JOHN ABBOTT
CEGEP/COLLEGE

(420-PS4-AB)
Developing ASP .NET Web Applications with ADO .NET – Introduction

Aref Mourtada

Summer 2018

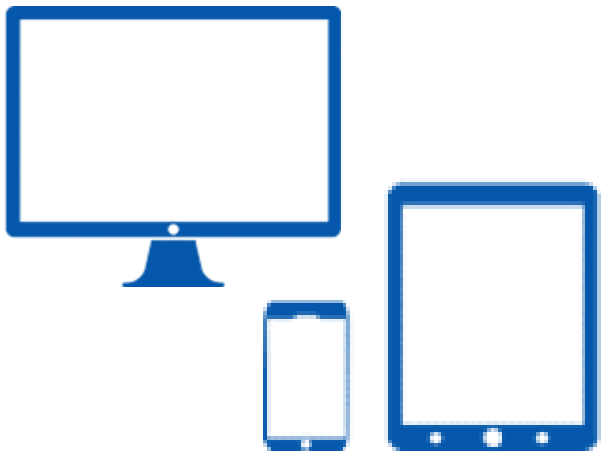
Outline

- Understanding the Web
- Creating ASP .NET Projects
- ASP .NET Basics
- Demos & Exercises

Program Courses

- Introduction to Computers, the Internet, and the Web
- HTML & XML
- Cascading Style Sheets (CSS)
- JavaScript
- Website Design
- Programming I
- Programming II
- Programming III
- Database I
- Database II
- PHP Programming
- Mobile Applications
- .NET Programming Fundamentals
- Visual Basic .NET and C# .NET Windows Development
- Data Structure and Algorithms
- Developing ASP .NET Web Applications
- Developing Web Services

Components of a Web Application



PCs, Tablets, Smartphones, Kiosks, ATMS,..

Internet

Web Server

Web Server & Browsers

- Web server: software that listens for web page requests
  - ***Apache***
  - Microsoft Internet Information Server (***IIS***) (part of Windows)
  - ***nginx*** and ***lighttpd*** -- Lighter weight open source alternatives to Apache

- Web browser: fetches/displays documents from web servers Firefox web browser
  - Mozilla Firefox
  - Microsoft Edge / Internet Explorer (IE)
  - Google Chrome
  - Opera
  - Apple Safari

Web Languages – Technologies

- Hypertext Markup Language (**HTML**): used for writing web pages
- Cascading Style Sheets (**CSS**): stylistic info for web pages
- PHP Hypertext Processor (**PHP**): dynamically create pages on a web server
- **JavaScript**: interactive and programmable web pages (with libraries like jQuery)
- Asynchronous JavaScript and XML (**Ajax**): retrieving data for web applications from pre-loaded web pages
- eXtensible Markup Language (**XML**) and JavaScript Object Notation (**JSON**): meta-languages for organizing data
- **C# / VB .NET**: Microsoft .NET languages used in web development as well as in other contexts

Uniform Resource Locator (URL)

- An identifier for the location of a document on a web site
- A basic URL:

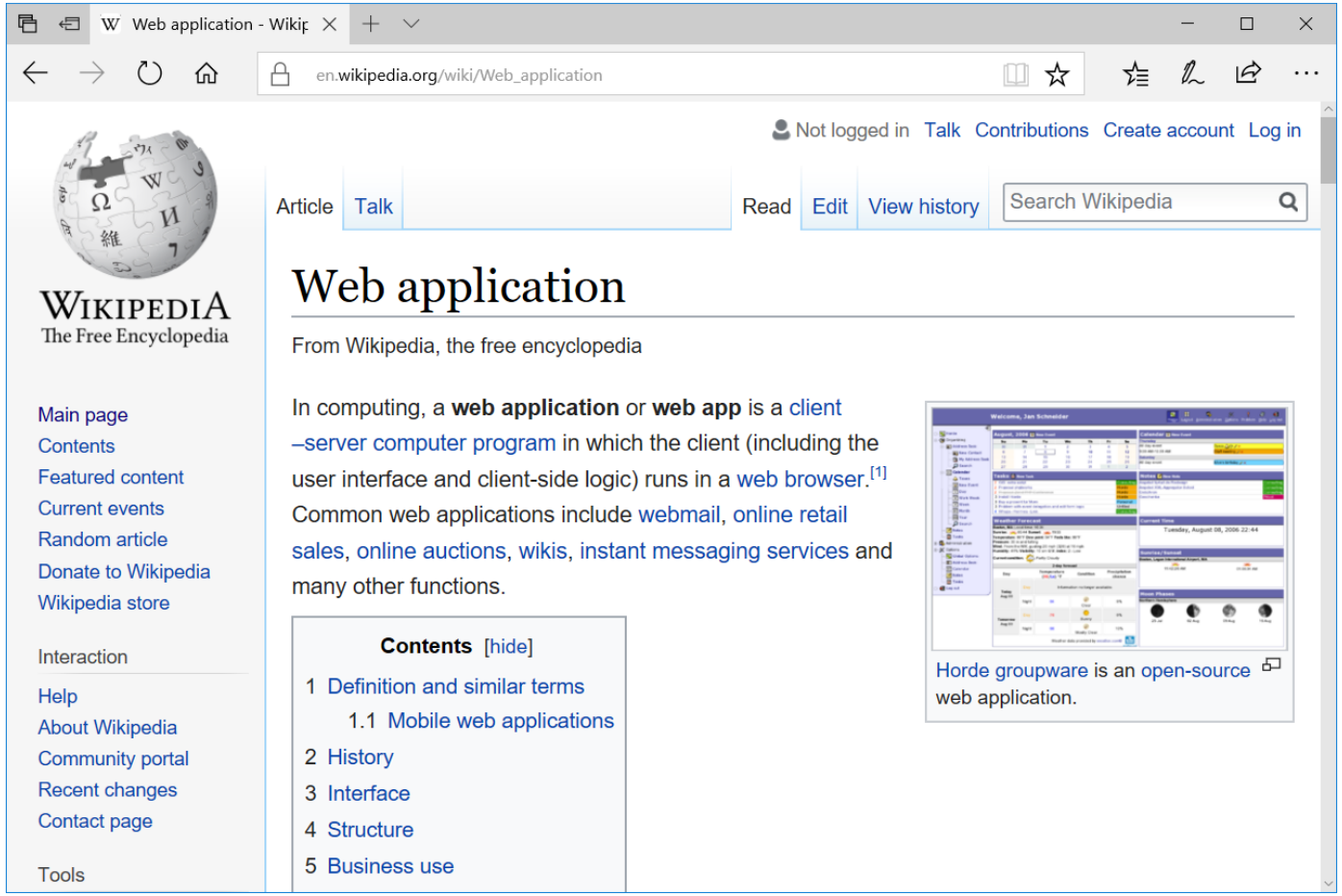
```
http://www.johnabbott.qc.ca/ info/index.html
~~~   ~~~~~~~~~~~~~~~~~ ~~~ ~~~~~~~
protocol    host                path  file name
```

- Upon entering this URL into the browser, it would:
  - Ask the DNS server for the IP address of www.johnabbott.qc.ca
  - connect to that IP address at port 80
  - ask the server to GET /info/index.html
  - display the resulting page on the screen

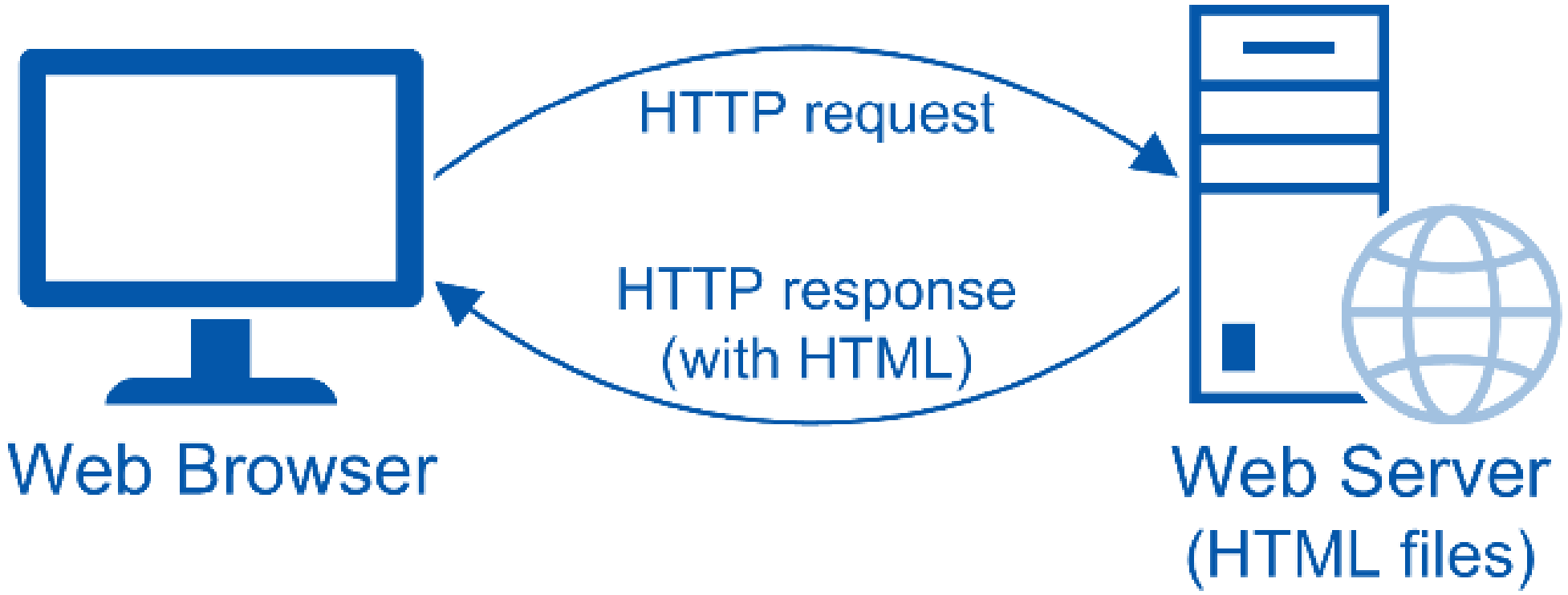
Static Web Pages

When a browser requests a `.html` file (static content)

Server just sends that file



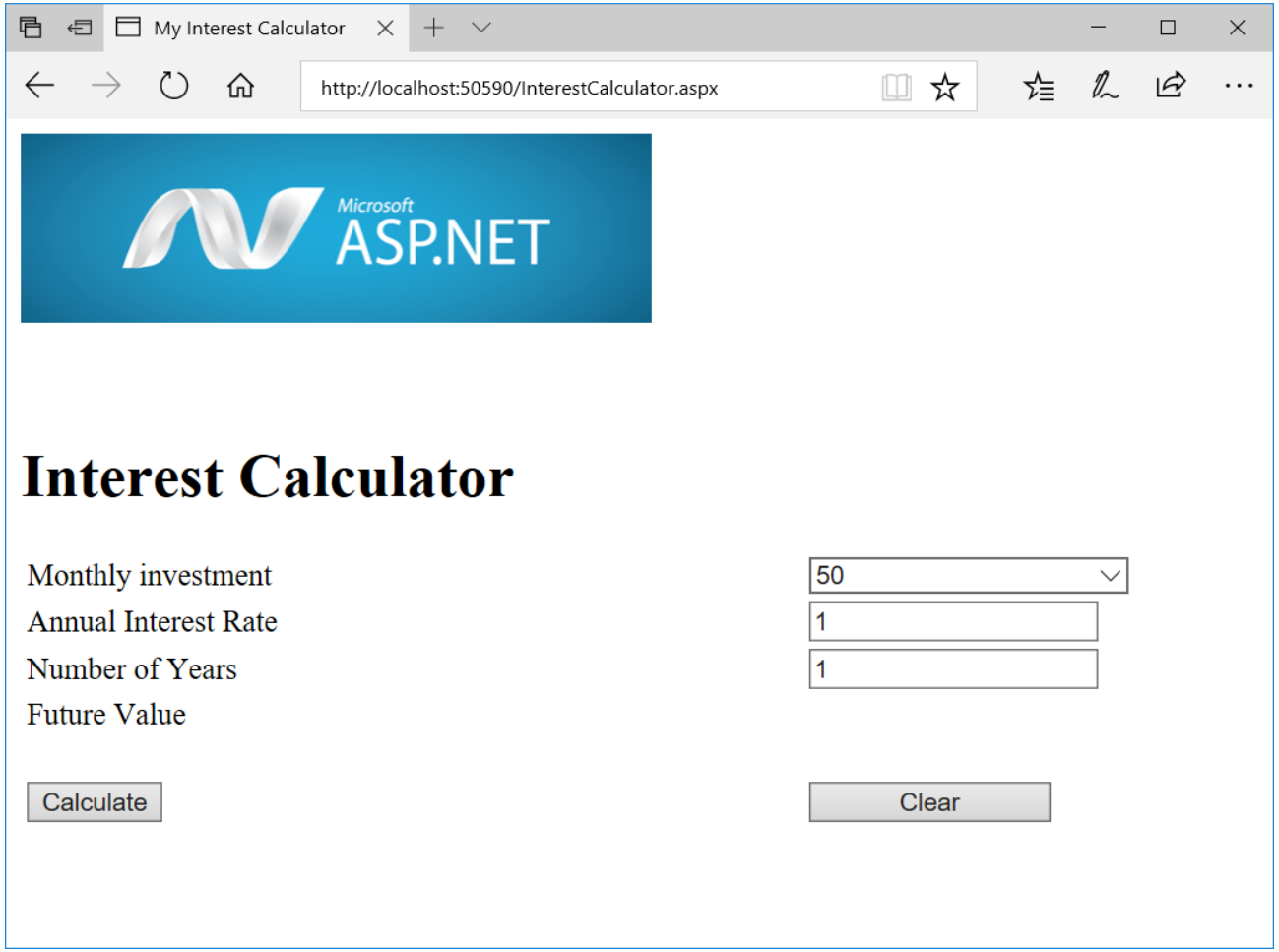
Processing a Static Web Page



Dynamic Web Pages

When a browser requests a `.aspx` file (dynamic content)

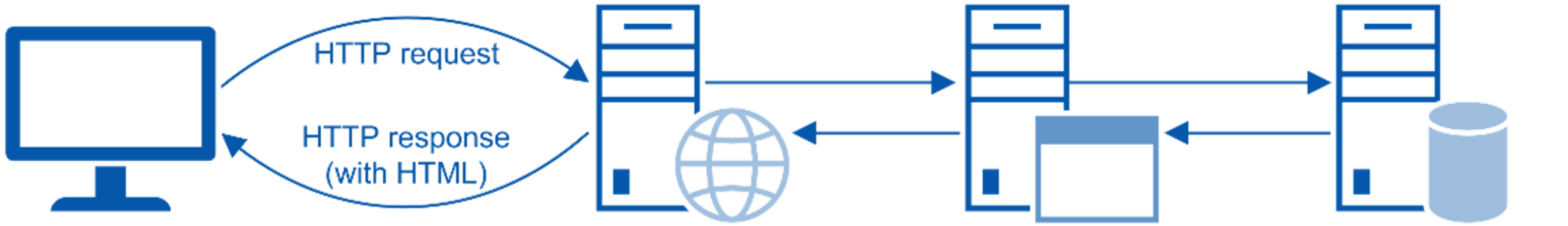
- Server reads it, runs any code inside it or that is called by the page, then sends result across the network
- aspx produces output that becomes the response sent back



ASP .NET Web Forms

- A programmable Web page called a Web Form
  - Goal is to server up HTML to browsers
  - Object-oriented
  - Derive from **Page** class
  - Display, collect and validate data using controls.

Processing Dynamic Web Pages



ASP .NET Web Forms Project

- ASP .NET support two different types of projects:
  - Web Site Project
    - Folder based
    - Compiled at web server
  - Web Application Project
    - Solution based
    - Tracks all components
    - Compiled before deployment.
    - DLL.

ASP .NET Code Separation

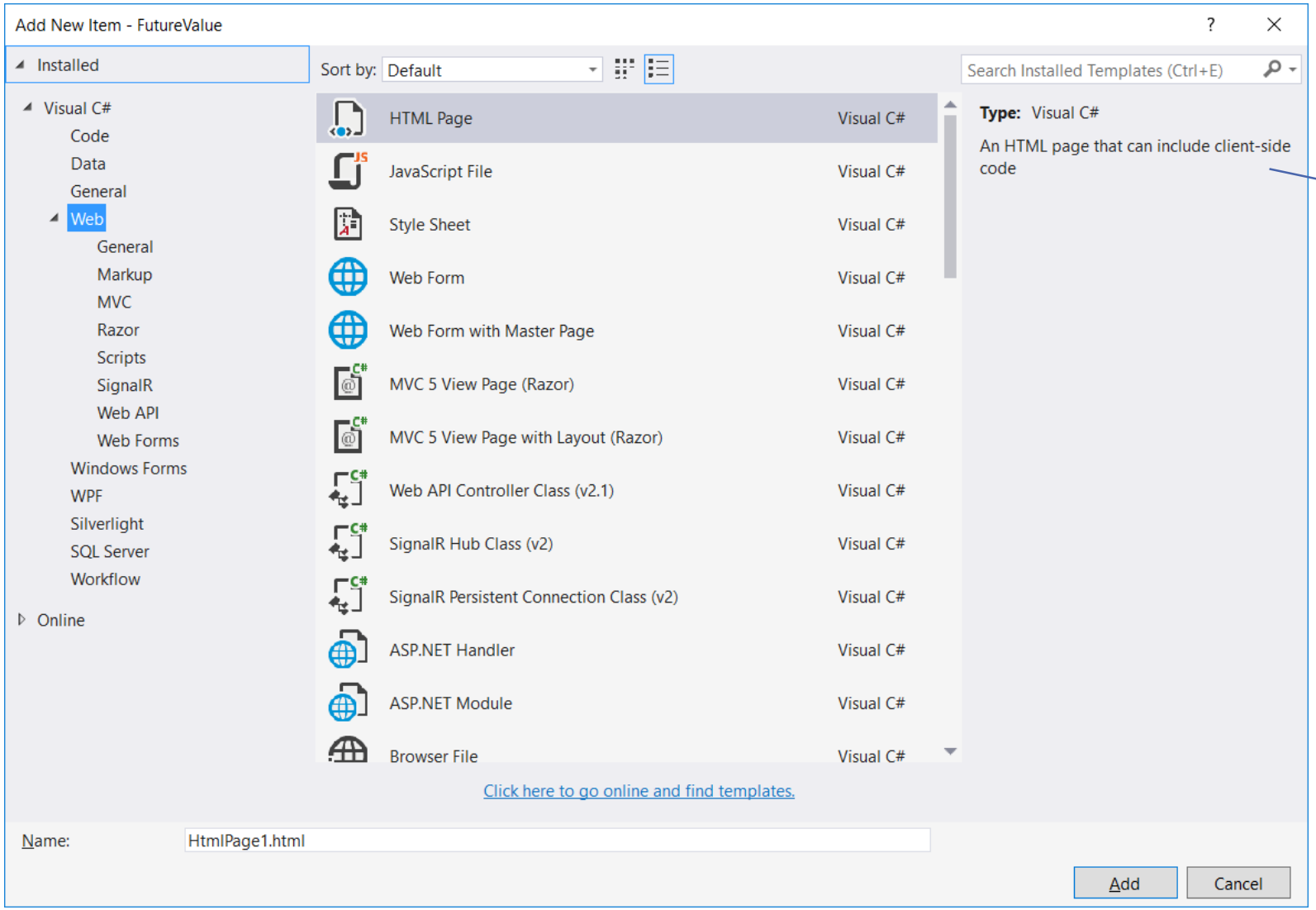
- Programming code can be placed into one file along with the HTML OR stored separately:

Working with Files in Your Project

- Many file types of an ASP .NET each offering a distinct functionality.

- To add files:
  - Solution Explorer: right-click your project and choose Add → Add New Item
  - Project menu: Add New Item

Add New Item



Description of item

Adding Existing Files

- Already existing file that you might have created before and want to reuse such as a images, logo, CSS, scripts, ..etc.

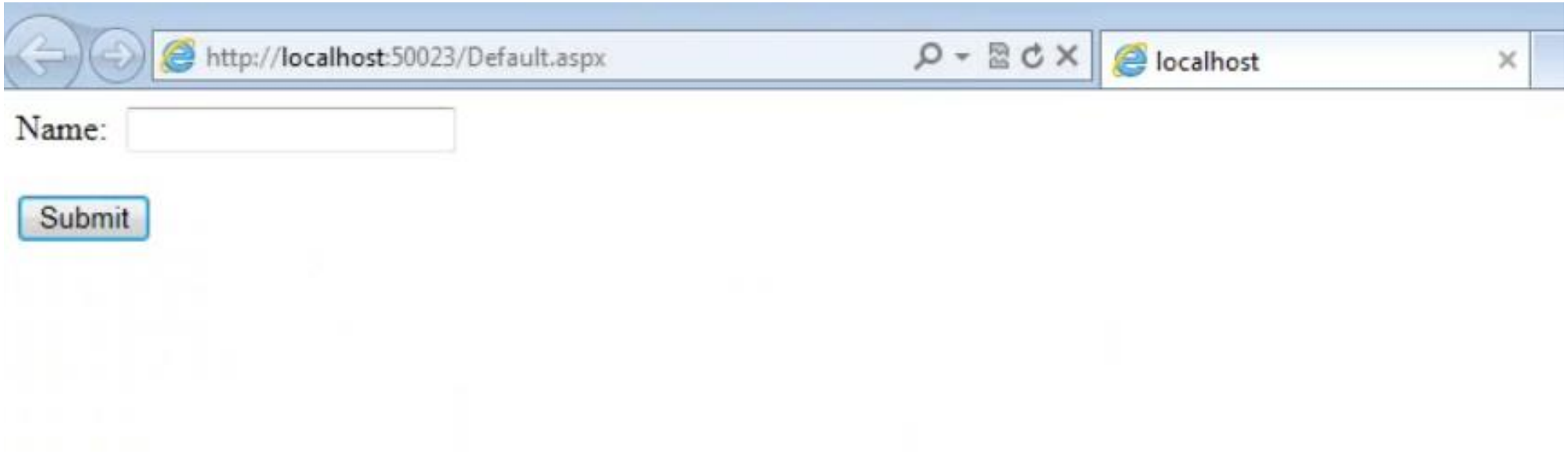
- To add such files, use any of the following
  - Project Menu: add existing item
  - Solution Explorer: right click → add existing item
  - Copy from file explorer and paste in the Solution Explorer
  - Drag from file explorer and drop into the Solution Explorer

Organizing Your Site

- Because of the many files that make up your site, it's often a good idea to group them by function in separate folders.
- Example:
  - Style sheet files could go in a folder called Styles
  - .js files containing JavaScript could go in Scripts
  - User controls could go in a Controls folder
  - Master pages could be stored in a folder called MasterPages.
- This is a matter of personal preference, but structured and well-organized sites are easier to manage and understand.

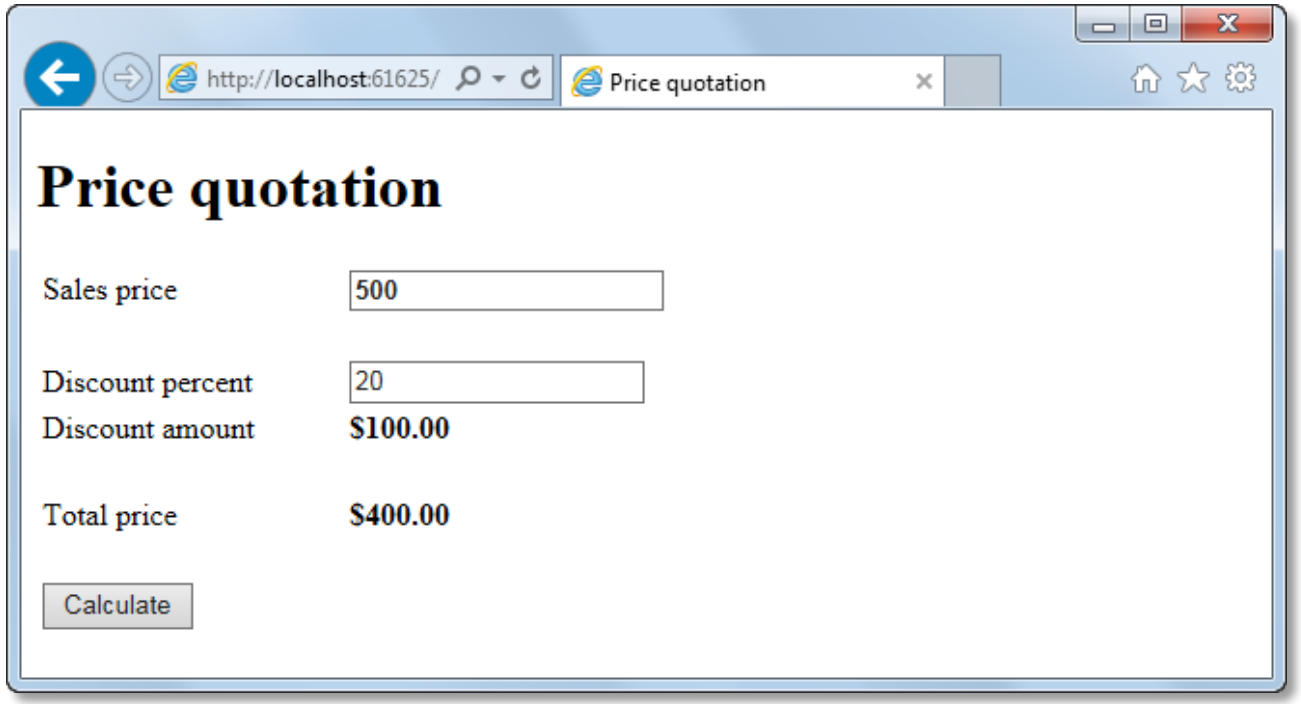
Demo: First Project

- Simple form:
  - Take the user “Name” and display it back on the same page.



Exercise: Quotation Builder

- Create a web form to generate a quotation.
- Inputs:
  - Sales price
  - Discount percent
- Output:
  - Discount amount
  - Total price



Q & A