

Resources for getting started with web development? [closed]

Asked 16 years, 4 months ago Modified 9 years, 7 months ago

Viewed 8k times



14



Closed. This question is seeking recommendations for software libraries, tutorials, tools, books, or other off-site resources. It does not meet [Stack Overflow guidelines](#). It is not currently accepting answers.



We don't allow questions seeking recommendations for software libraries, tutorials, tools, books, or other off-site resources. You can edit the question so it can be answered with facts and citations.

Closed 10 years ago.

[Improve this question](#)

Let's say I woke up today and wanted to create a clone of StackOverflow.com, and reap the financial windfall of millions \$0.02 ad clicks. Where do I start?

My understanding of web technologies are:

- HTML is what is ultimately displayed
- CSS is a mechanism for making HTML look pleasing
- ASP.NET lets you add functionality using .NET(?)

- JavaScript does stuff
- AJAX does asynchronous stuff
- ... and the list goes on!

To write a good website to I just need to buy seven books and read them all? Are Web 2.0 sites really the synergy of all these technologies?

Where does someone go to get started down the path to creating professional-looking web sites, and what steps are there along the way.

language-agnostic

Share

Improve this question

Follow

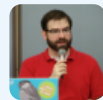
edited May 6, 2015 at 20:26



BJ Myers

6,813 ● 6 ● 35 ● 53

asked Aug 15, 2008 at 20:00



Chris Smith

18.7k ● 14 ● 61 ● 81

Use PHP instead of ASP.NET. Not that it's better, just easier/cheaper to find hosting.</my id=\$0.02>

– [JoeCortopassi](#) Feb 18, 2010 at 19:13

10 Answers

Sorted by:

Highest score (default)





12



While I have built my knowledge largely based on using the internet to search out what I want to know (w3schools.com helped a lot, as did A List Apart), a few good books have helped me along the way, though they have been platform/language-specific, so I'll avoid mentioning them unless someone is curious. For me, at least, having a book open so that I don't have to resize windows or switch between them is very valuable.

The first part of your list is ok, but the last few items need tweaking. ASP.NET adds *server-side* functionality (for the most part) to your application. This lives outside of the browser and is thus quite powerful and easily shared with a variety of end-users.

The problem (some say) with server-side processing is that your application must make a new HTTP request when you ask for an action to be performed. So if you click on a link to a page that yields a new set of data, you don't get instant results. The page reloads, or loads a separate page.

Javascript solves this to a degree--it allows you to respond to user input instantaneously. Do you want to display the sum of two numbers when the user clicks a button? You can do it with Javascript.

The problem with Javascript is that it can't talk directly to databases, or explore your server's file system, or other stuff like that. It lives in the browser--period.

AJAX bridges the gap between your user's browser and your server. With AJAX, *Javascript* makes the HTTP request without refreshing your page or loading a new one. Javascript talks to a server-side script (not necessarily ASP, either--works with PHP, Rails, Coldfusion, etc.) and sends and receives information. And because Javascript isn't dependent on page loads, a quick, snappy AJAX script can almost give the feeling of a common desktop application, in which you don't have to wait for HTTP requests when performing simple actions on your application's data.

Share Improve this answer

Follow

answered Aug 15, 2008 at 20:11



Brian Warshaw

23k ● 9 ● 54 ● 72



9

I think that this series of [Opera Articles](#) will give you a good idea of web standards and basic concepts of web development.



2014 update: the Opera docs were relocated in 2012 to this section of [webplatform.org](#):



http://docs.webplatform.org/wiki/Main_Page

Share Improve this answer

Follow

edited Mar 23, 2014 at 18:36




answered Aug 15, 2008 at 20:18



travis

36.4k ● 21 ● 72 ● 97

And your link is broken.. – [RP](#). Mar 23, 2014 at 12:01

- 1 @RP A lot has happened in nearly 6 years:
docs.webplatform.org/wiki/Main_Page – [travis](#) Mar 23, 2014 at 18:31 
-



3



Ian's answer has a lot of weight. You could buy all those books and read them all and know nothing about web development. What you really need to do is start with something that is not nearly as big as Stack Overflow. Start with your personal site. Read some web dev/css articles on [a list apart](#). Learn about doctypes and why to use them. Add some css and change the colors around. Go over to [quirksmode](#) and peruse the site. Add some js. Follow some links on [Crockfords site](#). You will probably stumble across his awesome video lectures, which you should watch. Then after that go back to all the js that you wrote and rewrite it. Then pick a server side language that you want to learn. [Python](#) is pretty easy, but it really doesn't matter what you pick. Then come back and integrate all those together in your site. At this point you will at least be getting started with web development and will have worked with several different technologies.

EDIT: I forgot to mention. **READ BOOKS.**

Many developers that I have worked with in the past have gotten through their career without really advancing after a certain point. I could be totally wrong, but I attribute it to

not reading enough books and relying on using their same bad code over and over.

Share Improve this answer

answered Aug 18, 2008 at 19:39

Follow



mk.

26.2k ● 13 ● 39 ● 41



2



You could go out and buy a bunch of books and start reading them and quickly get overwhelmed in the seemingly massive learning curve it takes to go from nowhere, which is where it appears you are, to a rich internet entrepreneur, which is where you want to be.



Alternatively, and what I would suggest is, you could define a problem you want to solve, and then go about finding the solution to that problem. Start with something small. "I have a problem: I don't have a web site about myself.". Define what you need to do to solve that problem, learn the basics, and do it. Then, define a new problem, which probably relies on the solution to the first problem, find what you need to do, and do it.

This is how all technology professionals evolve. My first website was a personal site with nothing but text. Then I added some jokes and some movie quotes. Then I got tired of man-handling all the updates to I learned how to put them into a database and retrieve them from the database for display. It goes on and on.

Call me when you've got more money from your financial windfall than you know what to do with.

Share Improve this answer

answered Aug 15, 2008 at 23:55

Follow



Ian Good

126 ● 1 ● 6



2



If you really just want to jump in with both feet, I would suggest looking at ColdFusion from Adobe. The developer edition is free and runs on windows, os x and linux. The documentation is authoritative and extensive, there is a very active developer community and only a few books you might want to dig into. The definitive guide is a series of books that can be found on [Amazon](#)

The nice thing about ColdFusion is that you can use it as a stepping stone to other languages and remain productive along the way. You can even mix it together with Java since it is itself written in java. There are also lots of goodies built in that you would have to scour the web for or pay more for in other languages. Things like full text indexing, graphing, server monitoring, ajax based controls, flash/flex integration, asynch os calls, etc.

You even have the choice of building object oriented code or procedural code, although some people would not count that as a benefit. Those people rarely agree on which style should win, though.

Cheers!

Share Improve this answer

answered Aug 21, 2008 at 15:40

Follow



mrankin



2



I think sitepoint is the best resource for learning best practices in web development. They have [great articles](#), [good references](#), and probably one of the [best forums](#).

However the people there can be a bit grumpy. ;) If you are a real nerd, reading the specs for HTML 5 and CSS is also a good way to learn.

Share Improve this answer

Follow

edited Nov 2, 2013 at 18:39



BenMorel

36.3k ● 51 ● 202 ● 334

answered Sep 29, 2008 at 12:00



olle

1,194 ● 7 ● 10



1



I'm with Ian on this one. Reading books is all well and good, but nothing beats getting stuck in. I actually started with a Dummies Guide to ASP (that'd be "classic" ASP), back in 1999.

If I was going to start from scratch today I'd be looking at something that covered a full stack solution, whether Apache/PHP/MySQL, RoR or whatever.

ATM I have no experience of Rails, but it might be a pretty good place to start as it includes a lot of stuff that you'd have to figure out early on otherwise (integration

with a Scriptaculous, a JS framework) - you can always learn what going on under the hood at a later date.

.NET is always an option, and if you're comfortable with Visual Studio it may be the way to go, but it's not the easiest thing to pick up otherwise.

If you know a bit of HTML but are basically new to server-side programming you might look at ColdFusion. It's actually extremely powerful and like Rails includes lots of "out of the box" benefits. There's a Swiss company called [Railo](#) who are currently in the process of releasing an Open Source ColdFusion engine that is affiliated with JBoss.

Last and not least - don't forget databases! Sooner or later you'll need to get to grips with some pretty serious SQL...

Share Improve this answer

edited Aug 19, 2008 at 10:22

Follow

answered Aug 19, 2008 at 10:16



[James Marshall](#)

620 ● 2 ● 7 ● 15



1

CFML (aka "ColdFusion" even though that's really an Adobe product, not the language) is definitely easy to learn, and if you want FOSS for CFML, in addition to Railo you can use [Open BlueDragon](#) which is a GPL CFML engine.



Share Improve this answer

answered Sep 29, 2008 at 11:54

Follow



[Matt Woodward](#)

234 ● 3 ● 4



Designing with Web Standards is a great first read!

1

<http://www.zeldman.com/dwws/>



Share Improve this answer

answered Sep 29, 2008 at 12:02

Follow



[danpickett](#)

2,046 ● 14 ● 18



I would recommend this book:

http://www.amazon.com/MCTS-Self-Paced-Training-Exam-70-528/dp/0735623341/ref=sr_1_1?ie=UTF8&s=books&qid=1218830714&sr=8-1

0



I have just read it to take the exam, and although I knew the web theory part, I found it to be of great value.

This of course is a ASP.NET specific book, but that is what I would recommend learning anyways.

After you learn all the ASP.NET stuff, I would suggest reading up on [jQuery](#).

Happy coding :)

Share Improve this answer

answered Aug 15, 2008 at 20:08

Follow



Lars Mæhlum

6,102 ● 3 ● 29 ● 32
