

Step 1: Installing and Configuring Apache 2

Introduction:

Before you begin, having a firewall up and running, such as Zone Alarm, can cause problems with Apache installations. I recommend that you turn off Zone Alarm before starting this tutorial. You do not need to turn off Microsoft's built in firewall. We will focus on configuring Zone Alarm after we get Apache 2 up and running. If you have communication problems down the road, turn off Zone Alarm and see if the problem persists. (NOTE: configuring zone alarm hasn't been addressed yet).

I'm using AVG free edition anti-virus and this installation works fine with AVG running.

The first step in this process is to visit the Apache 2 download pages at apache.org.

Apache Download Page:

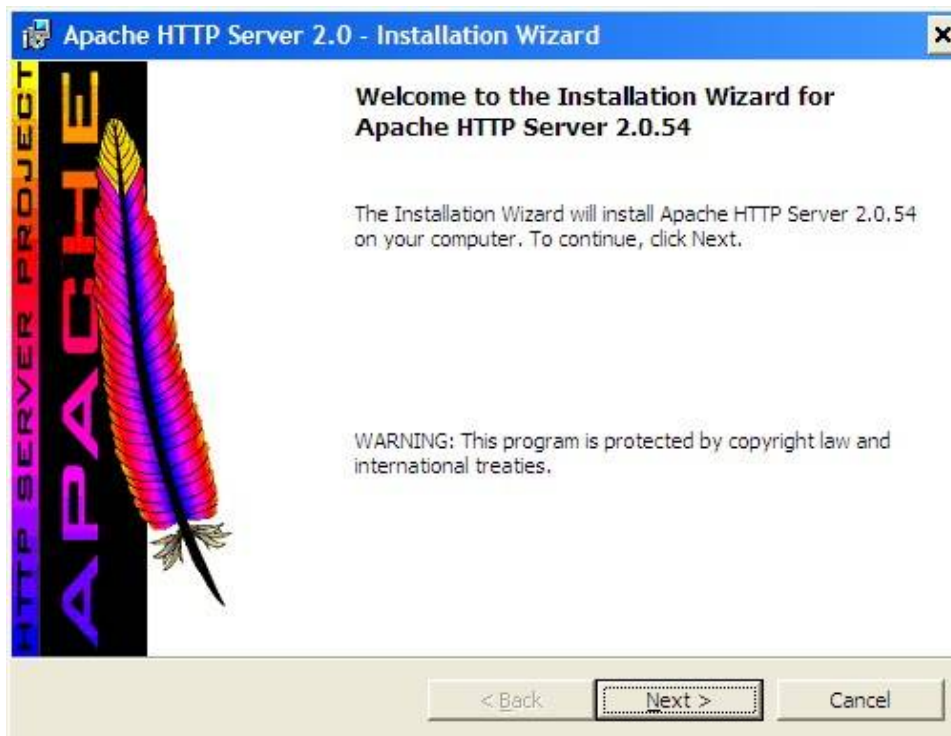
<http://httpd.apache.org/download.cgi> - Choose the binary install version.

In my case:

Win32 Binary (MSI Installer): [apache_2.0.55-win32-x86-no_ssl.msi](#) [PGP] [MD5]

Double click on the Apache 2 installer. You can download the MD5 and test it against the MD5 of the file.

You will see the following:

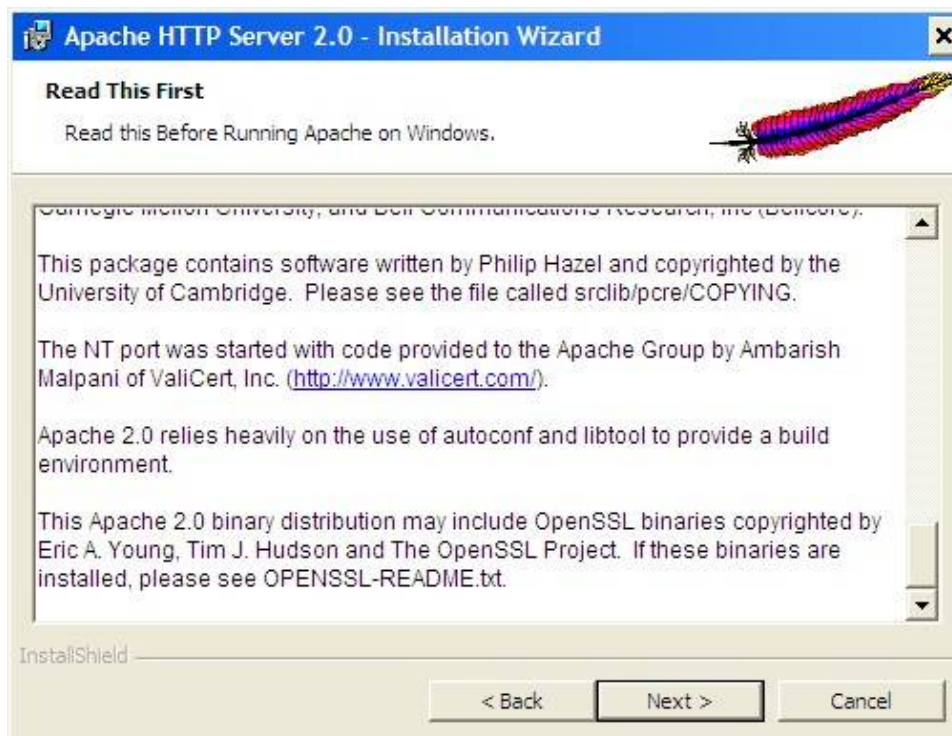


Press Next >



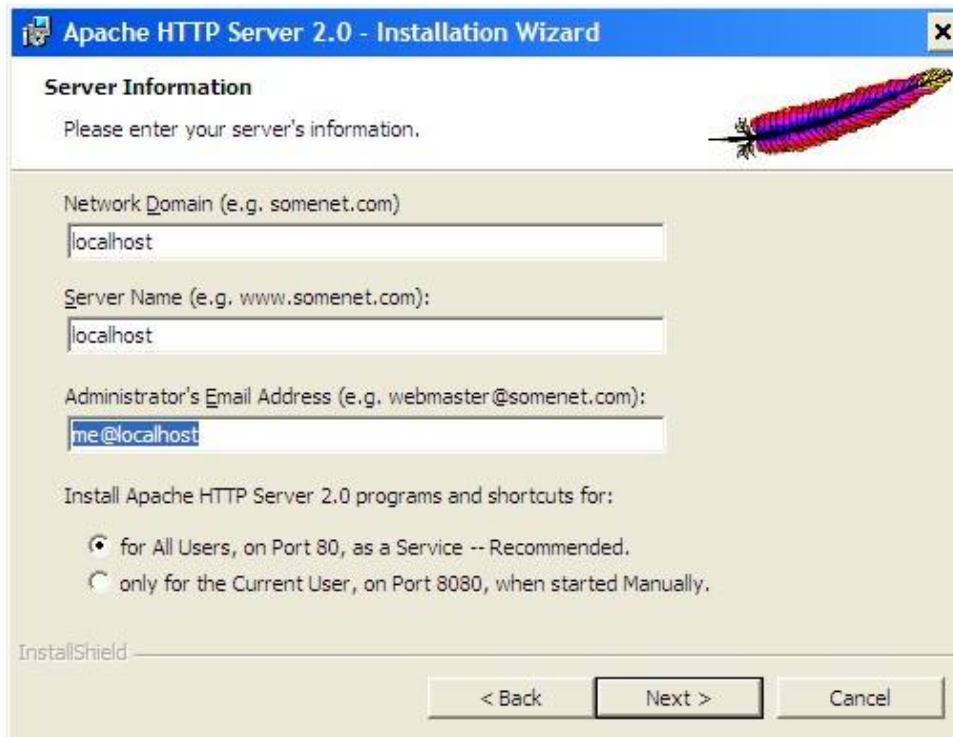
Click "I accept the terms in the license agreement" radio button (after you have read the license and agree to it).

Click Next >



Read the “Read This First” text.

Click Next >



I'm setting this up as a development machine – in other words, the server will reside on my own machine. I am not setting this up on a network. If I were setting this up on a network, I would enter something along the lines of:

Network Domain: mydomain.net

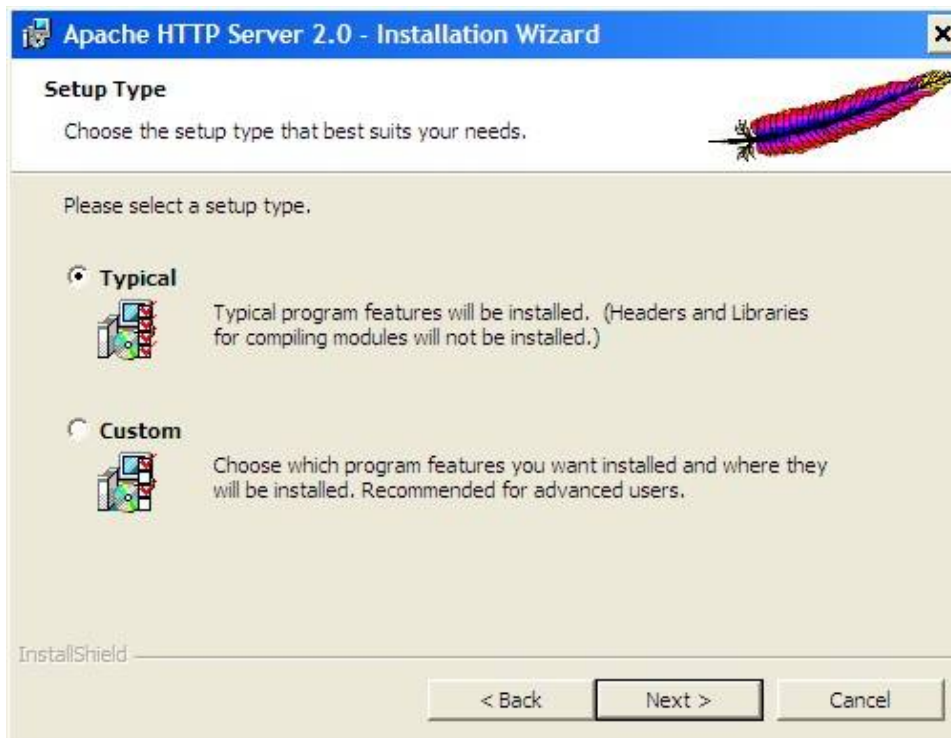
Server Name: servername.mydomain.net

Administrator's Email Address: admin@mydomain.net

I will go over setting up virtual hosts later in the tutorial.

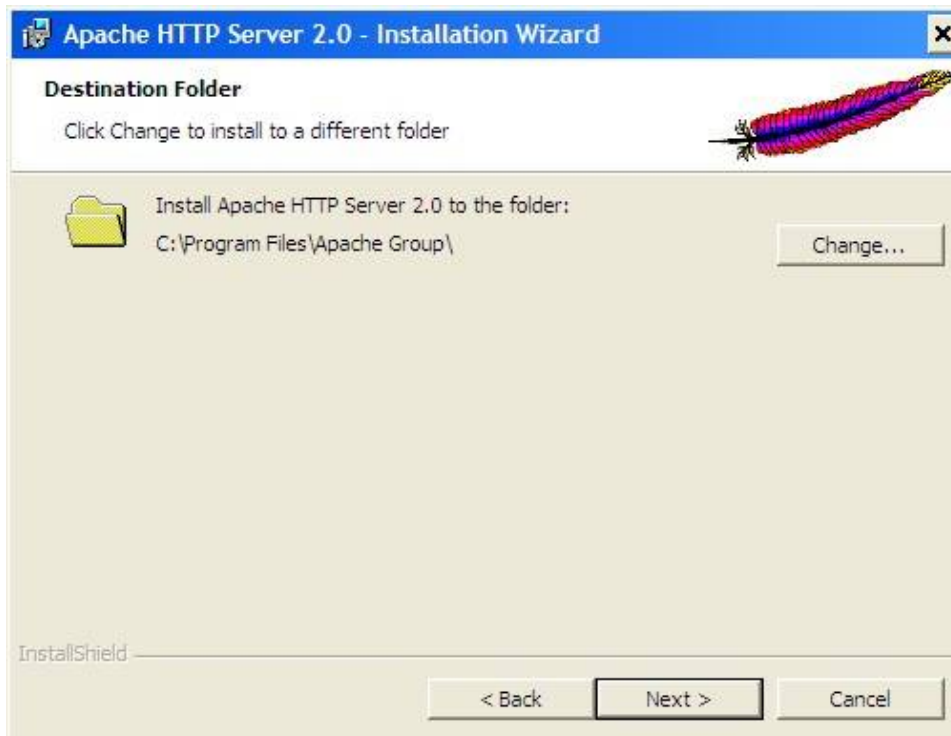
Notice, I'm also setting up Apache to run as a service. I've found this can make a lot of sense if one does a lot of development. My last installation of Apache 1.3 and PHP 4 was not set up as a service and I had to manually start up Apache every time I developed – which was almost all the time I fired up my laptop.

Click Next >

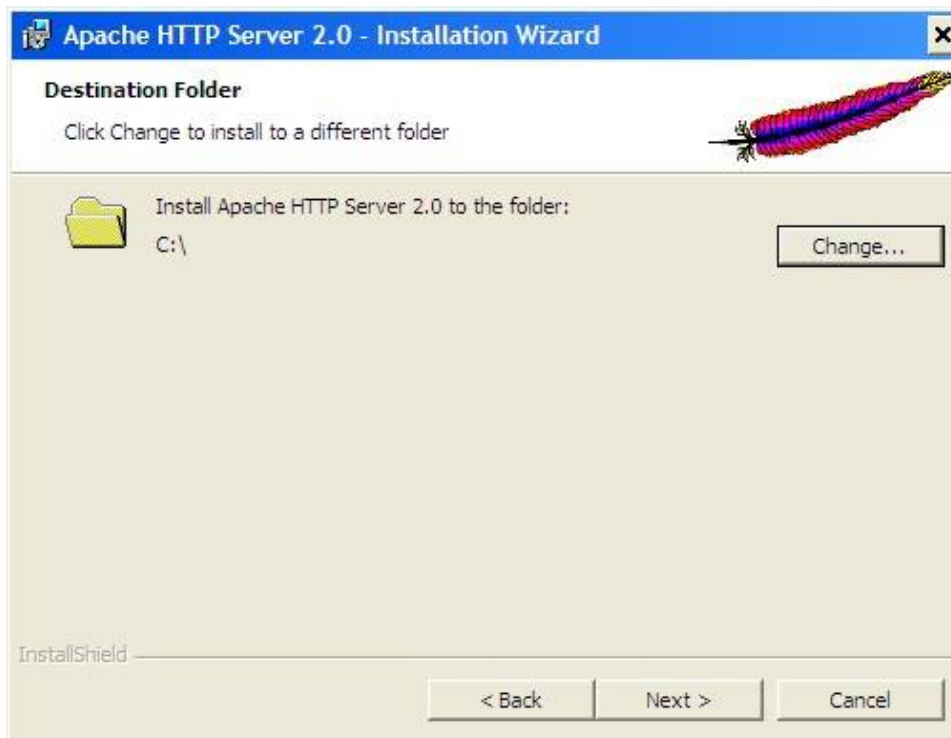


Choose the Typical setup radio button unless you really know what you are doing and have a need for a custom setup.

Click Next >

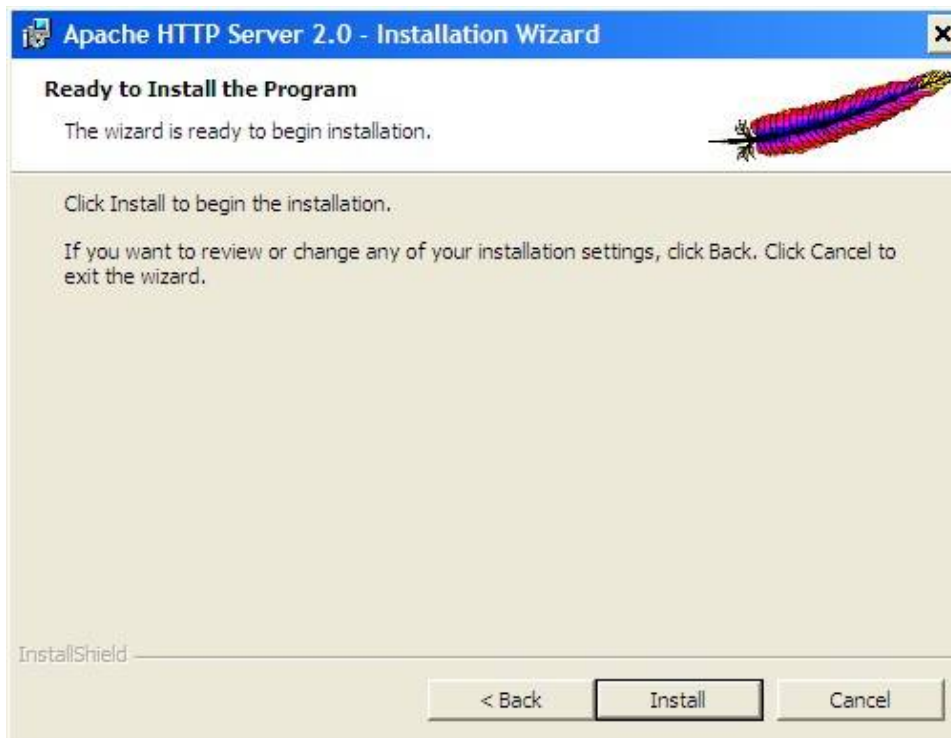


The default installation directory is C:\Program Files\Apache Group\. I have read where people have run into path problems due to the space character between “Program Files” and also between “Apache Group”. In order to be cautious, I'm going to change the installation directory to the “C:\” drive. The installation will create “C:\Apache2\” and install its files in that directory – the name is nice and clean with no spaces. Click “Change...”, browse to your C: drive, add an “Apache2” folder, enter the folder and click “OK.” You can install it in the program files, but make sure you know if and how you should quote “program files” in your path names.



That looks better!

Click Next >



Click Install to begin the installation. Let the installer do its thing. You should see several cmd style windows open and close. If you have a problem during installation, you can also [google](#) for the error messages that you receive (it works best if you enclose the error message in quotes and don't forget to leave off information specific only to your computer) – you will likely find out why the error occurred and what to do to fix it. If you have a problem during installation, you may have downloaded a corrupted file. Try downloading the file again and following these steps again.

Eventually, you may (or may not) see the following:



Click Unblock.

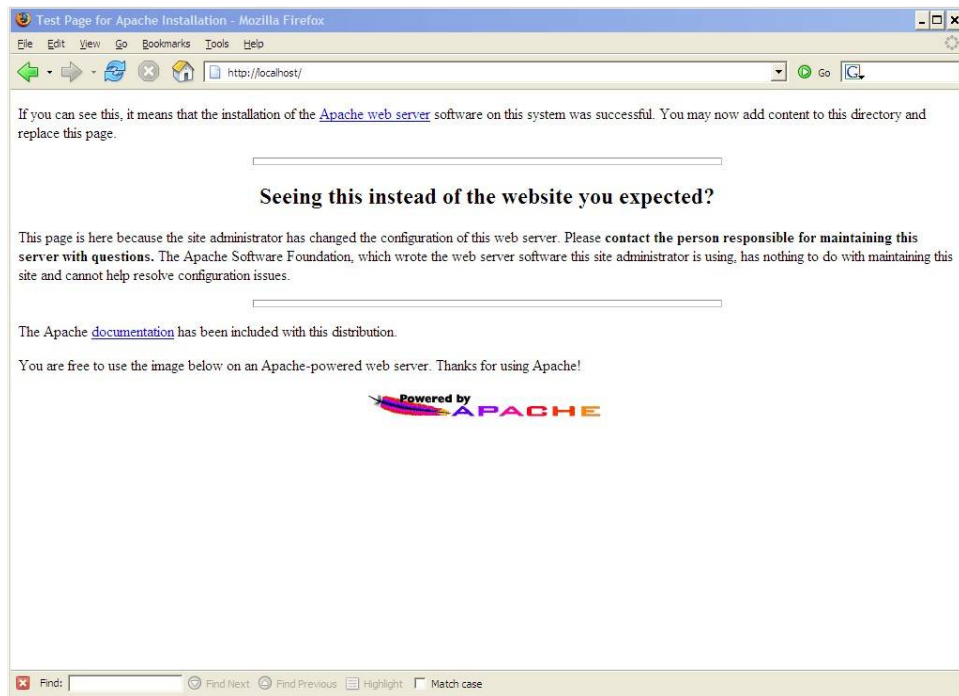
The following should be visible on your screen or should be available by switching to its window:



Click Finish.

If everything went well. You should see this feather  in your system tray (lower right hand corner of your screen just left of the clock).

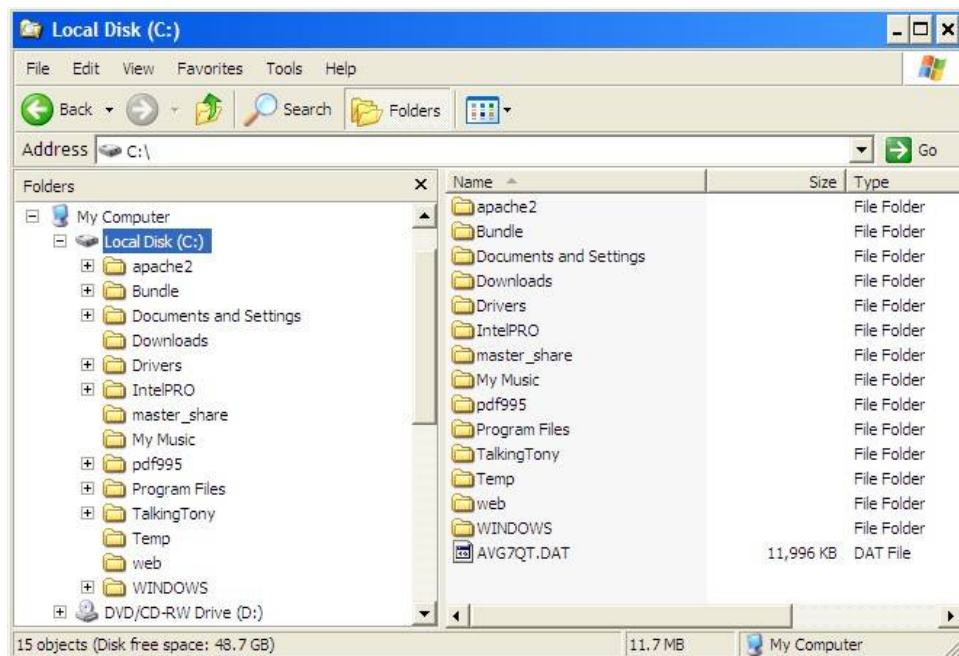
Open up a browser (you are using [firefox](#), right? ;-)) and type <http://localhost> into the address. You should see the following in your web browser:



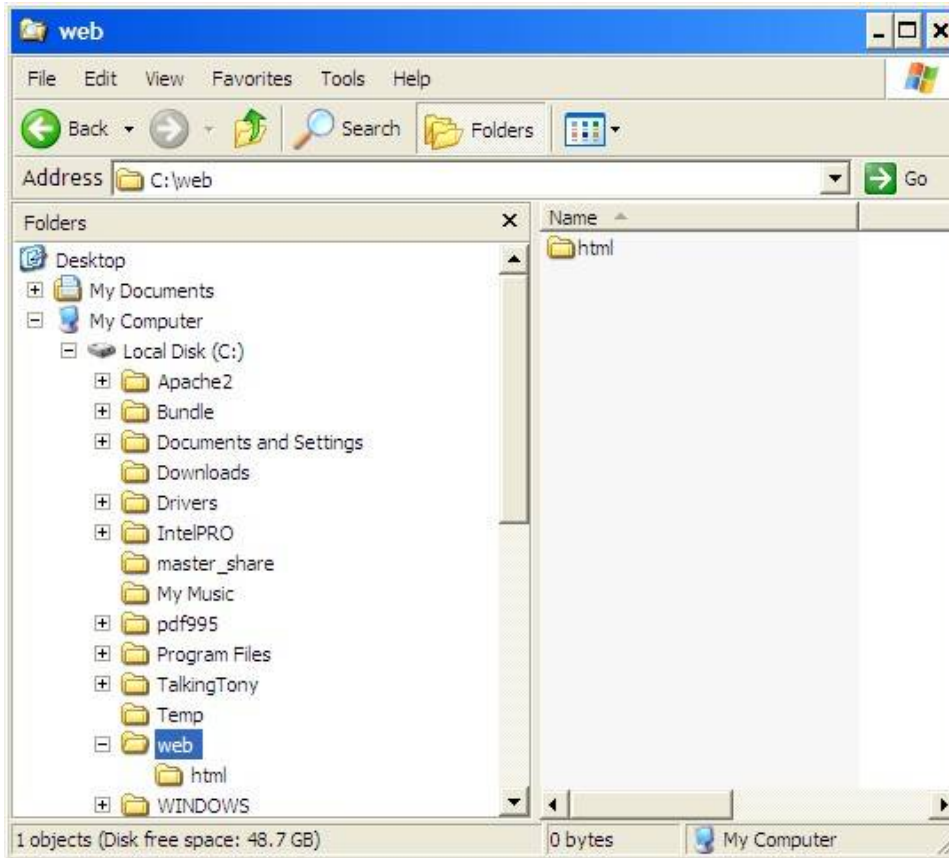
Did you make it this far? Whew! So far so good, but we still have some more things we still need to accomplish.

In order to make managing our web page(s) easier, we want to set up our web page(s) directory directly under the C: drive instead of its current location – C:\Apache2\htdocs\.

Create a directory named “web” in the C:\ drive yielding C:\web\



Create a directory named “html” under “C:\web\”. This yields “C:\web\html”.



We will store our html related files here (this will include php files, too). We will eventually create other directories under “C:\web\”.

Open notepad, copy the following green text and paste it into notepad.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">

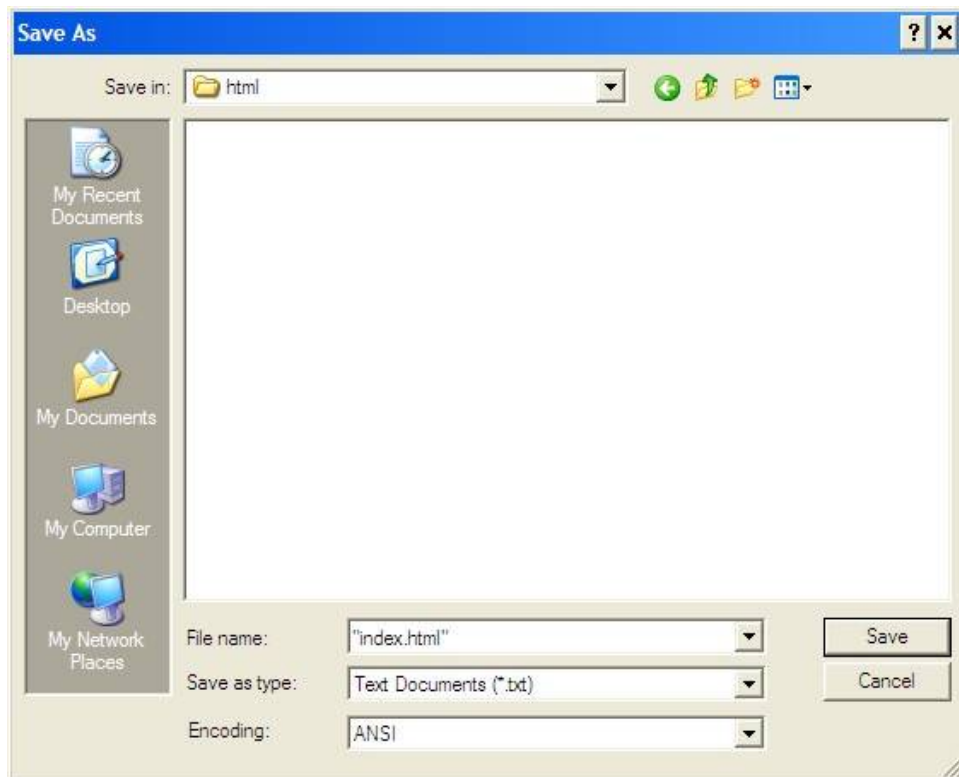
<html>
  <head>
    <title>Apache Rocks!</title>
  </head>

  <body>

    <p>My apache works in my new folder!</p>

  </body>
</html>
```

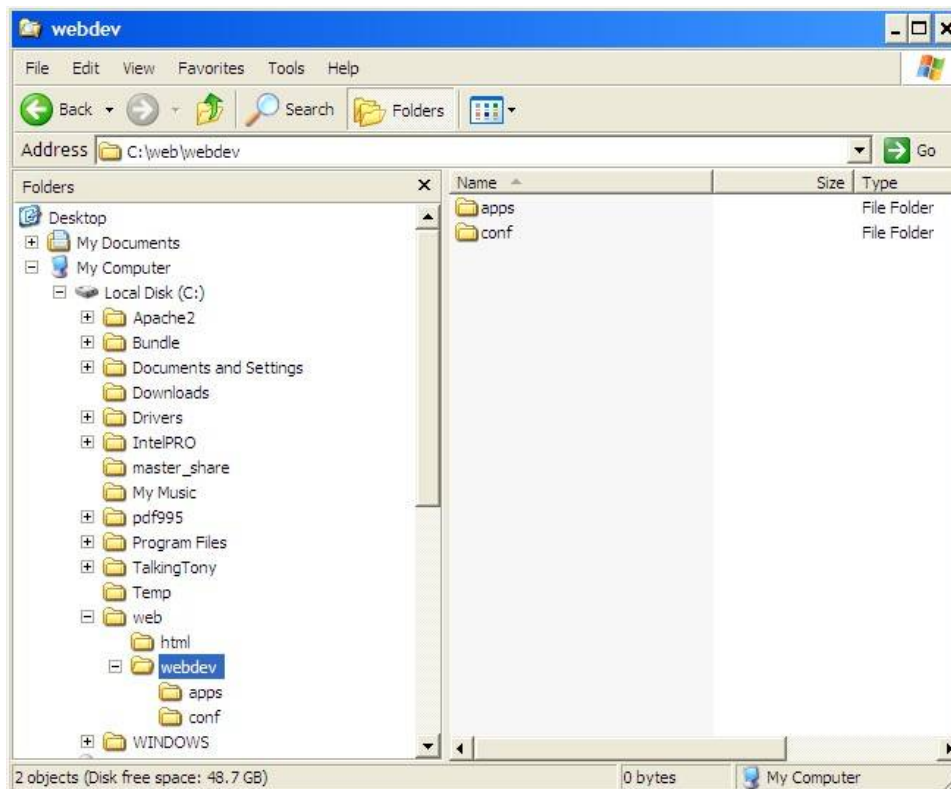
Name the file “index.html” (be sure to include the quotation marks or notepad will add a “.txt” extension and you browser won't recognize this as an html file). DO NOT save it as “index.htm” just yet. Verify it is “index.html”.



Click Save.

Organization and Time Saving Tip

I want all my web applications and configuration files to be quickly and easily accessible. Therefore, I'm going to create a second directory named "webdev" under "C:\web\". I'm going to then add two directories under "C:\web\webdev\", one named "apps" (web applications) and the other named "conf" (configuration files).



We now want to place the “webdev” folder on the Windows Taskbar (typically near the bottom of Windows – it displays “bar” indicating your active tasks). Right click on the Windows Taskbar and deselect Lock Taskbar, if it is checked. Right click on the Taskbar -> Toolbars -> New Toolbar... Maneuver to “C:\web\webdev\”, verify “C:\web\webdev\” is selected and then click OK. You should now have a nice toolbar that will eventually give you quick and easy access to your web development applications and configuration files. Let's be sure to relock the taskbar -> right click the taskbar -> click “Lock Taskbar”. Pretty cool, Huh?

The first configuration file link we will put in the “conf” directory is httpd.conf. Click Start -> All Programs -> Apache httpd server 2.x.x. (2.0.54 in my specific case) -> Configure Apache Server -> right click on Edit the Apache httpd Configuration File and copy the link to the clip board. Maneuver into “C:\web\webdev\conf\”, right click and click Paste Shortcut. Rename the newly pasted shortcut to “Shortcut to httpd.conf” (without the quotes, of course).

You can now try out your new taskbar. Taskbar -> webdev's “>>>” -> conf -> Shortcut to httpd.conf.

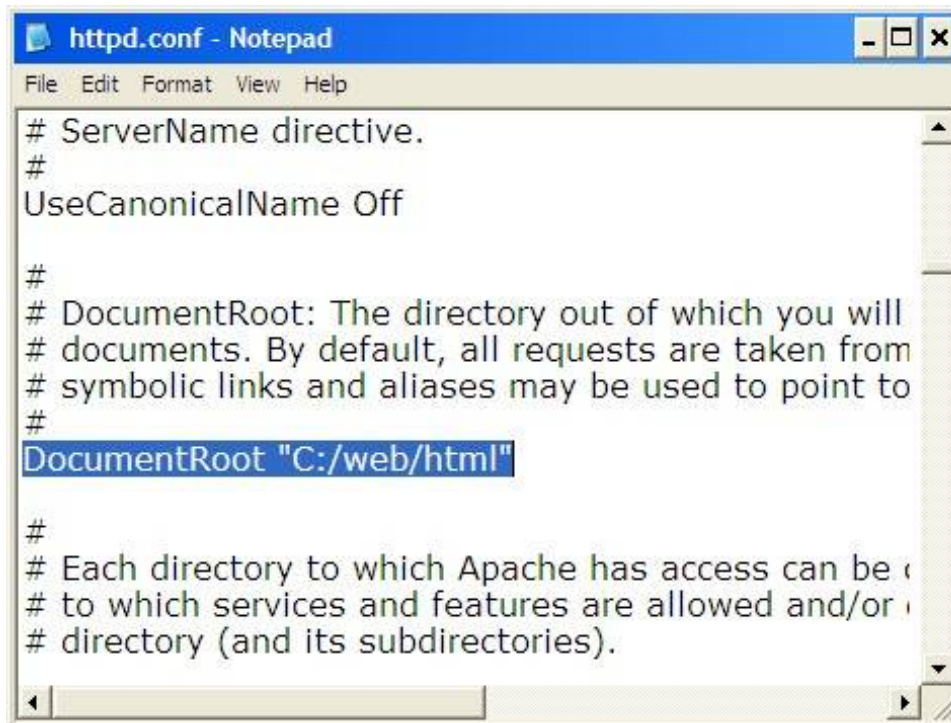
Nice!

End Organization and Time Saving Tip

Back to business...

Find DocumentRoot “C:/Apache2/htdocs” in httpd.conf and update it to your new document root:

```
DocumentRoot "C:/web/html/"
```



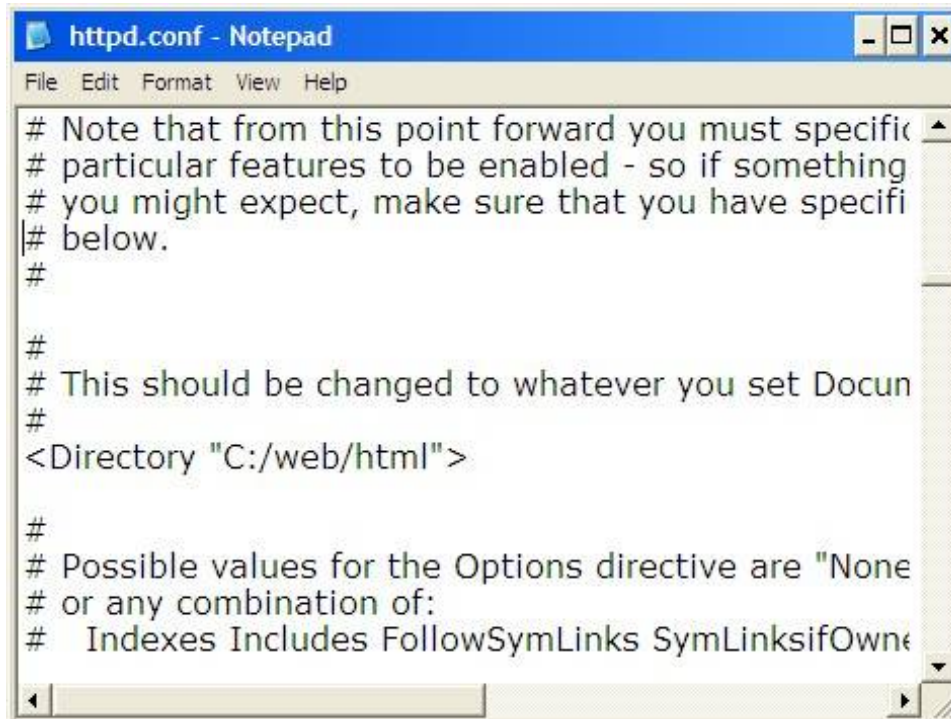
```
httpd.conf - Notepad
File Edit Format View Help
# ServerName directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you will
# documents. By default, all requests are taken from
# symbolic links and aliases may be used to point to
#
DocumentRoot "C:/web/html"

#
# Each directory to which Apache has access can be
# to which services and features are allowed and/or
# directory (and its subdirectories).
```

Find `<Directory "C:/Apache2/htdocs">` and update it to your new directory

`<Directory "C:/web/html">`



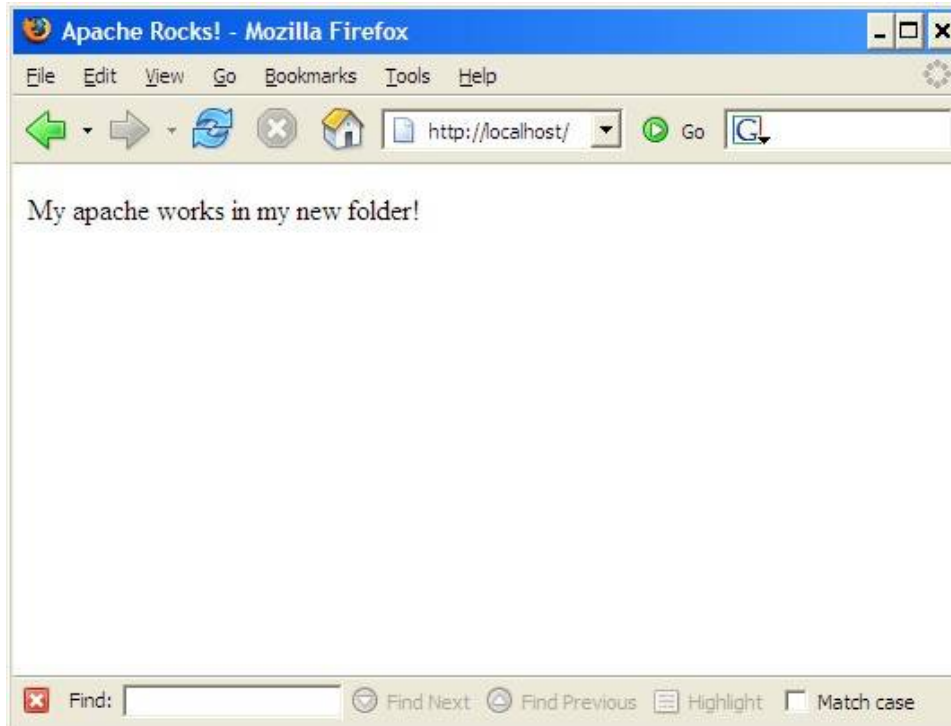
```
httpd.conf - Notepad
File Edit Format View Help
# Note that from this point forward you must specify
# particular features to be enabled - so if something
# you might expect, make sure that you have specified
# below.
#
#
# This should be changed to whatever you set DocumentRoot
#
<Directory "C:/web/html">

#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch
```

Save httpd.conf.

Restart your server. This can be accomplished by left clicking on the system tray's small feather icon in the system tray (lower right, to the left of the clock – you may have to expand the system tray by clicking on “<<”) -> Apache2 -> Restart. Make sure the server has been restarted.

Go to your browser and type in `http://localhost`. You should see “My apache works in my new folder!” displayed in the browser along with “Apache Rocks” in the browser title bar. Don't panic if you see the Apache default page you viewed earlier. If you do, refresh your browser. If you don't see the page below, it is now time to panic. ;-) Just kidding. Go to the Apache2 icon and choose “Stop.” Click on the icon again and make sure one of the live options is “Start.” Click “Start.” Remember to restart the apache2 web server every time you make changes to `httpd.conf`.



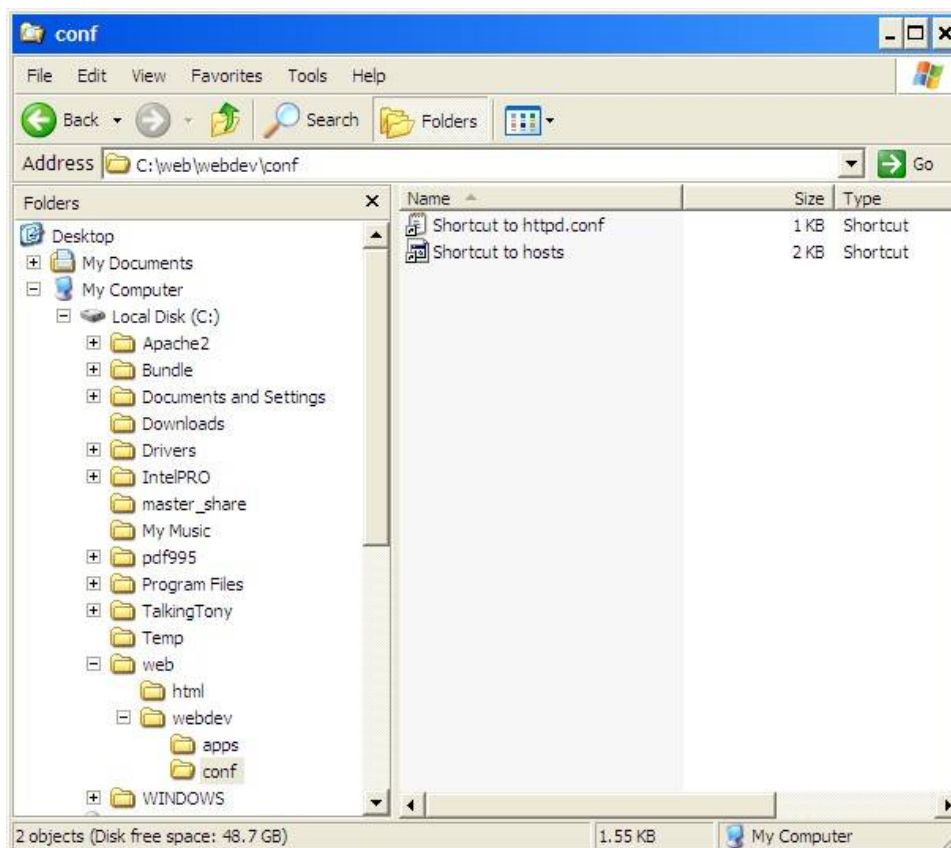
The default Apache page displayed when I typed in `http://localhost`. I refreshed and the correct page loaded.

So far... so good!

Before we move on to install PHP 5, let's take some time to set up virtual hosts. In order to set up virtual hosts, we need access to the Windows hosts file located in “`C:\windows\system32\drivers\etc`”. Virtual hosts allow you to type in a URL, for example, `mydomain.net`, and have the browser check your computer's directory for the file before checking the internet. This is an especially nice feature if you have an external web host. You can set up `mydomain.net` to check your hard drive first (Apache must be running for this to work) and allow `www.mydomain.net` to check the internet first. In other words, `mydomain.net` points to your computer's development files while `www.mydomain.net` points to your external web host's files.

Let's put “hosts” in “`C:\web\webdev\conf`”. Do a search in “`C:\windows`” for “hosts”. There are to files named “hosts”. One is “HOSTS” (uppercase) and is located in “`C:\windows\i386`”. Ignore this

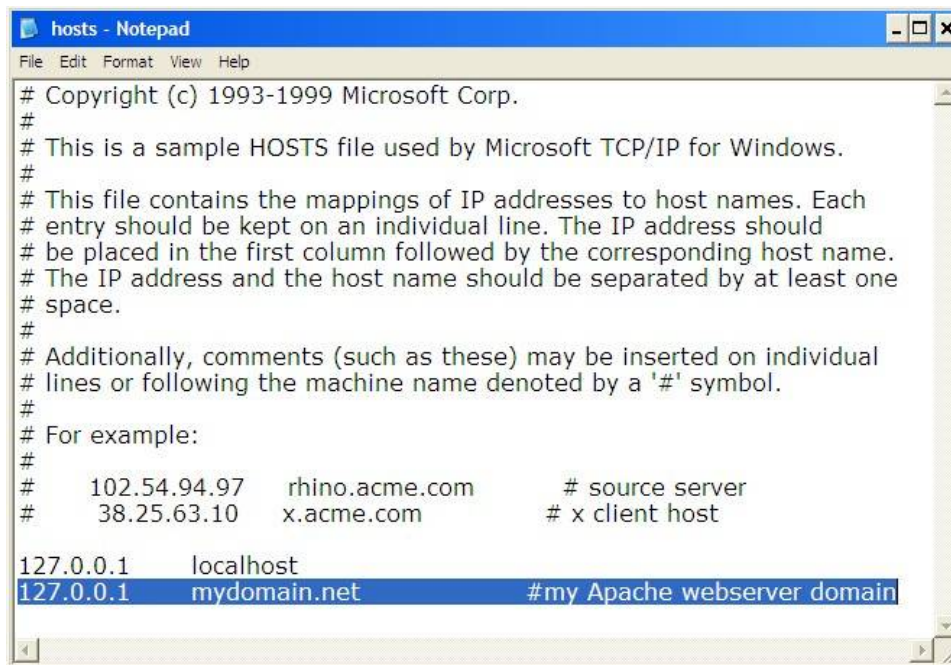
file. The other is “hosts” (lowercase) and is located in “C:\windows\system32\drivers\etc\”. This latter file is the one we want to copy and paste a shortcut into the “C:\web\webdev\conf\” directory.



Open up hosts in notepad: webdev toolbar -> conf -> Shortcut to hosts. If Windows doesn't know what program to use, it will give you an option to choose programs. Choose notepad.

Enter the following at the bottom of the hosts file (use spaces to create the spacing between elements, everything after “#” is a note:

```
127.0.0.1    mydomain.net          #my Apache webserver domain
```



```
hosts - Notepad
File Edit Format View Help
# Copyright (c) 1993-1999 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
# 102.54.94.97 rhino.acme.com # source server
# 38.25.63.10 x.acme.com # x client host
127.0.0.1 localhost
127.0.0.1 mydomain.net #my Apache webserver domain
```

Now we need to tell Apache what to do when Windows begins looking for mydomain.net. We do this by opening httpd.conf. You should know how to do this by using your toolbar folder.

Add the following at the end of httpd.conf (take note of the forward slashes!):

```
NameVirtualHost 127.0.0.1:80
```

```
<VirtualHost 172.0.0.1:80>
```

```
# ServerAdmin webmaster@dummy-host.example.com
```

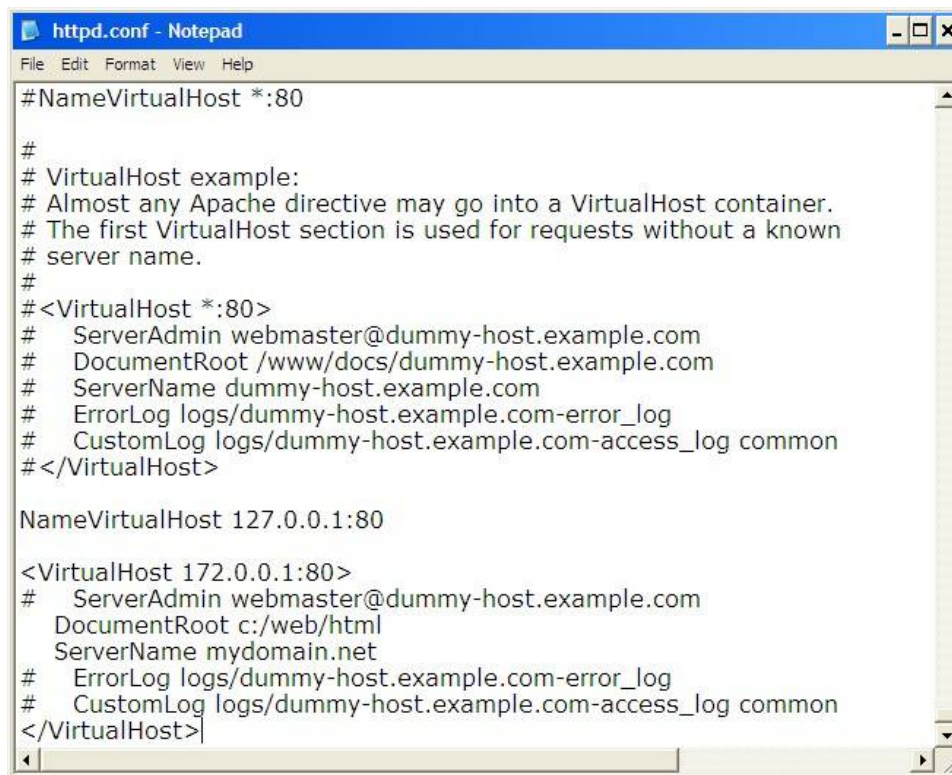
```
DocumentRoot c:/web/html
```

```
ServerName mydomain.net
```

```
# ErrorLog logs/dummy-host.example.com-error_log
```

```
# CustomLog logs/dummy-host.example.com-access_log common
```

```
</VirtualHost>
```



```
#NameVirtualHost *:80

#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for requests without a known
# server name.
#
#<VirtualHost *:80>
#   ServerAdmin webmaster@dummy-host.example.com
#   DocumentRoot /www/docs/dummy-host.example.com
#   ServerName dummy-host.example.com
#   ErrorLog logs/dummy-host.example.com-error_log
#   CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>

NameVirtualHost 127.0.0.1:80

<VirtualHost 172.0.0.1:80>
#   ServerAdmin webmaster@dummy-host.example.com
#   DocumentRoot c:/web/html
#   ServerName mydomain.net
#   ErrorLog logs/dummy-host.example.com-error_log
#   CustomLog logs/dummy-host.example.com-access_log common
</VirtualHost>
```

Save httpd.conf, restart your server, go to your browser and enter “mydomain.net” into the location bar. Your browser should display:

“My apache works in my new folder!”

just as you saw earlier.

Not only is Apache working in your new directory, but so is your virtual host! Notice that if you enter www.mydomain.net that you will be taken directly to the net and should hit a commercial site by that name. Only mydomain.net will check your virtual host set up in Apache based on our configuration. If you wanted www.mydomain.net to access your virtual host, you need to set it up in both the hosts and httpd.conf files.

Note: It is a good idea to change “my new folder” (folder = directory) in your html to the actual name of your new directory if you are setting up more than one virtual host. If you don't, you won't know which directory is actually being accessed because the html output is exactly the same in every directory.

Apache is officially set up! Now on to Step 2: Installing and Configuring PHP 5.