



Color Coding Your Items

How to Dynamically Change the Color
of Objects on Your Template

Supports the following BarTender software versions:
BarTender 2016, BarTender 2019

Contents

- Overview 3**
- Using Layers to Change Object Color 4**
 - How Layers Work in BarTender 4
 - Conditional Printing 4
 - Example 4
- Using VBScript to Change Object Color 8**
 - About VBScript 8
 - Using VBScript with a Database 8
 - Document-Level Events 8
 - Examples of How to Use VBScript to Dynamically Change Colors 9
 - Support for Data-Sourced VBScript 10
- Appendix A: Object Properties for Dynamic Color 11**
- Appendix B: BtColor Constants 12**
- Related Documentation 15**

Overview

Many businesses use *color coding* to display and distinguish information by color. For example, the chemical industry uses color-coded labels for chemicals that are stored in the laboratory, so that users can tell at a glance whether the stored chemical is hazardous and, if so, what kind of hazard it poses. Electrical wiring is color-coded so that technicians can easily and safely identify the wire type and avoid accidents.

Color coding your items can improve organization, inventory management, scheduling and more. A fitness gym, for example, might want to print color-coded membership cards, one for each level of membership that they offer. A moving company might want to color-code moving box labels according to room.



Printing color-coded items in single-color batches can be time consuming. However, you can use BarTender to print differently color-coded items *dynamically* (on the fly), at print time. When you do this, you have to set up the template only once, by using layers and/or Visual Basic Script (VBScript), to print items with colors that vary from one item to the next. You never have to go back and edit the template to get different color results.

Using Layers to Change Object Color

One way to change the color of objects on your template at print time is to combine layers with conditional printing. A *layer* is an object or group of objects that occupy a particular plane and that can be stacked on top of each other.

How Layers Work in BarTender

In BarTender, layers can include one or more text objects, images, barcodes, shapes and/or lines on the same plane. If you need to dynamically print multiple objects on your template at different times, you can put objects on different layers and then conditionalize the layers to be printed when certain conditions are met. You use each layer's **When to Print** dialog to determine when that layer is printed.

Conditional Printing

Use *conditional printing* to specify exactly when you want parts of your template to print.

Conditional printing is based on conditional operators. For example, your conditions for printing might be based on whether the conditionalized item **contains** or **does not contain** certain text, numeric values or images, or whether the conditionalized item is **less than**, **greater than**, or **equals** a set value.

In situations where you need to change colors for one or more objects on your template, you can conditionalize an entire layer rather than each individual object. To do this, on a new or copied layer, change the color of the objects that you want to be printed differently, and then conditionalize the layers so that they print when the conditions are met by the referenced database field or named data source value.

For more information about layers and conditional printing, refer to the following topics in the BarTender help system:

- [Using Layers](#)
- [Conditional Printing](#)
- [Building Conditional Expressions](#)

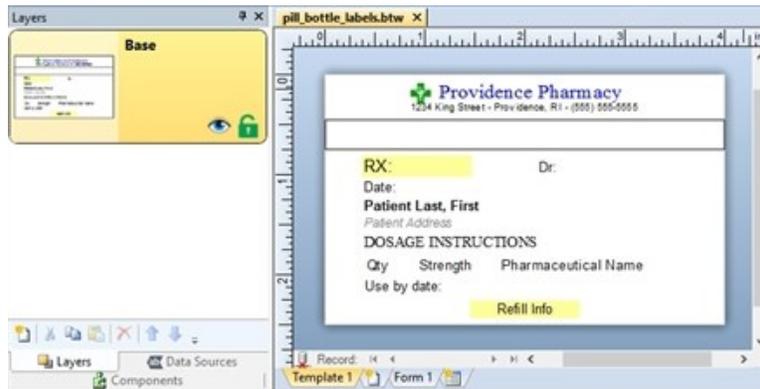
Example

Suppose that you are creating a prescription label. The pharmacy uses a color-coded stripe at the top of the bottle to quickly indicate the type of medication. Red is for cardiac medicines, blue is for analgesics, and green is for antibiotics. You want the colored stripes to be printed based on the type of medication that is listed in the database. To implement this scenario, follow these steps:

1. Create a base layer that has the label information, and then add the "stripe" to the design as a plain rectangle object.
2. Create a new layer (Layer 2) and then paste the rectangle object onto the new layer.
3. Copy and paste the new layer twice more to create Layers 3 and 4.
4. Change the color of the rectangle object for each layer.
5. Conditionalize each layer by connecting it to a data source's value.

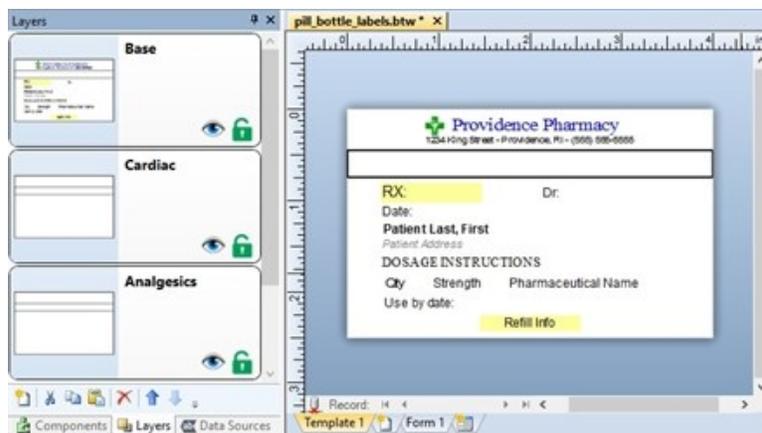
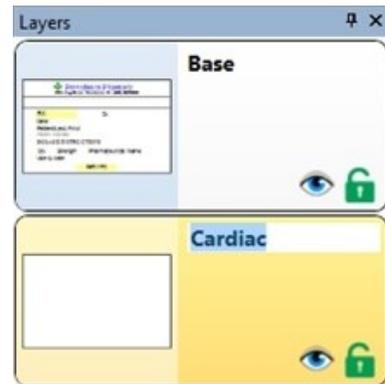
Creating the Base Layer

1. Create the base layer, including the objects whose color you want to change. For this example, use a rectangle object as a "stripe."
2. At the bottom of the Toolbox, click the **Layers** tab. The **Layers** pane opens and displays your design in "Layer 1."
3. To rename this layer, click **Rename** in the layer's context menu.



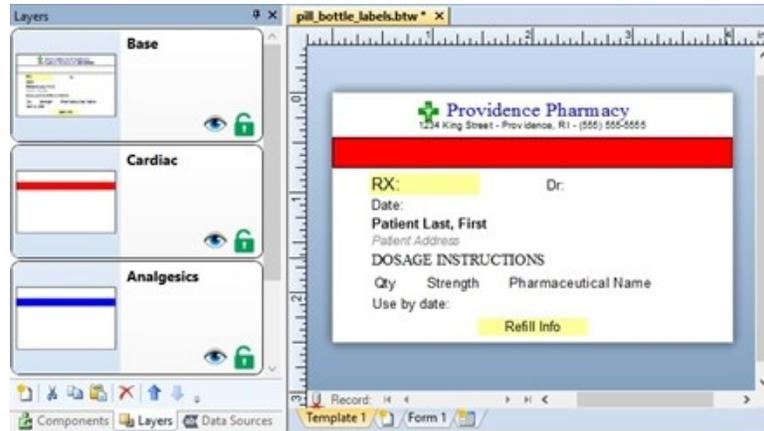
Creating New Layers for Each Color Variation

1. On the **Layers** pane context menu, click **New Layer**. The **Layer Properties** dialog appears.
2. Click **OK** to close the dialog. You now have a blank Layer 2 in the **Layers** pane.
3. On the new layer's context menu, click **Rename** to assign a name to the layer.
4. Repeat steps 1-3 for the remaining number of color variations.
5. On each of your new layers, copy and paste the object that you want to color code. In this case, copy and paste the rectangle object from the base layer.



Setting Each Layer's Object Color

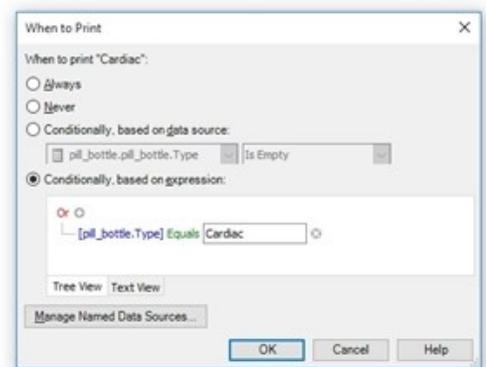
1. Double-click the rectangle object in Layer 2 ("Cardiac"). The **Box Properties** dialog opens.
2. In the left navigation pane, click **Box**.
3. Under **Line Properties**, use the **Color** list to change the color of the border.
4. Under **Fill Properties**, use the **Color** list to change the fill color of the object.
5. Repeat steps 1-4 to change the box colors on the subsequent layers.
6. When you have finished assigning colors, click **Close**.



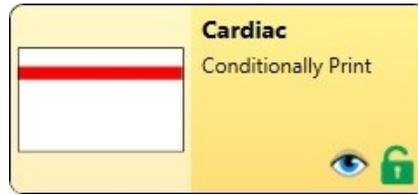
Conditionaling the Layers

By creating a separate condition for each colored layer, you can control which colors are printed and when. In this example, BarTender determines which layer to print based on the "Type" of medicine that is listed in the database.

1. In the **Layers** pane, double-click on Layer 2 ("Cardiac"). Alternatively, click **Properties** in the layer's context menu. The **Layer Properties** dialog opens.
2. Click the **Print Options** tab.
3. At the right of the **Print When** field, click . The **When to Print** dialog opens.
4. Click to select **Conditionally, based on expression**.
5. Create an expression that is specific to the condition under which you want the layer to be printed, based on one or more data fields. For example, "[Type] Equals Cardiac" causes the layer (which contains a red stripe) to be printed when the "Type" database field returns the value "Cardiac".
6. Conditionalize the other two layers to be printed when "Type" equals "Analgesic" and "Antibiotic."



Note that conditionalized layers display the words "Conditionally Print."



The final output from this example pulls information from the database. When the medicine type matches "Cardiac," the stripe on the label is colored red. Analgesics are colored blue, and antibiotics are colored green.

<p> Providence Pharmacy 1234 King Street - Providence, RI - (555) 555-5555</p> <hr/> <p>RX: 111683836A Dr: KENDALL, J Date: 2015-10-15 Patient: Smith, Maxwell</p> <p>TAKE ONE TABLET EVERY DAY</p> <p>Qty: 5 Warfarin Sodium 5mg Use by date: 2016-10-15 **Refills require authorization**</p>	<p> Providence Pharmacy 1234 King Street - Providence, RI - (555) 555-5555</p> <hr/> <p>RX: 111683837A Dr: WHITE, L Date: 2015-10-15 Patient: Willis, Katherine</p> <p>TAKE ONE TABLET EVERY SIX HOURS AS NEEDED FOR PAIN</p> <p>Qty: 20 APAP/Codeine 300/30mg Use by date: 2016-10-15 REF: 5</p>
<p> Providence Pharmacy 1234 King Street - Providence, RI - (555) 555-5555</p> <hr/> <p>RX: 111683838A Dr: CARLSON, M Date: 2015-10-15 Patient: Doe, Jane</p> <p>TAKE ONE CAPSULE THREE TIME S DAILY</p> <p>Qty: 30 Amoxicillin 500mg Use by date: 2016-10-15 **No refills remaining**</p>	

Using VBScript to Change Object Color

You can set up your BarTender document to run VBScript that changes the color of an object in response to certain document-level events. You can apply VBScript to any template object, including text, barcode and shape objects.

You can create VBScript for your document by using the Script Editor. For more information, refer to the [Script Editor](#) topic in the BarTender help system.



VBScript is an advanced method of managing your template and should be used by those who have some technical knowledge of scripting.

About VBScript

VBScript is a subset of Microsoft's Visual Basic programming language. It is specifically designed to add functionality to existing programs rather than to write programs.

You can use VBScript to programmatically modify the color of template objects at different times throughout the document design and print process. To do this dynamically, you need to create a database and then connect it to your BarTender document.

Using VBScript with a Database

When you connect your BarTender document to a database, you can access data that is contained within the data source. The data can be used to do a variety of things, including changing the color of a template object. By linking a specific database field to an object on the template, the value of the object is replaced with the data that is contained within the database. Additionally, you can refer to colors that are stored in database fields or change the color based on the data that is inside your database.

For more information about how to use databases in BarTender, refer to the following topics in the BarTender help system:

- [Reading Data from Databases](#)
- [Referencing Database Fields](#)

Document-Level Events

Certain events occur on the document level, such as OnPrintStart or OnPrintCancel. From the Script Assistant, you can select a document-level event that will trigger your VBScript to run and change an object's color.

One of the easiest document-level events to use for changing color is "OnNewRecord," which causes the script to run each time a new record is read from the database. For a complete list of document-level events, refer to the [Document Level Events](#) topic in the BarTender help system.

Before you can use VBScript with a BarTender document, you must first enable the use of VBScript for that document. To do this, follow these steps:

1. In the **BarTender Document Options** dialog, click the **Scripting** tab.
2. Click to select **Enable document level script events**.
3. Select the event that you want to use.
4. Click **Edit** to open the Script Editor.
5. In the **Script** pane, select the event that you want.
6. In the **Editor** pane, enter your custom script.

Examples of How to Use VBScript to Dynamically Change Colors

The following examples describe how you might dynamically change an object's color by using VBScript.

Example 1

Suppose that you have a business that uses customer membership levels, and you want to change the color of a text object to match each customer's membership level. To do this, reference the data source that specifies membership type (in this example, the data source is called MembershipType). To make a text object turn gold, silver or blue depending on membership level, enter a script that is similar to the following in the **Script Assistant** pane of the Script Assistant:

```
ReferenceField("MembershipType")
Set Notice = Objects("Text 1")
If (Field("MembershipType") = "Gold") Then
    Notice.TextColor = btColor.Gold
Else
    If (Field("MembershipType") = "Silver") Then
        Notice.TextColor = btColor.Silver
    Else
        If (Field("MembershipType") = "Platinum") Then
            Notice.TextColor = btColor.Blue
        Else
            Notice.TextColor = btColor.Black
        End If
    End If
End If
```

Example 2

If you have a field in your database that contains the name of a color, you can use VBScript to set the color of an object.

For example, if you have a database field that is called "ConditionColor" and database records that contain color names like "BtColor.Gold" and "BtColor.Silver," you can use VBScript to determine when a certain color is printed. If you wanted to change the color of a box shape object, your script might resemble the following.

```
ReferenceField("ConditionColor")  
Objects("Box 1").FillColor = Eval(Field("ConditionColor"))
```

- For a list of object properties for dynamic color (such as TextColor and BarCodeColor), refer to [Appendix A: Object Properties for Dynamic Color](#) of this white paper.
- For a list of BtColor constants, refer to [Appendix B: BtColor Constants](#) of this white paper.

Support for Data-Sourced VBScript

BarTender can read VBScript from external sources, such as a database or a program that is controlling BarTender. For example, you could have a field in a database that contains VBScript. Then, when BarTender reads the data record that contains that field, the VBScript code in BarTender can use the Visual Basic “Execute” statement or the “Eval” function to run the imported VBScript.

Appendix A: Object Properties for Dynamic Color

The Object object in BarTender's library of VBScript objects represents an object in a document, such as a barcode or text object. This table lists color-related properties that you can change in VBScript for each type of object. For a complete of object properties, refer to the [Template Objects \(Object Object\)](#) topic in the BarTender help system.

For a list of btColor constants, refer to [Appendix B: BtColor Constants](#) of this white paper.



All Object properties are to be used in document scripts, which you can access from the **BarTender Document Options** dialog. They cannot be used in data source scripts or transform scripts.

Examples

`MyTextObject.TextColor = btColor.Blue` turns the text in a text object blue.

`MyShapeObject.FillColor = btColor.Black` sets a shape's fill color to black.

Name	Description
BarCodeColor	Sets or returns a barcode object's bar color.
FillColor	Sets or returns any shape's fill color.
LineColor	Sets or returns the color of a line object.
TextBackgroundColor	Sets or returns the background color of an object's text.
TextColor	Sets or returns the color of an object's text.

Appendix B: BtColor Constants

When you set the color of an object, you must reference a BtColor constant.

Example

`MyTextObject.TextColor = btColor.LimeGreen` turns the text in a text object *lime green*.

The following constants are available.

Color	Value	Color	Value
AliceBlue	#F0F8FF	LightSalmon	#FFA07A
AntiqueWhite	#FAEBD7	LightSeaGreen	#20B2AA
Aqua	#00FFFF	LightSkyBlue	#87CEFA
Aquamarine	#7FFFD4	LightSlateGray	#778899
Azure	#F0FFFF	LightSteelBlue	#B0C4DE
Beige	#F5F5DC	LightYellow	#FFFFE0
Bisque	#FFE4C4	Lime	#00FF00
Black	#000000	LimeGreen	#32CD32
BlanchedAlmond	#FFEBCD	Linen	#FAF0E6
Blue	#0000FF	Magenta	#FF00FF
BlueViolet	#8A2BE2	Maroon	#800000
Brown	#A52A2A	MediumAquamarine	#66CDAA
BurlyWood	#DEB887	MediumBlue	#0000CD
CadetBlue	#5F9EA0	MediumOrchid	#BA55D3
Chartreuse	#7FFF00	MediumPurple	#9370DB
Chocolate	#D2691E	MediumSeaGreen	#3CB371
Coral	#FF7F50	MediumSlateBlue	#7B68EE
CornflowerBlue	#6495ED	MediumSpringGreen	#00FA9A
Cornsilk	#FFF8DC	MediumTurquoise	#48D1CC
Crimson	#DC143C	MediumVioletRed	#C71585
Cyan	#00FFFF	MidnightBlue	#191970
DarkBlue	#00008B	MintCream	#F5FFFA
DarkCyan	#008B8B	MistyRose	#FFE4E1
DarkGoldenrod	#B8860B	Moccasin	#FFE4B5
DarkGray	#A9A9A9	NavajoWhite	#FFDEAD

DarkGreen	#006400	Navy	#000080
DarkKhaki	#BDB76B	OldLace	#FDF5E6
DarkMagenta	#8B008B	Olive	#808000
DarkOliveGreen	#556B2F	OliveDrab	#6B8E23
DarkOrange	#FF8C00	Orange	#FFA500
DarkOrchid	#9932CC	OrangeRed	#FF4500
DarkRed	#8B0000	Orchid	#DA70D6
DarkSalmon	#E9967A	PaleGoldenrod	#EEE8AA
DarkSeaGreen	#8FBC8F	PaleGreen	#98FB98
DarkSlateBlue	#483D8B	PaleTurquoise	#AFEEEE
DarkSlateGray	#2F4F4F	PaleVioletRed	#DB7093
DarkTurquoise	#00CED1	PapayaWhip	#FFefd5
DarkViolet	#9400D3	PeachPuff	#FFDAB9
DeepPink	#FF1493	Peru	#CD853F
DeepSkyBlue	#00BFFF	Pink	#FFC0CB
DimGray	#696969	Plum	#DDA0DD
DodgerBlue	#1E90FF	PowderBlue	#B0E0E6
FireBrick	#B22222	Purple	#800080
FloralWhite	#FFFaf0	Red	#FF0000
ForestGreen	#228B22	RosyBrown	#BC8F8F
Fuchsia	#FF00FF	RoyalBlue	#4169E1
Gainsboro	#DCDCDC	SaddleBrown	#8B4513
GhostWhite	#F8F8FF	Salmon	#FA8072
Gold	#FFD700	SandyBrown	#F4A460
Goldenrod	#DAA520	SeaGreen	#2E8B57
Gray	#808080	Seashell	#FFF5EE
Green	#008000	Sienna	#A0522D
GreenYellow	#ADFF2F	Silver	#C0C0C0
Honeydew	#F0FFF0	SkyBlue	#87CEEB
HotPink	#FF69B4	SlateBlue	#6A5ACD
IndianRed	#CD5C5C	SlateGray	#708090
Indigo	#4B0082	Snow	#FFFAFA
Ivory	#FFFFF0	SpringGreen	#00FF7F

Khaki	#F0E68C	SteelBlue	#4682B4
Lavender	#E6E6FA	Tan	#D2B48C
LavenderBlush	#FFF0F5	Teal	#008080
LawnGreen	#7CFC00	Thistle	#D8BFD8
LemonChiffon	#FFFACD	Tomato	#FF6347
LightBlue	#ADD8E6	Turquoise	#40E0D0
LightCoral	#F08080	Violet	#EE82EE
LightCyan	#E0FFFF	Wheat	#F5DEB3
LightGoldenrodYellow	#FAFAD2	White	#FFFFFF
LightGray	#D3D3D3	WhiteSmoke	#F5F5F5
LightGreen	#90EE90	Yellow	#FFFF00
LightPink	#FFB6C1	YellowGreen	#9ACD32

Related Documentation

White Papers

- *Creating Intelligent Templates*

To view and download white papers, visit:

<https://www.bartendersoftware.com/resources/white-papers/>

BarTender Help System

- [Changing an Object's Color](#)
- [Using Layers](#)
- [Conditional Printing](#)
- [Scripting in BarTender](#)

Other Resources

Please visit the BarTender website at <https://www.bartendersoftware.com>.

