

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.....	3
3 Reference Guide.....	4
Win32 DLL Encoder Functions	4
DMEncode Function.....	4
DMGetCharAt Function.....	5
DMGetCols Function.....	5
DMGetRows Function.....	5
.NET Class Library Encoder Methods	5
Encode Method.....	5
GetCols Method.....	7
GetRows Method.....	7
GetRowStringAt Method.....	7
Part III Crystal Reports	8
1 How To Use It.....	8
2 UFL Functions.....	13
DMFontEncode Function	13
DMFontGetBlock Function	14
3 Legacy UFL.....	14
How To Use It	14
How To Distribute It	20
UFL Functions	20
DatUFLMW6Encoder Function.....	20
DatUFLMW6GetBlock Function.....	21
Part IV Office 2007 & 2010	21
1 Word.....	21
Install Template File	21
Create Single Barcode	22
Create Multiple Barcodes	23
Mail Merge	25
2 Access.....	30
Part V Office 2000 & 2003	33
1 Word Demo.....	33
Install Template File	33
Create Single Barcode	35
Create Multiple Barcodes	35
Mail Merge	37

2	Access Demo.....	38
	Part VI DataMatrix Formats and Data Capacity	41
	Part VII License	41
	Index	0

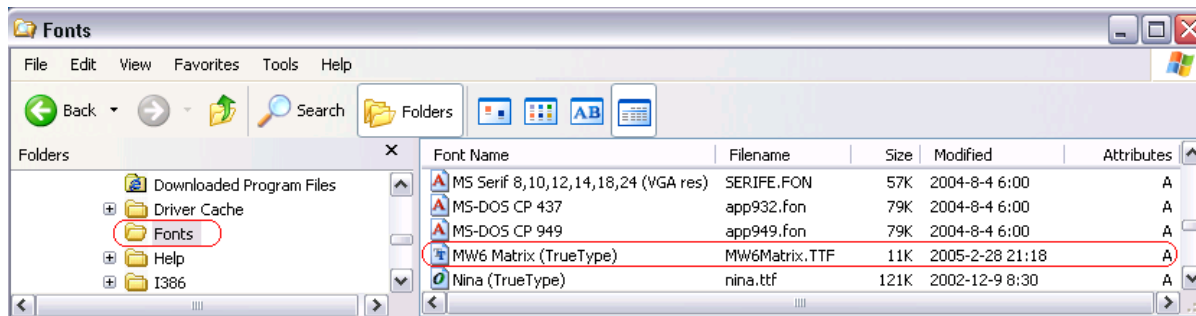
1 Introduction

MW6 DataMatrix font package can print DataMatrix 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 DataMatrix.

There is only 1 TrueType font file (MW6Matrix.ttf) bundled with the font package:

Font name	Height/Width of module at 12 points
MW6 Matrix	1/30 inch

The above font file is same for the trial version package and the full version package, copy MW6Matrix.ttf file to the Windows Fonts folder.



2 How To Use Encoders

2.1 Win32 DLL Encoder

If you build DataMatrix 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 DataMatrix font format string.

1. For 32-bit OS such as Windows XP and NT, copy "DataMatrixFont.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 "DataMatrixFont.dll" to the folder "C:\windows\system32".
3. For 64-bit version of Windows Vista and above, copy "DataMatrixFont.dll" to the folder "C:\windows\SysWow64".
4. If you want to generate DataMatrix barcodes inside 64-bit Office Word, Excel, or Access, copy "DataMatrixFont_x64.dll" to the folder "c:\windows\system32".

2.2 .NET Class Library Encoder

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

2.3 Reference Guide

2.3.1 Win32 DLL Encoder Functions

2.3.1.1 DMEncode Functiona

Encodes a string using DataMatrix format.

```
void DMEncode(
  LPCTSTR Message,
  WORD Mode,
  WORD PreferredFormat,
  BOOL HandleTilde);
```

Parameters

Message

String to be encoded using DataMatrix format.

Mode

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

Value	Comment
0	Auto mode for optimized encoding
1	ASCII encoding mode
2	C40 encoding mode
3	Text encoding mode
4	Base256 encoding mode

PreferredFormat

Indicates which format is used, the values of all formats are listed here.

HandleTilde

Indicates whether to process the tilde character "~" or not. If it is set to TRUE, non-printable characters can be passed to the encoder library 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).

"~1" is used to indicate FNC1. For example, "~10107612345678900~117100503" can be used to generate GS1 DataMatrix "(01)0107612345678900(17)100503", and "~110AC34563G3" can be used to generate GS1 DataMatrix "(10)AC34563G3".

"~5" is used to indicate Macro 5. For example, "~5ABCDEF[GS]123456" can be used to generate DataMatrix "[>][RS]05[GS]ABCDEF[GS]123456[RS][EOT]".

"~6" is used to indicate Macro 6. For example, "~6ABCDEF[GS]123456" can be used to generate DataMatrix "[>][RS]06[GS]ABCDEF[GS]123456[RS][EOT]".

[RS] is the record separator with ASCII value 30, [GS] is the group separator with ASCII value 29,

and [EOT] is the end of transmission with ASCII value 4.

2.3.1.2 DMGetCharAt Function

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

```
WORD DMGetCharAt(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 DataMatrix font data matrix.

2.3.1.3 DMGetCols Function

Returns the number of columns in DataMatrix font data matrix.

```
WORD DMGetCols();
```

Return Value

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

2.3.1.4 DMGetRows Function

Returns the number of rows in DataMatrix font data matrix.

```
WORD DMGetRows();
```

Return Value

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

2.3.2 .NET Class Library Encoder Methods

2.3.2.1 Encode Method

Encodes a string using DataMatrix format.

```
[Visual Basic .NET]  
Public Sub Encode(ByVal Message As String, _  
                 ByVal Mode As Integer, _
```

```
ByVal PreferredFormat As Integer, _
ByVal HandleTilde As Boolean)
```

```
[C#]
public void Encode(string Message,
    int Mode,
    int PreferredFormat,
    bool HandleTilde);
```

Parameters

Message

String to be encoded using DataMatrix format.

Mode

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

Value	Comment
0	Auto mode for optimized encoding
1	ASCII encoding mode
2	C40 encoding mode
3	Text encoding mode
4	Base256 encoding mode

PreferredFormat

Indicates which format is used, the values of all formats are listed here.

HandleTilde

Indicates whether to process the tilde character "~" or not. If it is set to TRUE, non-printable characters can be passed to the encoder library 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).

"~1" is used to indicate FNC1. For example, "~10107612345678900~117100503" can be used to generate GS1 DataMatrix "(01)0107612345678900(17)100503", and "~110AC34563G3" can be used to generate GS1 DataMatrix "(10)AC34563G3".

"~5" is used to indicate Macro 5. For example, "~5ABCDEF[GS]123456" can be used to generate DataMatrix "[>][RS]05[GS]ABCDEF[GS]123456[RS][EOT]".

"~6" is used to indicate Macro 6. For example, "~6ABCDEF[GS]123456" can be used to generate DataMatrix "[>][RS]06[GS]ABCDEF[GS]123456[RS][EOT]".

[RS] is the record separator with ASCII value 30, [GS] is the group separator with ASCII value 29, and [EOT] is the end of transmission with ASCII value 4.

2.3.2.2 GetCols Method

Returns the number of columns in DataMatrix 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 DataMatrix font data matrix.

2.3.2.3 GetRows Method

Returns the number of rows in DataMatrix 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 DataMatrix font data matrix.

2.3.2.4 GetRowStringAt Method

Concatenates characters for a row in DataMatrix 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

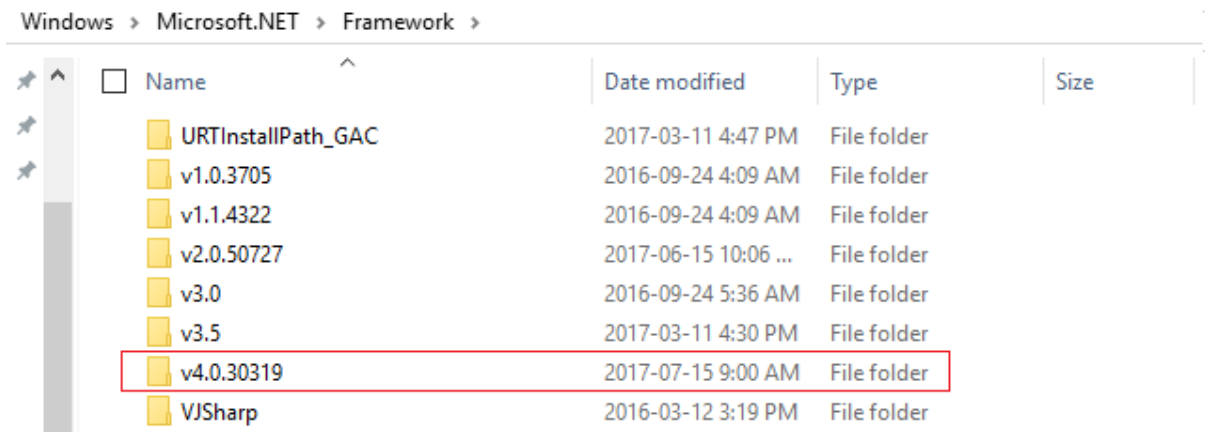
The return value is a row string for DataMatrix font data matrix.

3 Crystal Reports

3.1 How To Use It

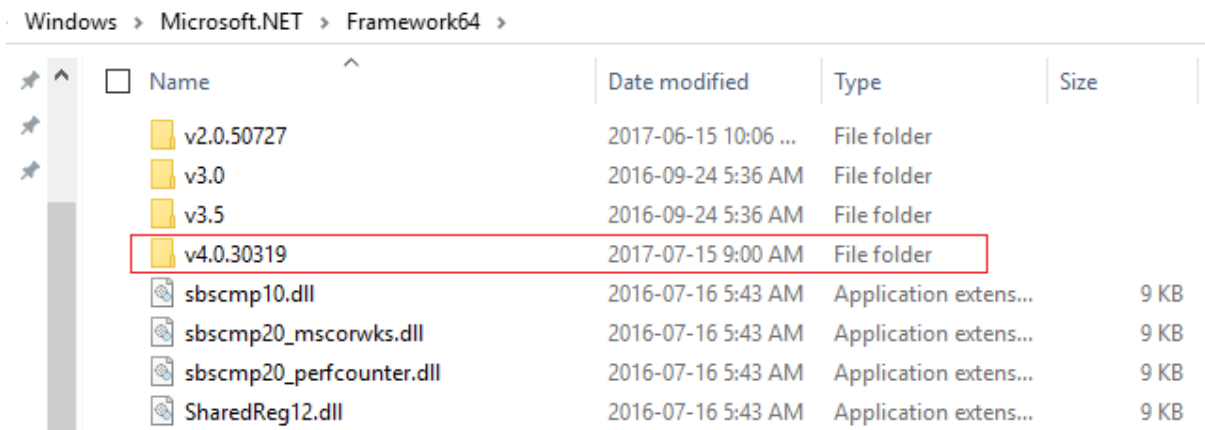
1. For 32-bit version of Crystal Reports, unzip UFL_32.ZIP and put CRUFLDM.dll in SysWow64 folder (64-bit version of Windows OS) or system32 folder (32-bit version of Windows OS). Run the following command as "**Run as administrator**", and the **ver_num** is the version number for 32-bit version of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

```
C:\Windows\Microsoft.NET\Framework\ver_num\RegAsm.exe /codebase C:\Windows
\SystemWow64\CRUFLDM.dll
```



2. For 64-bit version of Crystal Reports, unzip UFL_64.ZIP and put CRUFLDM.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\CRUFLDM.dll
```



3. Open up Crystal Reports, go to "**Field Explorer**", right click on "**Formula Fields**", click on "**New**", enter "**DataMatrix 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 DataMatrix font related functions.

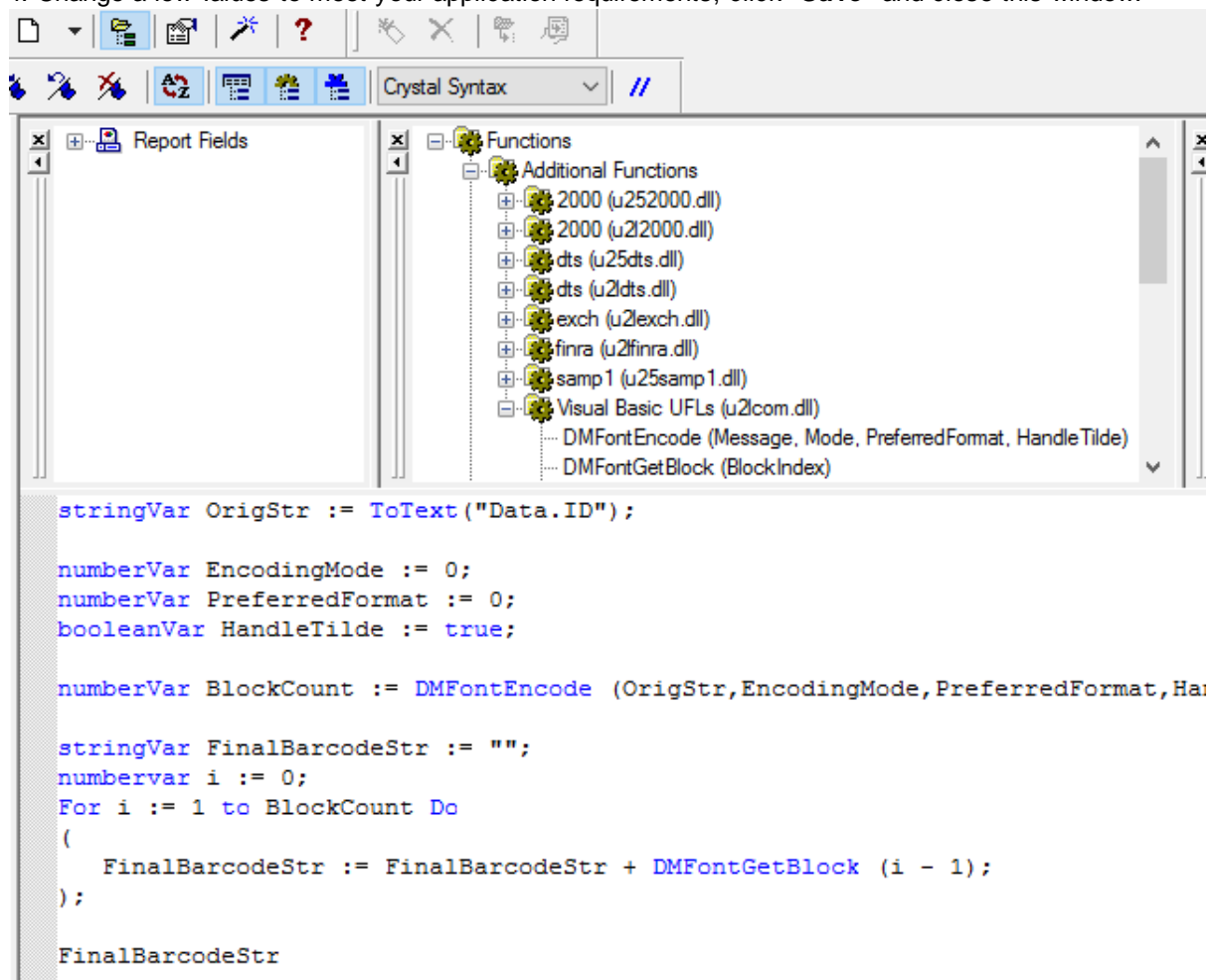
```
stringVar OrigStr := ToText({TableName.FieldName});
numberVar EncodingMode := 0;
numberVar PreferredFormat := 0;
booleanVar HandleTilde := true;

numberVar BlockCount := DMFontEncode (OrigStr,EncodingMode,PreferredFormat,HandleTilde);

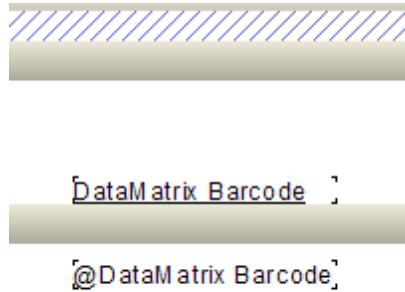
stringVar FinalBarcodeStr := "";
numberVar i := 0;
For i := 1 to BlockCount Do
(
    FinalBarcodeStr := FinalBarcodeStr + DMFontGetBlock (i - 1);
);

FinalBarcodeStr
```

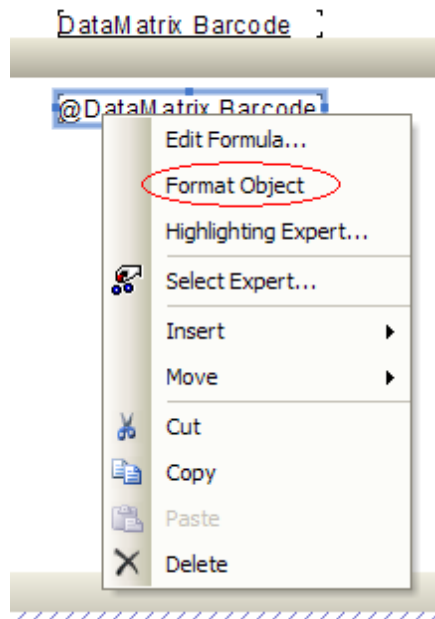
4. Change a few values to meet your application requirements, click "**Save**" and close this window.



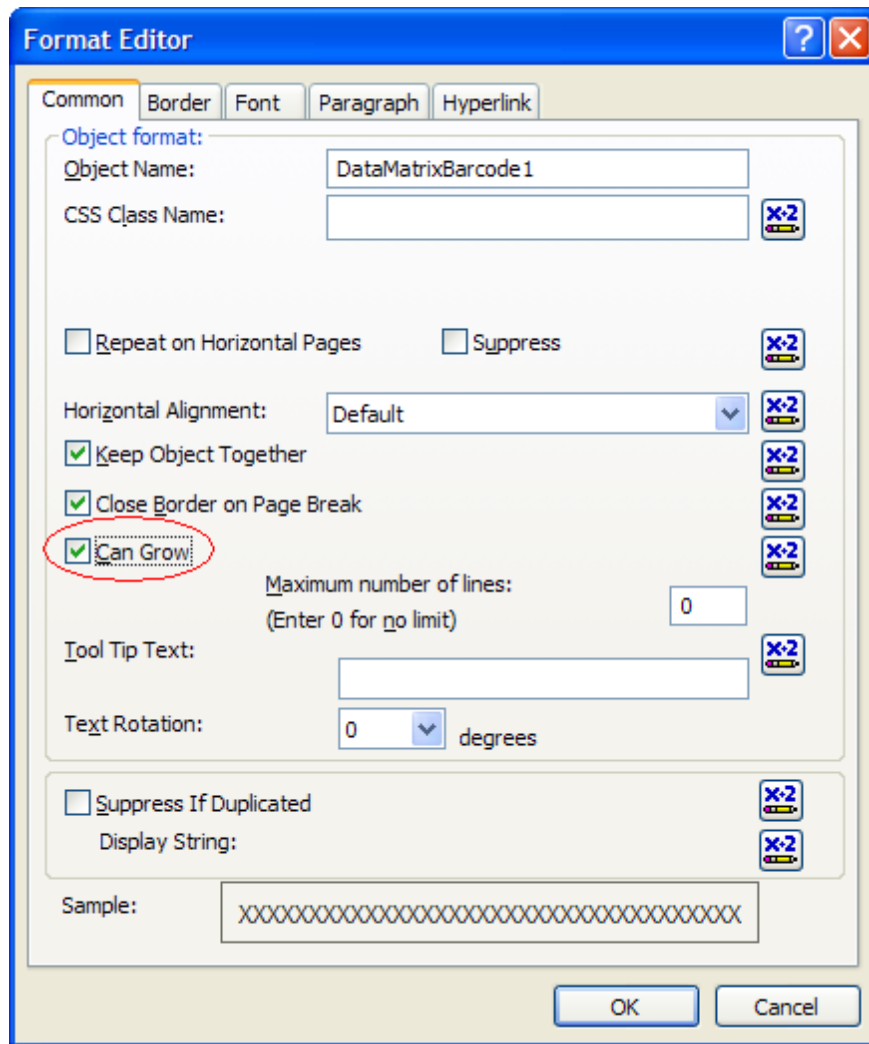
5. Click on the formula field "**DataMatrix Barcode**" and drag it on the report.



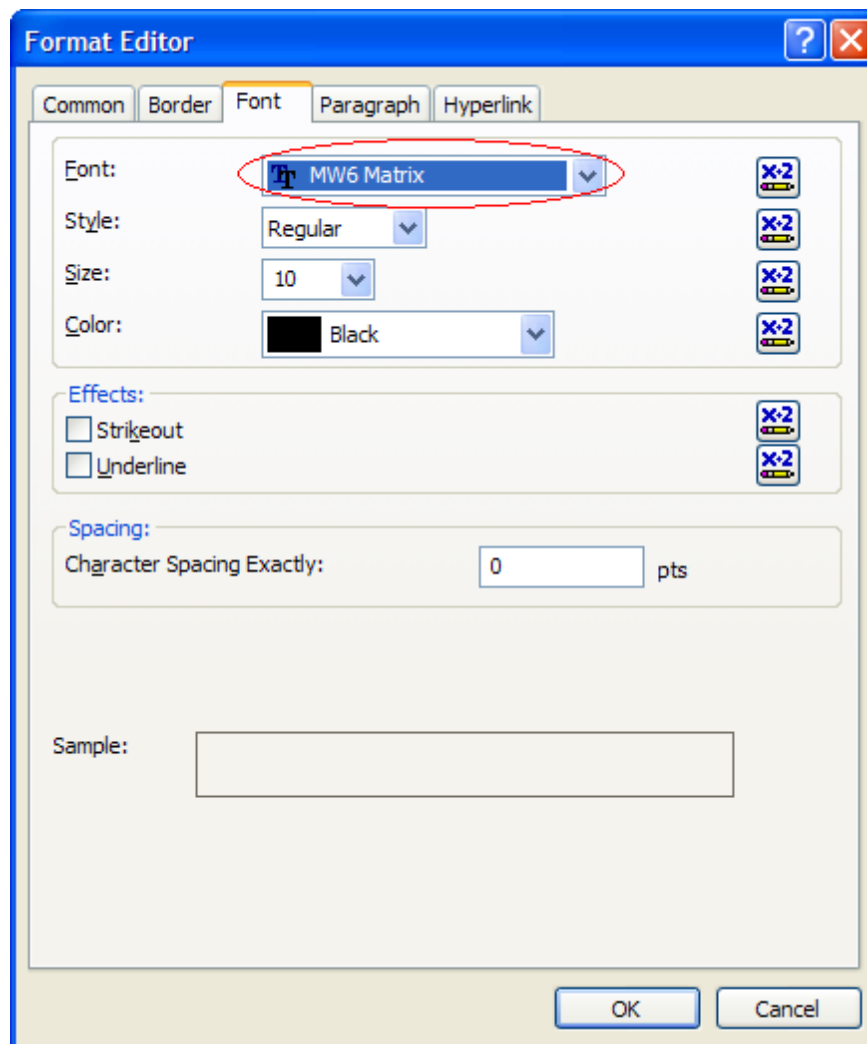
6. Right-click "**@DataMatrix Barcode**" and choose "**Format Object**".



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



8. Choose "MW6 Matrix" as the font name under "Font" tab.



9. Run the report.

DataMatrix Barcode



3.2 UFL Functions

3.2.1 DMFontEncode Function

Encodes a string using DataMatrix format.

```
Public Function DMFontEncode(ByVal Message As String, _  
                             ByVal Mode As Integer, _  
                             ByVal PreferredFormat As Integer, _  
                             ByVal HandleTilde As Boolean) As Integer
```

Parameters

Message

String to be encoded using DataMatrix format.

Mode

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

Value	Comment
0	Auto mode for optimized encoding
1	ASCII encoding mode
2	C40 encoding mode
3	Text encoding mode
4	Base256 encoding mode

PreferredFormat

Indicates which format is used, the values of all formats are listed here.

HandleTilde

Indicates whether to process the tilde character "~" or not, the detailed description is:

- "~1" is used to represent the FNC1 code.
- "~2" is used to represent the Structured Append and must be followed by a 3-digit number between 1 and 255.
- "-3" is used only at the very beginning of the symbol for the reader programming purpose.
- "~5" is used only at the very beginning of the symbol, the header []> + ASCII 30 + ASCII 05 + ASCII 29 will be transmitted by the barcode reader before the data in the message and the trailer ASCII 30 + ASCII 4 will be transmitted afterwards.
- "~6" is used only at the very beginning of the symbol, the header []> + ASCII 30 + ASCII 06 + ASCII 29 will be transmitted by the barcode reader before the data in the message and the trailer ASCII 30 + ASCII 4 will be transmitted afterwards.
- "~7NNNNNN" is used to specify the Extended Channel and NNNNNN is a value between 000000 and 999999.

- o "~dNNN" is used to represent the ASCII character with the value of NNN.

Return Value

Number of DataMatrix 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 DataMatrix format string by concatenating all blocks together.

3.2.2 DMFontGetBlock Function

Retrieves a block data of DataMatrix format string.

```
Public Function DMFontGetBlock(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 DataMatrix 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 DataMatrix 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 DataMatrix barcode into your reports.
2. Go to the folder where u2lcom.dll is located and copy CRUFLDat.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

3. 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.
4. For 32-bit version Windows OS, run "regsvr32 CRUFLDat.dll" to register it. Copy "DataMatrixFont.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32") and move to the step 6.
5. For 64-bit version Windows OS, run "C:\windows\SysWOW64\regsvr32 CRUFLDat.dll" to register it. Copy "DataMatrixFont.dll" to the windows 32-bit system folder, which is "C:\windows\SysWOW64".
6. Open up Crystal Reports, go to **"Field Explorer"**, right click on **"Formula Fields"**, click on **"New"**, enter **"DataMatrix Barcode"**, copy the following code into the Formula Editor area.

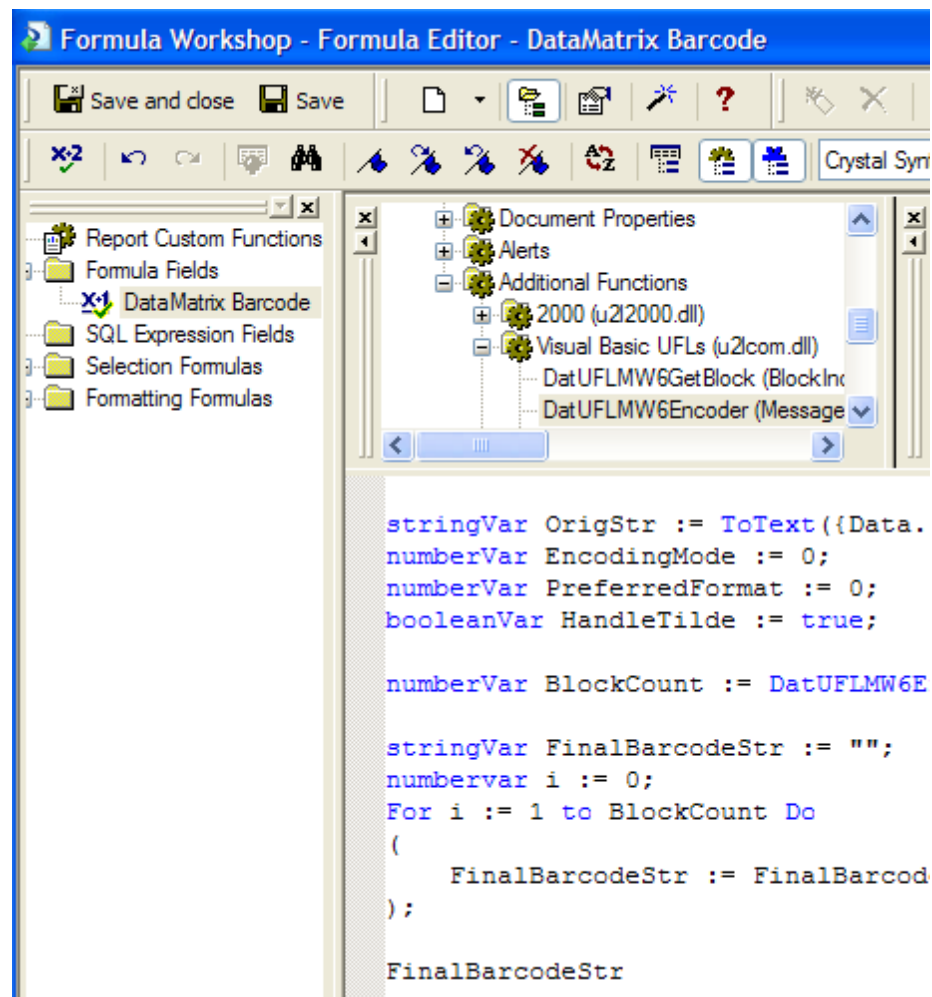
```
stringVar OrigStr := ToText({TableName.FieldName});
numberVar EncodingMode := 0;
numberVar PreferredFormat := 0;
booleanVar HandleTilde := true;

numberVar BlockCount := DatUFLMW6Encoder (OrigStr,EncodingMode,PreferredFormat,HandleTilde);

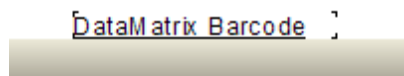
stringVar FinalBarcodeStr := "";
numbervar i := 0;
For i := 1 to BlockCount Do
(
    FinalBarcodeStr := FinalBarcodeStr + DatUFLMW6GetBlock (i - 1);
);

FinalBarcodeStr
```

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

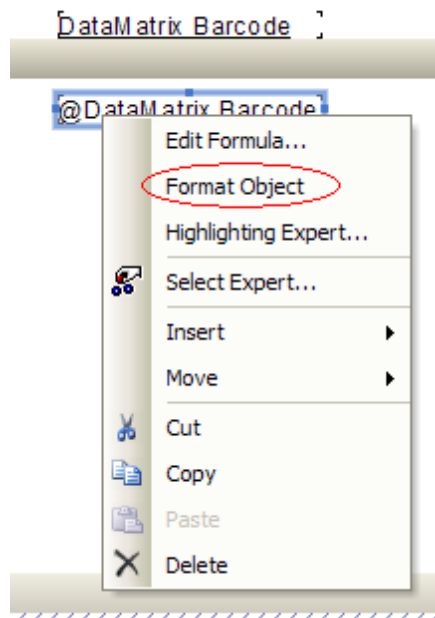


8. Click on the formula field "**DataMatrix Barcode**" and drag it on the report.

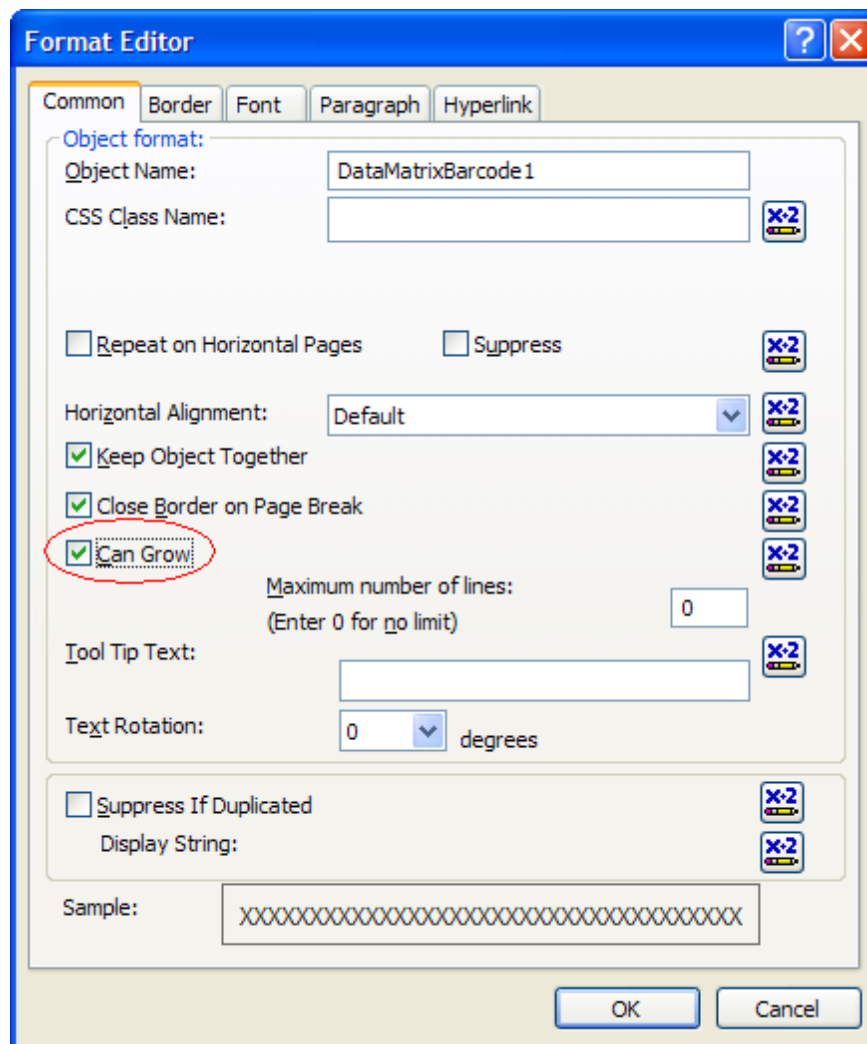


[DataMatrix Barcode]

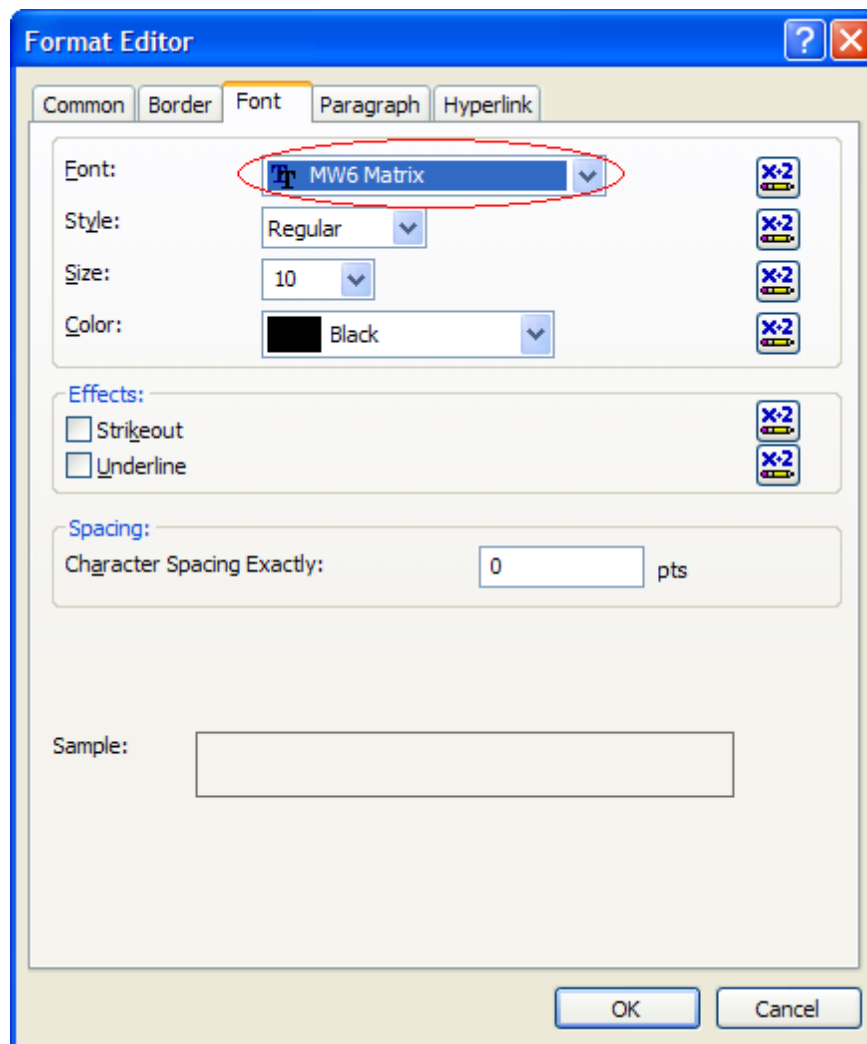
9. Right-click "**@DataMatrix Barcode**" and choose "**Format Object**".



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



11. Choose "MW6 Matrix" as the font name under "Font" tab.



12. Run the report.

DataMatrix Barcode



3.3.2 How To Distribute It

For the distribution purpose, you need to distribute MW6 DataMatrix font .ttf file (MW6Matrix.ttf), Win32 DLL Encoder (DataMatrixFont.dll), Crystal Reports UFL (CRUFLDat.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 DatUFLMW6Encoder Function

Encodes a string using DataMatrix format.

```
Public Function DatUFLMW6Encoder(ByVal Message As String, _
                                ByVal Mode As Integer, _
                                ByVal PreferredFormat As Integer, _
                                ByVal HandleTilde As Boolean) As Integer
```

Parameters

Message

String to be encoded using DataMatrix format.

Mode

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

Value	Comment
0	Auto mode for optimized encoding
1	ASCII encoding mode
2	C40 encoding mode
3	Text encoding mode
4	Base256 encoding mode

PreferredFormat

Indicates which format is used, the values of all formats are listed here.

HandleTilde

Indicates whether to process the tilde character "~" or not, the detailed description is:

- "~1" is used to represent the FNC1 code.
- "~2" is used to represent the Structured Append and must be followed by a 3-digit number between 1 and 255.
- "~3" is used only at the very beginning of the symbol for the reader programming purpose.
- "~5" is used only at the very beginning of the symbol, the header []> + ASCII 30 + ASCII 05 + ASCII 29 will be transmitted by the barcode reader before the data in the message and the trailer ASCII 30 + ASCII 4 will be transmitted afterwards.

- "~6" is used only at the very beginning of the symbol, the header []> + ASCII 30 + ASCII 06 + ASCII 29 will be transmitted by the barcode reader before the data in the message and the trailer ASCII 30 + ASCII 4 will be transmitted afterwards.
- "~7NNNNNN" is used to specify the Extended Channel and NNNNNN is a value between 000000 and 999999.
- "~dNNN" is used to represent the ASCII character with the value of NNN.

Return Value

Number of DataMatrix 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 DataMatrix format string by concatenating all blocks together.

3.3.3.2 DatUFLMW6GetBlock Function

Retrieves a block data of DataMatrix format string.

```
Public Function DatUFLMW6GetBlock(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 DataMatrix format string.

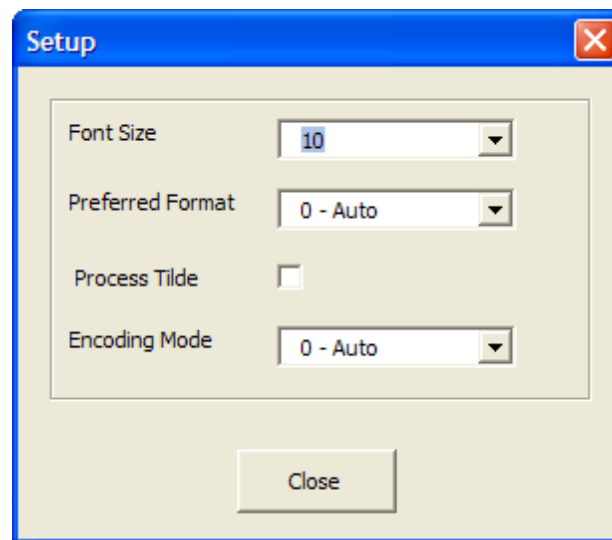
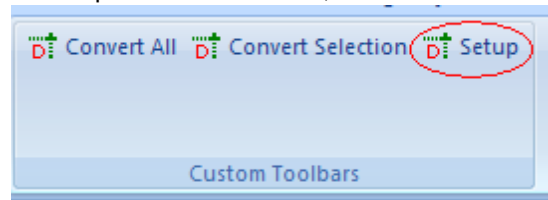
4 Office 2007 & 2010

4.1 Word

4.1.1 Install Template File

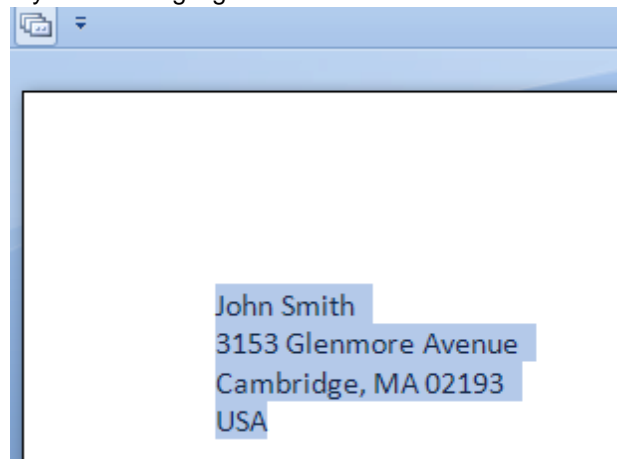
1. Locate Microsoft Word Startup folder, which usually is "C:\Documents and Settings\\Application Data\Microsoft\Word\STARTUP" for Windows XP or "C:\Users\\AppData\Roaming\Microsoft\Word\STARTUP" for Windows Vista and above.
2. Copy MW6_DataMatrix_Font.dotm for 32-bit Office or MW6_DataMatrix_Font_x64.dotm for 64-bit Office to this folder.
3. For 32-bit Office, copy "DataMatrixFont.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").

- For 64-bit Office, copy "DataMatrixFont_x64.dll" to the windows 32-bit system folder (e.g., "C:\windows\system32").
- Click on "**Add-Ins**", then click on "**Setup**". Change the configurations for DataMatrix format, if you want to encode a string with non-printable characters, click on "**Process Tilde**" checkbox.

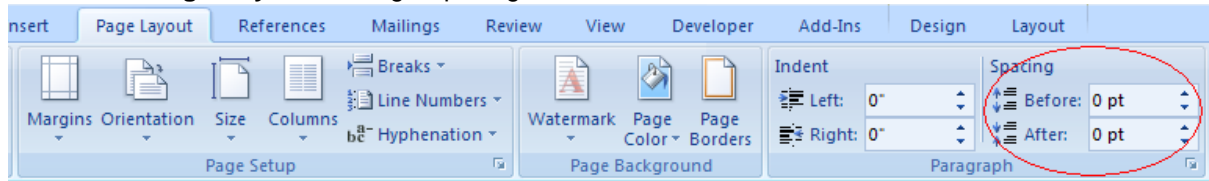


4.1.2 Create Single Barcode

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



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

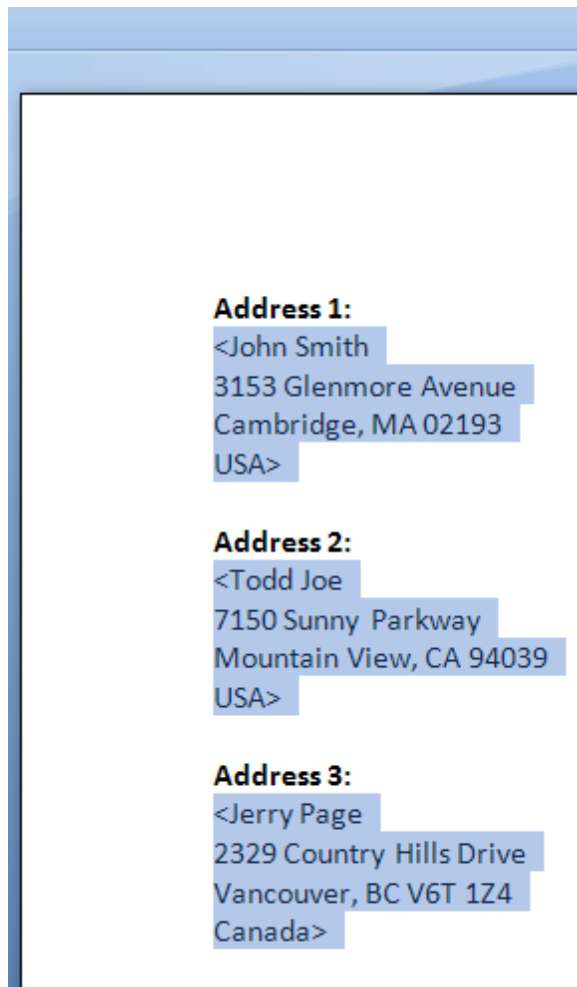


3. Click on "**Add-Ins**", then click on "**Convert Selection**" to create a DataMatrix barcode.

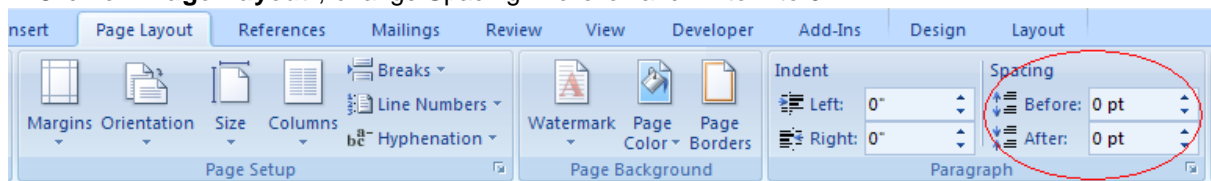


4.1.3 Create Multiple Barcodes

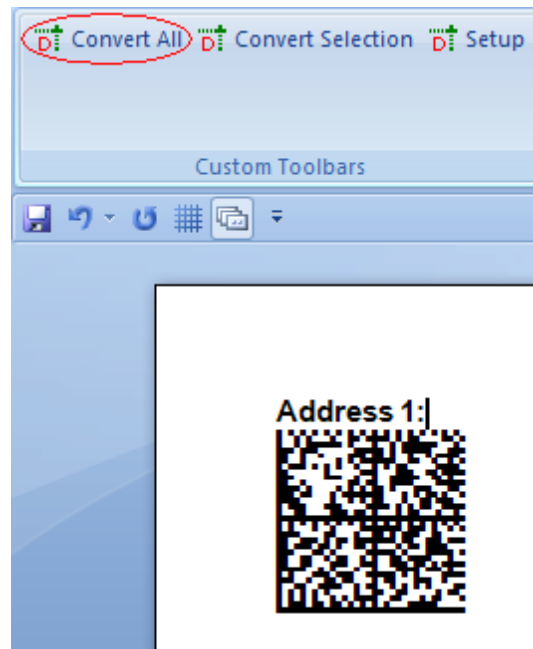
1. Enter a few string sections, surround those sections which will be converted to barcodes with the "<" and ">" characters, highlight those sections.



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

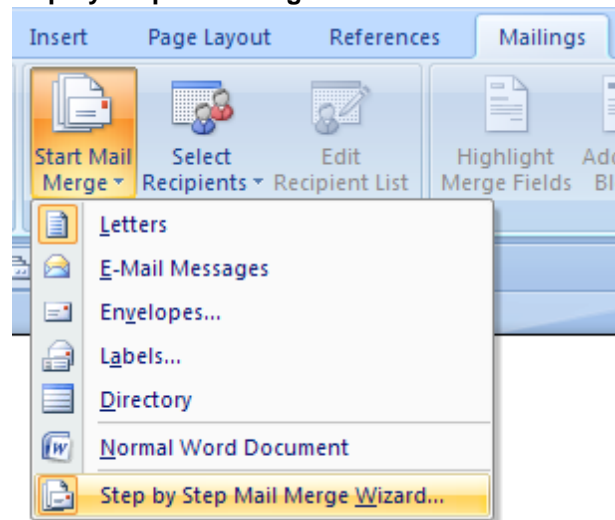


3. Click on "**Add-Ins**", then click on "**Convert All**" to create barcodes for the string sections surrounded with the "<" and ">" characters.

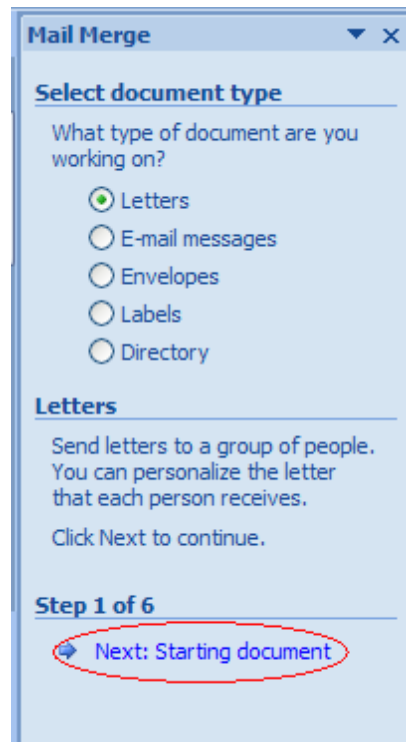


4.1.4 Mail Merge

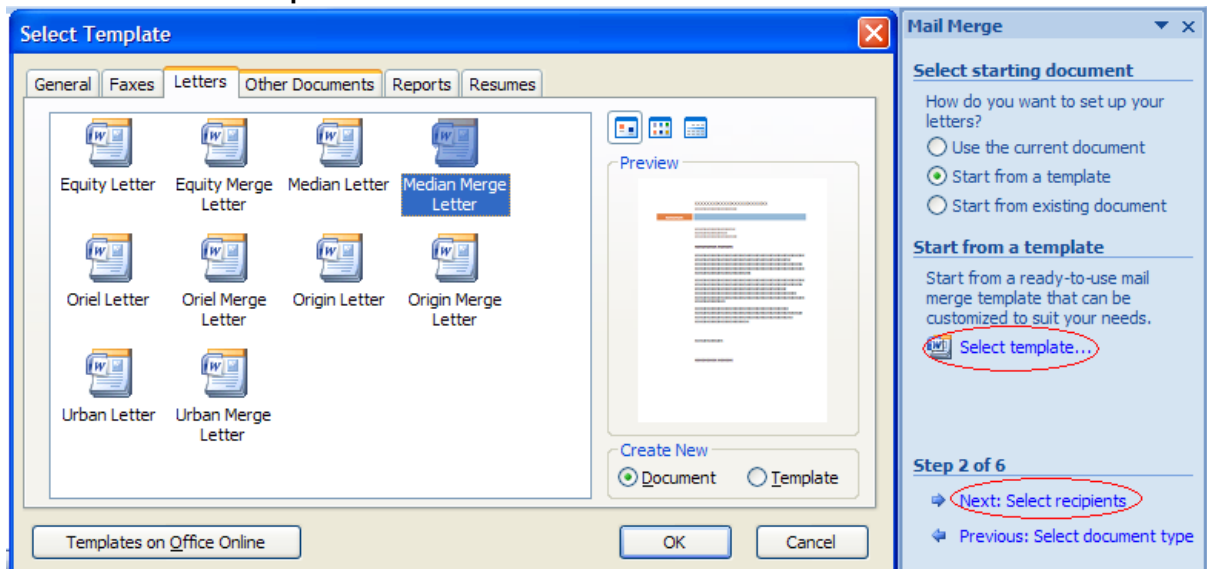
1. Click on "**Mailings**", then click on "**Start Mail Merge**". A drop-down list appears as shown below, select the last option "**Step by Step Mail Merge Wizard**".



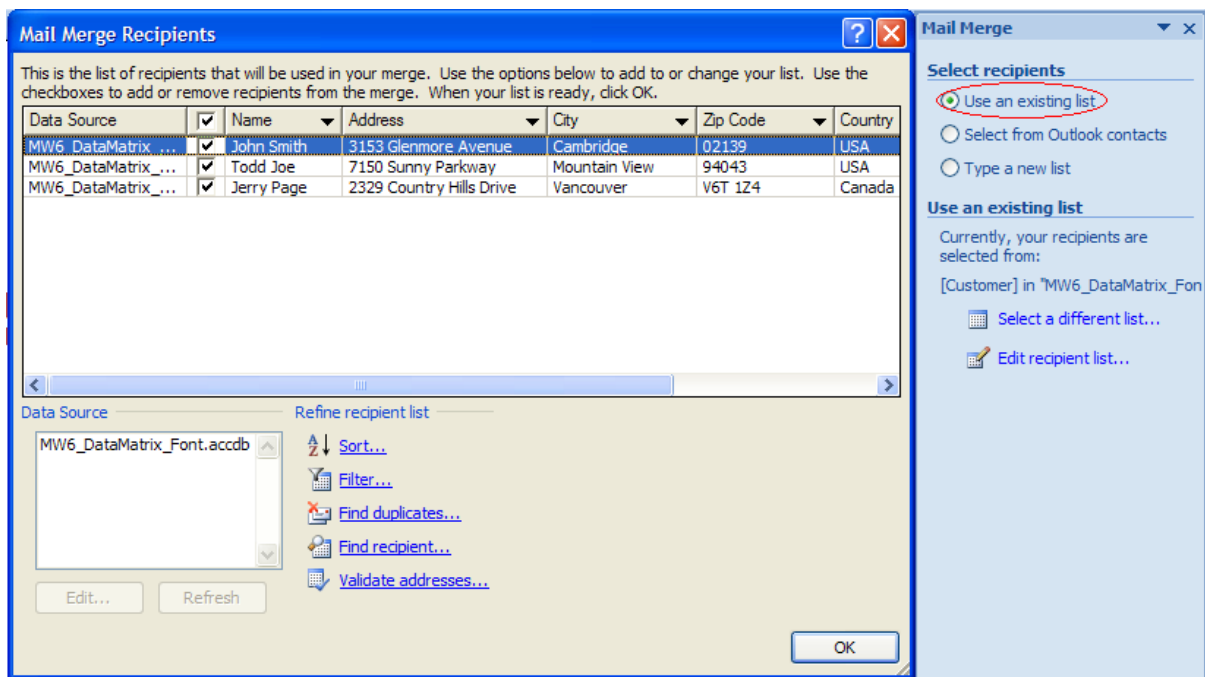
2. Select a document type and click on "**Next: Starting document**".



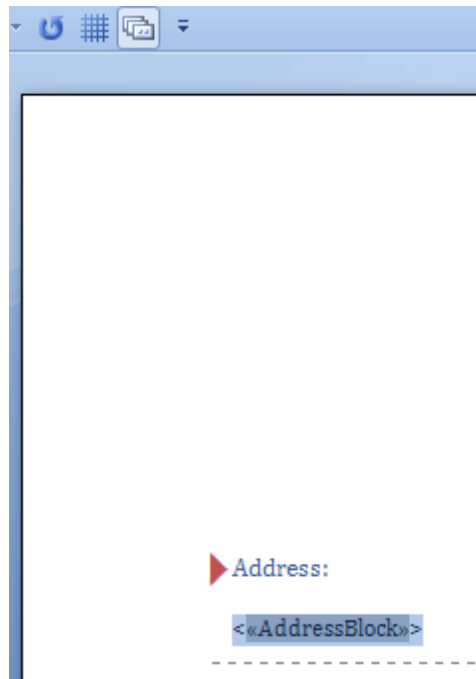
3. Click on **"Start from a template"**, then click on the link **"Select template"**, choose a template, click on **"Next: Select recipients"**.



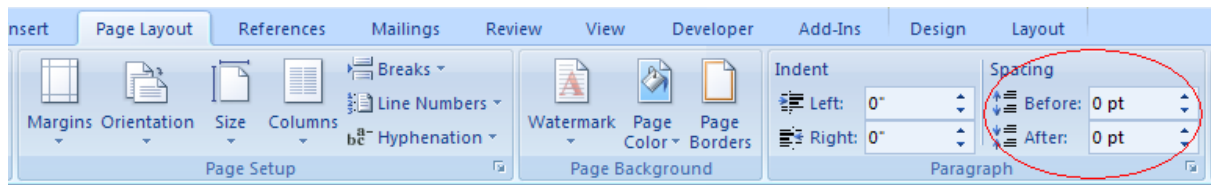
4. Select **"Use an existing list"** and click on **"Browser"** link, choose **"MW6_DataMatrix_Font.accdb"** database as an existing list, click **"Next: Write your letter"**.



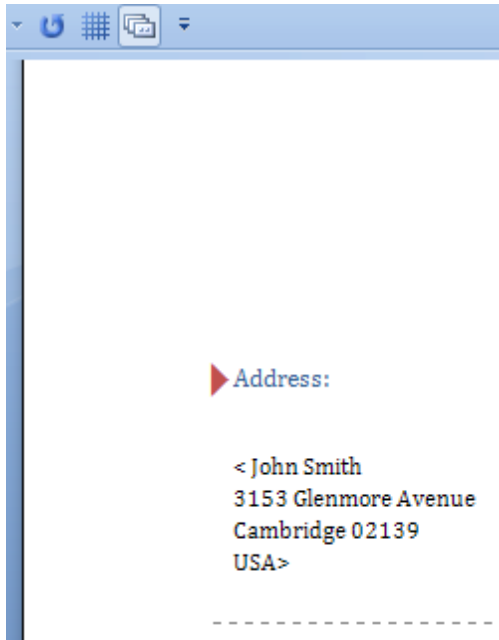
5. Surround the section which will be converted to DataMatrix barcode with "<" and ">" characters and highlight it.



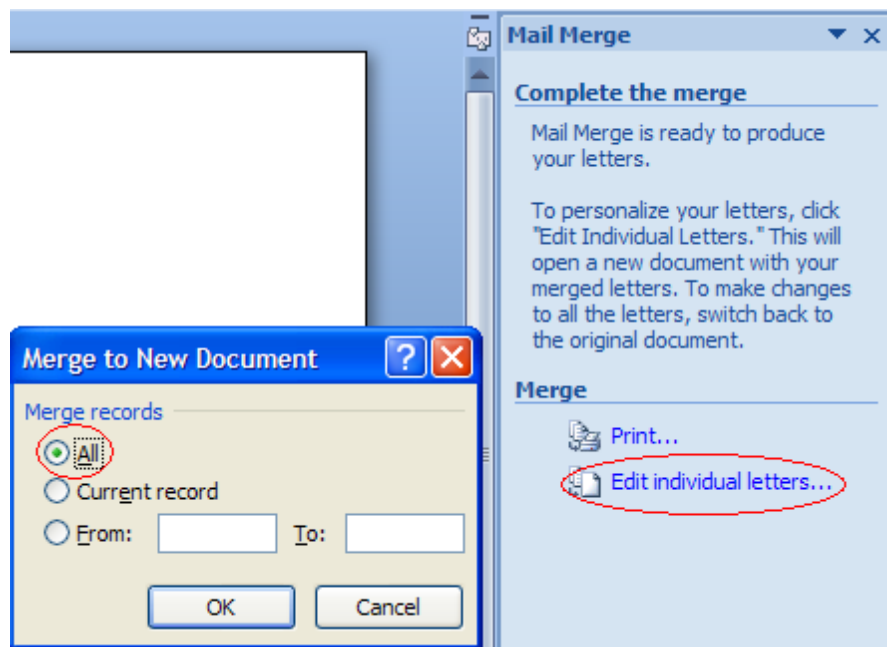
6. Click on "Page Layout", change Spacing "Before" and "After" to 0.



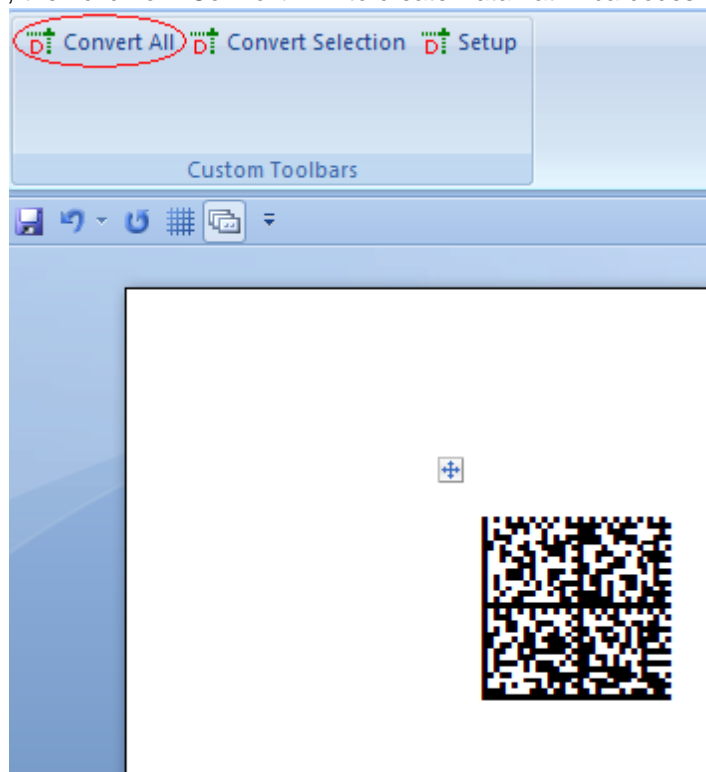
7. Click on "**Next: Preview your letters**", then click on "**Next: Complete the merge**".



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

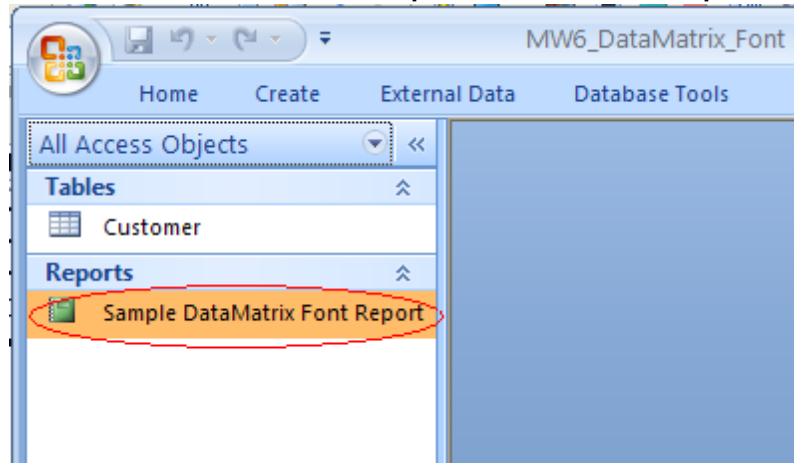


9. Click on "Add-Ins", then click on "Convert All" to create DataMatrix barcodes.

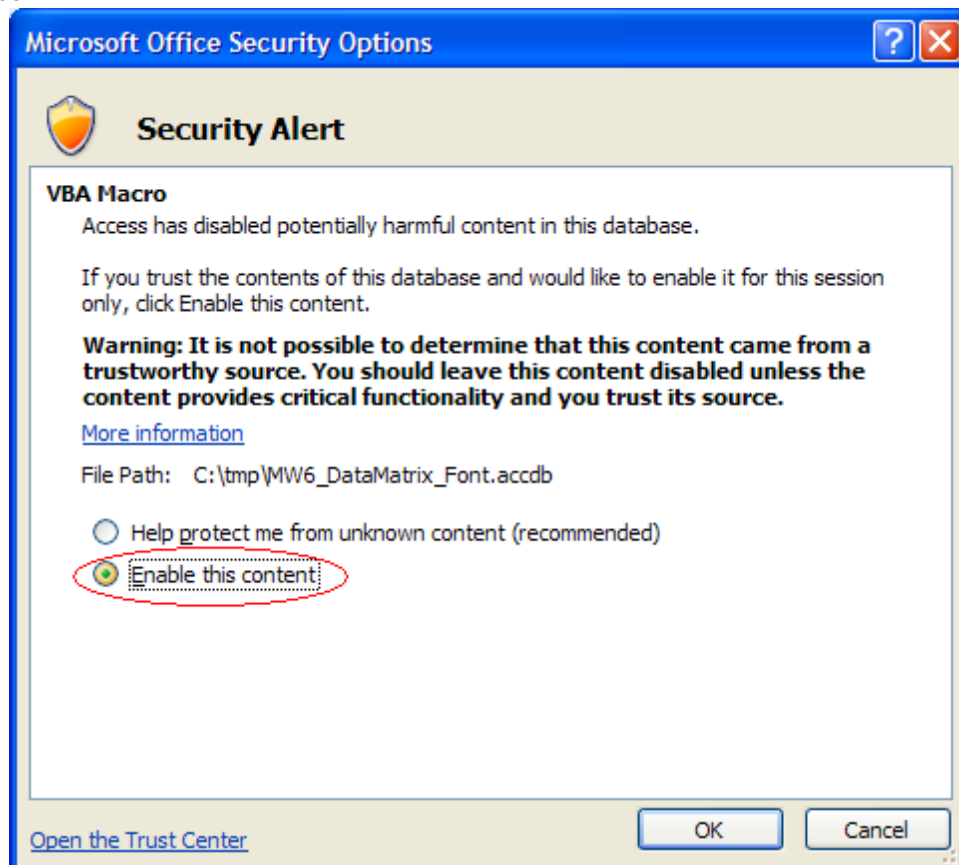


4.2 Access

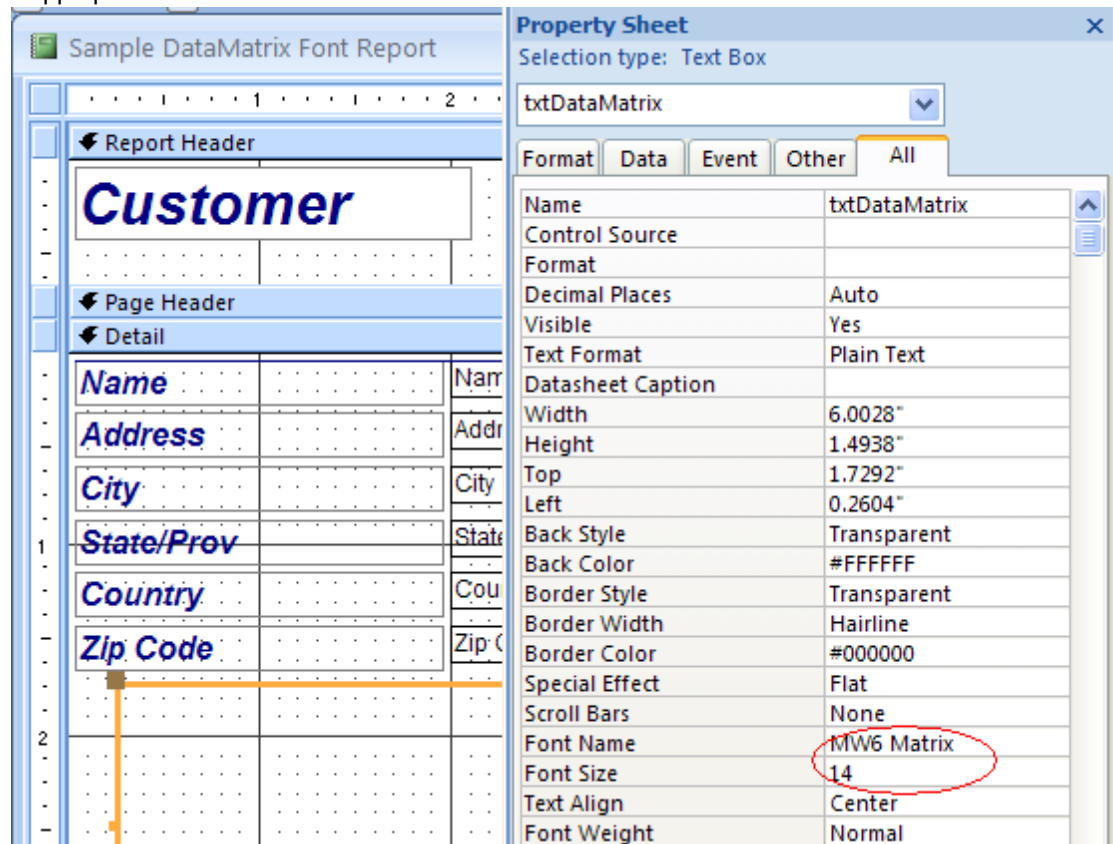
1. Copy "DataMatrixFont.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
2. Open MW6_DataMatrix_Font.accdb, select "**Sample DataMatrix Font Report**".



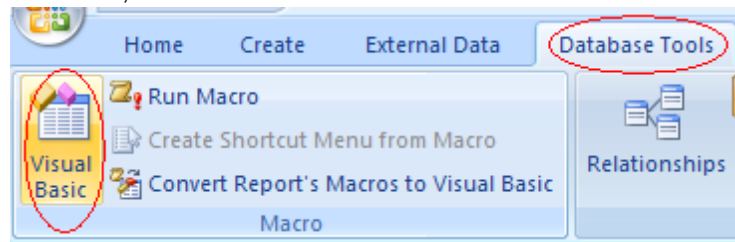
3. If you see "**Security Warning, Certain content in the database has been disabled**", click on "**Options**" to open "**Microsoft Office Security Options**" dialog, toggle on "**Enable this content**" check box.



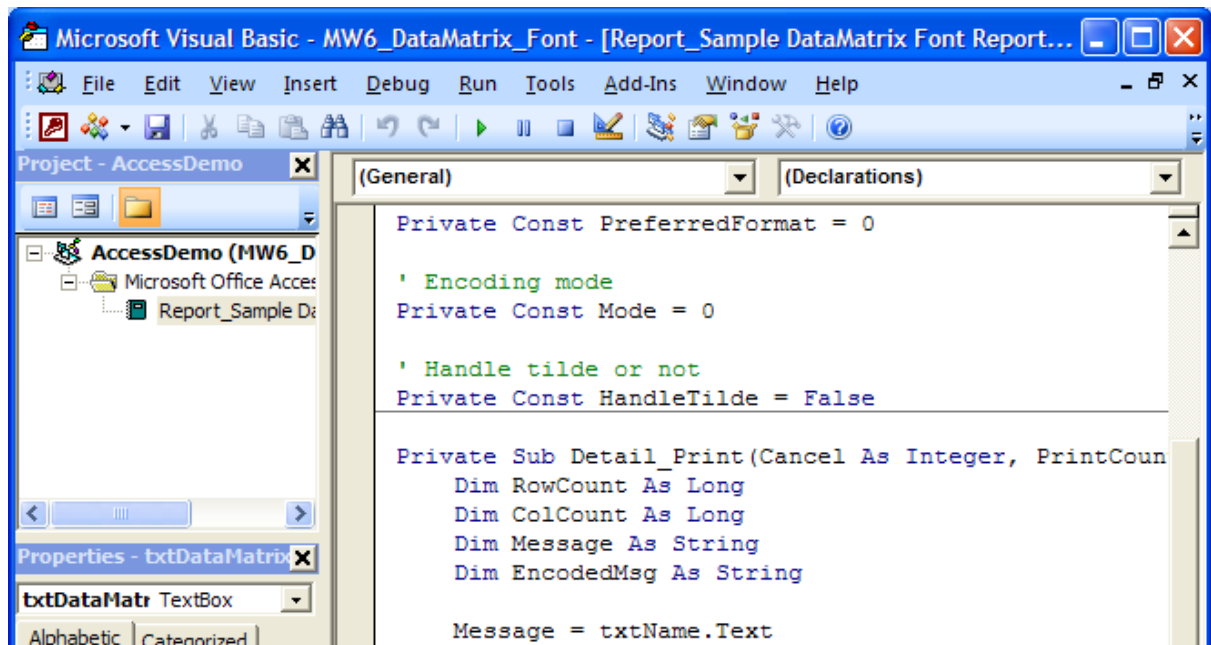
- Click on "**Design View**", insert a Text Box into the report, set its font name to "MW6 Matrix", choose an appropriate font size.



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



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



7. Click on "**Preview**" to view DataMatrix barcodes.

Customer

Customer

Name	John Smith
Address	3153 Glenmore Avenue
City	Cambridge
State/Prov	MA
Country	USA
Zip Code	02139



5 Office 2000 & 2003

5.1 Word Demo

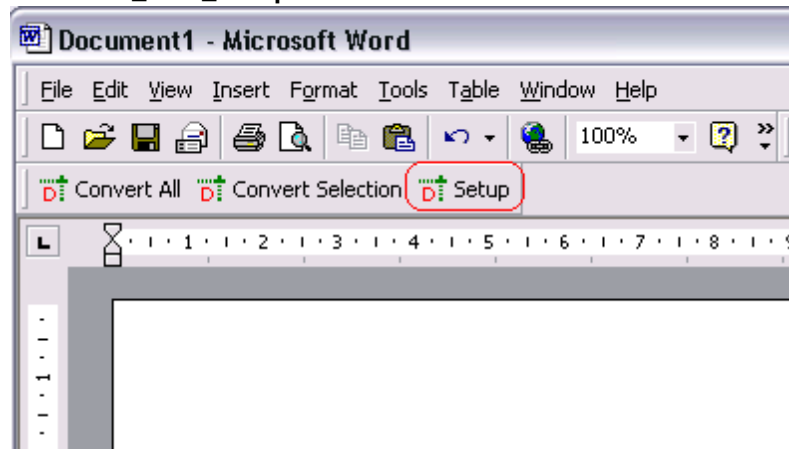
5.1.1 Install Template File

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

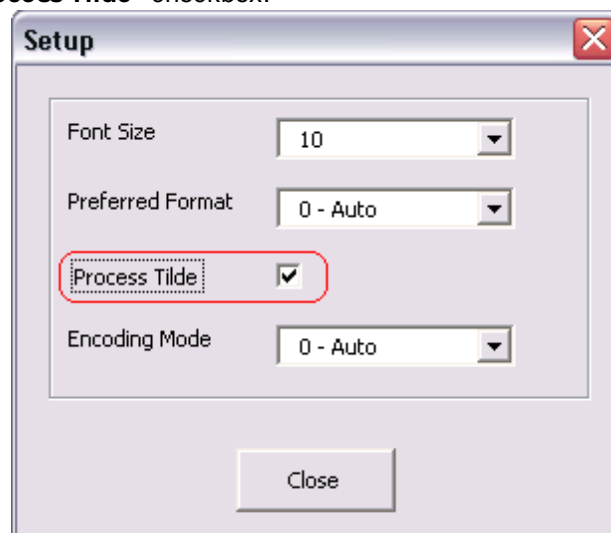
OS	Location
Windows Vista and above	"C:\Users\ <user name="">\AppData\Roaming\Microsoft\Word\STARTUP"</user>
Windows 2000/XP	"C:\Documents and Settings\ <user name="">\Application Data\Microsoft\Word\STARTUP"</user>
Windows NT4	"C:\Winnt\Profiles\ <user name="">\Application Data\Microsoft\Word\STARTUP"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

Office 2000/97: "C:\Program Files\Microsoft Office\Office\STARTUP"

- Copy MW6_DataMatrix_Font.dot to this folder.
- Copy "DataMatrixFont.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
- 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_DataMatrix_Font_Setup"** and run it.

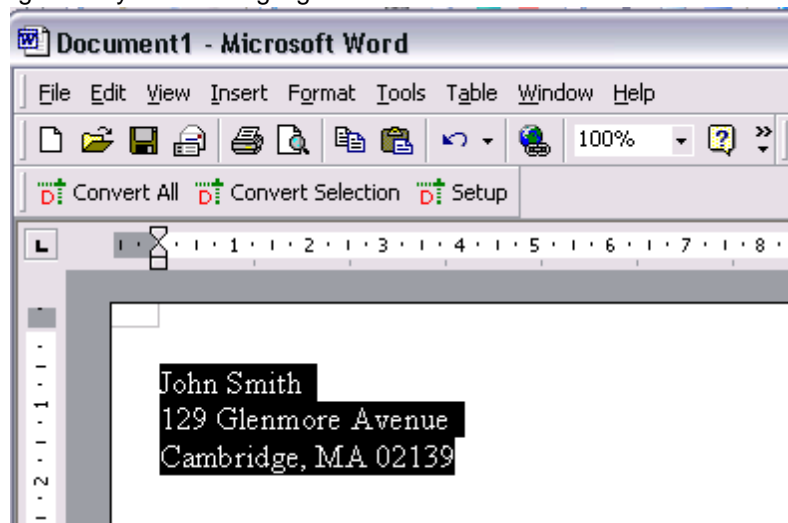


- Change the configurations for DataMatrix format, if you want to encode a string with non-printable characters, click on **"Process Tilde"** checkbox.

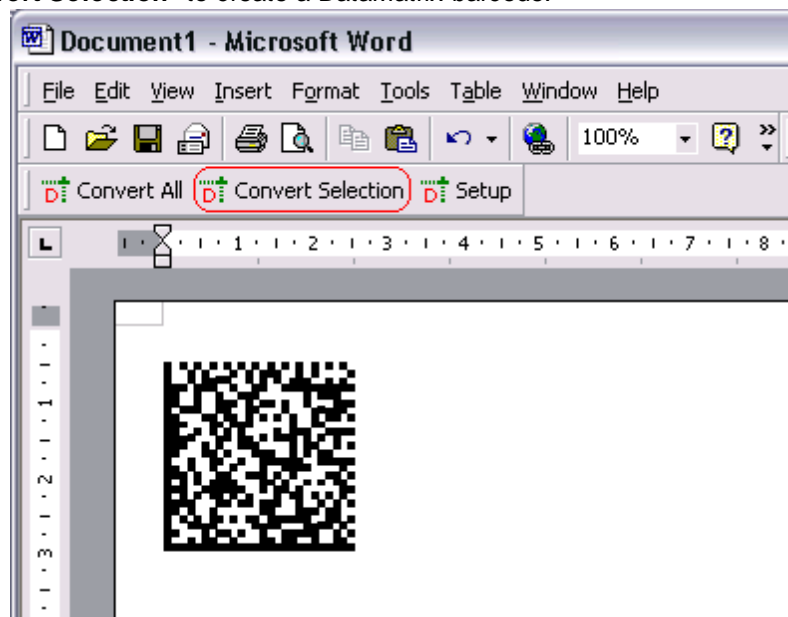


5.1.2 Create Single Barcode

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

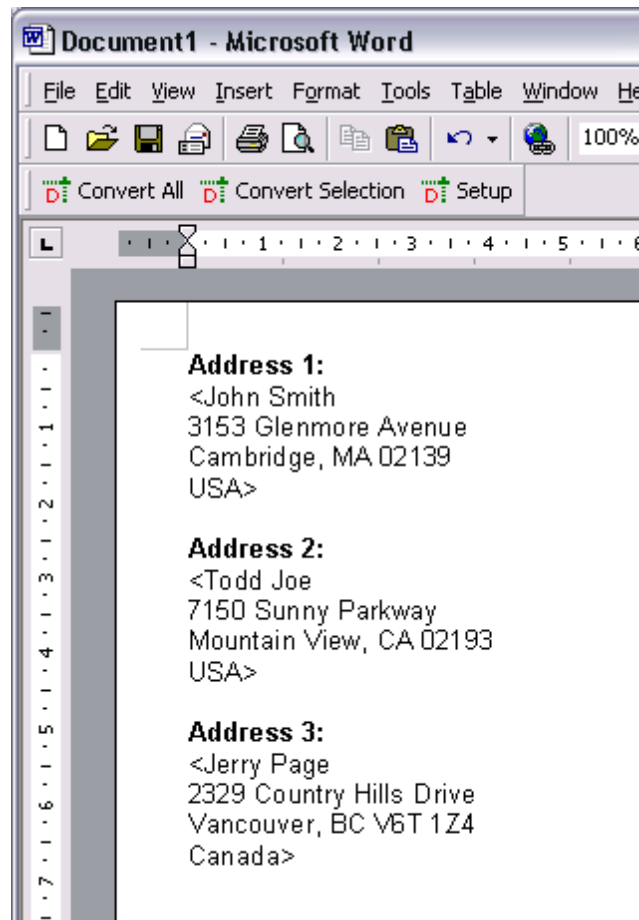


2. Click on "Convert Selection" to create a DataMatrix barcode.

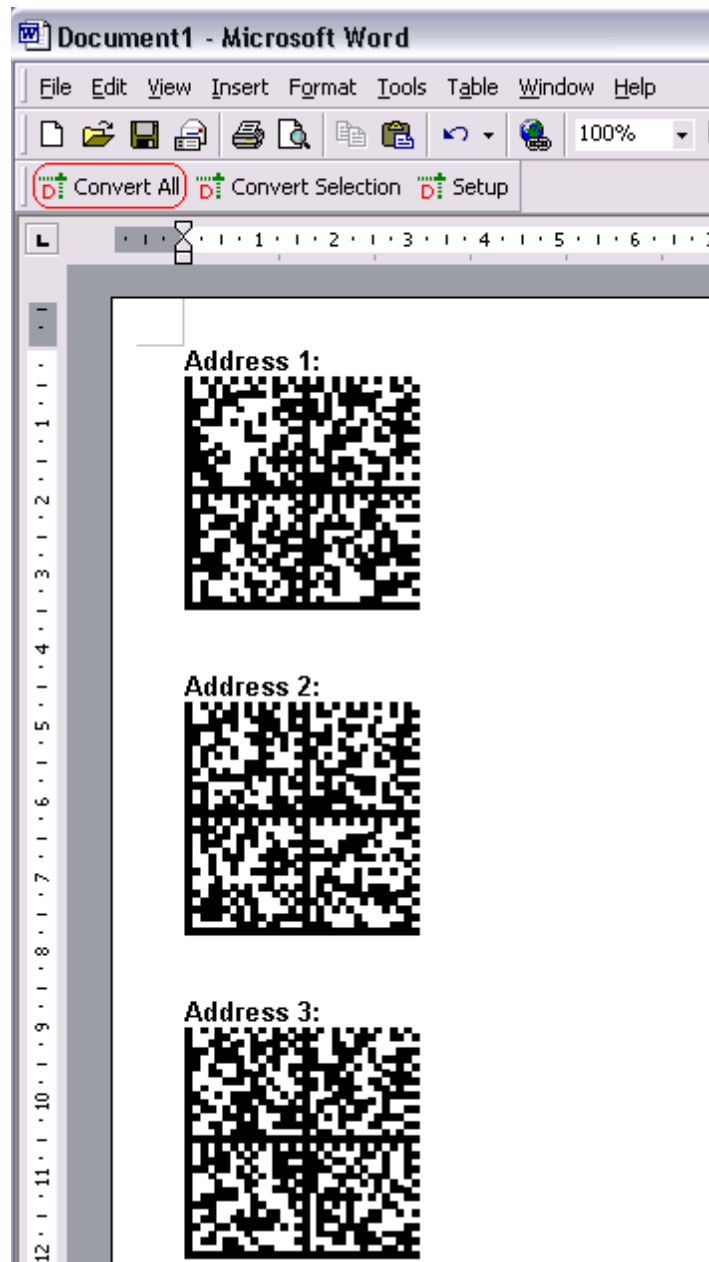


5.1.3 Create Multiple Barcodes

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

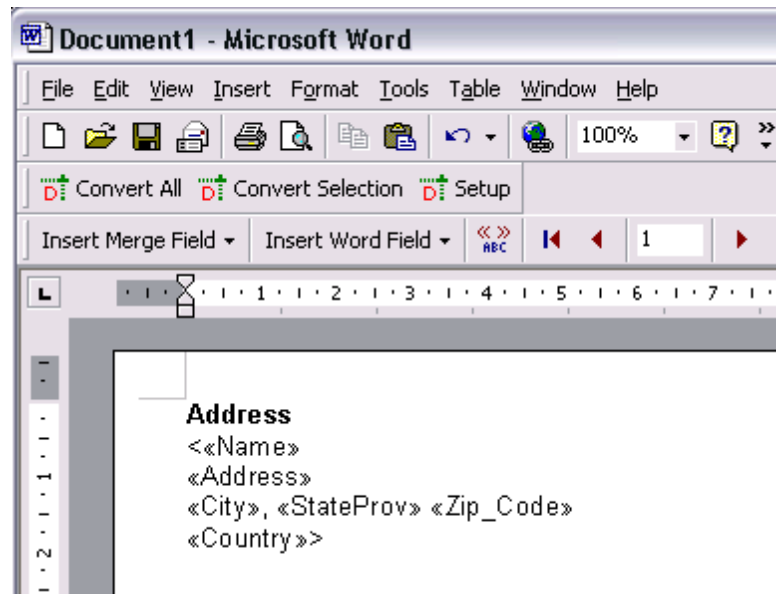


2. Click on "**Convert All**" to create the barcodes for paragraphs surrounded with the "<" and ">" characters.



5.1.4 Mail Merge

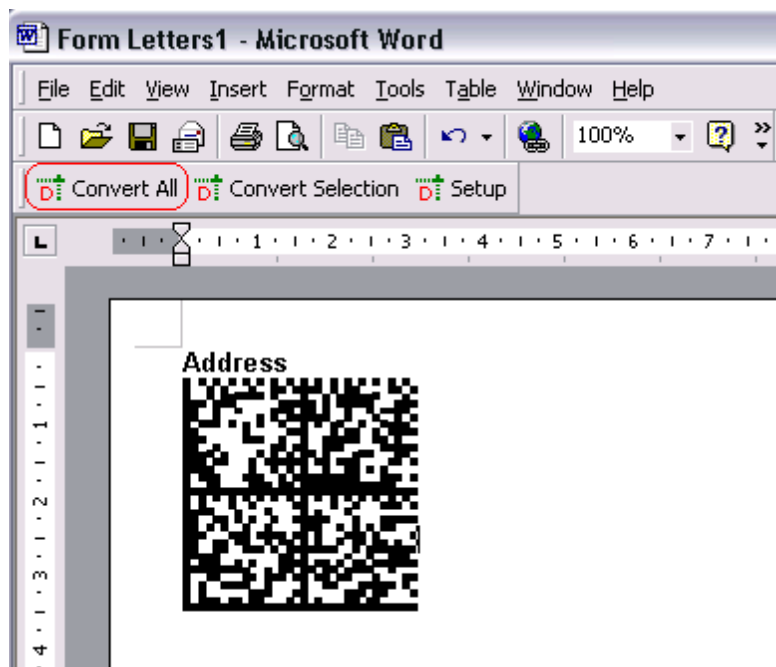
1. In Mail Merge, choose MW6_DataMatrix_Font.mdb as Data Source, surround the paragraph which will be converted to DataMatrix barcode with the "<" and ">" characters.



2. Click on "Merge ..."

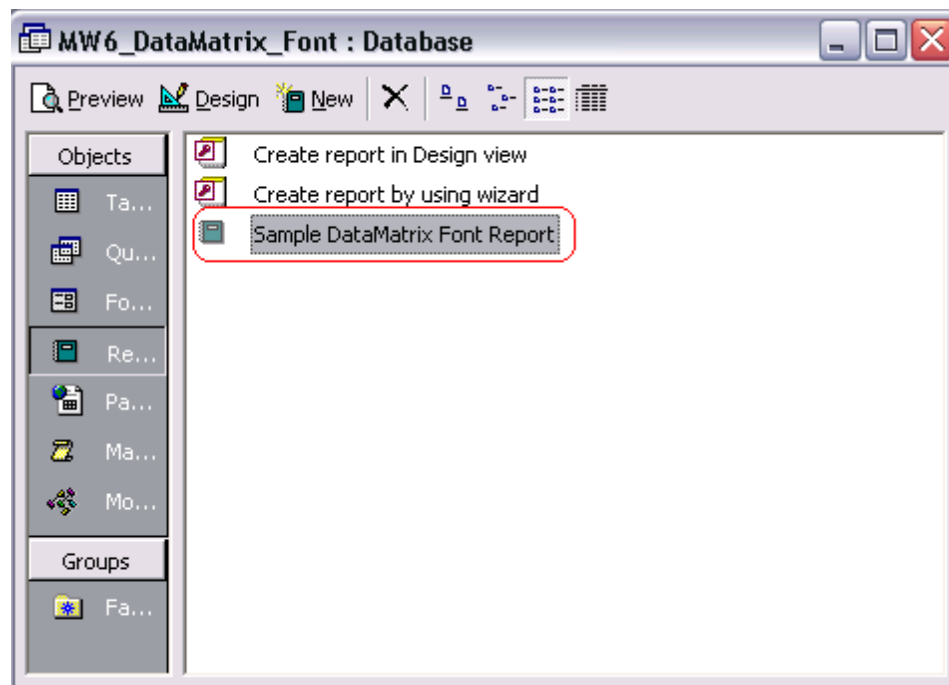


3. Click on "Convert All" to create DataMatrix barcodes for the paragraphs surrounded with the "<" and ">" characters.

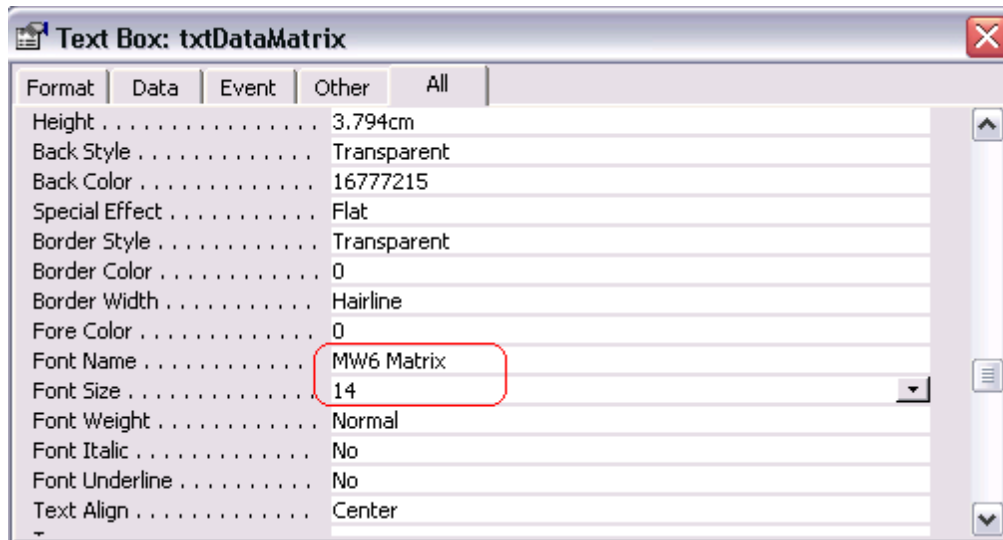


5.2 Access Demo

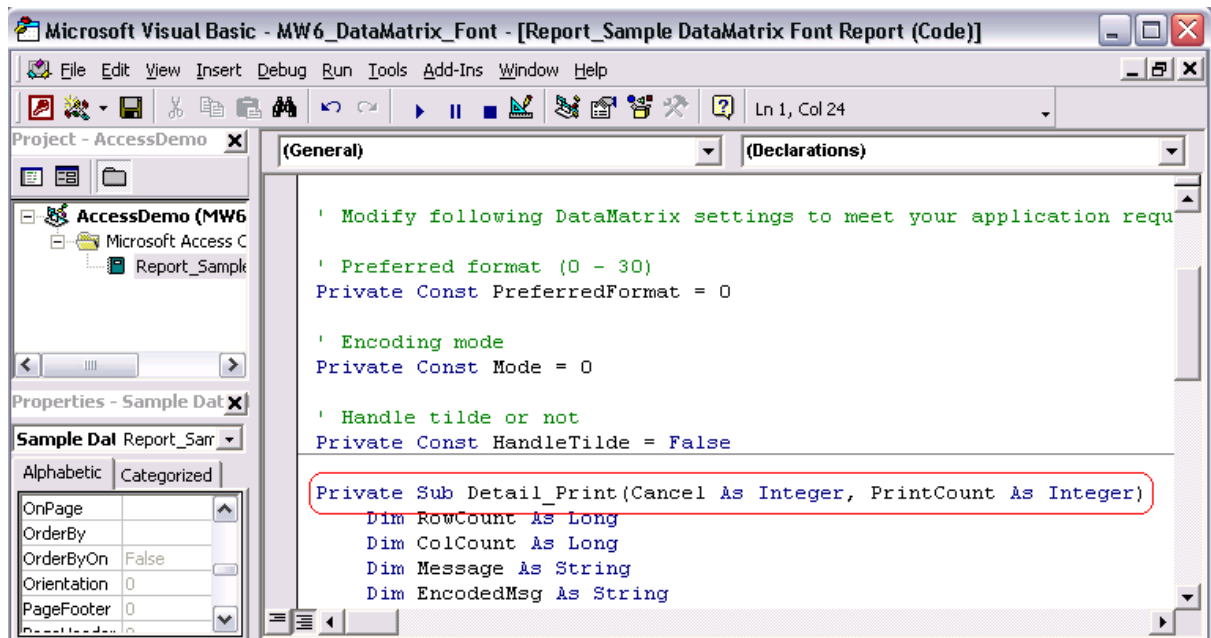
1. Copy "DataMatrixFont.dll" to the windows 32-bit system folder (e.g. "C:\winnt\system32" or "C:\windows\system32").
2. Open MW6_DataMatrix_Font.mdb, select "Sample DataMatrix Font Report".



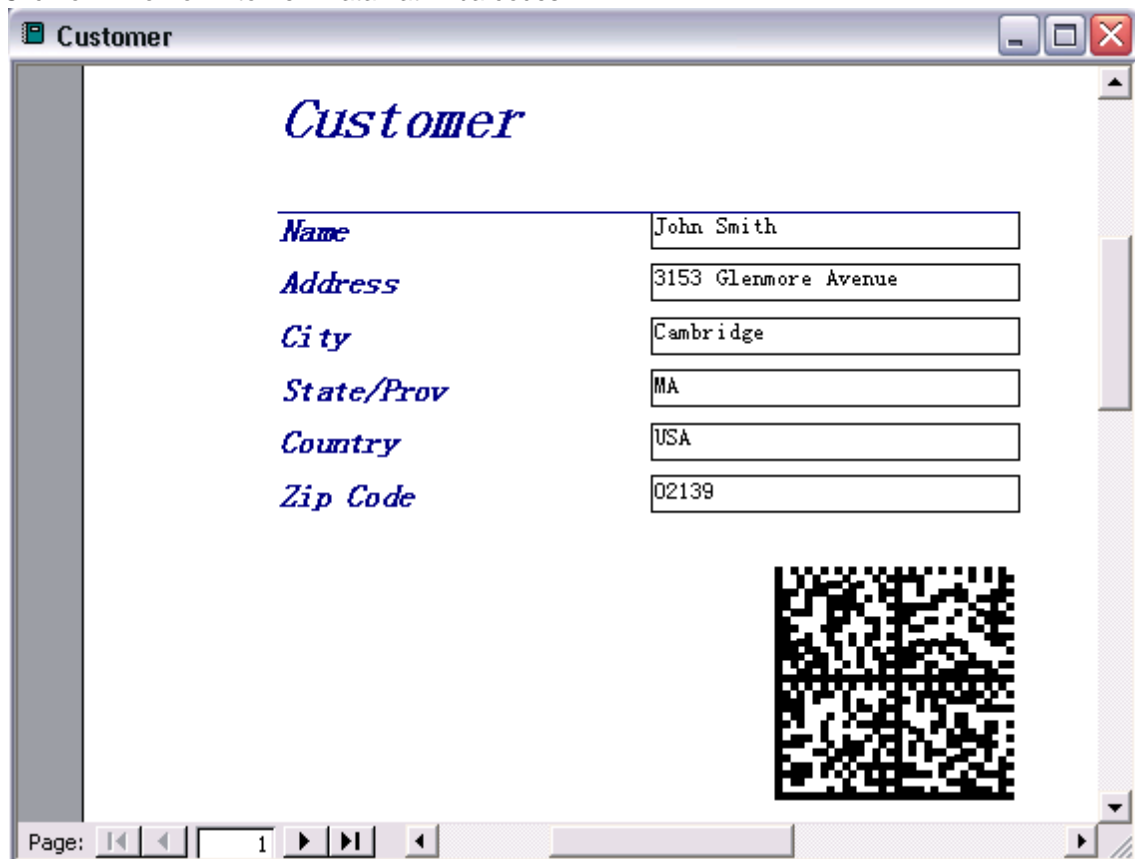
3. Click on "**Design**", insert a Text Box into the report, set its font name to "MW6 Matrix", choose an appropriate font size.



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 DataMatrix barcodes.



6 DataMatrix Formats and Data Capacity

The following table lists all formats of DataMatrix barcode and their data capacity, keep in mind, the trial version encoders (Win32 DLL encoder and .NET encoder) append "MW6 Demo" to the string encoded with DataMatrix.

Value	Description	Data Capacity		
		Numeric	Alphanumeric	Byte
0	Auto format			
1	10 X 10 format	6	3	1
2	12 X 12 format	10	6	3
3	14 X 14 format	16	10	6
4	16 X 16 format	24	16	10
5	18 X 18 format	36	25	16
6	20 X 20 format	44	31	20
7	22 X 22 format	60	43	28
8	24 X 24 format	72	52	34
9	26 X 26 format	88	64	42
10	32 X 32 format	124	91	60
11	36 X 36 format	172	127	84
12	40 X 40 format	228	169	112
13	44 X 44 format	288	214	142
14	48 X 48 format	348	259	172
15	52 X 52 format	408	304	202
16	64 X 64 format	560	418	278
17	72 X 72 format	736	550	366
18	80 X 80 format	912	682	454
19	88 X 88 format	1152	862	574
20	96 X 96 format	1392	1042	694
21	104 X 104 format	1632	1222	814
22	120 X 120 format	2100	1573	1048
23	132 X 132 format	2608	1954	1302
24	144 X 144 format	3116	2335	1556
25	8 X 18 format	10	6	3
26	8 X 32 format	20	13	8
27	12 X 26 format	32	22	14
28	12 X 36 format	44	31	20
29	16 X 36 format	64	46	30
30	16 X 48 format	98	72	47

7 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.
