

ELFRING FONTS

UPC BAR CODES

This package includes five UPC-A and five UPC-E bar code fonts in both TrueType and PostScript formats, a Windows utility, BarUPC, which helps you make bar codes, and Visual Basic macros for Excel, Access, and Word. This package also includes a copy of our eBook, “The Truth About UPC Bar Codes”. Each UPC-A bar code encodes a 12 digit number (11 data + 1 digit checksum), while UPC-E bar codes encode 6 digit numbers. Both bar code fonts also support optional 2 or 5 digit supplemental codes.

UPC-A bar codes are complicated and you can not simply type numbers in the bar code font. However, they can be calculated by hand. UPC-E codes are more involved and require our utility, our Visual Basic functions, or a program of your own to implement. All UPC bar codes require a checksum (*check digit*). Each of these UPC bar codes comes in five different aspect ratios, allowing you to print bar codes with the same width but at different heights.

Table 1














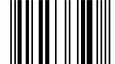

Typeface Name	UPCA a	UPCA b	UPCA c	UPCA d	UPCA e
UPC-A	 7 14569 00173 5	 7 14569 00173 5	 7 14569 00173 5	 7 14569 00173 5	 7 14569 00173 5
Typeface Name	UPC Plain a	UPC Plain b	UPC Plain c	UPC Plain d	UPC Plain e
UPC-A					

Table 2

Typeface Name	UPCE a	UPCE b	UPCE c	UPCE d	UPCE e
UPC-E	 145695 9	 145695 9	 145695 9	 145695 9	 145695 9

INSTALLATION

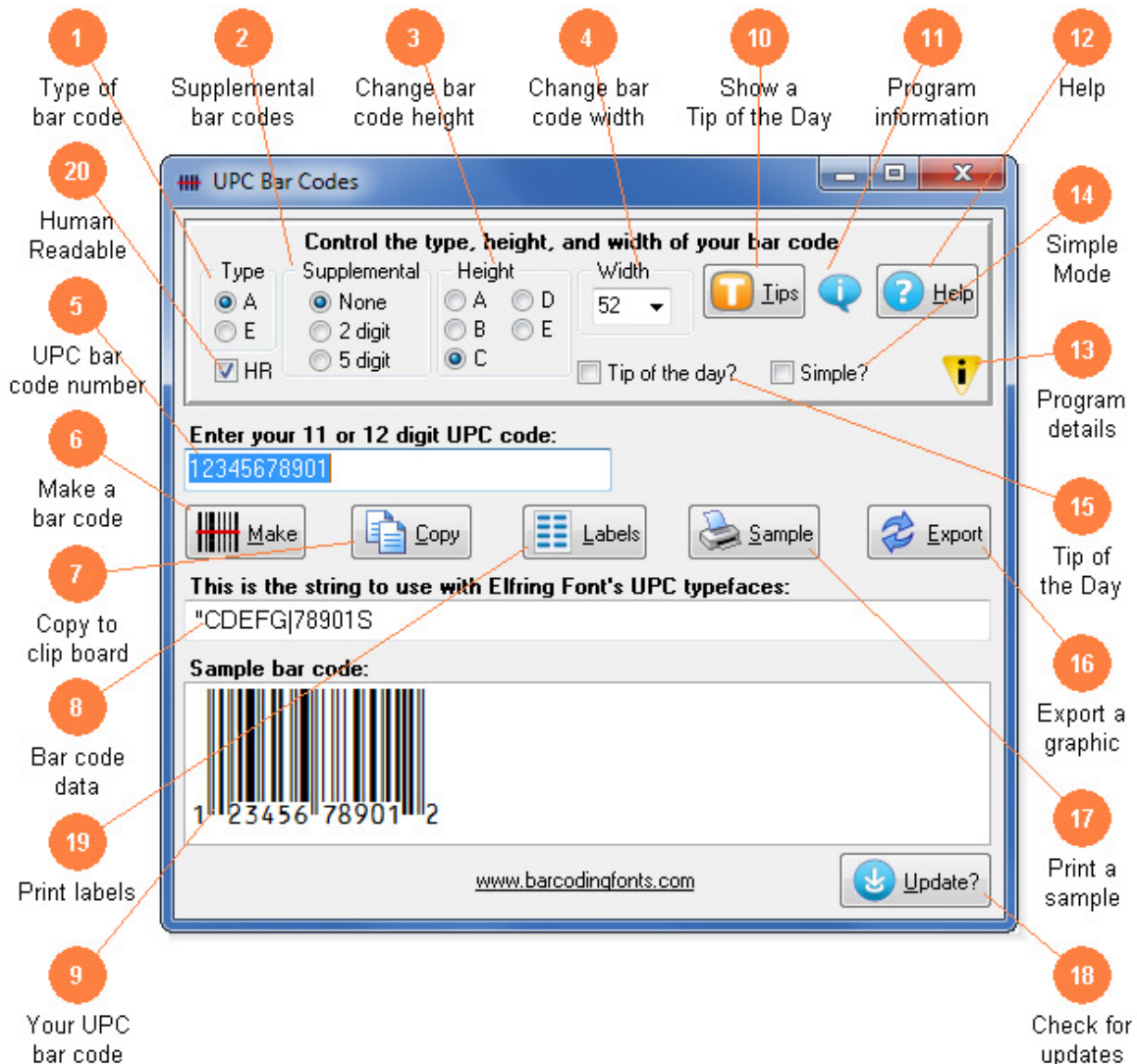
Our BarcdUPC.exe program will install this package and font set in your copy of Windows. The fonts will be installed into the proper Windows font folder automatically. Our bar code utility program, BarUPC.exe, this documentation, and matching files will also be installed. The bar code fonts will appear in all Windows font menus. You can select any one of the bar code just as you normally select any other font in your

application. For help in building bar codes, run the BarUPC.exe utility program. To access the utility program or the documentation, click on Start, Programs, UPC Bar Codes, and select the program or documentation item you want to view.

BARUPC UTILITY PROGRAM

The BarUPC utility program was automatically added to your system when you ran our install program. You can access the BarUPC utility from the Desktop icon, or by clicking on: Start, Programs, Elfring UPC Bar Codes, Utility program. This program lets you enter your data and build a bar code. You can print a sample page of the resulting bar code, transfer that bar code to another Windows program, or export the bar code as a bmp, gif, jpg, or png file.

This utility converts your UPC bar code numbers into actual UPC bar codes. Use this utility to build UPC bar codes and then to copy and paste them into other Windows programs for printing. Bar codes can be copied into Word for label printing, or into a desktop publishing program to display a bar code on your packaging. Or build your bar code and export it as a graphic file in bmp, gif, or jpg format. These bar code fonts are compatible with virtually any Windows program. For programs like Excel and Access see the documentation section on Visual Basic macros that follows.



- (1) **Type of bar code.** Select the type of UPC bar code to create. UPC-A bar codes are 12 digits long and are used on virtually every retail product sold in North America. UPC-E bar codes are smaller shortened versions of UPC-A bar codes. All UPC-E bar code numbers must be specially assigned by the Uniform Code Council.
- (2) **Supplemental bar codes.** Add a 2 or 5 digit supplemental bar code after your UPC bar code. This adds extra digits of information to your bar code but should not be used unless it is required.
- (3) **Change bar code height.** Modify the height of a UPC bar code without changing the width. UPC bar codes can range from .25 to 1 inch in height.
- (4) **Change bar code width.** Scale the bar code width up or down by increasing or decreasing the point size. UPC-A bar codes should be about 1.5 inches wide.
- (5) **UPC bar code number.** Type in your 11 digit UPC number here. Our program will calculate the checksum for you, or verify yours if you enter all 12 digits.
- (6) **Make a bar code.** Click this button to make a UPC bar code from your number.
- (7) **Copy to clip board.** Copy your completed UPC bar code to the Windows clip board. (*The Windows clip board is invisible- don't worry about where it is or how it works.*) You can then paste this bar code directly into virtually any Windows program by pressing Ctrl-V or using the Edit, Paste menu in that program.
- (8) **Display converted bar code data.** This is the string of characters to use with our bar code fonts. Note you can not just type in the UPC bar code fonts. Your UPC bar code number is converted into a data string and this data string must be used with our UPC bar code fonts.
- (9) **This is your UPC bar code.** This is a sample of what your bar code will look like. Note that Windows screen resolution is fairly low, so the bar code displayed may not exactly match what will be printed.
- (10) **Show a Tip of the Day.** Display the next program Tip of the Day. From this display you can page forwards or backwards through the list of all available tips.
- (11) **Program information.** Display program and contact information including the version number of this software.
- (12) **Help.** Launch the Windows Help application. The Help form will give you detailed answers to most questions about the program.
- (13) **Program details.** Display this help screen.
- (14) **Simple Mode.** Toggle between Simple Mode (UPC-A with no options) and full mode.
- (15) **Tip of the Day.** Display the next *Tip of the Day* on each start up when this is checked. Uncheck the box to stop this start up behavior.
- (16) **Export a Graphic.** Export the bar code you just created as a graphic image in bmp, gif, or jpg format.

- (17) **Print a sample.** Print a sample page of your UPC bar code to see just what it will look like. This sample page shows you your UPC bar code in a number of different sizes.
- (18) **Check for updates.** Check the web for a program update. You must have an active internet connection running. A web page will be displayed with information about the current version of this program and whether or not you need to update.
- (19) **Labels.** Print a sheet of labels in your word processor. Select a label template and open it with your word processor, then paste your bar code on to any label.
- (20) **Human Readable.** Create bar codes with human readable numbers below when checked.

USER ACCESSIBLE FILES

In Windows 10, 8, 7, & Vista users are not allowed to access any files stored in the Programs folder. The Visual Basic macros, label templates, and sample Excel spread sheet are placed into a folder named “BarUPC” in each user’s Documents area. You can find these files as follows:

- Windows XP: My Documents\BarUPC
- Vista: Libraries\Documents\ BarUPC
- Windows 7: Libraries\Documents\ BarUPC
- Windows 8: Libraries\Documents\ BarUPC
- Windows 10: Libraries\Documents\ BarUPC

UPC BAR CODES

UPC-A bar codes are used to identify all retail products in the US and Canada. You must purchase a number assignment from www.GS1.org (expensive!), www.mybarcodestore.com, or some other source to make this bar code. For details on how to get bar code numbers assignments see the eBook: “The Truth About UPC Bar Codes at” www.mybarcodestore.com. This UPC bar code set includes TrueType and PostScript fonts in five separate variations for each of the two UPC bar code types (*UPC-A and UPC-E*): UPC-A a, UPC-A b, UPC-A c, UPC-A d, UPC-A e, and UPC-E a, UPC-E b, UPC-E c, UPC-E d and UPC-E e,. Each bar code font can print either the UPC-A or the UPC-E bar codes, plus an optional 2 or 5 digit supplemental code. If you just want to print a few bar codes in your existing Windows programs, try our BarUPC utility program. (*Skip ahead to the BarUPC section.*) If you want to use Access, Excel, or Word to print bar codes, see the section on our Visual Basic functions. UPC-A bar codes always have a Start character, 5 data characters, a guard bar, 5 more data characters, and a checksum / Stop character.



Start/Left/Right/Checksum 7 14569 00173 5
 Add guard bar 7 14569 | 00173 5
 Map characters (BEFGJ | 00173 V
 Complete string (BEFGJ|00173V



UPC-A checksums are calculated as follows. Take the 11 digit UPC number and assign the right-most digit an “odd” value. Next alternate even/odd assignments for the rest of the digits. Add all the odd position digits and multiply that sum by 3. Add to that first sum the sum of all the even position digits. The checksum is the smallest number that can be added to that sum to make it an even multiple of 10.

Data: 7 1 4 5 6 9 0 0 1 7 3
 Position: O E O E O E O E O E O
 Odd sum: 7 + 4 + 6 + 0 + 1 + 3 = 21, 21 * 3 = 63
 Even sum: 1 + 5 + 9 + 0 + 7 = 22
 Sum odd plus even: 63 + 22 = 85
 Checksum: 5 (85 + 5 = 90, which is an even multiple of 10)

UPC-E

UPC-E bar codes are an abbreviated form of UPC-A codes. UPC-E bar codes are designed for use in small areas. To create a UPC-E bar code, start with a UPC-A bar code number. Your UPC-A bar code *must* start with a zero, and it *must* have at least 4 zeros in the rest of the bar code data. UPC-E bar codes appear to be 6 characters long, with a checksum character encoded as a parity pattern in the bar code data. Table 4 identifies which ASCII characters represent the individual numbers 0-9 for each bar code pattern. Our bar code fonts also include representations of Short/Odd and Even patterns for the 2 and 5 digit supplemental codes.

Table 4

← UPC-E Codes →						← 2 & 5 digit supplemental →			
Label		Left / Odd		Right / Even		Short Odd		Short Even	
Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position
0	0	0	A	0	Q	0	a	0	q
1	1	1	B	1	R	1	b	1	r
2	2	2	C	2	S	2	c	2	s
3	3	3	D	3	T	3	d	3	t
4	4	4	E	4	U	4	e	4	u
5	5	5	F	5	V	5	f	5	v
6	6	6	G	6	W	6	g	6	w
7	7	7	H	7	X	7	h	7	x
8	8	8	I	8	Y	8	i	8	y
9	9	9	J	9	Z	9	j	9	z
Start	[Stop]			Start 5	/	Space 5	.

UPC-E bar codes always begin with the *Start Code* [, have 6 data characters using a mixture of the Left/Odd (*numbers 0-9 are represented by ASCII characters A-J*), and Right/Even characters (*numbers*

0-9 are represented by ASCII characters Q-Z), and a Stop Code J. So to make a UPC-E bar code from the UPC-A data “07110000455”, we first calculate a checksum digit (7). Then you use the following rules to determine which digits from the UPC-A code go into the UPC-E code. Once you have determined which digits go into the UPC-E code, you use the checksum to generate a parity pattern among those digits. See Table 5 for the checksum to parity pattern conversion.

UPC-A code
0 71100 00455 7

0 5 digit manufacturer number 5 digit item number checksum

- 1) If the 5 digit manufacturer’s number ends in 000, 100, or 200 the 5 digit item number may range from 00000 to 00999. The 6 characters in the UPC-E code are: the first 2 characters of the manufacturer’s number, the last 3 characters of the item number, followed by the 3rd character of the manufacturer’s number.
- 2) If the 5 digit manufacturer’s number ends in 300, 400, 500, 600, 700, 800, or 900 the 5 digit item number may range from 00000 to 00099. The 6 characters in the UPC-E code are: the first 3 characters of the manufacturer’s number, the last 2 characters of the item number, followed by the character “3”.
- 3) If the 5 digit manufacturer’s number ends in 10, 20, 30, 40, 50, 60, 70, 80, or 90 the 5 digit item number may range from 00000 to 00009. The 6 characters in the UPC-E code are: the first 4 characters of the manufacturer’s number, the last character of the item number, followed by the character “4”.
- 4) If the 5 digit manufacturer’s number does not end in zero, the 5 digit item number may range from 00005 to 00009. The 6 characters in the UPC-E code are: all 5 characters of the manufacturer’s number and the last character of the item number.

Table 5


Checksum Value	Character Parity vs Position					
	1	2	3	4	5	6
0	E	E	E	O	O	O
1	E	E	O	E	O	O
2	E	E	O	O	E	O
3	E	E	O	O	O	E
4	E	O	E	E	O	O
5	E	O	O	E	E	O
6	E	O	O	O	E	E
7	E	O	E	O	E	O
8	E	O	E	O	O	E
9	E	O	O	E	O	E

UPC-E Example 1

Data: 07110000455
 Manufacturer number 71100
 Item number 00455

Since the last 3 characters of the manufacturer’s number are 100, the 6 digits of the UPC-E code are:


First 2 characters of manufacturer’s number 71

Last 3 characters of the item number	455	
Third character of manufacturer's number	1	
Checksum (from UPC-A number)	7	(see UPC-A for example)
Bar Code data	714551	7
Parity pattern for digits, from Table 5	EOEOEO	(look up parity pattern for checksum of 7)
Map characters using even/odd parity	XBUFVB	
Add Start, Stop, and checksum display	[XBUFVB]	7
Bar Code		

UPC-E Example 2

Data:	09135700009
Manufacturer number	91357
Item number	00009

Since the manufacturer's number does not end in zero, the 6 digits of the UPC-E code are:

First 5 characters of manufacturer's number	91357	
Last character of the item number	9	
Checksum (from UPC-A number)	6	(see UPC-A for example)
Bar Code data	913579	6
Parity pattern for digits, from Table 5	E000EE	(look up parity pattern for checksum of 6)
Map characters using even/odd parity	ZBDFXZ	
Add Start, Stop, and checksum display	[XBUFVB]	6
Bar Code		

SUPPLEMENTAL CODES

Both UPC-A and UPC-E bar codes support the same 2 and 5 digit supplemental codes. These supplemental codes let you add a two or a five digit number to any UPC bar code.

5 DIGIT SUPPLEMENTAL CODES

All supplemental calculations are the same for both UPC-A and UPC-E bar codes. First, take your 5 digit supplemental number and calculate a parity pattern for it. Assign the right-most digit an "odd" value, and then alternate even/odd assignments with the rest of the digits. Add all the odd position digits and multiply that by 3. Add all the even position digits and multiply that by 9. Add those two sums together. The parity pattern is the number in the ones position (*the remainder when you divide by 10*).

Data:	7 1 3 0 5
Position:	O E O E O
Odd sum:	$7 + 3 + 5 = 15, 15 * 3 = 45$
Even sum:	$1 + 0 = 1, 1 * 9 = 9$
Odd plus even:	$45 + 9 = 54$
Parity Pattern:	4

ef161.rtf	4.00 x 1.00	20
ef162.rtf	4.00 x 1.33	14
ef163.rtf	4.00 x 2.00	10
ef197.rtf	4.00 x 1.50	12
ef164.rtf	4.00 x 3.33	6

Our BarUPC.exe utility also lets you use your own custom label template. See your word processor manual for details on how to create a template, or download one from your label supplier. Once you have your own custom template, save the template file (*in rich text format*) in the Libraries\Documents\BarUPC folder with the file name “custom.rtf”. Our BarUPC.exe utility will directly access that template to let you easily print labels.

EMBEDDING BAR CODE FONTS IN PDF FILES

The fonts in this bar code set are not embeddable. When you embed a font in a PDF document, you are actually distributing a copy of that font with every single PDF file you generate. Your PDF generator glues a copy of the bar code font to the end of each PDF file it builds. This font is then installed on every computer that views the PDF document.

Embeddable versions of these fonts are available, at additional cost. Pricing for this add-on font set is based on how many computers the embeddable fonts will be installed on and how many people will view the PDF files. When you purchase an embeddable add-on font set you receive a new version of these fonts, and the new fonts will embed in PDF files. You must contact Elfring Fonts to order an embeddable add-on font set.

VISUAL BASIC MACROS FOR ACCESS, EXCEL, AND WORD

This package contains Visual Basic macros that let you automatically build UPC bar code strings in Excel, Access, and indirectly in Word. These macros, and sample Excel spread sheets in both Excel 2003 and Excel 2010 formats, are located in a folder named BarUPC under your Libraries\Documents folder. (**Libraries\Documents\BarUPC**)

Table 8

Function	Details
BarUPCa(Text)	Converts the input 11 or 12 digit number into a complete UPC-A bar code compatible with our UPC-A fonts. The function adds the Start code, translates and appends the data plus guard bar, calculates and adds the checksum, and puts the Stop code at the end. Non-numeric data is ignored, as are numbers greater than 12 digits in length. This result must be formatted with one of the following typefaces: UPCA a, UPCA b, UPCA c, UPCA d, UPCA e.
BarUPCe(Text)	Converts the input 7, 11, or 12 digit number into a complete UPC-E bar code compatible with our UPC-E fonts. UPC-E numbers have limits on the manufacturer’s number, and other limits on the product number. See previous sections above for details on these limits. This function translates the 7, 11, or 12 digit number into the abbreviated UPC-E format. Non-numeric data is ignored, as are numbers greater than 12 digits in length. This result must be formatted with one of the following typefaces: UPCE a, UPCE b, UPCE c, UPCE d, UPCE e. Note that virtually no error checking is done- make sure your data fits the limits of UPC-E!
BarUPCcheck(Text)	Take an 11 digit UPC-A number and calculate the checksum for it. Add the checksum to the end of the 11 digit number and return the resulting 12 digit UPC-A bar code number. (<i>Not a valid bar code string, just the actual bar code number including the checksum.</i>)
UPCatoe(Text)	Convert an 11 digit UPC-A number into a 7 digit UPC-E number. (<i>Not a valid bar code string, just the actual UPC-E bar code number including the checksum.</i>)

Warning!

Unlike our utility program, *BarUPC*, these Visual Basic functions do **very** limited error checking. You must make sure that the data you send to the function is correct! If you send bad data, your bar codes may be unreadable or they may not encode the data you think they have. Please use the *BarUPC* utility program to verify that your data is correct before using these Visual Basic functions to mass produce bar codes. UPC-E bar codes must follow a very strict formula (*see BarUPC.rtf*). If your UPC-E bar code data is not correct you will produce scannable UPC-E bar codes that do not match your manufacturer's number!

Using Visual Basic Functions in Excel

Open the spread sheet where you want to add bar code functions (*or create a new spread sheet*). Click on Tools, Macros, then Visual Basic Editor. In the Visual Basic Editor tool, click on File, Import File, and select the drive and user data folder. (Libraries\Documents\BarUPC or My Documents\BarUPC) The Visual Basic file, **VBbarUPC.bas**, should appear there. Select this file and open it. This will add a new function module, EFbarUPC, to your spread sheet. This module is automatically saved along with your spread sheet and adds the macro functions shown in Table 6.

These bar code functions can be used in any formula or cell to build working bar codes. For example, if cell H9 is defined as a text cell (*Format, Cell, Number, Text*) and cell I9 has the formula, =EAN13(H9), then any text entered in cell H9 will be converted into a bar code string in cell I9. Note that you also need to select the proper typeface for that bar code type, using Format, Cell, Font. See that bar code description for applicable font names.

Using Visual Basic Functions in Access

Open the database where you want to add EAN bar code functions (*or create a new database*). Under your database Objects, click on Modules, then click on the New icon at the top of the box. This will bring up the Visual Basic Editor tool. Click on File, Import File, and select the drive and user data folder where you installed our UPC Bar Codes. (Libraries\Documents\BarUPC or My Documents\BarUPC) The Visual Basic file, **VBbarUPC.bas**, should appear there. Select this file and open it. This will add a new function module, EFbarUPC, to your database. This module is automatically saved along with your database and adds the macro functions shown in Table 6.

These bar code functions can be used in a report to build working bar codes. To use these functions, pass data to them from your table fields and return the result in a report. You do this by entering a formula =BarUPCa ([table.field]) in the control source field of the report. Note that you need to select the proper typeface for that bar code subset. See Table 1 for a complete listing of font names.

Using Visual Basic Functions in Word

While Word does have the ability to use Visual Basic functions in macros, you need to be a macro expert to use these functions. You can, however, combine Word and Excel to print UPC bar codes! The basic idea is to keep your data in Excel, and to have Excel format the data for bar code printing. You then do a mail merge in Word, selecting the formatted string from Excel as the data source.

Place your data in Excel in a specific column. Now apply one of our bar code functions to the data in that column, producing a second column that contains the formatted string. This second column is passed to Word in the mail merge. You must make sure to select the proper subset typeface for your mail merge field. See Table 1 for a complete list of font names. Also note that the on-screen display of the bar code will not look correct. Only the data printed by the mail merge will be in the correct UPC bar code format.

Glossary

Application Program: a computer program that performs useful work not related to the computer itself. Examples are word processors, spreadsheets, accounting systems, and engineering programs.

ASCII: American Standard Code for Information Interchange – a standard code for representing characters as numbers that is used on most microcomputers, computer terminals, and printers.

Aspect ratio: the ratio of the bar code height to bar code width.

Character string: a sequence of characters stored in a computer and treated as a single data item.

Checksum: a number that accompanies data transferred from one place to another and helps to ensure that the data was transferred correctly.

Clipboard: a holding area to which information can be copied in order to transfer it from one application to another.

CPI: characters per inch: the number of characters a fixed width font will print per inch.

DPI: dots per inch: the number of pixels or printer dots per linear inch.

Font: a complete collection of characters, in a consistent style and size. This includes upper and lower case letters, numerals, punctuation, ligatures, and reference marks.

Modulo: a mathematical operation that gives the remainder when one number is divided by another.

Parity pattern: a clever way of encoding extra data, like a checksum, in the individual characters of a bar code.

Pixels: one of the individual dots that make up a graphical image.

Radio button: small circles in a dialog box, only one of which can be chosen at a time. Choosing any button with the mouse causes all the other buttons in the set to be cleared.

Scalable: able to be used on a large or small scale without major changes.

Scalable font: a font that can be used to print characters of any size. Many newer laser printers include scalable fonts.

Symbology: a method of representing information by printed characters.

Scanner: a device that enables a computer to read printed or handwritten page.

Start Code: the leading character of a bar code that identifies what kind of bar code it is.

Stop Code: the last character of a bar code that tells the bar code reader when it has reached the end of the code.

TrueType Font: an outline typeface that can be scaled to practically any size. Useful in Windows, some versions of Linux, and the Mac.

UPC: Uniform Product Code- the bar code used on all retail products in North America. For detailed information see: www.mybarcodestore.com

UPC-A: Uniform Product Code, version A- the standard 12 digit bar code used on all retail products in North America. Includes 11 data digits plus a 1 digit checksum.

UPC-E: Uniform Product Code, version E- an abbreviated form of the 12 digit UPC-A code. UPC-E bar codes are designed for use in small areas and through zero suppression have only 7 digits.

Utility: a program that assists in the operation of a computer but does not do the main work for which the computer was bought.

Visual Basic Function: a small program written in Visual Basic and used in products like Excel, Access, or Word to help create bar codes.

Copyright 1992-2015 by Elfring Fonts, Inc.
2020 Dean St, Unit N, St Charles, IL 60174
Phone: 1-630-377-3520

www.barcodingfonts.com

www.mybarcodestore.com