



How to Easily Make Web Content That Actually Gets Traffic...

In this eManual I will walk you through the basics of creating web content. Whether you just want a personal website or you want to make widgets for your friends, this eManual will show you how.

In addition to showing you how to create the content, I will be showing you how to create content that gets real traffic by using simple techniques.

Table of contents

- [Introduction](#): *So you want to make web content.*
- [Chapter I](#): *Nvu, the easy way!* In this chapter we will get started in actually making your web content. We will be using the Nvu editor so we can make our content graphically, the easy way!
- [Chapter II](#): *HTML Basics*. I know, I know...you don't need to know HTML, but understanding the basics helps...a lot!
- [Chapter III](#) (Chapter 2 extended). *How to build content for search engines*. In this chapter I will go over what needs to be done in order to help your content do well in search engines like Google and Yahoo.
- [Chapter IV](#). *Advanced Topics*. In this chapter I will show you how to turn your web content into portable widgets. We also will go over the basics to Javascript.

Introduction

So, you want to make web content. There are many reasons people get involved with online projects. Some people like to build online family trees, others like to have hobby websites. Some individuals take the fairly lucrative step to setting up an online business. The latter is beyond the scope of this manual as the purpose of this manual is to strictly help you learn to build content that will get traffic.

Web content comes in many different forms. In this manual I am going to discuss the three most popular:

- *Website*
- *Blog*
- *Widget/Gadget* (more about this later)

Some of the topics we cover will require some actual personal "study time" in order for you to become familiar with them. Don't worry though, if you spend a small amount of time each week on this you will be making all kinds of web based content.

When it comes to making web content there are generally two routes to follow. You can either write the HTML code yourself or you can build graphically using an HTML editor. I will go over some of the basics to HTML (more about this later) but for all intents and purposes I will be showing you how to make the content using the WYSIWYG editor *Nvu*. WYSIWYG is an acronym for 'what you see is what you get'. This is a simple graphical interface you can use to build your content. You can get Nvu by [clicking here](#).

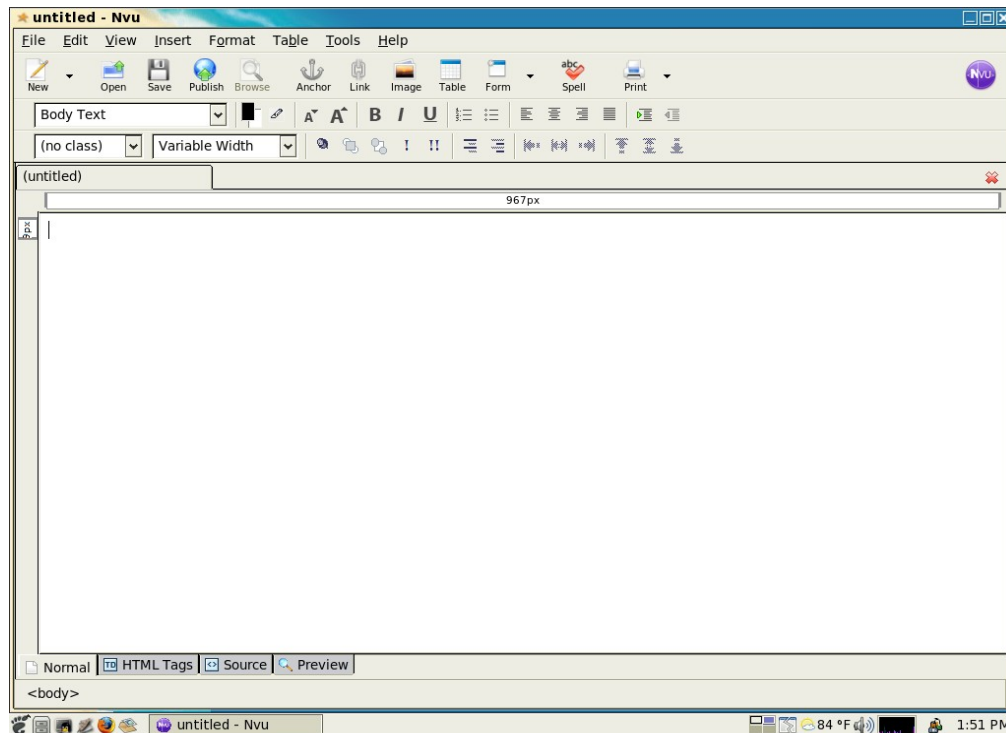
You should download and install Nvu so you can follow along with the instructions in this manual. We used many images in order to help you "see" what you need to do to get started. So...

Let's get started making web content!

Chapter one

In this chapter we will be using Nvu extensively to build HTML content. If you haven't already obtained a copy of this program [click here](#) to do so now. We also will be using the example web template provided with this eManual as we build our example web content.

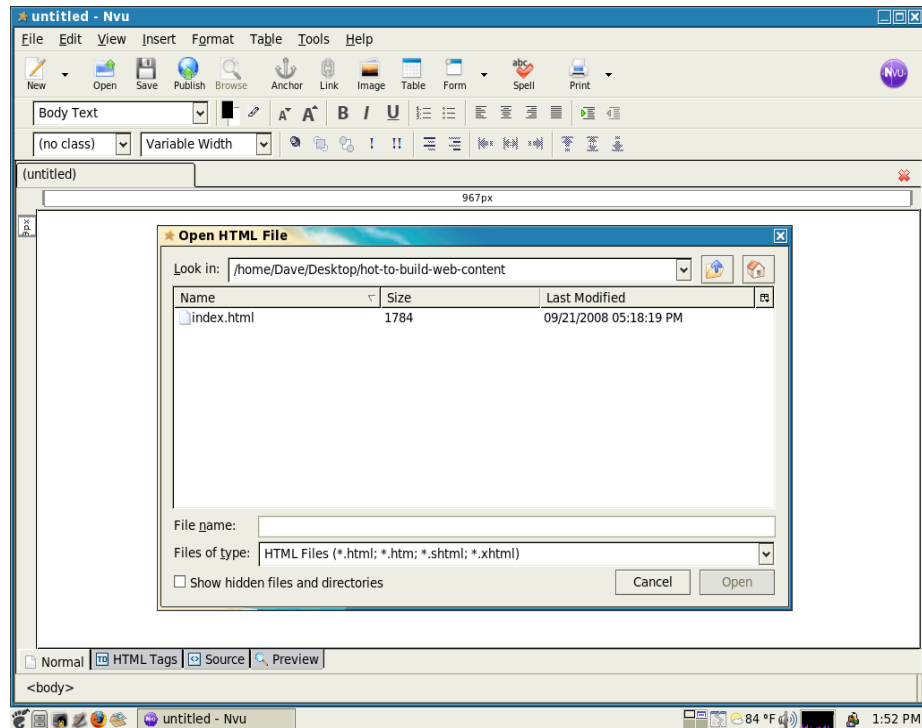
Now, run the Nvu program. The program has a toolbar that contains buttons which aid you in accomplishing the most often completed tasks.



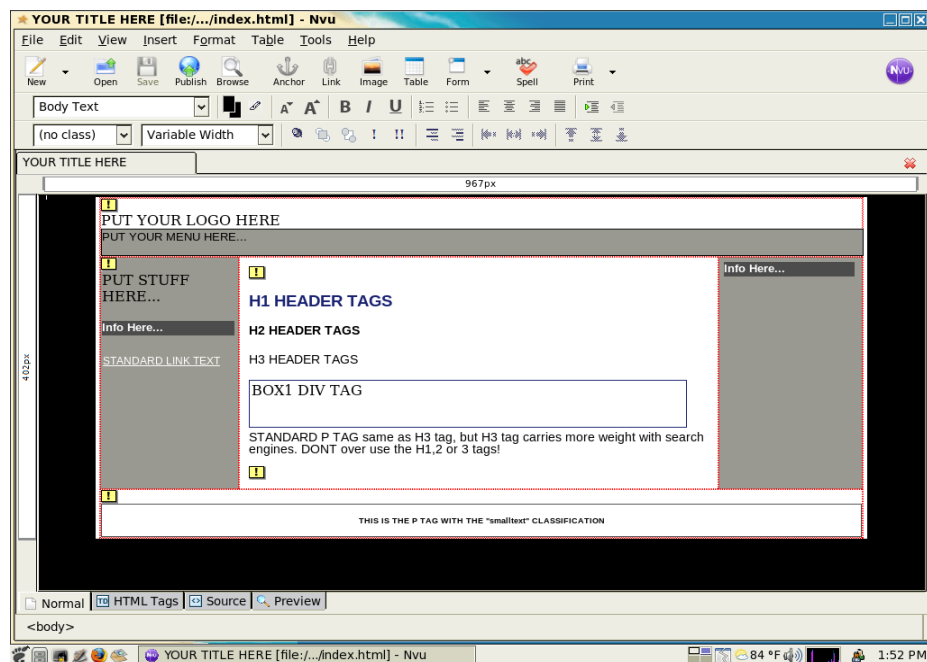
Now, we will go over a few of the basic features to this program to help you navigate around. Remember, it will require a bit of personal study to get proficient with Nvu. Look at the above image, the toolbar across the top of the interface contains buttons that will help you do things like, insert images, add links and create tables to help you organize your content. On the bottom of the interface you will see four tabs; Normal, HTML Tags, Source, Preview. In general, when creating your content you should use the "Normal" mode. This allows you to see what your content looks like as you make it. We will go over the other tabs a bit later.

We have provided a web template for you to use to be able to quickly get started with this. You can alter the template as you see fit, but please make a copy/backup of the template files BEFORE you edit them! This way, if you mess something up you can easily start over again. I can't stress this enough! Always back up your files, especially before editing them.

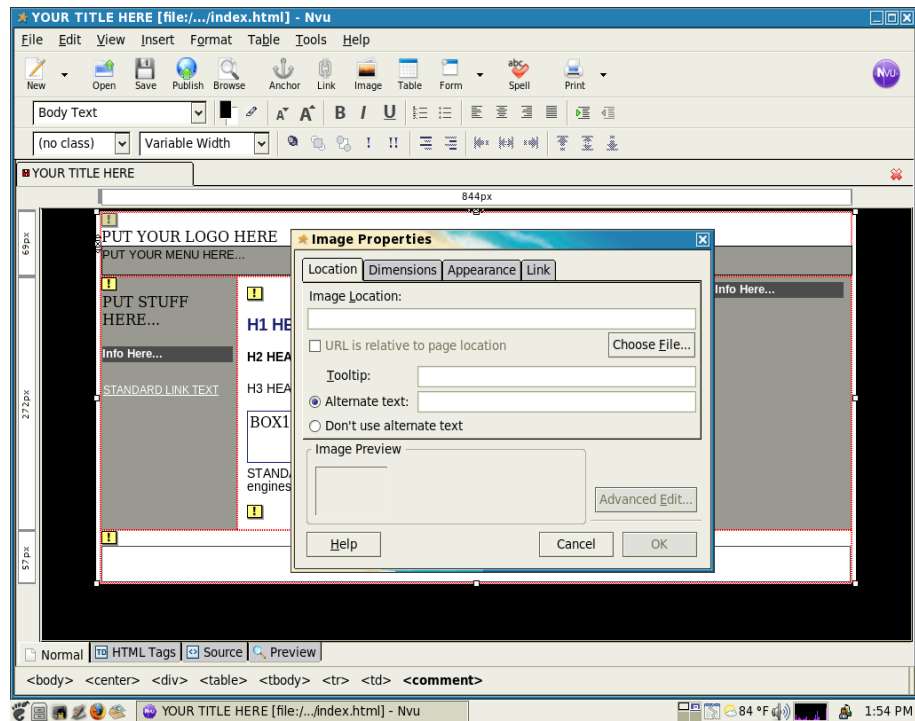
Now let's open up the example template. Click the "Open" button located on the toolbar. A window will open asking you what file you want to open. Using the drop-down list, navigate to the *index.html* file located in the *example-template* folder.



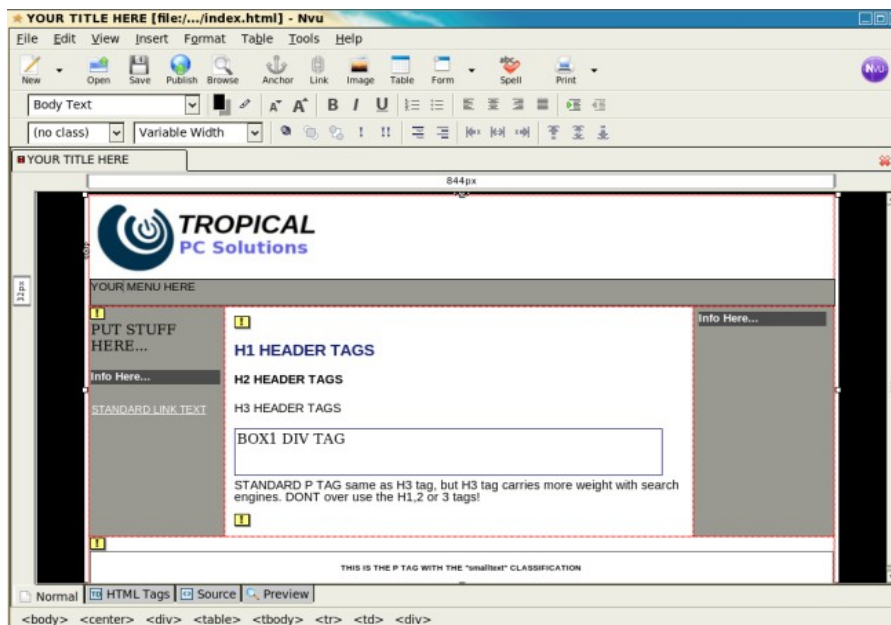
Once you have located the file, click the 'open' button on the bottom of the little window. You will now see the example web template. It is the same template we use here at [Tropical PC Solutions](http://TropicalPC.com).



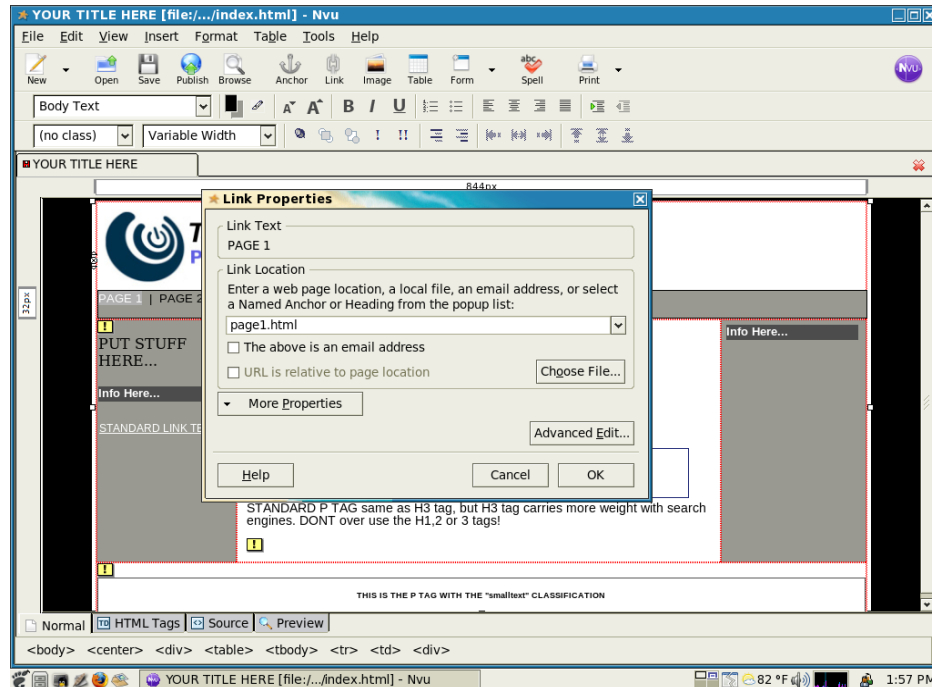
Once you have the example template loaded into the Nvu interface you can begin to modify the template to suit your needs. For instance to add your logo, click your mouse where you see the words "PUT YOUR LOGO HERE". Once your cursor is located at the place you want the image to go click the 'Image' button on the tool bar:



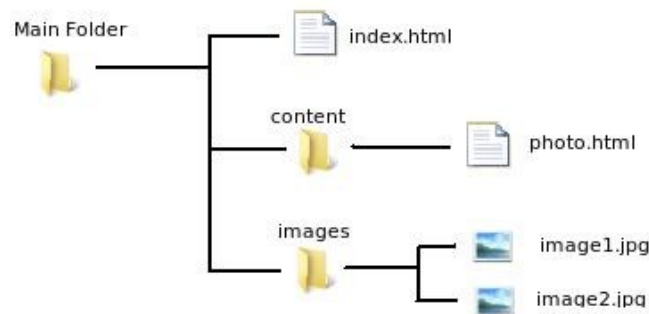
It is very similar to loading the template. A small window will open asking you what image you want to load. Click on "Choose File" and navigate to your image. Once you have chosen your image, click the "OK" button on the bottom of the small window.



Your logo will now appear in the template. To create a link, click your mouse cursor where ever you want the link to be. For this example we will be making our site menu. So, click on the "Link" button located on the Nvu toolbar.

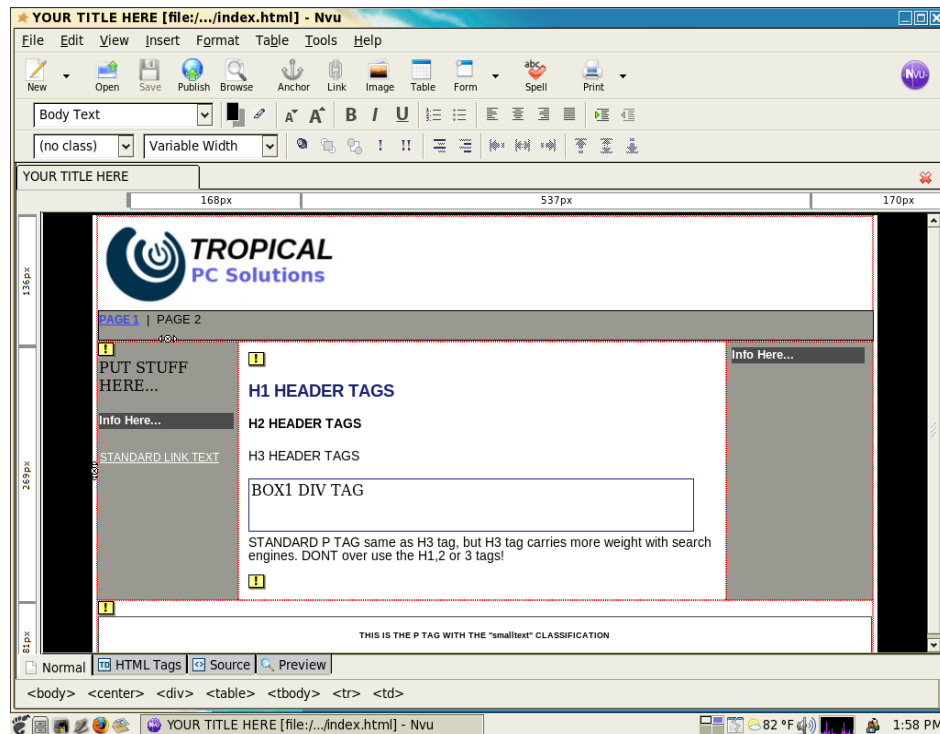


A small window will open asking you where you want the link to point to. In other words, when people click the link, where will they go. Put the name (or location) of the page you want to link to. If you want to link to Google, put <http://www.google.com>. This is called an absolute URL. If you want to link to another page within your own website you can use relative URL's. So, to link to your photo page you would put *photo.html* if photo.html is in the same folder as your index.html file. If your photo.html file is located in another folder called *content* you would then put *content/photo.html*.



Let's say you were building your *photo.html* page. If you wanted to load *image2.jpg* into the content of that page you would put: `../images/image2.jpg`. The `../` part means, "go back one directory."

Once you have created your link you will see it in the Nvu interface. the link will appear where ever you had your mouse cursor at.



You can copy and paste text into the interface or you can simply type what you want. When you have your page the way you want it click the "Save" button on the Nvu toolbar. This will save your changes. You just made a webpage!

You should spend a few days getting familiar with these actions. It may seem a bit overwhelming at first but it is really quite simple with a bit of practice.

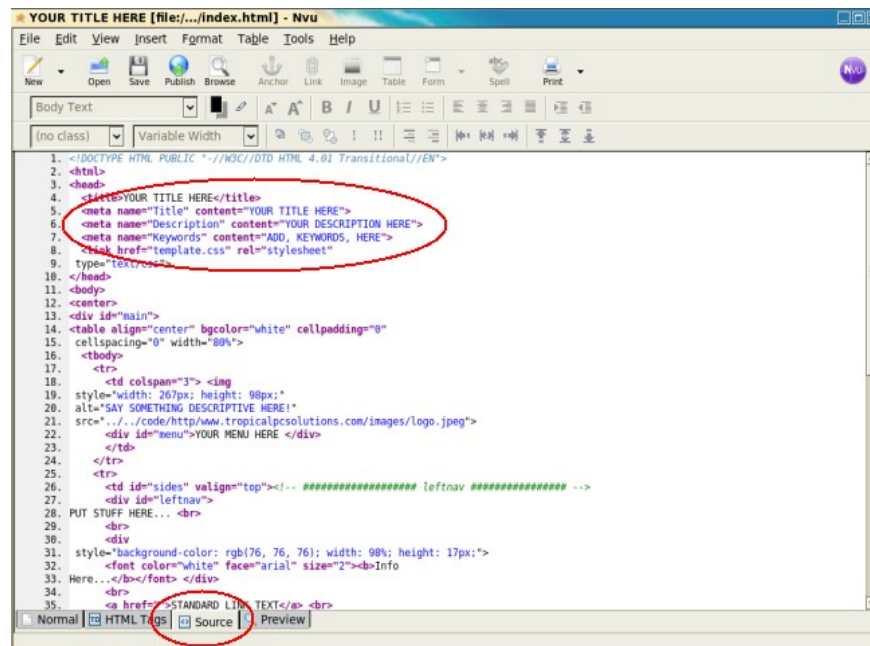
Another place you can practice your newly found webdesign skills is a blog. Blogs can be a primary project or an extension to an existing project. A blog gives you the option to set up email newsletters which is another place you can practice your design/coding skills. If you would like to start blogging I recommend [blogger](#).

Now, go and take a well deserved coffee break :)

Chapter two

Although the purpose of this manual is to show you how to graphically build content, you still should be familiar with HTML basics. Doing small things can help you with search engines greatly! This chapter and the next are closely related and tend to flow together.

HTML stands for *Hyper Text Markup Language*. It is not really considering programming. HTML is for formatting text and other web content. To see the HTML of your Nvu files click the "Source" tab on the bottom of the Nvu interface:



What you are looking for here is the META tags located in the *head* section of the document. You see, all HTML files have two main sections:

- *Head - Informative tags go here*
- *Body - The pages content goes here*

Search engines look for certain META tags in order to determine what the content is about. A "tag" is anything between the < and > symbols. HTML tags also need closing tags. For instance, an example HTML skeleton looks like:

<html> - this tag begins the html document

<head> - this tag begins the head section

</head> - this tag ends the head section

<body> - this tag begins the body section

</body> - this tag ends the body section

</html> - this tag ends the html document.

META tags belong in the head section. The search engines will use your TITLE, DESCRIPTION and KEYWORDS tags to determine a number of factors about your content and its relative placement within the search engines results. Let's go over these tags:

- TITLE
- DESCRIPTION
- KEYWORDS

There is two 'title' tags. The standard title tag and the META title tag. Both should say the same thing. The standard title tag will be displayed on the top of the browser window and the META title tag will be shown in the search engine results. The (META) title tag is very important, as are the other tags, so be sure to include them in ALL of your HTML documents.

The description tag should describe your content. This will most likely change from page to page depending on your various content.

The keyword tags is one of the most important and abused! There are a number of rules to correctly using keywords in your document which are very important. Keep in mind that keywords are to help search engines match your content with relevant searches. In other words, If your content is about Florida beaches, your keywords might be: florida, beaches, oceans, sun. You are hoping that when people go to Google and type in "getting sun in florida" or "florida beaches" that Google will consider your keywords and page content relevant and then match your content with search queries.

- Your keywords should be relevant to your page content
- Both your TITLE and DESCRIPTION tags should include keywords
- You should use your keywords in the body of your content.
- You should use keywords in the "Alt" attribute of the image tags
- DON'T over use your keywords (keyword stuffing), 10% of content at most.

There are a few other tags that are of some importance to your content. These other tags belong in the body section because they are directly formatting your content.

- `` - Bold text
- `<h1>` and `<h2>` - Headers
- The "alt" attribute in the image tag ``

When using your keywords within the content of your pages it is very helpful to include them in either `` or `<h1>` (also `<h2>`) tags. When text is made bold, the search engines will assume it is important to the overall content. The same goes for the header tags. Your page titles and such should be formatted using the `<h1>` or `<h2>` tags.

When placing images within your content there is an option to add "Alternate text". This is text that would otherwise appear if the browser couldn't load your image. In other words, if a person visits your page and the image doesn't appear, your "alt" text will. The alt attribute gets added to the image tag like so:

```

```

Chapter three (Chapter 2 extended)

When you consider search engines there is more to think about than just your content. How other people view your site carries a lot of weight to search engines. In other words, if a lot of other sites link to yours, then Google will assume your content is important (generally speaking). These external, 3rd party link are called backlinks. You don't always have control over how someone else will link to your content but when you do have some say in the matter, you should make sure that the words they use to link to you also include your keywords. When you link to your own content always be sure to use very descriptive anchor text to describe the content that is being linked to.

Links are also really important to search engines. Search engines companies use a program called a spider, crawler or robot. I will use these terms interchangeably. These programs will search your website for it's description, content and links. These robots like to find sites that are easy for them to navigate. The best two tips to helping the robots navigate your page are:

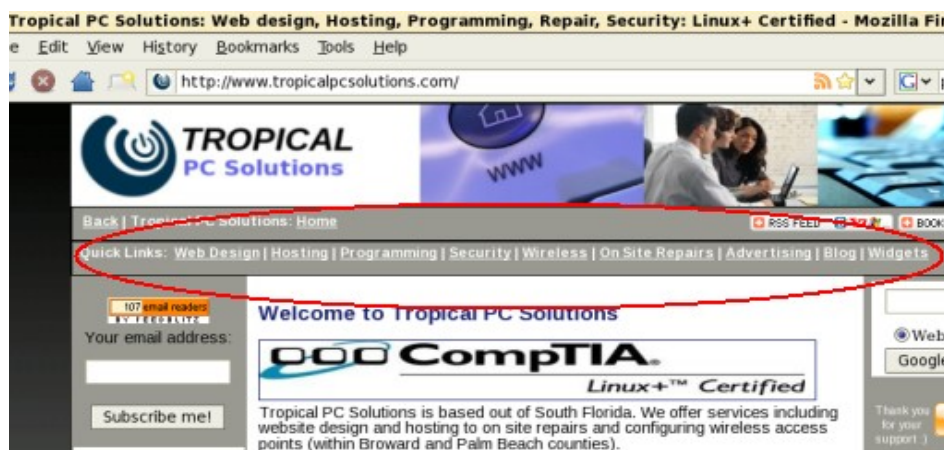
- Use simple URL's
- Create a sitemap

A simple URL is a link that looks like: www.yoursite.com/content/mypage.html

A NON simple URL looks like: www.yoursite.com/content/mypage.php?your=foo&data=bar

Some spiders have trouble following the latter so be sure to make an effort to use simple URL's. Another important aspect of linking to your content is the sitemap. The sitemap is a collection of all the internal links that are found in your content. There are a number of online resources that will generate a sitemap for you. I generally use [this site](#) a lot. This site will create XML and HTML sitemaps so you can add the HTML content to your site and the XML version is for [Google webmaster tools](#).

Another important aspect to consider is that you want all of your pages to link to all of your other pages. Take our main site as an example:



Look at the menu on the top. Each page has that same menu so, if a spider came to that page, it could crawl those links and find most of our content from that one location.

Now, go and take a well deserved coffee break :)

Chapter four

Have you caught the widget wave? Widgets are everywhere! Exactly what is a widget and why should you care?

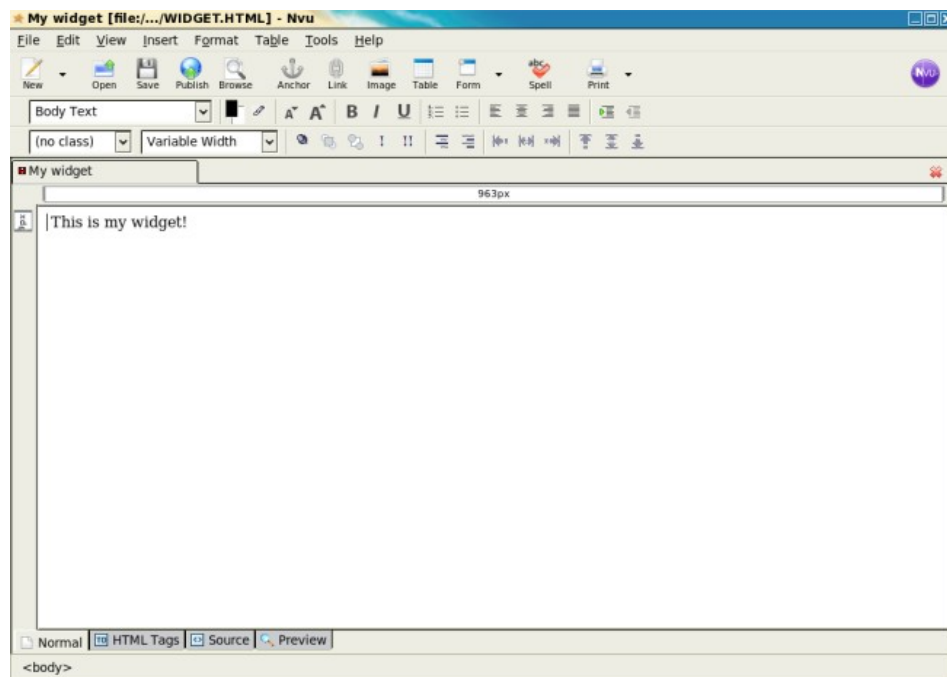
Widgets (also known as gadgets) are small portable web content. You can make a little webpage and wrap that up in an XML widget container. Others can then place this 'container' in their website, blog, or public profile. In turn this can generate a lot of free traffic!

Here we will be making a widget from start to finish. To get started you will need to locate the example widget files located in the *example-widget* folder provided with this download.

In this folder you will see two files:

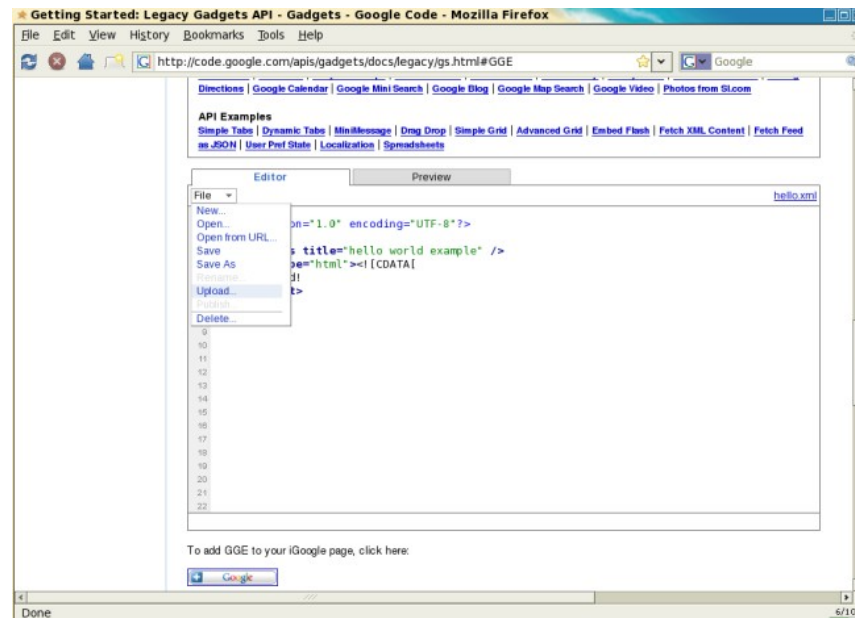
- *widget-template.xml*
- *WIDGET.html*

The general idea for building a simple widget is the same as a webpage. In fact it IS a webpage, just a very small one. So you make a small website, 300 pixels wide by 500 pixels high (you can size it what ever you want, but that is pretty standard. So, let's open Nvu and load the WIDGET.html file:

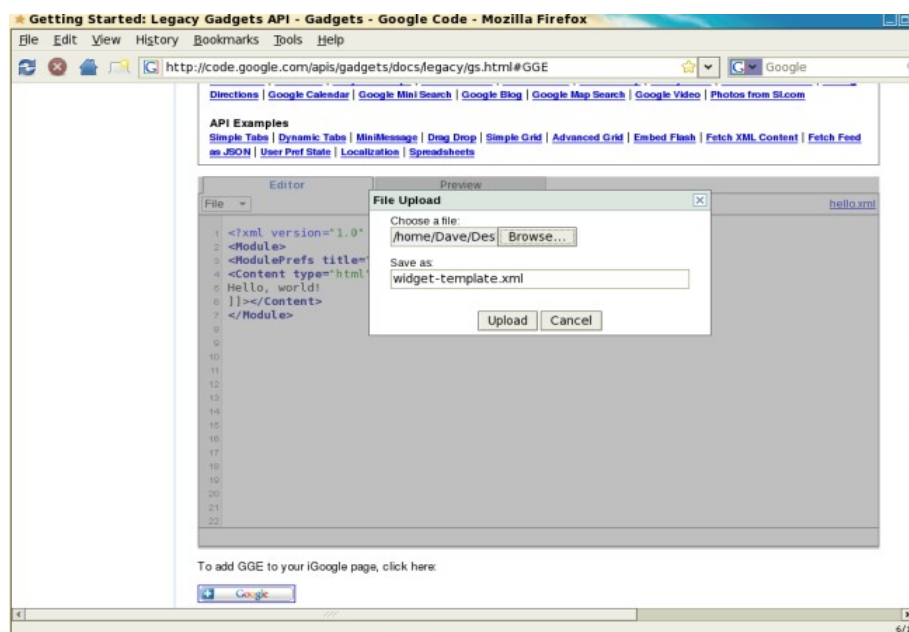


You can make what ever you want, but for the sake of this manual we will keep it as is...

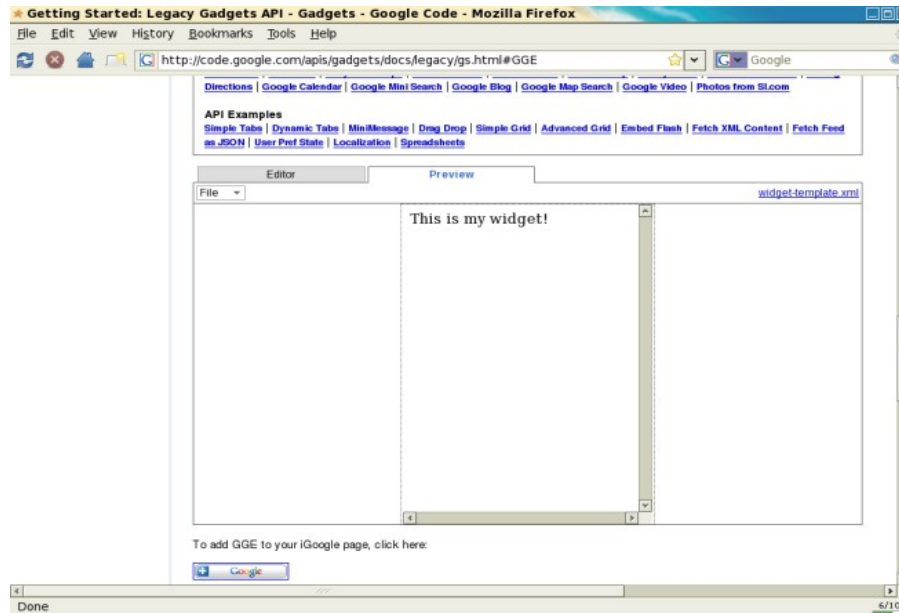
After you have the WIDGET.html file the way you want it, upload/transfer it to your host. Now open the *widget-template.xml* file. make the necessary changes (where you see all CAPS). Make sure to put the correct URL in the content url section! If you put your widget file at <http://www.yoursite.com/widget.html> than use that address for the content url. Once you have finished editing the *widget-template.xml* file you can open the Google Gadget Editor by [click here](#). When you have the GGE open you will see the following screen:



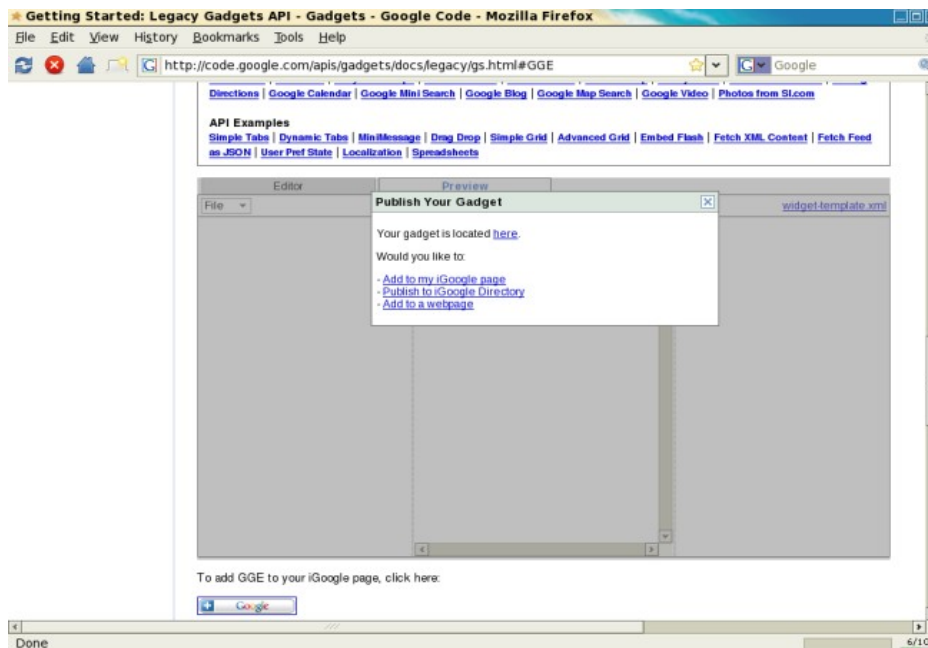
Click on the drop down menu called "File". Then choose "Upload". A little window will open asking you what xml file you want to upload:



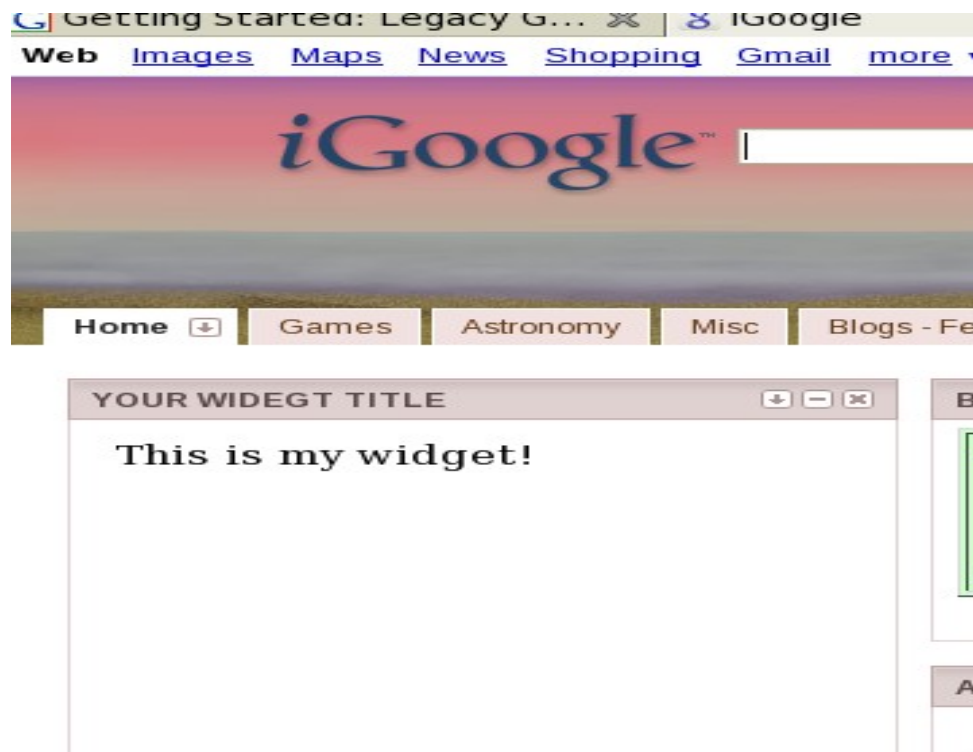
Once you have located the file you want to upload, click the upload button. This will upload your xml file to the Google servers and it will then let you modify or preview the widget:



Once you have your widget (gadget) the way you want it you can publish it. To publish your widget click on the drop down menu called "File", choose the "Publish" option. Google will give you three options.



Here we will publish our widget to our [iGoogle](#) page:

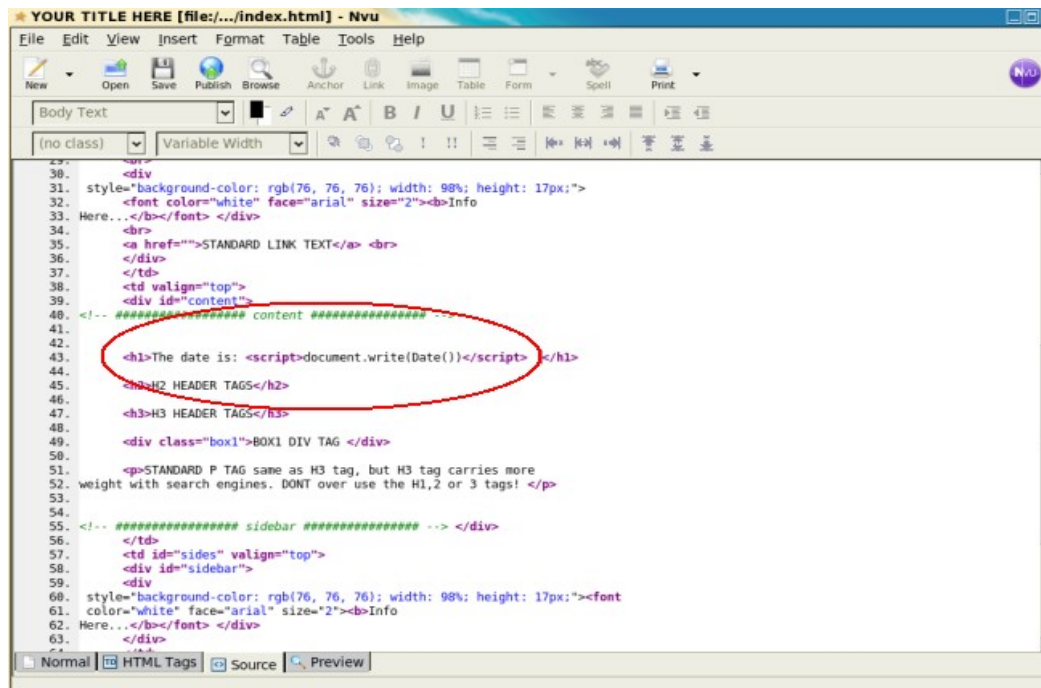


If you like what you see, go back and choose to publish your widget to the iGoogle directory. Here millions will see it and hopefully thousands will use it! Not only can you place these widgets on your iGoogle pages. You can embed them into your website or blog too!

There will inevitably come a time when you want to add some kind of interactivity or dynamic content to your webpages, blogs and/or widgets. This is easily done with Javascript. Javascript is a programming language used within web content. Although Javascript programming is beyond the scope of this manual it does need to be mentioned to some extent so you at least know what it is and where to look when you are ready to add these features.

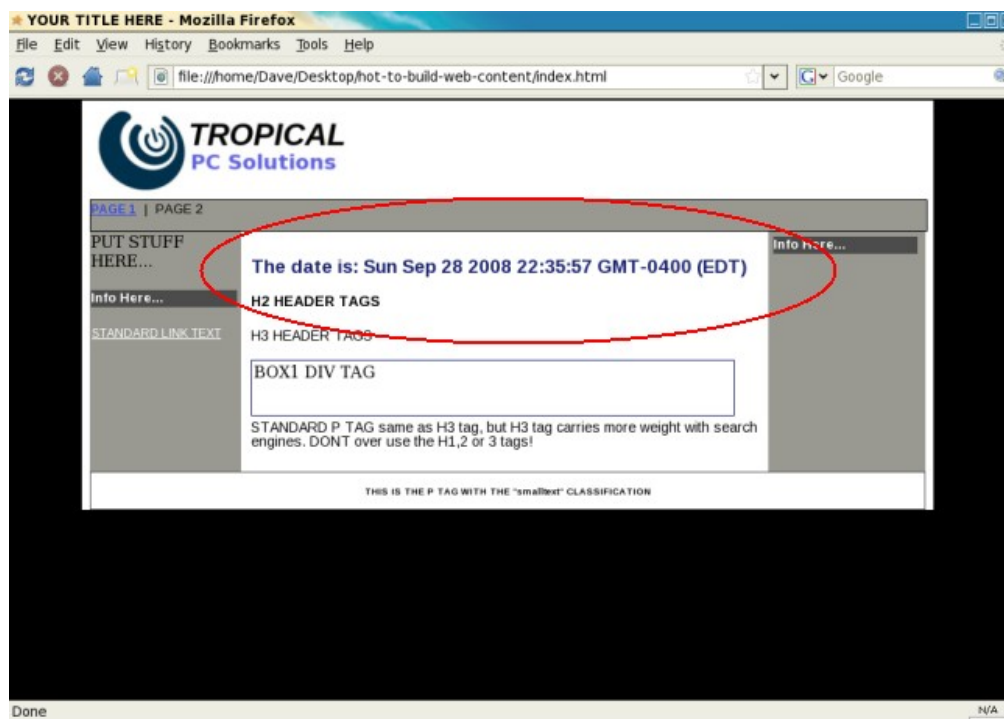
We will go over some simple example code and brief explanations as to what the code does.

- Display current date: `<script>document.write(Date())</script>`
This code will display the current date. You add this code to the body section of your HTML source (see images on next page)

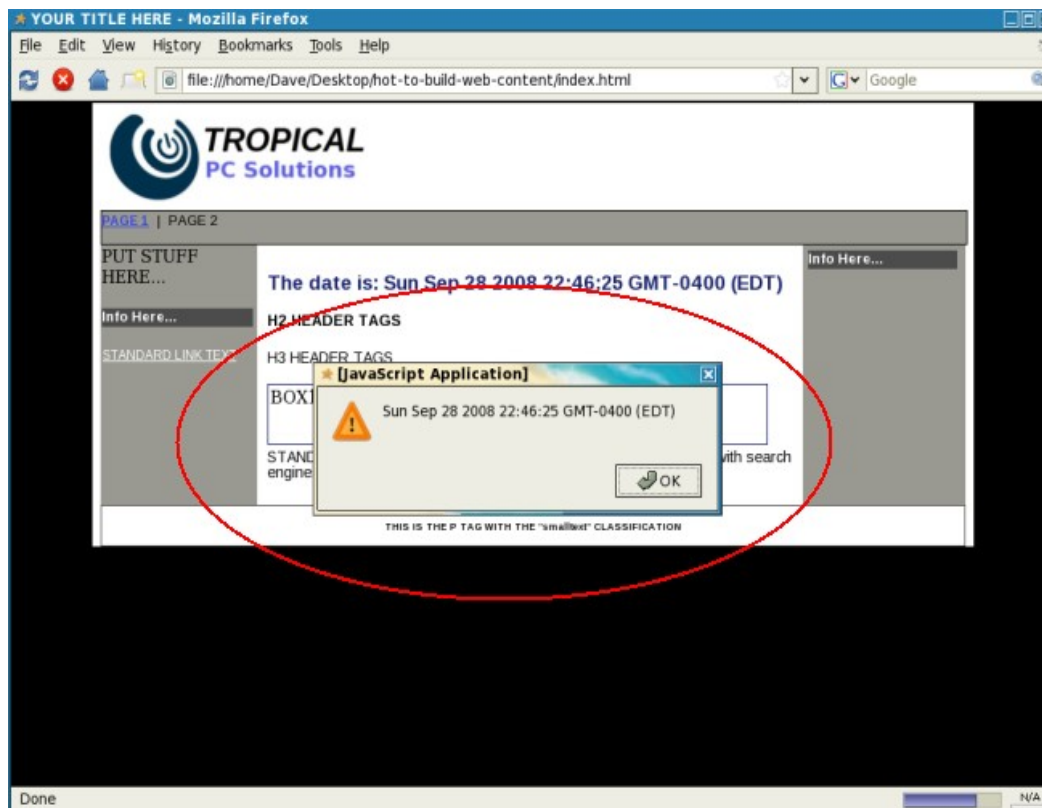


```
30. <div style="background-color: rgb(76, 76, 76); width: 98%; height: 17px;">
31. <font color="white" face="arial" size="2"><b>Info
32. Here...</b></font> </div>
33. <br>
34. <a href="">STANDARD LINK TEXT</a> <br>
35. </div>
36. </td>
37. <td valign="top">
38. <div id="content">
39. <!-- ##### content ##### -->
40.
41. <h1>The date is: <script>document.write(Date())</script> </h1>
42.
43. <h2>H2 HEADER TAGS</h2>
44.
45. <h3>H3 HEADER TAGS</h3>
46.
47. <div class="box1">BOX1 DIV TAG </div>
48.
49. <p>STANDARD P TAG same as H3 tag, but H3 tag carries more
50. weight with search engines. DONT over use the H1,2 or 3 tags! </p>
51.
52. <!-- ##### sidebar ##### --> </div>
53.
54. </td>
55. <td id="sides" valign="top">
56. <div id="sidebar">
57. <div
58. style="background-color: rgb(76, 76, 76); width: 98%; height: 17px;"><font
59. color="white" face="arial" size="2"><b>Info
60. Here...</b></font> </div>
61.
62. </div>
63.
```

Once you have entered your date script (we used the header tag here) you can save your document. Remember, Javascript is interpreted by your browser. In other words, Nvu doesn't know how to read Javascript. You have to open the file with your web browser to see the Javascript work:



- The Javascript 'alert' function: `<script>alert("Hello from Javascript")</script>`
Or you could use the date function: `<script>alert(Date())</script>`



Those little pop up windows you see on the web are a result of Javascript programming. There is a lot more you can do with both HTML and Javascript.

When you are ready to begin adding complex Javascripts to your web content you will want to look at a well established Javascript resource site. We highly recommend [HotScripts](#). HotScripts is a huge script site with anything you can think of. To visit HotScripts [click here](#). You can also visit our Javascript website by [clicking here](#).

Now, get busy building content!