1. **Why should we learn ASP.NET?**

ASP.NET helps build website using Windows framework. Some of the different ways to build a website is through console applications, windows application, etc.

ASP.net is a web framework that has been created by Microsoft to develop web applications.

If you want a .NET application with a web interface, then ASP.NET is the way to do it.

Can be used to make many different applications

Easier to set up, as it is already built int the windows server environment, so it requires less configurations

.Net is a platform used for making all kinds of applications made up of tools, programming languages and libraries that are used to build those applications. It’s built into windows server environment, so it is easier to use than other environments.

Significantly reduces the amount of code required for building large and complex applications which increases overall development speed and reduce development costs.

Just-in-time compilation, smart caching technologies and native optimization dramatically increase overall application performance.

We should learn ASP.NET in order to be able to create all kinds of applications made up of tools, programming languages and libraries. It’s built into the Windows Server Environment and it’s easier to learn than other environments.

We should learn ASP.net because it is collaborative platform which includes many different libraries, tools, where we can build web apps for different platforms.

There are plenty of good reasons to use ASP.NET when developing a website or an application. High speed, low cost, and vast language support are among the most significant benefits. ASP.NET is built into the familiar Windows server environment, requiring less setup and configuration than other web development platforms that must be installed and configured separately. The popularity of ASP.NET makes online resources and skilled developers easy to find.

Websites and applications built with ASP.NET can be faster and more efficient than a website build with PHP, for example. ASP.NET applications are compiled, which means the code is translated into object code, which is then executed. This compilation process takes a small amount of time, but happens only once. After compilation, the code can be executed over and over by the .Net platform very quickly.

1. **Where can we find the best resources for learning ASP.NET?**

The best place to find resources for learning ASP.NET is Microsoft website/manual.

https://dotnet.microsoft.com/learn

1. **What is MVC architecture and what does each layer of this pattern represent?**

This pattern is used to separate application's concerns.

* **Model** - Model represents an object or JAVA POJO carrying data. It can also have logic to update controller if its data changes.
* **View** - View represents the visualization of the data that model contains.
* **Controller** - Controller acts on both model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps view and model separate.
* MVC stands for Model view controller and is a design pattern that allows developers to use interfaces which divide related program logic in to three categories. Model defines data structure, view defines the user interface, controller defines control logic.

1. **What are some benefits of using MVC?**
   * Enables clean separation of concerns, testability, and TDD by default.
   * It is highly extensible and pluggable. Everything in the MVC framework is designed so that it can be easily replaced/customized
   * It includes a very powerful URL mapping component that enables you to build applications with clean URLs. URLs do not need to have extensions within them, and are designed to easily support SEO and REST-friendly naming patterns. For example, I could easily map the /products/edit/4 URL to the "Edit" action
   * The MVC framework supports using the existing ASP.NET .ASPX, .ASCX, and. Master markup files as "view templates" (meaning you can easily use existing ASP.NET features like nested master pages, <%= %> snippets, declarative server controls, templates, data-binding, localization, etc). It does not, however, use the existing post-back model for interactions back to the server.
   * The ASP.NET MVC framework fully supports existing ASP.NET features like forms/windows authentication, URL authorization, membership/roles, output and data caching, session/profile state management, health monitoring, configuration system, the provider architecture, etc.
   * More security (e.g. validation, sanitization, and normalization), Reusability, More maintainable due to each process being compartmentalized
   * Faster web application to development
   * Supports asynchronous technique
   * Able to create multiple views of the Model
   * Good for developing large web applications
   * No need to format the data-Modifications never affect the whole model
   * Benefits It makes development fast. Offers multiple views for a model. Can handle large size development. MVC does not require format and can work on any interface. Modification can be made to specific parts of the model.
   * Easier to update the applications
   * Easier to debug since each level is written separately in the application
   * Significantly reduces the amount of code required for building large and complex applications which increases overall development speed and reduce development costs
   * Just-in-time compilation, smart caching technologies and native optimization dramatically increase overall application performance.
   * MVC architecture helps us to control the complexity of application by dividing it into three components i.e. model, view and controller.
   * MVC does not use server-based forms, that’s why it is ideal for those developers who want full control over their application behavior.
   * Test driven development approach is supported by MVC architecture.
   * MVC use front controller pattern. Front controller pattern handles the multiple incoming requests using single interface (controller). Front controller provides centralized control. We need to configure only one controller in web server instead of many.
   * Front controller provides support rich routing communications to design our web application.

**MVC Pattern**



**Q. What is the most popular version control system?**

-GITHUB

**Q. What is version control?**

- Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later. For the examples in this book, you will use software source code as the files being version controlled, though in reality you can do this with nearly any type of file on a computer.

**Q. Why do we use version control?**

* + Version control is important for all code, files, and assets that multiple team members will collaborate on. Using version control software helps you keep track of changes and keep every team member working off the latest version. It needs to do more than just manage and track files.