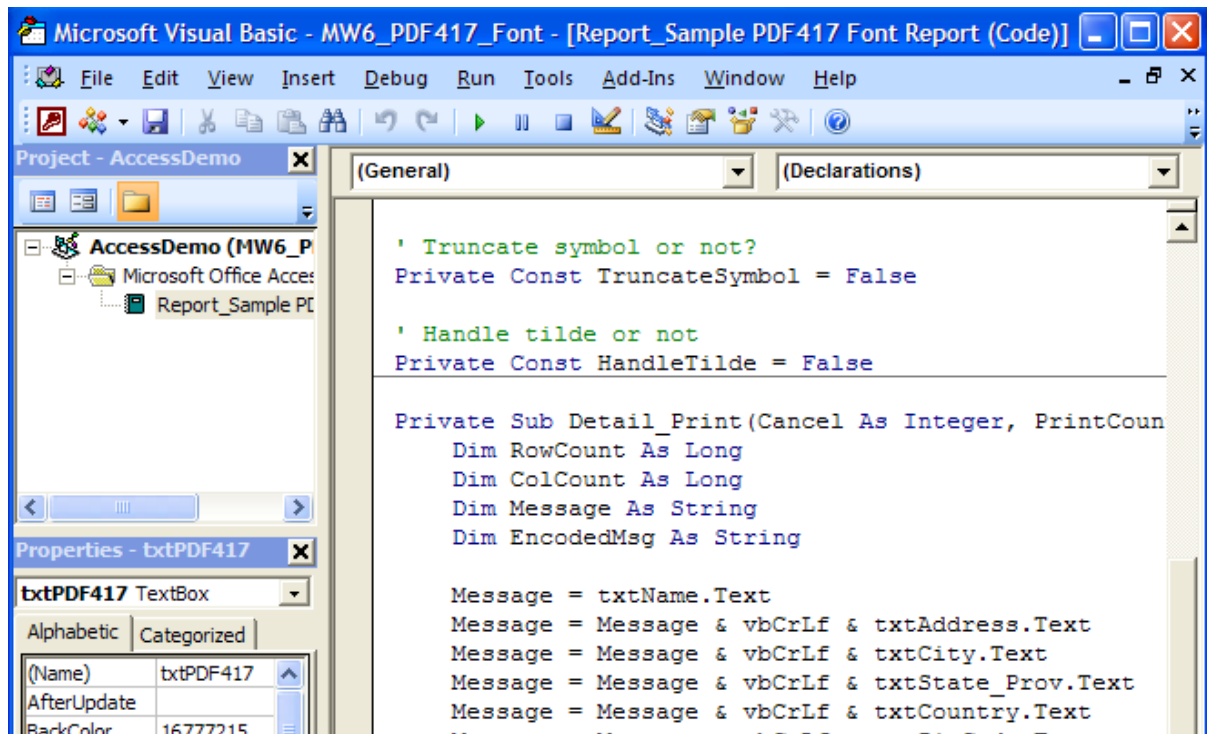
4. Click on "**Design**", insert a Text Box into the report, set its font to one of 4 PDF417 font options, choose an appropriate font size.



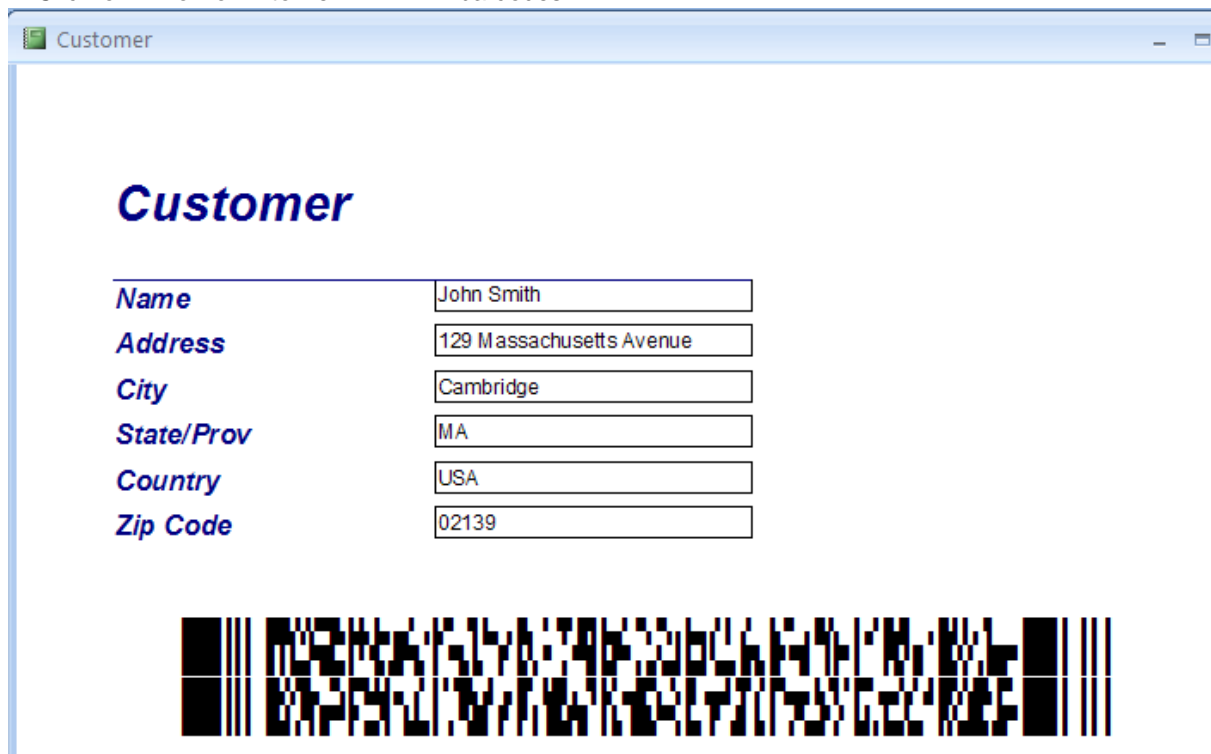
5. Click on "**Database Tools**", then click on "**Visual Basic**".



6. Convert a regular string to an PDF417 format barcode string in "*Private Sub Detail_Print(Cancel As Integer, PrintCount As Integer)*".



7. Click on "Preview" to view PDF417 barcodes.



5 Office 2000 & 2003

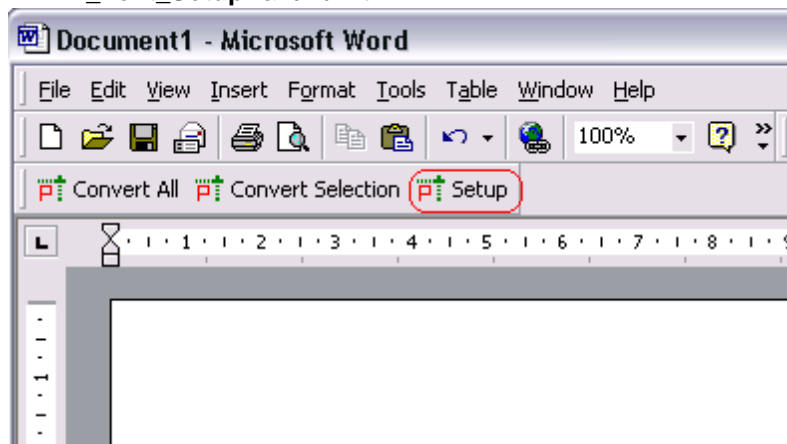
5.1 Word Demo

5.1.1 Install Template File

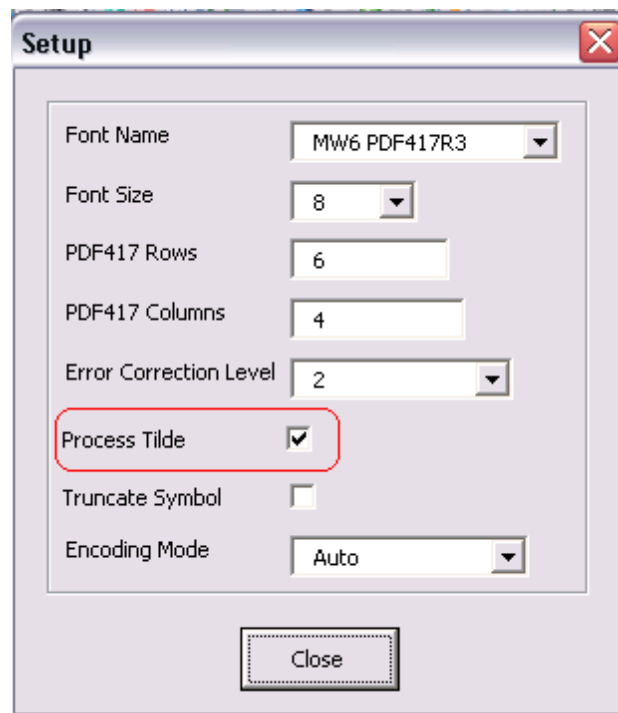
1. Locate the Word Startup folder, the Startup folder can be found in the following locations:

OS	Location
Windows Vista and above	"C:\Users\ <user name="">\AppData\Roaming\Microsoft\Word\STARTUP"</user>
Windows 2000/XP	"C:\Documents and Settings\ <user name="">\Application Data\Microsoft\Word\STARTUP"</user>
Windows NT4	"C:\Winnt\Profiles\ <user name="">\Application Data\Microsoft\Word\STARTUP"</user>
Windows 95, 98, ME	Office XP: "C:\Program Files\Microsoft Office\Office10\STARTUP" Office 2000/97: "C:\Program Files\Microsoft Office\Office\STARTUP"

2. Copy MW6_PDF417_Font.dot to this folder.
3. Copy "PDF417Font.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
4. Open up Word, click on "**Setup**". If you keep getting the error message "**The macro cannot be found or has been disabled because of**", download Office 2000 or 2003 Service Pack 3 from Microsoft website and install it to fix this issue. Or simply click "**Tools**" > "**Macro**" > "**Macros**", select "**MW6_PDF417_Font_Setup**" and run it.

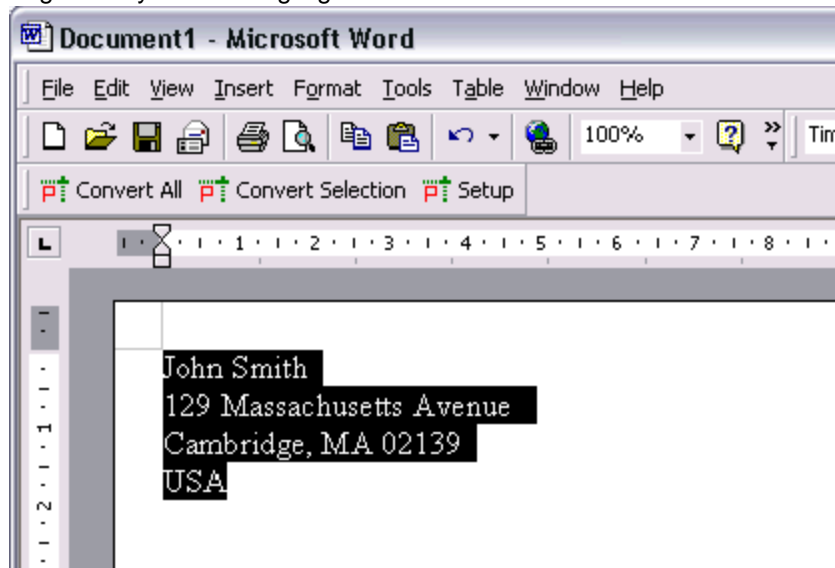


5. Change the configurations for PDF417 format, if you want to encode a string with non-printable characters, click on "**Process Tilde**" checkbox.

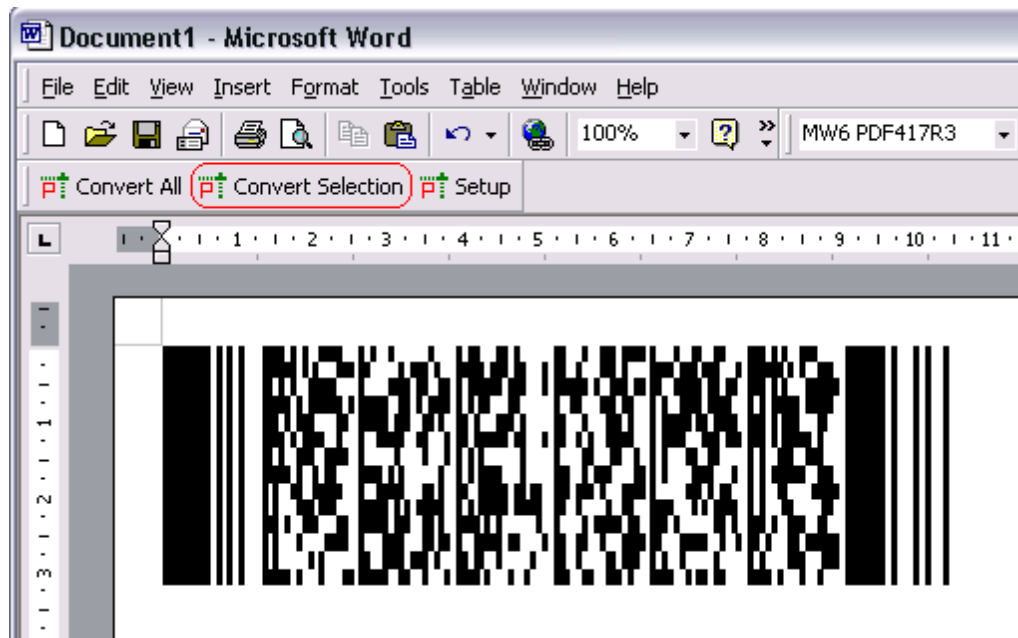


5.1.2 Create Single Barcode

1. Enter a few strings line by line and highlight them.

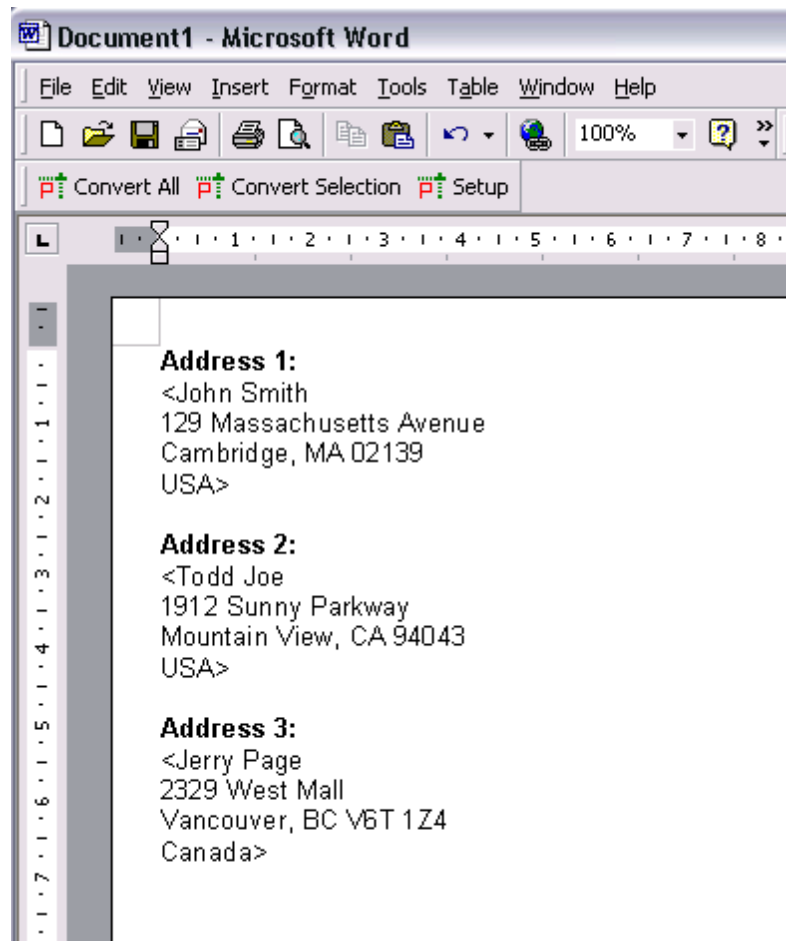


2. Click on "**Convert Selection**" to create a barcode.

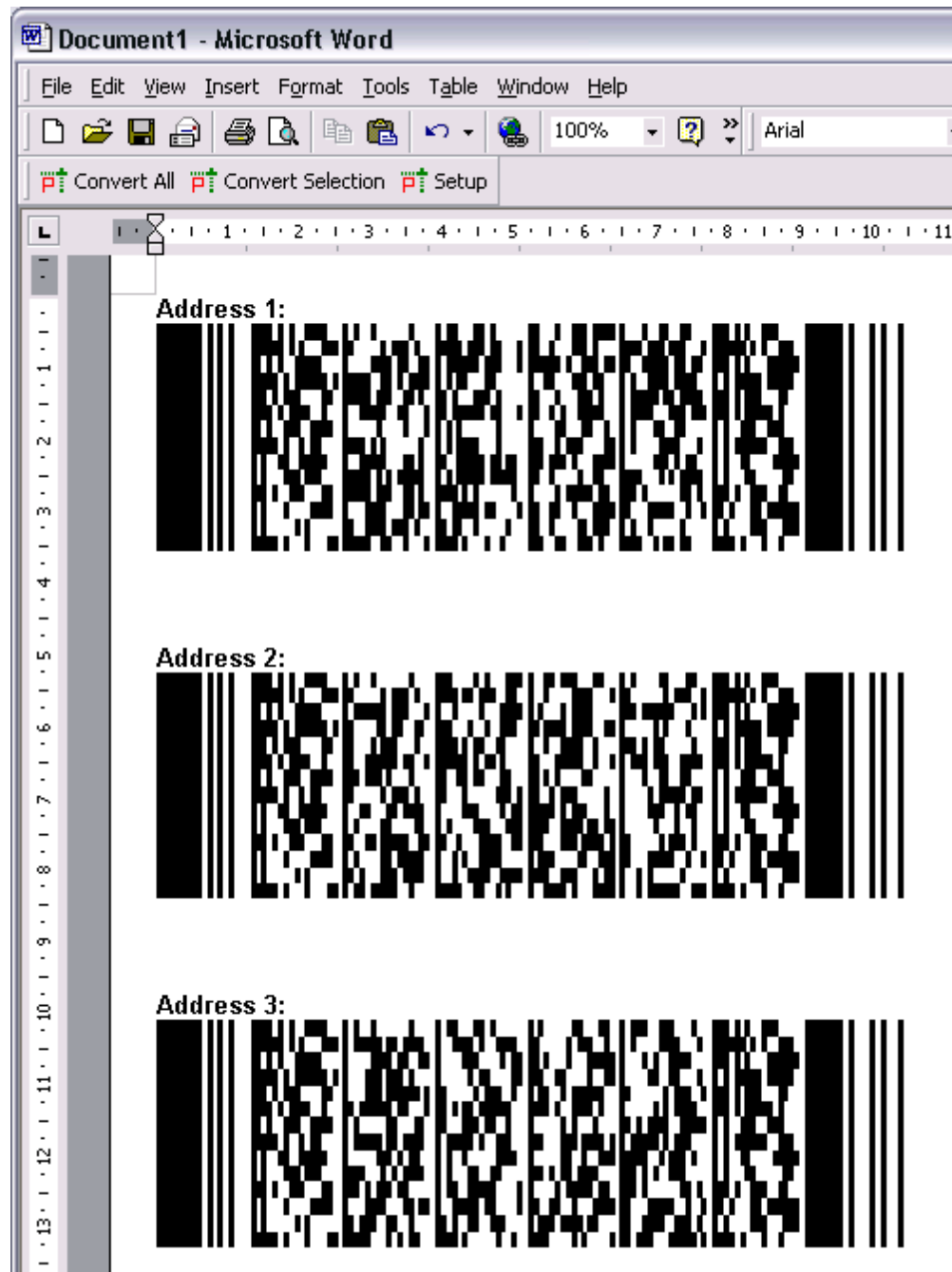


5.1.3 Create Multiple Barcodes

1. Enter a few paragraphs, surround those paragraphs which will be converted to the barcodes with the "<" and ">" characters.

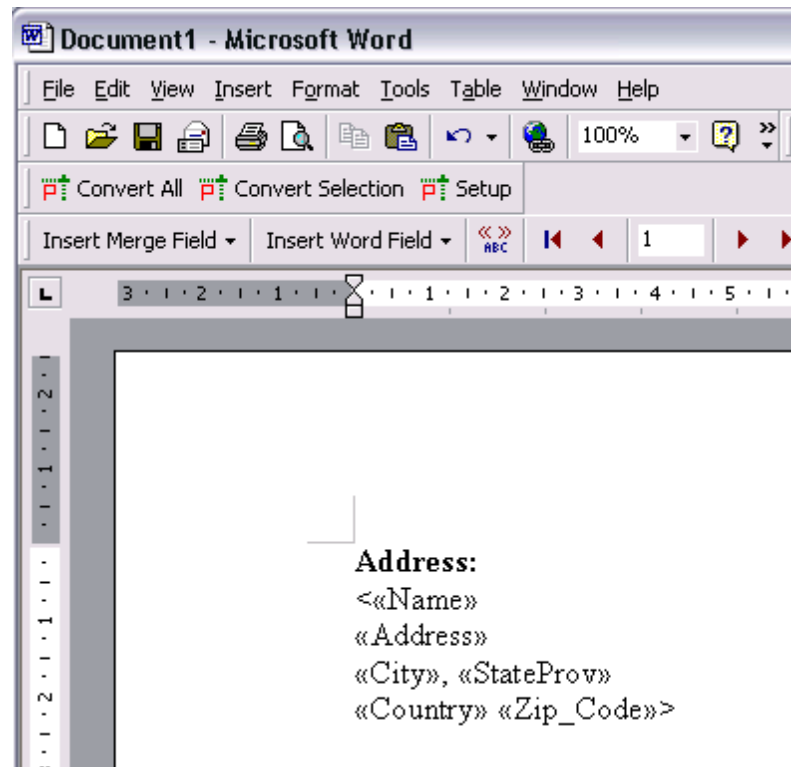


2. Click on "**Convert All**" to create barcodes for the paragraphs surrounded with the "<" and ">" characters.



5.1.4 Mail Merge

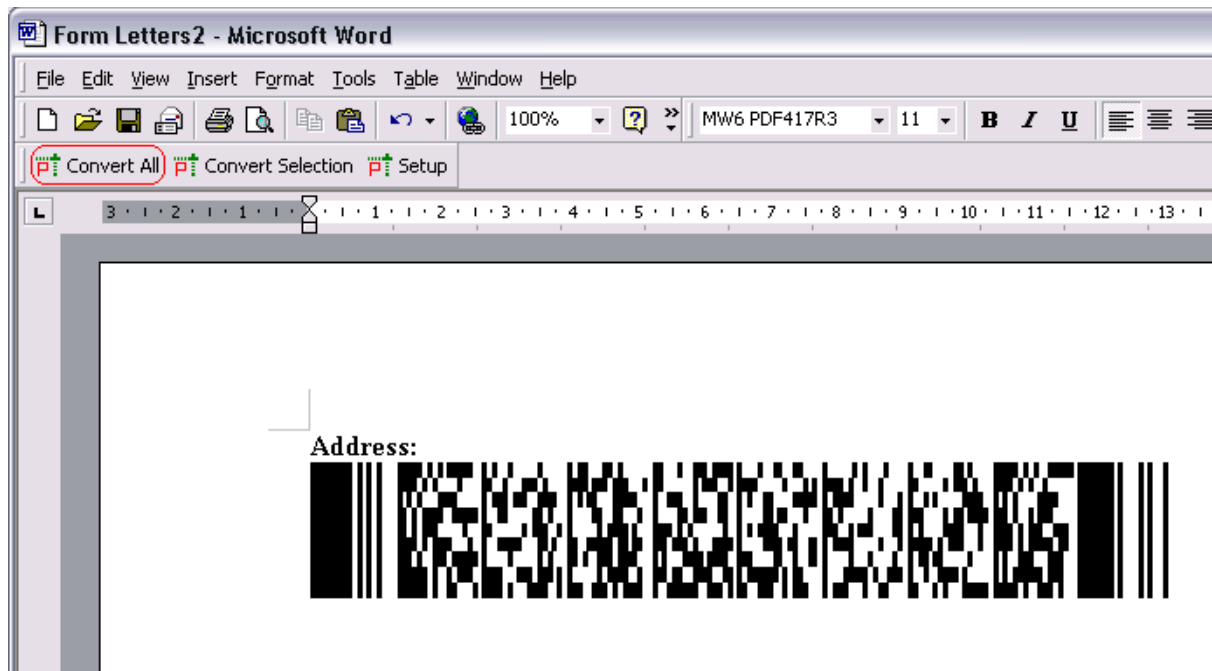
1. In Mail Merge, choose MW6_PDF417_Font.mdb as the Data Source, surround the paragraph which will be converted to PDF417 barcode with the "<" and ">" characters.



2. Click on "**Merge ...**"

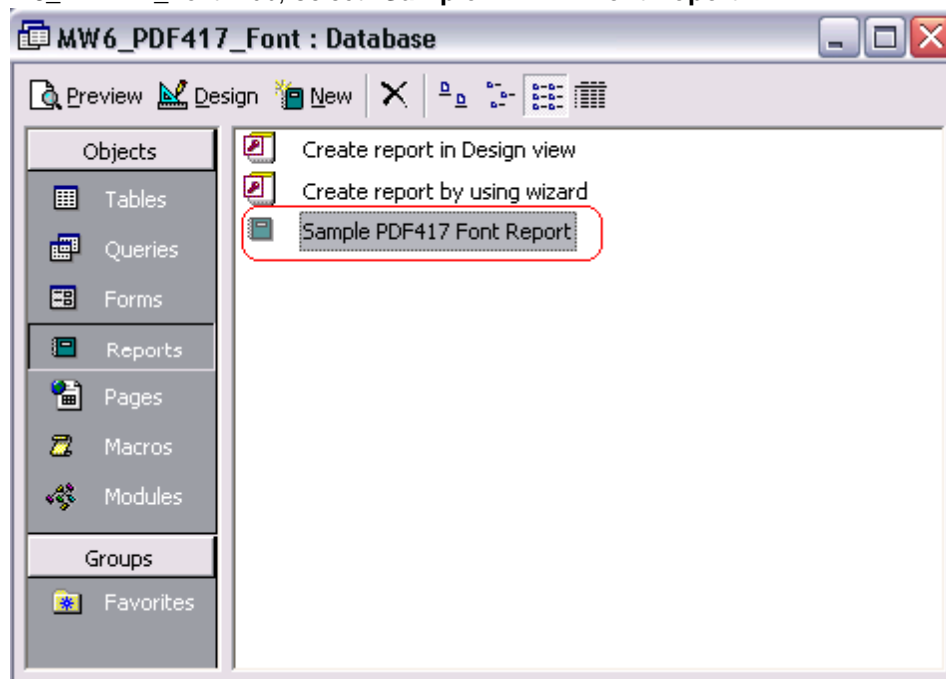


3. Click on "**Convert All**" to create PDF417 barcodes for the paragraphs surrounded with the "<" and ">" characters.

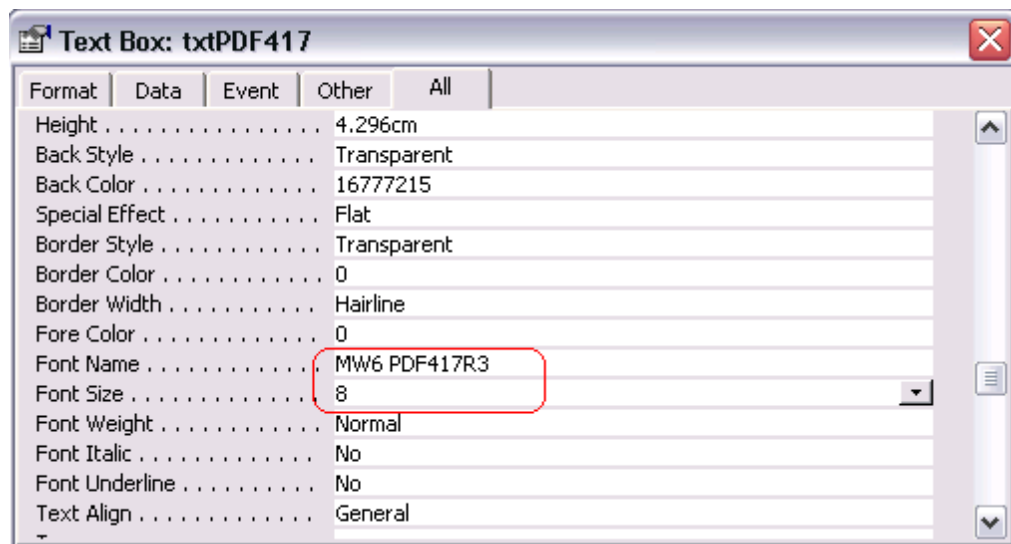


5.2 Access Demo

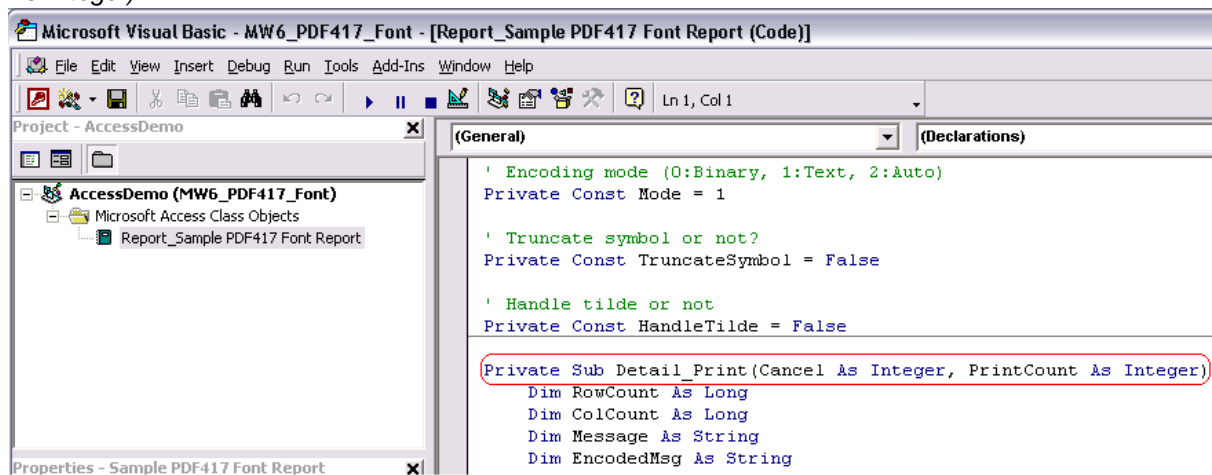
1. Copy "PDF417Font.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
2. Open MW6_PDF417_Font.mdb, select "**Sample PDF417 Font Report**".



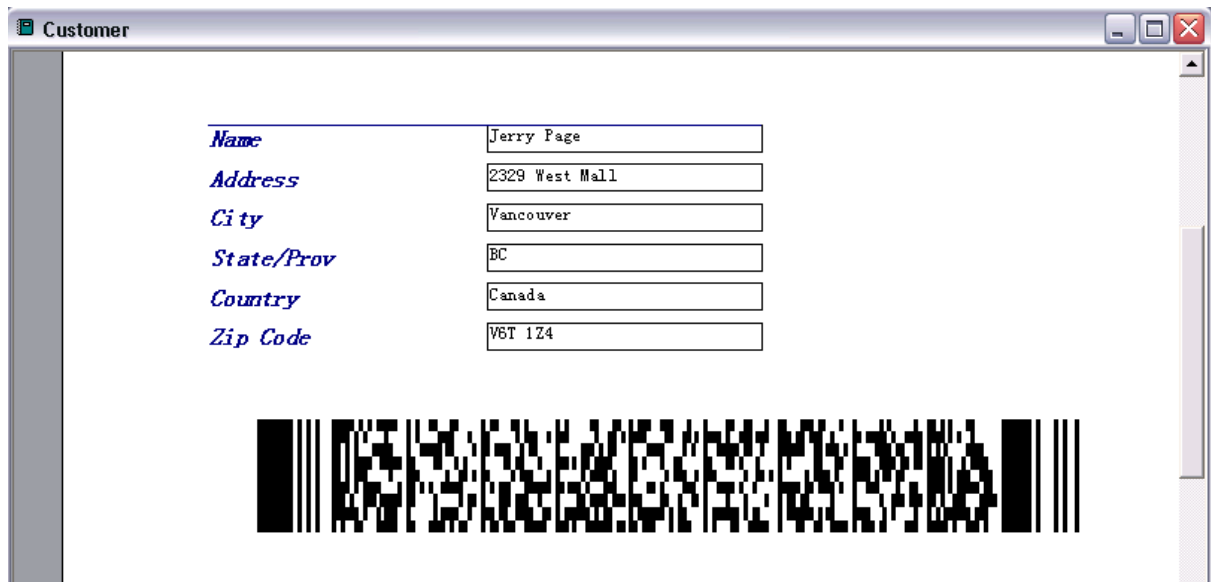
3. Click on "**Design**", insert a Text Box into the report, set its font to one of 4 PDF417 font options, choose an appropriate font size.



4. Convert a regular string to a barcode string in "*Private Sub Detail_Print(Cancel As Integer, PrintCount As Integer)*".



5. Click on "**Preview**" to view PDF417 barcodes.



The screenshot shows a window titled "Customer" with a form containing the following fields:

<i>Name</i>	Jerry Page
<i>Address</i>	2329 West Mall
<i>City</i>	Vancouver
<i>State/Prov</i>	BC
<i>Country</i>	Canada
<i>Zip Code</i>	V6T 1Z4

Below the form is a large barcode.

6 License

License agreement

This License Agreement ("LA") is the legal agreement between you and MW6 Technologies, Inc. ("MW6") for the font, and any electronic documentation ("Package"). By using, copying or installing the Package, you agree to be bound by the terms of this LA. If you don't agree to the terms in this LA, immediately remove unused Package.

1. License

* The Single User License allows the use of the font on **ONE** computer by **ONE** person in your organization.

* The Site License allows the use of the font at exactly 1 physical site by up to 10,000 users in your organization.

* The Single Developer License allows 1 developer in your organization the royalty-free distribution (up to 10,000 users) of the font to the third parties, **each individual developer requires a separate Single Developer License as long as he or she needs access to MW6's product(s) and document(s).**

* The 2 Developer License allows 2 developers in your organization the royalty-free distribution (up to 10,000 users) of the font to the third parties.

* The 3 Developer License allows 3 developers in your organization the royalty-free distribution (up to 10,000 users) of the font to the third parties.

* The 4 Developer License allows 4 developers in your organization the royalty-free distribution (up to 10,000 users) of the font to the third parties.

* The 5 Developer License allows 5 developers in your organization the royalty-free distribution (up to 10,000 users) of the font to the third parties.

* The Unlimited Developer License allows unlimited number of developers in your organization the royalty-free distribution (unlimited number of users) of the font to the third parties.

2. User Disclaimer

The font is provided "as is" without warrant of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or noninfringement. MW6 assumes no liability for damages, direct or consequential, which may result from the use of the font. Further, MW6 assumes no liability for losses caused by misuse or abuse of the font. This responsibility rests solely with the end user.

3. Copyright

The font and any electronic documentation are the proprietary products of MW6 and are protected by copyright and other intellectual property laws.
