
Scope
Time
Quality
Cost
A
B
PICK TWO!
GOOD QUALITY
FAST DELIVERY
CHEAP
What are your steps?

The Ladder View - B4 - n – After

1. You (Student)
    a. Are "here" today...
        i. You (Student) want to be where?
2. Footsteps in the garden of IT…
    a. What is the commitment to the goal called "Results"?
3. You (Student) have you read…?
    a. Goals vs. Microsoft certs supported by syllabus?
    b. Microsoft Learn web site
        i. https://www.microsoft.com/en-us/learning/
        ii. Exam track review?
        iii. Your current Microsoft transcript of certifications.
    c. LinkedIn profile is current: NO MORE FAKEING CERTS –
        i. HR will know to look for the digital badge from https://www.youracclaim.com/
4. You (Student) determine "Expectation Management & Schedule" to your path of success.
    a. Determine the path steps: "Yes" or "No" or "Be true to my world" …
        i. I know what I wish to accomplish with this training?
        ii. I know the exam track requirements because I looked it up on the web.
        iii. I plan to dedicate _____ amount of personal time to achieve my goal.
        iv. I realize my knowledge level is here " ? "…
        v. My knowledge path and timeframe to (goal) is " ? "...
    b. What do I not understand?
        i. What am I trying to do at this step?
            1. Did I follow the instructions in the book or am I "Free Thinking" my way to an error(working "Out of Scope" or the on the wrong thing)
    c. Today is Day__of 24 - Week__ of 6; in the course.
        i. Will my current "Pace and Purpose" fulfill my "Expectation Management Quota"?
            1. What are my expectations the next ___ weeks?
                a. For interviews?
                b. For employment
            2. Is this obtainable with my schedule?

Be Kind, Honest & Fair to yourself…

Technologies & Languages: Things ALWAYS Grow & Change – Stay Current

Links change often and/or get chopped in the pdf creation process – especially if there is a "hyphen – " – so if the click doesn't work – copy n paste to notepad – check link – or just Google the topic…

1. ASP Defined
    a. https://en.wikipedia.org/wiki/ASP
    b. https://en.wikipedia.org/wiki/ASP_(disambiguation)
    c. https://en.wikipedia.org/wiki/Active_Server_Pages
2. .NET Framework
    a. https://en.wikipedia.org/wiki/.NET_Framework
    b. Open sourced in
3. The ASP. (NET – CORE – whatever)
    a. Is an acronym..
    b. https://www.asp.net/
    c. ASP.NET is an open source web framework for building modern web apps and services with .NET. ASP.NET creates websites based on HTML5, CSS, and JavaScript that are simple, fast, and can scale to millions of users.
4. Active Server Pages originally written in VBScript (Visual Basic Lite) – was a cross over from applications that installed on a PC to applications delivered through a "browser" like Internet Explorer, Firefox, Opera or Chrome for example.
    a. Office is a good example – we can install it on our own PC and pay for a license or use it with Office 365 online – and still pay a fee… 😊
5. 3rd Party Tools & "THE CORE!!!!"
    a. Microsoft has recently "Open Sourced" the .Net Framework which allows for any hybrid the developer may be involved within. NOT A NEW THING…. HENCE.
        i. The "Core" is marrying to other open source products (Linux, Apache, Big Data, R Server, NoSQL among others) in a more streamlined fashion – in a seriously small nutshell.
6. ASP.Net vs. CORE on the web – today is Monday 07/23/2018
    a. Choose between ASP.NET and ASP.NET Core:
        i. https://docs.microsoft.com/en-us/aspnet/core/fundamentals/choose-aspnet-framework?view=aspnetcore-2.1#aspnet
    b. Choosing between .NET Core and .NET Framework for server apps:
        i. https://docs.microsoft.com/en-us/dotnet/standard/choosing-core-framework-server

Client vs. Server-Side: What is Compiling?

Compiled code can be executed directly by the computer's CPU. That is, the executable code is specified in the CPU's "native" language (assembly language). The code of interpreted languages however must be translated at run-time from any format to CPU machine instructions.

Compiling is one of the things tools like Visual Studio do to convert programming text such as C# or VB into PC understandable instructions.

The browser – IE, Firefox, etc. compiles JavaScript, PHP, CSS among others without the need for tools such as Visual Studio.

Server-side languages are executed on the web server then sent to the browser – Client side execute in the browser – whose CPU is doing the "math" …

The "General" Language List * – OOP vs. Architecture: There is no longer ONE language to learn…..

1. Angular JS:
    a. https://www.w3schools.com/angular/angular_intro.asp
2. *CSS: Cascade Style Sheets – Bolding, Styling, Positioning the HTML parts.
    a. https://www.w3schools.com/css/default.asp
3. *C# - OOP – Server Side "Stuff" – Object Oriented Programming – IE Relational Algebra – I use variables so I can re-use code…. I compile and run on the server before going to a browser….. My server does the "math".
4. *HTML: Hyper Text Markup Language
    a. DOM – Document Object Model – Structures & Containers on a web page
    b. https://www.w3schools.com/html/default.asp
5. *JavaScript- OOP – Client Side "Stuff" – Object Oriented Programming – IE Relational Algebra – I use variables so I can re-use code…. I compile and run in the browser – so the users CPU does the "math".
    a. https://www.w3schools.com/js/default.asp
    b. *AJAX: Asynchronous JavaScript and XML – Forms that do not refresh if a field is not filled in is a sample… Doing stuff behind the User and Browsers "back" so to speak.
    c. *JQuery: JavaScript from a library for "Cool Stuff" like drop down calendars for a date field typically called USER Experience (UX).
    d. *JSON: JavaScript Object Notation Used for transferring data without rules.
6. Python:
    a. https://www.w3schools.com/python/
7. *Razor: The "Extra Code" that works with server-side code to create web pages.
    a. Look for the @ sign.
8. Visual Basic & VBScript
9. *T-SQL: Transactional Structure Query Language
    a. LINQ 2 SQL: Connecting from OOP to SQL programmatically.
        i. Uses Lambda expressions.
10. *XML: Extended Markup Language
    a. Used for transferring data with data typing rules.

General Development Components

1. SignalR is a framework to allow the server to push to the client: https://www.asp.net/signalr
    a. SignalR can be used to add any sort of "real-time" web functionality to your ASP.NET application. While chat is often used as an example, you can do a whole lot more.
    b. Any time a user refreshes a web page to see new data, or the page implements long polling to retrieve new data, it is a candidate for using SignalR.
    c. Examples include dashboards and monitoring applications, collaborative applications (such as simultaneous editing of documents), job progress updates, and real-time forms.
    d. SignalR supports Web Sockets, and falls back to other compatible techniques for older browsers.
    e. SignalR includes APIs for connection management (for instance, connect and disconnect events), grouping connections, and authorization.

2. Web Sockets and Web API - two entirely different concepts.

    a. Web API is a type of Web Services which handles HTTP requests and will be something that you actually make your request to when you need to access your data (or make changes to it).
    b. Web Sockets are a type of connection that can be used within browsers to create a persistent connection between the client and server. This can be used for a wide variety of things such as handling various real-time activities (like chat, notifications, etc.) but I don't believe that it would provide any benefit with regards to accessing your data quicker.

3. ASP.NET Web API is focused on RESTful web services: https://msdn.microsoft.com/en-us/library/hh833994(v=vs.108).aspx
    a. Use ASP.NET Web API to create a web API that returns a list of products. The front-end web page uses jQuery to display the results: https://docs.microsoft.com/en-us/aspnet/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api

4. WebSocket: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API
    a. an advanced technology that makes it possible to open an interactive communication session between the user's browser and a server. With this API, you can send messages to a server and receive event-driven responses without having to poll the server for a reply.
    b. There is a persistent connection between the client and the server and both parties can start sending data at any time.
5. "Single-page application" (SPA)
    a. The general term for a web application that loads a single HTML page and then updates the page dynamically, instead of loading new pages. After the initial page load, the SPA talks with the server through AJAX requests.: https://docs.microsoft.com/en-us/aspnet/single-page-application/overview/introduction/knockoutjs-template

6. ASP.NET Web Forms are pages that are based on Microsoft ASP.NET technology, in which code that runs on the server dynamically renders the correct HTML for features such as styles, layout, and so on.
    a. Web Forms are compatible with any language supported by the .NET common language runtime, such as Microsoft Visual Basic (.asp) and Microsoft Visual C# (.aspx).
    b. WingTipToys: Getting Started with ASP.NET 4.5 Web Forms: entity framework w aspx - not schtml
    c. https://docs.microsoft.com/en-us/aspnet/web-forms/overview/getting-started/getting-started-with-aspnet-45-web-forms/introduction-and-overview

7. Web pages: For discussion…
    a. https://docs.microsoft.com/en-us/aspnet/web-pages/overview/getting-started/introducing-aspnet-web-pages-2/getting-started

The Sandbox to MCSA/MCSE/MCSD 

Tools & the "Place to Start" …

1. **Web Dev & SQL**
    a. **Visual Studio & VS Code:**
    b. **https://visualstudio.microsoft.com/downloads/**
2. **SQL Server: Admin, Management & the "Azure Cloud"**
    a. **SQL Express or Developer:**
        i. **https://www.microsoft.com/en-us/download/details.aspx?id=52679**
    b. **SQL Server Management Studio (SSMS is now standalone):**
        i. **https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-2017**
3. **What is your exam track?**
4. **Have you read the Skills Measured for each one?**
5. **Can you login and see your profile and transcript?**
    a. **Microsoft Learn:**
        i. **https://www.microsoft.com/en-us/learning/default.aspx**
    b. **View "Preparation options" – Self-paced training links on each exam page.**
6. **Get building:**
    a. **http://www.asp.net/**
7. **Creating Your First Application:**
    a. **MVC: http://www.asp.net/mvc/overview/getting-started/introduction/getting-started**
    b. **MVC CORE: https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnetcore-2.1&tabs=aspnetcore2x**