# Use the Heading 1 style for the main title of your article

After the heading, the majority of your text should use the style called Body Text. **Never use the style called Normal!** Everything in the *CODE Magazine* template is based on the Body Text style.

You always place a space (empty line, in the Body Text style) between all paragraphs, before and after all Code snippets, before and after figures and tables, before and after headings, etc.

Body text can use **bold** and *italic* styles. Generally, bold for commands and italic for almost anything else that you want to emphasize.

## A Few Words about Writing Style (Heading 2 style)

It’s your article and we’d like to keep your words, but we’re competing for our reader’s attention and we need to sound like a quality publication – that means we enforce minimum writing standards. The editorial staff at *CODE Magazine* will edit your article, and they will make changes. Don’t get defensive about it – all good writers have good editors in the background to support them. If you write poorly and we determine that it takes too much effort to edit your article, we will return it to you to rewrite it. Use a style guide if you have basic grammar questions (any one will do). Pay special attention to consistent capitalization of key programming terms in particular because the person who copy edits your article isn’t a developer.

You can distinguish good writing from worse writing because good writing minimizes sentences in the passive voice. Passive voice makes your article dull and we want you to write an interesting article, not a dull article about an interesting topic. Here’s a good rule of thumb to avoid passive voice: If your sentence has any form of the verb **to be** in it (is, am, are, was, were, be, being, been), that sentence, by definition, is passive. Do a search through your article once in a while for one of these words and you’ll be surprised how often you use them. Then think about how you can rewrite your sentence to add a subject near the front and add a verb followed by the action performed. You can usually catch that you’re going to write a passive sentence when you start a sentence with one of these words: “This, These, That, The”. To avoid passive sentences, tell the reader what the subject of the sentence is doing. For example:

*Active:* Incorporating stored procedures into your application logic offers many advantages.

*Passive:* There are many advantages to incorporating into your application logic.

*Active:* You’ll find GraphicsPath, a GDI+ class, in the *System.Drawing.Drawing2D* namespace.

*Passive:* *GraphicsPath* is a GDI+ class that can be found in the *System.Drawing.Drawing2D* namespace.

*Active:* This article will show you how to use stored procedures in conjunction with the SqlDataAdapter.

*Passive:* The purpose of this article is to demonstrate how stored procedures can be used in conjunction with the SqlDataAdapter.

*CODE Magazine’s* editors will spend 80% of their effort on your article correcting capitalization and reducing passive voice. We contact good writers and ask them to write again and again, and after they prove themselves we pay them more. When book publishers contact us, we recommend good writers. We’d like to see your name in the group of good writers.

## Handling Visual Elements within Your Article (Note: This is the second Heading 2. You need at least two Heading 2 style headlines in each article.)

Remember that we need a blank line before and after your heading, and you need to make that blank line in the Body Text style.

### Figures, Screen Shots, and Diagrams

You may have some body text where you have screen shots or diagrams. *CODE* refers to all of these as a Figure. You always need to have a callout to a figure from within the text as **Figure 1** or **Figure X**. The figure references are always bold.

For example, “The result of this scripting technique produces the output you see in **Figure 5**.”

*Do not* make references to “the *next* figure/table/listing, etc” or “the *previous* figure/table/listing” because the figure/table/listing could be two pages away from your text callout. Make references to the next figure/table/listing by indicating the number of that item. So for example, “You can see the result in **Figure 5** because you checked the setting in **Figure 4**.” Don’t assume that the figure will appear on the same page as the figure reference–it’s a layout issue Advertisers pay to have their ads inserted into an article, not between articles, so advertiser’s artwork gets higher precedence than your article’s figure, layout-wise. And if a Code Listing takes a lot of space, it could bump your figure to a later point in the article, or vice versa.

You need to provide all figures as non-compressed TIFF files externally to the Word document (***not*** inside the document). You may paste a (low res) version of the image in the article if you wish, but that’s optional. We expect to see your article with non-compressed TIFF files of screenshots, etc. in addition to your Word file. Name them YourLastName\_Fig001.tif, YourLastName\_Fig002.tif, YourLastName\_Fig003.tif, etc.

### Subheadings like This Are in the Heading 3 Style

When you’re using subheadings think about parallelism. You shouldn’t ever have just one subhead – you need to have two or more subheads.

*CODE Magazine* does NOT support a Heading 4 style. If you think you need it, your article is drilling down TOO MUCH. Think about another way to organize your thoughts. Heading 1 it the title of your article. You have Heading 2 and Heading 3 to organize all of your thoughts. Nothing deeper.

### Inserting Code Examples in Your Article

Like many technical publications, *CODE Magazine* prints code in the article. You’re writing to other developers and they like to see code that illustrates your point. *CODE Magazine* uses Code Snippets (15 lines or fewer) and Code Listings (longer than 15 lines). You set short listings (up to a max. of 15 lines) “inline” using the Code Snippet style. Here is a sample. Note that we use color formatting and you are required to turn your articles in using color formatting for VB and C# code. :

Code Snippet (15 lines or less)

 1 2 3 4 5

12345678901234567890123456789012345678901234567890

public bool LoadStrings(string SomeString)

{

 // Some Comment

 string encoding="utf-8" ?>

}

<configuration>

 <system.web>

 <!-- DYNAMIC DEBUG COMPILATION -->

 <compilation defaultLanguage="vb" />

 </system.web>

</configuration>

The maximum width of the line is important here: 50 characters! A Code Snippet has to fit within a column of text on a printed page. NO LINE OF CODE CAN AUTOMATICALLY WRAP TO THE NEXT LINE. YOU MUST PUT A HARD RETURN AT THE END OF EACH LINE AND THEN USE THE SPACE BAR TO LINE UP INDENTED LINES. You may want to turn on the Show/Hide Paragraph Marks in Word to see which lines wrap and which don’t. (Search Help if you need to figure out how to turn it on or off.)

When pasting code from Visual Studio .NET, syntax coloring is automatically preserved. Therefore, always use Visual Studio .NET to copy source code from. We want to retain the color coding for all source code if at all possible in your Code Snippets and longer Code Listings. If your Code Snippet is longer than 15 lines, make it a Listing and give it a caption.

You can place lines of code that you want to highlight as “Code Snippet + Pattern: Clear (Gray-20%)”.

Please make sure your code has no background colors or bold/italic fonts.

There is a known Word bug when copy/pasting your code into Word and applying a style to them. Sometimes the color changes work, sometimes they don’t. It has something to do with whether you grab the last paragraph mark. We don’t have a guaranteed workaround. If you copy/paste a sample from the VS IDE into Word and apply the Code Snippet style and all of the code becomes one color, you (as author) are responsible to hand-color that snippet to match what was in the IDE.

It is your responsibility to apply the style and then fix any color coding problems before you send the article to us. Yes, it could mean tediously applying color coding by hand, a few colors per line. If you don’t do it, that’s how we have to do it. Here’s a tip. Highlight something in your Code Snippet and change the color (blue for key words for example). Then go to the next key word and press F4. That will apply the last style to the next text you highlight. Then go through the rest of the snippet and apply blue to all keywords. Then make a pass through the snippet to apply the next color. For example, after blue, make a pass for black text. Then make a pass for all comments in green, then terms in teal, then things in quotation marks in red. That’s five passes through a snippet or listing to fix color coding – It seems tedious but using F4 to apply the previous formatting to the currently highlighted text makes it go pretty fast. Color formatting has to be right and that guarantees it is correct when it gets to layout.

*CODE Magazine* uses LOTS of white space to make it easier to read the articles and create out unique layout. Put a blank line in Body Text style before and after Code Snippets, paragraphs, and all headings.

### Listings

You should set longer code examples (20 or more lines) in separate Listing blocks. Like in the case of figures, you should insert a bolded callout from within your paragraph, and listings have a caption. Note that separate listings support up to 67 characters per line. Also like figures, DO NOT make references to “next” or “previous” listings. It is often the case that a long listing won’t appear on the same page as the paragraph where you discuss it. In fact, the layout tends to stack listings up toward the end of the article. You’ll put Listings at the end of the article because they’re awkward to handle so they get placed separately in layout.

Note: Articles should not have a disproportionate amount of listings. If you find yourself writing a 4 page articles with an additional 5 pages of code listings, it will not end up as a nice reading experience for our readers. In that case, some of those listings should instead just be part of the sample download (see below) and the text should refer to the download.

So you’d use a body text callout like this: See **Listing 1** for an example. (And the listing itself goes at the end of the article.)

*Hint:* To keep code and the text describing it together, write shorter code examples so that you can use the code snippets style. Your reader will find it easier to read several code snippets with text in between that describes the snippets.

### Downloads

Articles should always be accompanied by code downloads that contain sample projects (if at all applicable), source code, and similar assets. Downloads have to be packaged into a single ZIP file that is submitted together with your main article and images.

Note: Not having downloads is the biggest criticism we hear from readers. Your article is very likely to be much higher rated if you provide a download. Not to mention that we won’t pay your author’s fee if your article has examples but the download is not provided ;-)

### Numbered Lists and Bullet Lists

*CODE Magazine’s* layout supports numbered lists (use the “Numbered Body Text” style from the CODE Magazine template instead of the numbering icon that appears in the Office Ribbon. Like everything else in CODE Magazine, put a blank line before and after a block that is a numbered list or a bullet list. For example, you might have step-by-step directions that would start like this:

1. From the Tools menu choose the Options menu item.
2. When the Options dialog box comes up, click the list item on the left for the Windows Forms Designer.
3. Change the GridSize property from 8,8 (the default) to 12,12.
4. Click OK to finish.

Use a numbered list for procedures, step-by-step directions or other sequential lists. Introduce a procedure with an infinitive phrase or imperative, and avoid explanatory text after the phrase. Capitalize the first word of each entry. Numbered lists should have a sentence and a verb, thus a period at the end. Note that before and after lists you need an empty line in the Body Text style, but ***not*** within each list (after items).

*CODE* *Magazine* also supports bulleted lists (“Bulleted Body Text”) and you should use CODE Magazine’s style instead of Word’s built-in bullets in the Office Ribbon:

* Item 1
* Item 2
* Item 3

In general, unless all of the items in the bullet list are sentences, bullet list items do not end with a period. (This is an example of a PullQuote. It goes inline and use the PullQuote style from the CODE Magazine template.)

Our layout also supports tables and figures which we’ll cover next.

### Figures and Tables

Figures and their captions go inline. Always have a callout to a figure or table from within the body text of the article. For example, “Look at the Watches window in **Figure 1** and see how the values change as you step through the code.” Put the figure and the caption both in the Figure Caption Style. In particular, note that you have **Figure X**: (with the figure number and the colon) in BOLD, followed by the text of the caption. Please make the caption a complete sentence with a period at the end.

Copy the figure into the document at its full size. (You can change the size of the image so it fits the width of the document. Do NOT change the size of the image before you paste it in a different graphics program). It is best to use standard colors. We know a lot of people like to create their own color schemes in their development environments and such, but for a magazine, we want to print screen shots in a way that most people are familiar with and thus have an easier time to follow. So please use standard color themes for Visual Studio and other IDEs. Also, it is preferable to stick to a light color theme for print. Dark images print badly (sometimes they “bleed through” to the next page, even) and they tend to be hard to read in print.

The image itself shall use the style “Figure”. The paragraph following the figure paragraph shall hold the image caption and shall use the “Figure Caption” style, as shown in the following example:



Figure 1: The XAML Editor in Visual Studio.

Remember to provide a high resolution version of each image, numbered correctly, in the zip file that you send with your Word document. And don’t forget to include the photo of yourself to run with the article!

Tables are a great way to define terms or to take any list that is more than about five items and organize them. You can see **Table 1** as an example of the right way to apply the styles for tables.

**Table 1:** Albacore tasks. (Note that table captions use the Table Caption style)

|  |  |
| --- | --- |
| Task (note the Table Heading style) | Description |
| AssemblyInfo (Note the Table Text style) | Generate assembly information in Visual Basic or C#. |
| Docu | Build project and assembly documentation from XML code comments. |
| Exec | Execute any arbitrary command with parameters. |
| ExpandTemplates | Use file templates with special data markers and YAML files to generate app config and other files during the build. |
| MSBuild | Build a Visual Studio solution or project, or MSBuild project file. |
| MSpec | Execute unit tests with the MSpec (Machine.Specification) framework. |
| NCoverConsole | Run code coverage analysis with NCover. |
| NCoverReport | Report the code coverage results with various options. |
| NUnit | Execute unit tests with the NUnit framework. |

### Hyperlinks

We’ve searched through major IT publications to try and come up with a standard for handling hyperlinks. The fact of the matter is that there isn’t a standard. Go figure. Interestingly, no print publications that we would consider competitors allow the use of Word’s automatic hyperlinks of blue text, underlined. However, they use them online. And publications even vary whether they have the http:// ahead of a www, and some of them put the Web address in Italic. So here’s the *CODE* standard:

* No bold or italic.
* Please insert the fully qualified name, so http://www.codemag.com but remove the hyperlink that Word tries to automatically insert.

If you want to disable them permanently, go to the Tools menu, choose AutoCorrect options, choose the AutoFormat as you type tab, and clear the check box, Internet and network paths with hyperlinks.

### Checking that Your Article Applied Styles Correctly

If you use the *CODE Magazine* template, you can either take our template, delete the example text and write your article, or you can take your article and apply our template to your article. To apply our template to your article, click the Office Ribbon button, at the bottom center, click Word Options. From the menu on the left choose Add-ins. At the bottom of the dialog box you’ll see a drop down box with the types of Add Ins you can choose. Select Templates and then click Go. The Templates and Add-Ins dialog box appears. Click the Attach button to choose the CODE Magazine Style Guide and Template from whatever directory you put it in. IMPORTANT: Make sure you click the checkbox to Automatically update template styles. Then click OK.

Your article should apply the styles. Note that the Developer menu appears with the Word’s other menus. Click the Developer top menu. Then click the Macros button. You’ll see two macros that you should run before you send the article to the Editor in Chief. The one called CheckCodeMagazineGuidelines is the one that does most of the real work. It will identify all of the things you need to fix to make sure you applied styles correctly in your article. Please FIX ALL THINGS IDENTIFIED. You may need to break out of the macro, fix the item it wants to fix, then restart the macro. You should run this macro a few times until you’ve fixed all things it identifies.

The other macro called FormatHelper where you need to insert a blank line (in the Body Text style) into your article. It also fixes typographical issues in your Code Snippets and Code Listings and replaces tabs with three spaces.

### The Stuff that Goes at the End of Your Article

Sidebars Start with a Title in the Sidebar Caption Style

SideBar body text goes in the SideBar style. We’d like one or two sidebars per article. 2-3 paragraphs at the most.

Note: Sidebars can NOT contain code snippets!

Your Author Name

Author Email and Phone
Blog URL
Twitter URL

The bio from you should be at least three sentences long, and at most 3 paragraphs long. If you’ve written books or you write for a magazine, the publication titles go in Italic. If you speak at conferences, those do NOT go in Italic.

**Listing 1:** CODE listing header style is like the caption of a figure, but no period at the end

code Listing (more than 15 lines)

 1 2 3 4 5 6

1234567890123456789012345678901234567890123456789012345678901234567

public bool LoadStrings(string SomeString)

{

 // Some Comment

 string oldLine = ""; string newLine = "";

 string oldLine2 = ""; string newLine2 = "";

}

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

 <system.web>

 <!-- DYNAMIC DEBUG COMPILATION -->

 <compilation defaultLanguage="vb" />

 </system.web>

</configuration>

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

 <system.web>

 <!-- DYNAMIC DEBUG COMPILATION -->

 <compilation defaultLanguage="vb" debug="true"/>

 </system.web>

</configuration>

### Editorial Pet Peeves

* Since the introduction of the computer and desktop publishing packages, we stopped placing two spaces at the end of each sentence. You should only put one space after a period, questions mark, colon, etc. In typing class way back when the standard was two spaces. In the modern computer age we use one.
* You didn’t provide a photo of yourself so we have to chase you down for it. Make it a high resolution image (at least 300 DPI). Wear something a bit nicer than a t-shirt please.
* You’ve got a lot of passive voice in your article. You can’t eliminate all passive voice, but we want 80% of it gone. If you rewrite every sentence to eliminate the words “is” and “are” you’ll catch most of it. Our copy editor has problems figuring out what IS doing the action, especially if you refer to it in the next sentence. If you can change the sentence to state what method, the function, the .NET Framework, Visual Basic, or something else does, you’ll catch most of the problems.
* You didn’t provide the sidebars or pull quotes. We’ll kick the article back to you to provide them. They give us a way to visually break up a page, especially if it is very text heavy. We may not use all pull quotes, especially if they make the page look really busy. We can’t make that decision until we start layout.
* You didn’t put hard returns to break lines in your code snippets and code listings. Assume that you need to break about two characters sooner than the right margin of the grey or blue box. They are there to help you know where a line must break. Indent your code properly using the space bar to make it readable.
* Your Code Snippets and/or Code Listings don’t have the color formatting from the VS IDE. It is your responsibility to apply the style and then fix any color coding problems before you send the article to us. Yes, it could mean tediously applying color coding by hand, a few colors per line. If you don’t do it, that’s how we have to do it. Here’s a tip. Highlight code and change the color (blue for key words for example). Then go to the next key word and press F4. That will apply the last style to the next text you highlight. We hand-change all the keywords in blue in a snippet or listing. Then go through and make a pass for black text. Then for comments in green, then terms in teal, then things in quotation marks in red. That’s 5 passes through a snippet or listing to fix color coding. But it has to be right and that guarantees it is correct when it gets to layout.
* You want to emphasize something with a dash – it has a single space around it.
* You were inconsistent with your spelling or capitalization of terms. Please double-check this. The copy editor is not a programmer so he’ll miss whatever you miss. When in doubt about the capitalization of a term, follow the product documentation on MSDN Online. Remember to look at stuff written by the editors at Microsoft, not at blogs (which are a capitalization consistency nightmare from an editor’s point of view).
* You used styles in your article that aren’t provided in this template, such as styles from your book publisher. DON’T DO THAT. That takes TONS of time to fix on our side, and styles we don’t catch screw up the layout.

If you have questions, ask. That’s what we’re here for. We want you to look good in print.