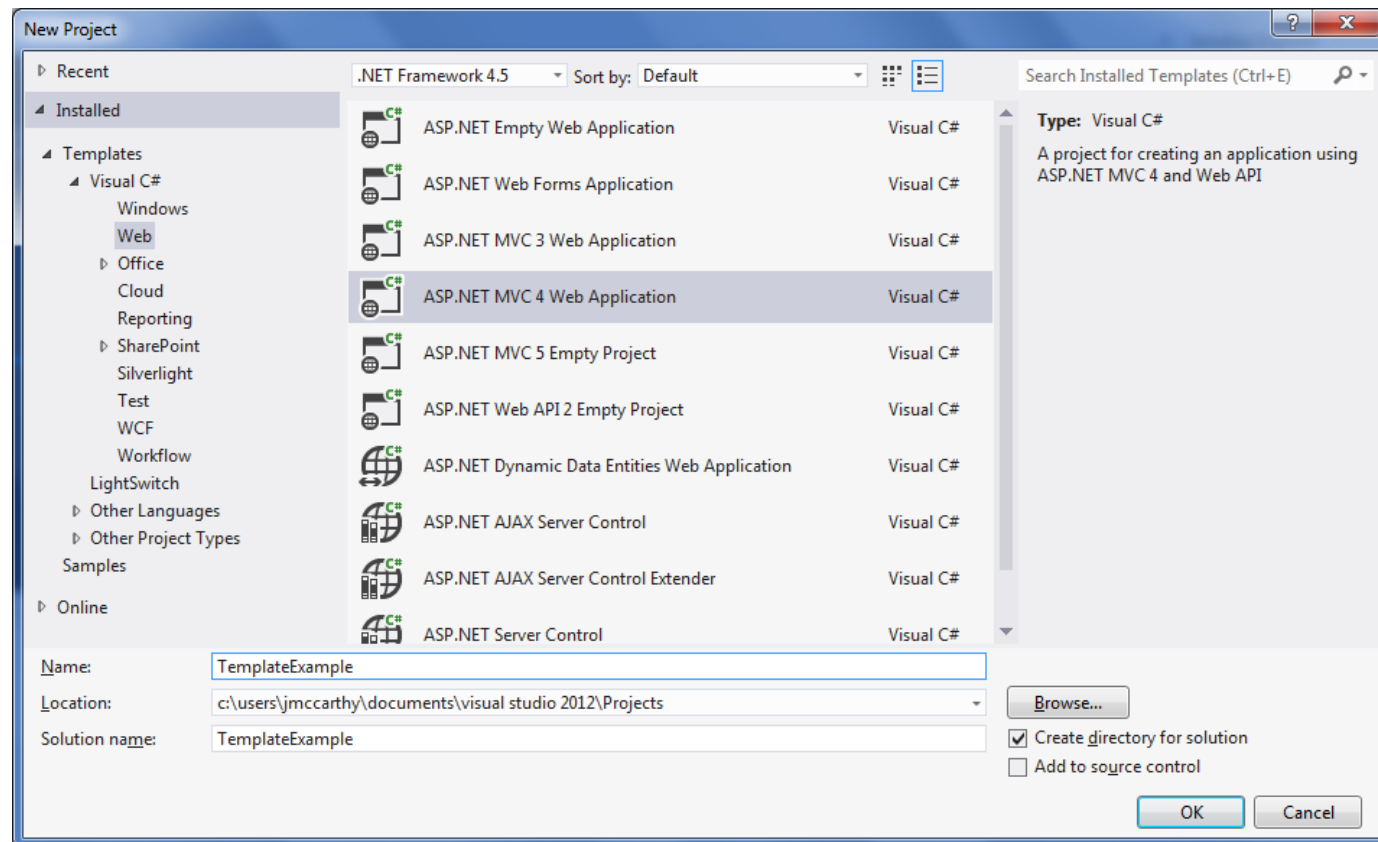


# Adding HTML, CSS and JavaScript to an ASP.NET MVC Web App

# Create a new Project

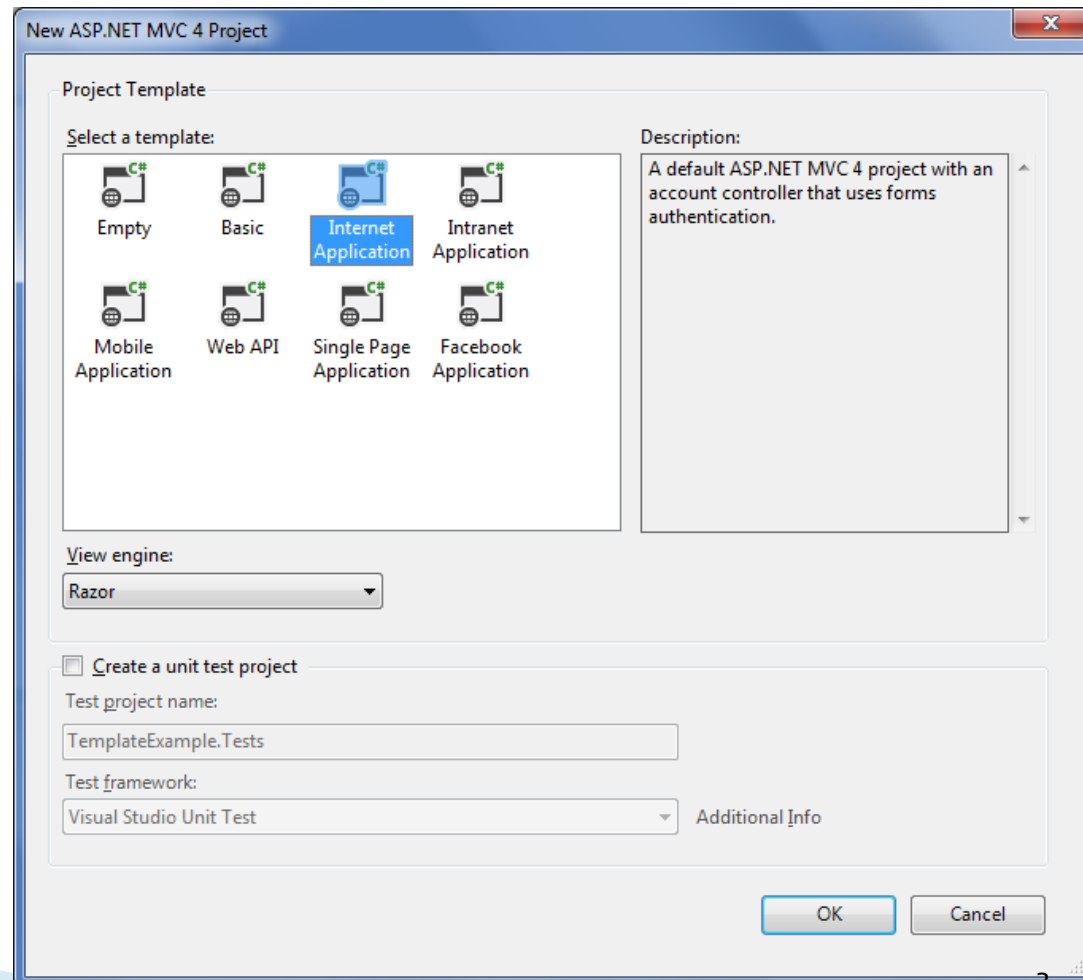
- ▶ Type: ASP.NET MVC 4 Web Application

Name:  
TemplateExample



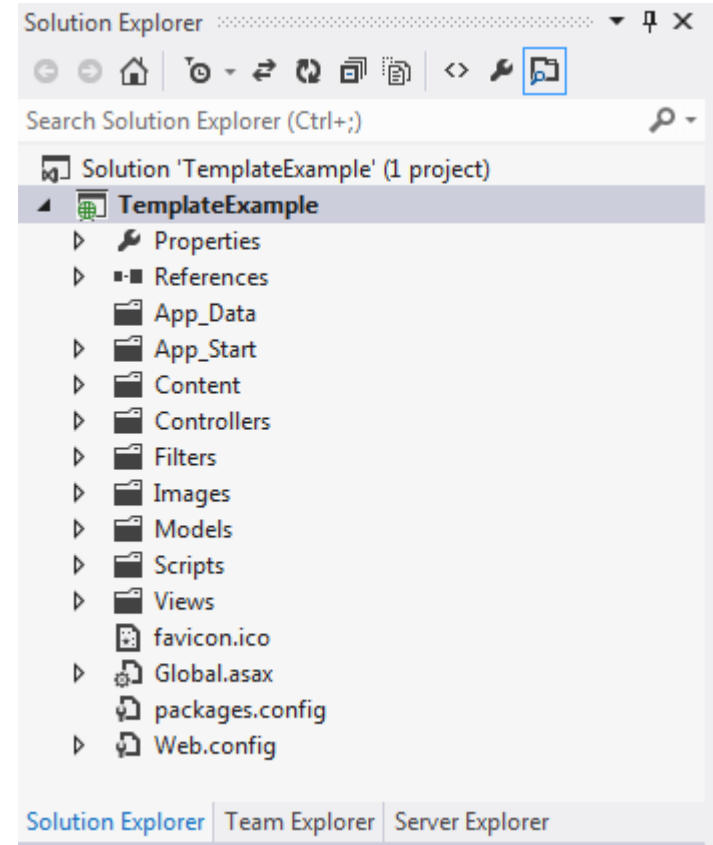
# Create a new Project

- ▶ Select Internet Application
- ▶ Click OK.



# Project Layout

- ▶ The Solution Explorer shows the hierarchy of the project folder.
- ▶ The **Content** folder contains a file named **Site.css**.
- ▶ The **Scripts** folder contains all the JavaScript files and libraries\frameworks for the project.
- ▶ The **Views** folder contains a **Shared** folder with a file named **\_Layout.cshtml**. This contains the layout template for the given application.



# \_Layout.cshtml

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>@ViewBag.Title - My ASP.NET MVC Application</title>
    <link href="~/favicon.ico" rel="shortcut icon" type="image/x-icon" />
    <meta name="viewport" content="width=device-width" />
    @Styles.Render("~/Content/css")
    @Scripts.Render("~/bundles/modernizr")
  </head>
```

- ▶ Bundling and Minification techniques are incorporated in ASP.NET 4.5 to improve load times.

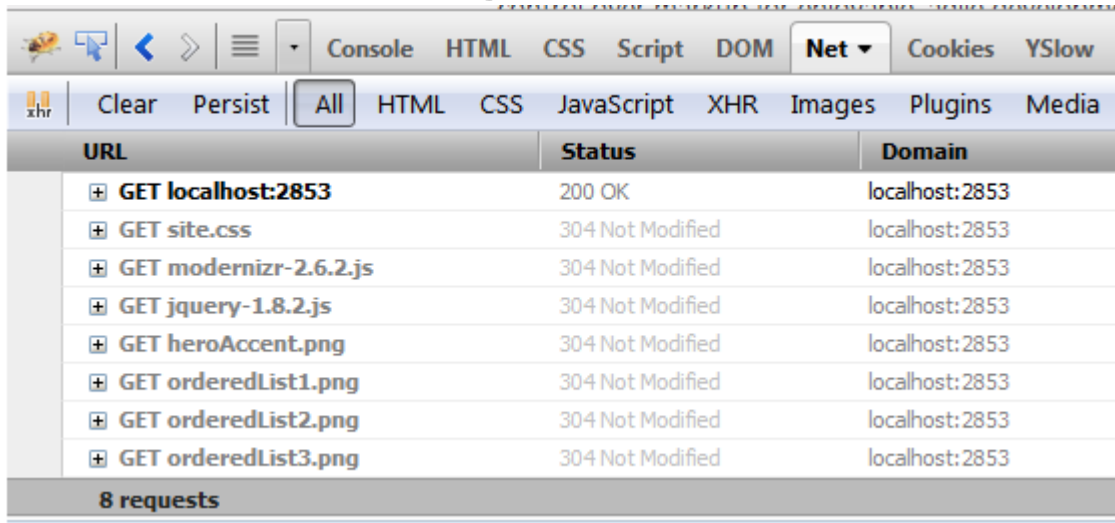
# CSS in \_Layout.cshtml

- ▶ To add a CSS file to the web application:
- ▶ `@Styles.Render("~/Content/css")`
- ▶ `~/Content/css` is a reference to a bundle created in the file `App_Start\BundleConfig.cs`  

```
bundles.Add(new StyleBundle("~/Content/css").Include("~/Content/site.css"));
```
- ▶ For a given bundle, files will be added to the bundle. Eg the file `site.css` has been added to this bundle. When the page is loaded in a browser, the following is observed in FireBug.

# CSS in \_Layout.cshtml

- ▶ The following files were loaded.



The screenshot shows a browser's developer console with the 'Net' tab selected. The console displays a list of network requests. The first request is a successful GET request to localhost:2853. The subsequent seven requests are for various files (site.css, modernizr-2.6.2.js, jquery-1.8.2.js, heroAccent.png, orderedList1.png, orderedList2.png, and orderedList3.png), all of which returned a 304 Not Modified status, indicating they were already cached.

URL	Status	Domain
GET localhost:2853	200 OK	localhost:2853
GET site.css	304 Not Modified	localhost:2853
GET modernizr-2.6.2.js	304 Not Modified	localhost:2853
GET jquery-1.8.2.js	304 Not Modified	localhost:2853
GET heroAccent.png	304 Not Modified	localhost:2853
GET orderedList1.png	304 Not Modified	localhost:2853
GET orderedList2.png	304 Not Modified	localhost:2853
GET orderedList3.png	304 Not Modified	localhost:2853

8 requests

- ▶ The site.css file has loaded as expected.

# Add a CSS file to \_Layout.cshtml

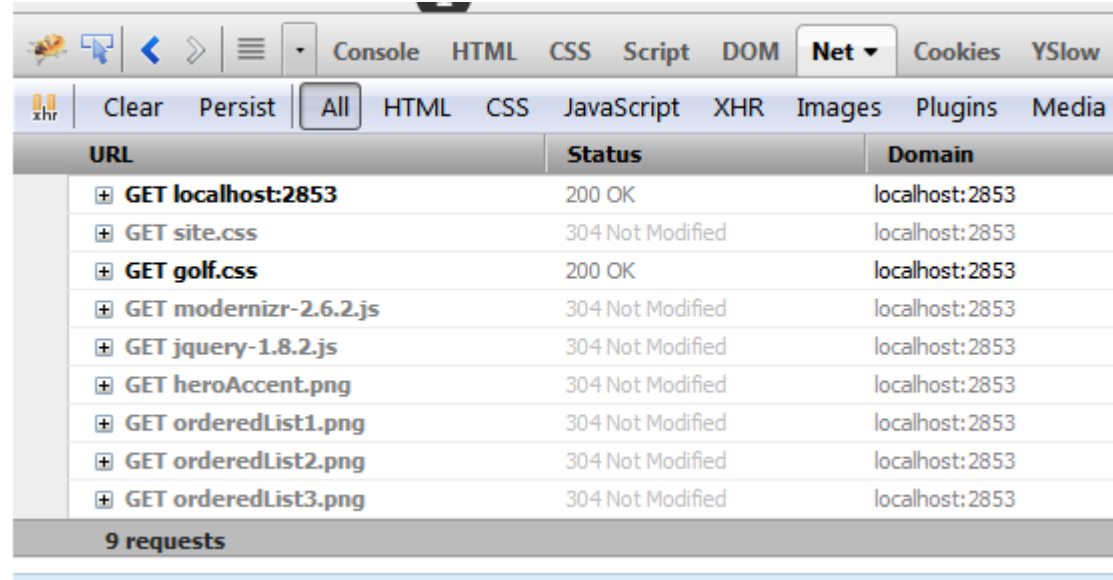
- ▶ Add a new file named Golf.css to the Content folder. Right click the Content folder and select Add -> New Item.
- ▶ To add a new CSS file we can update the BundleConfig.cs file.
- ▶ Update the css bundle to look as follows:

```
bundles.Add(new StyleBundle("~/Content/css").Include("~/Content/site.css", "~/Content/golf.css"));
```

- ▶ The Include method can take multiple files.
- ▶ When the application is run the Golf.css file will be loaded when the page is viewed in a browser. Check FireBug to make sure it appears. The following slide shows the FireBug screenshot.

# Add a CSS file to \_Layout.cshtml

- ▶ The two CSS files are now added to the \_Layout.cshtml file.
- ▶ Next we will try optimise the page by combining all CSS to a single file that has been Minified.



The screenshot shows a browser's developer console with the 'Net' tab selected. The console displays a list of network requests. The first request is a successful GET for 'localhost:2853'. The subsequent requests are for 'site.css' and 'golf.css', both successful (200 OK). The remaining requests are for 'modernizr-2.6.2.js', 'jquery-1.8.2.js', 'heroAccent.png', 'orderedList1.png', 'orderedList2.png', and 'orderedList3.png', all of which returned a 304 Not Modified status. The total number of requests is 9.

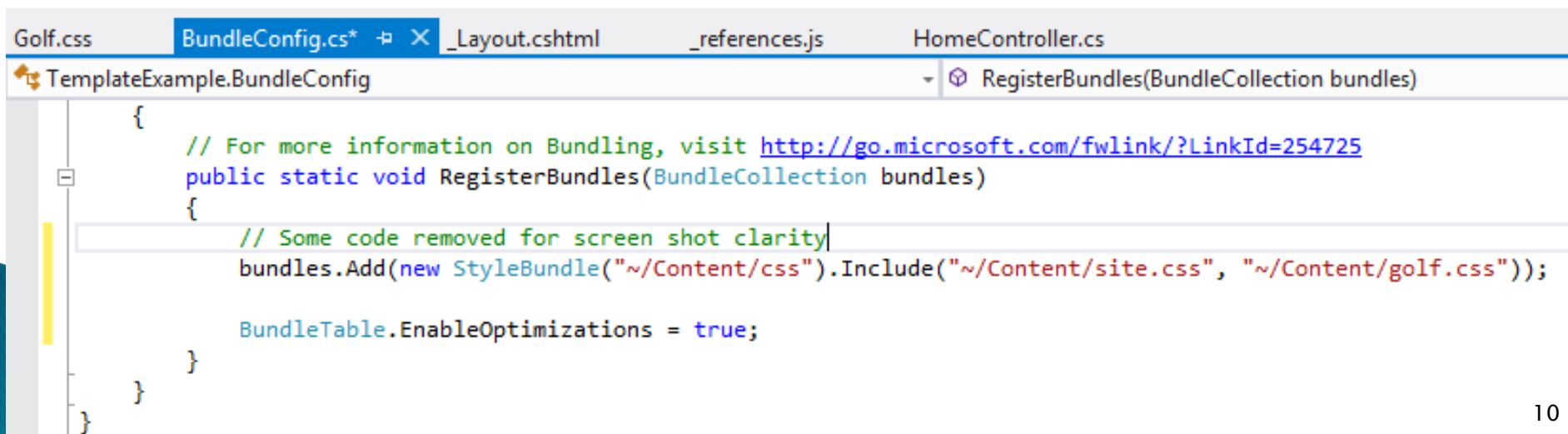
URL	Status	Domain
GET localhost:2853	200 OK	localhost:2853
GET site.css	304 Not Modified	localhost:2853
GET golf.css	200 OK	localhost:2853
GET modernizr-2.6.2.js	304 Not Modified	localhost:2853
GET jquery-1.8.2.js	304 Not Modified	localhost:2853
GET heroAccent.png	304 Not Modified	localhost:2853
GET orderedList1.png	304 Not Modified	localhost:2853
GET orderedList2.png	304 Not Modified	localhost:2853
GET orderedList3.png	304 Not Modified	localhost:2853

9 requests

# How to Minify CSS and JavaScript and combine to a single file.

- ▶ To combine all CSS and JavaScript to a single file, add the following line to the BundleConfig.cs file, inside the RegisterBundles method. (example shown below)

**BundleTable.EnableOptimizations = true;**

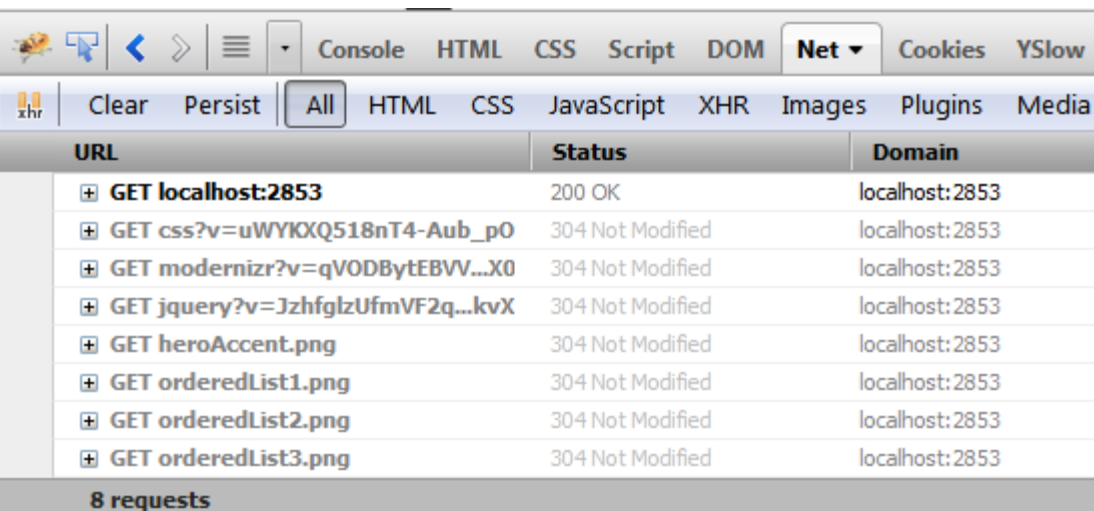


```
Golf.css  BundleConfig.cs*  _Layout.cshtml  _references.js  HomeController.cs
TemplateExample.BundleConfig  RegisterBundles(BundleCollection bundles)
{
    // For more information on Bundling, visit http://go.microsoft.com/fwlink/?LinkId=254725
    public static void RegisterBundles(BundleCollection bundles)
    {
        // Some code removed for screen shot clarity
        bundles.Add(new StyleBundle("~/Content/css").Include("~/Content/site.css", "~/Content/golf.css"));

        BundleTable.EnableOptimizations = true;
    }
}
```

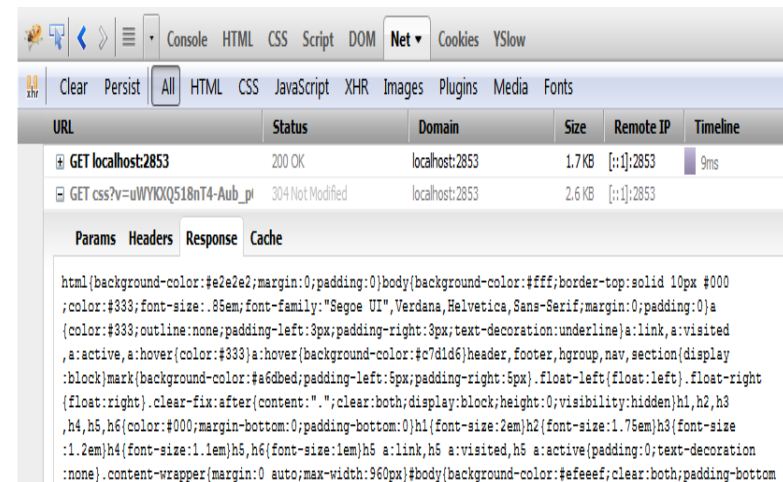
# How to Minify CSS and JavaScript and combine to a single file.

- ▶ Run the application and inspect the page load in FireBug. It should look as follows:
- ▶ The two files (site.css and golf.css) have been combined and Minified.
- ▶ Test this in FireBug to ensure it is working correctly.



8 requests

URL	Status	Domain
GET localhost:2853	200 OK	localhost:2853
GET css?v=uWYKXQ518nT4-Aub_p0	304 Not Modified	localhost:2853
GET modernizr?v=qVODBytEBVV...X0	304 Not Modified	localhost:2853
GET jquery?v=JzhfglzUfmVF2q...kvX	304 Not Modified	localhost:2853
GET heroAccent.png	304 Not Modified	localhost:2853
GET orderedList1.png	304 Not Modified	localhost:2853
GET orderedList2.png	304 Not Modified	localhost:2853
GET orderedList3.png	304 Not Modified	localhost:2853



Params Headers Response Cache

```
html{background-color:#e2e2e2;margin:0;padding:0}body{background-color:#fff;border-top:solid 10px #000;color:#333;font-size:.85em;font-family:"Segoe UI", Verdana, Helvetica, Sans-Serif;margin:0;padding:0}a{color:#333;outline:none;padding-left:3px;padding-right:3px;text-decoration:underline}a:link,a:visited,a:active,a:hover{color:#333}a:hover{background-color:#c7d1d6}header,footer,hgroup,nav,section{display:block}mark{background-color:#6dbed;padding-left:5px;padding-right:5px}.float-left{float:left}.float-right{float:right}.clear-fix:after{content:"";clear:both;display:block;height:0;visibility:hidden}h1,h2,h3,h4,h5,h6{color:#000;margin-bottom:0;padding-bottom:0}h1{font-size:2em}h2{font-size:1.75em}h3{font-size:1.2em}h4{font-size:1.1em}h5,h6{font-size:1em}h5 a:link,h5 a:visited,h5 a:active{padding:0;text-decoration:none}.content-wrapper{margin:0 auto;max-width:960px}body{background-color:#efefef;clear:both;padding-bottom
```

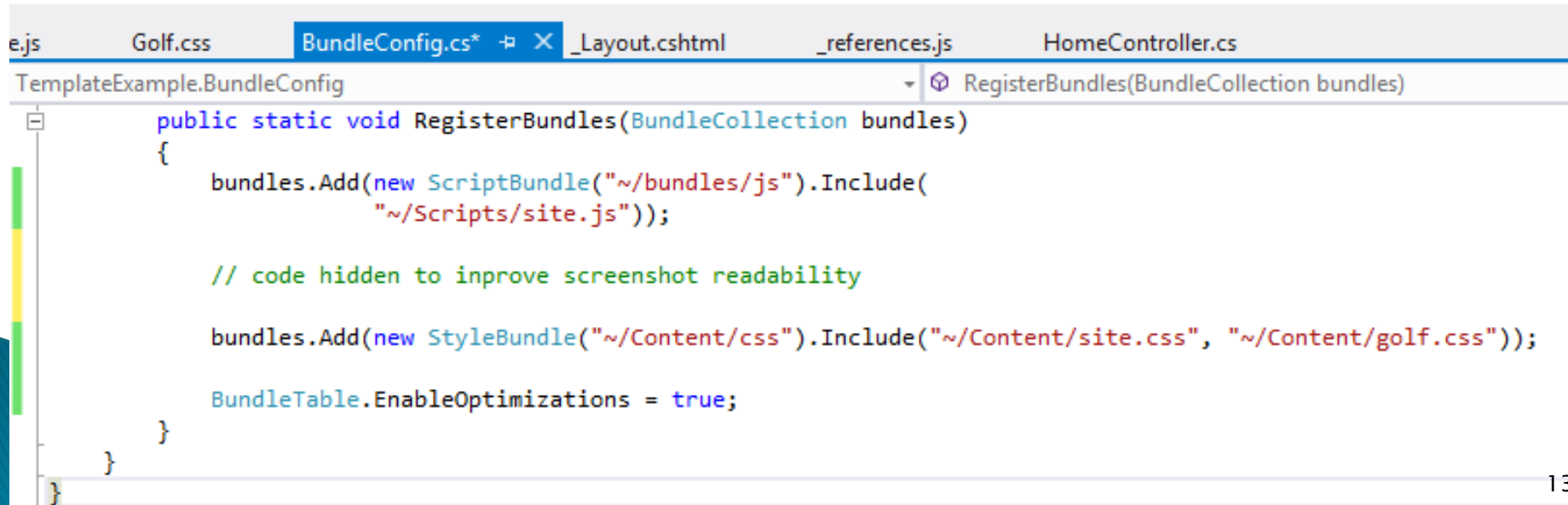
# Add a JavaScript file to \_Layout.cshtml

- ▶ Create a new file named Site.js in the Scripts folder.
- ▶ Add the following to the new file:
- ▶ `console.log('Hello World');`
- ▶ Next update the BundleConfig.cs to cater for our JavaScript files. The following slide will show what to add.

# Add a JavaScript file to \_Layout.cshtml

- ▶ Add the following code to the top of the RegisterBundles method in the BundleConfig.cs file.

```
bundles.Add(new ScriptBundle("~/bundles/js").Include(
    "~/Scripts/site.js"));
```



The screenshot shows the Visual Studio IDE with the following tabs: e.js, Golf.css, BundleConfig.cs\* (selected), \_Layout.cshtml, \_references.js, and HomeController.cs. The BundleConfig.cs file is open, showing the RegisterBundles method. The code is as follows:

```
public static void RegisterBundles(BundleCollection bundles)
{
    bundles.Add(new ScriptBundle("~/bundles/js").Include(
        "~/Scripts/site.js"));

    // code hidden to improve screenshot readability

    bundles.Add(new StyleBundle("~/Content/css").Include("~/Content/site.css", "~/Content/golf.css"));

    BundleTable.EnableOptimizations = true;
}
```

# Add a JavaScript file to \_Layout.cshtml

- ▶ Next we have to add a reference to \_Layout.cshtml to call the new JavaScript bundle.
- ▶ Add the following line to the bottom of the \_Layout.cshtml file, above the jQuery bundles reference.

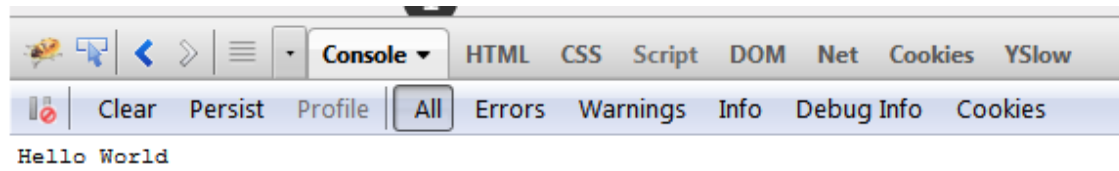
`@Scripts.Render("~/bundles/js")`

- ▶ It should look as follows:

```
    </footer>
    @Scripts.Render("~/bundles/js")
    @Scripts.Render("~/bundles/jquery")
    @RenderSection("scripts", required: false)
</body>
</html>
```

# Add a JavaScript file to \_Layout.cshtml

- ▶ Run the application to ensure it is working correctly.
- ▶ Use FireBug to test if the Hello World message is appearing in the Console window.



- ▶ The JavaScript code has been Minified, similar to the CSS example.

# Adding a HTML template to a ASP.NET MVC Web Application

- ▶ Following on from the previous example, when the application is launched it looks as follows:

your logo here

Register Log in

Home About Contact

**Home Page.** Modify this template to jump-start your ASP.NET MVC application.

To learn more about ASP.NET MVC visit <http://asp.net/mvc>. The page features [videos, tutorials, and samples](#) to help you get the most from ASP.NET MVC. If you have any questions about ASP.NET MVC visit [our forums](#).

**We suggest the following:**

- 1 Getting Started**  
ASP.NET MVC gives you a powerful, patterns-based way to build dynamic websites that enables a clean separation of concerns and that gives you full control over markup for enjoyable, agile development. ASP.NET MVC includes many features that enable fast, TDD-friendly development for creating sophisticated applications that use the latest web standards. [Learn more...](#)
- 2 Add NuGet packages and jump-start your coding**  
NuGet makes it easy to install and update free libraries and tools. [Learn more...](#)
- 3 Find Web Hosting**  
You can easily find a web hosting company that offers the right mix of features and price for your applications. [Learn more...](#)

© 2014 - My ASP.NET MVC Application

# Adding a HTML template to a ASP.NET MVC Web Application

- ▶ Download the OnlineBookStore.zip file from Moodle. This is a static html page using css and JavaScript.



# Adding a HTML template to a ASP.NET MVC Web Application

- ▶ Step 1: Update the HTML
- ▶ Open ProjectExample.htm and copy the html markup.
- ▶ Paste the html into \_Layout.cshtml. (Ie replace whatever is there).

# Update the head section

- ▶ Update the head section to look as follows:

```
Site.js Golf.css BundleConfig.cs _Layout.cshtml x _references.js Ho
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <title>@ViewBag.Title - My ASP.NET MVC Application</title>
  <link href="~/favicon.ico" rel="shortcut icon" type="image/x-icon" />
  <meta name="viewport" content="width=device-width" />
  @Styles.Render("~/Content/css")
  @Scripts.Render("~/bundles/modernizr")
</head>
```

# Update the Navigation

- ▶ Update the navigation section to look as follows:

```
<div id='menu'>
  <!-- Header Image Displayes here -->
  <div id="navbar">
    <ul>
      <li>@Html.ActionLink("Home", "Index", "Home")</li>
      <li>@Html.ActionLink("About", "About", "Home")</li>
      <li>@Html.ActionLink("Contact", "Contact", "Home")</li>
    </ul>
  </div>
</div>
```

# Update the Body Section

- ▶ Update the content div to look as follows:

```
<div id='content'>  
  @RenderSection("featured", required: false)  
  <section class="content-wrapper main-content clear-fix">  
    @RenderBody()  
  </section>  
</div>
```

# Add the JavaScript references to the bottom of the page.

- ▶ Add the JavaScript references as shown:

```
</div>
<div id='footer'>
  <h6 class="footertext">&copy; Online Stores NCI&trade; 2014</h6>
</div>

@Scripts.Render("~/bundles/js")
@Scripts.Render("~/bundles/jquery")
@RenderSection("scripts", required: false)

</body>
</html>
```

# Add the required images

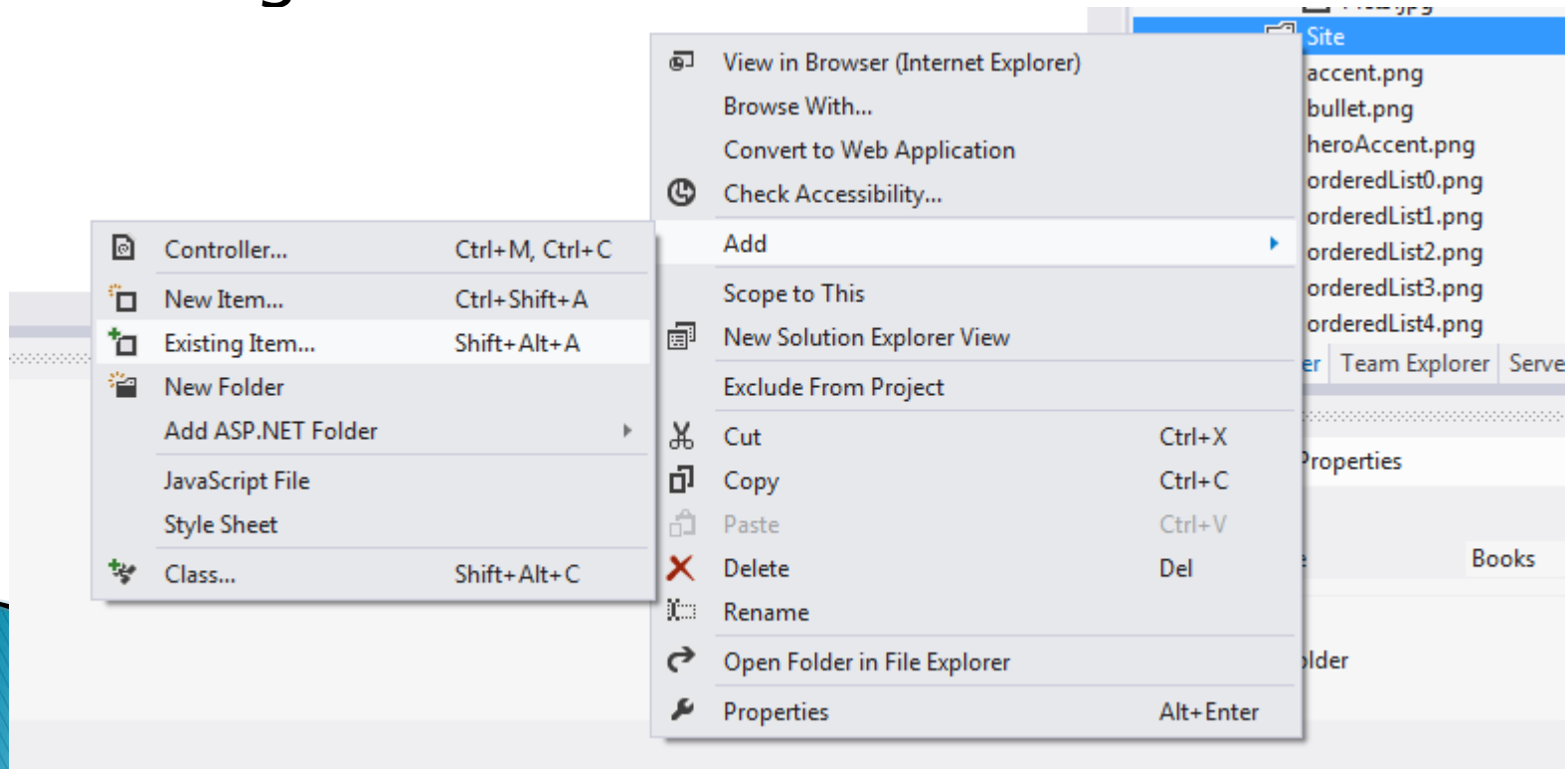
- ▶ There are a number of different images required for this html page.
- ▶ There are two folders within the Images folder for this OnLineBookStore template.
- ▶ We will add these two folders to the Images folder in our TemplateExample Project.

# Create the image folders

- ▶ Right click the Images folder in the Solution Explorer and select Add -> New Folder
- ▶ Name the folder Books
  
- ▶ Right click the Images folder in the Solution Explorer and select Add -> New Folder
- ▶ Name the folder Site

# Add images to the folders

- ▶ We now need to add the images to the Books and Site folders.
- ▶ Right click the Books folder, select Add – Existing Item:



# Add images to the folders

- ▶ Browse to the folder where the Books images are stored (ie the Books folder from the OnLineBookStore.zip file).
- ▶ Select all images (Ctrl A)
- ▶ Click Add
  
- ▶ Repeat for the Site images folder.

# Add the CSS

- ▶ Paste the content of SiteExample.css and two-column.css into Site.css (ie replace the current css in the Site.css file).

# Add the JavaScript

- ▶ If any CSS is required, it can be added to the Site.js file.

# Run the Web Application

- ▶ Run the App, it should now look as follows:

http://localhost:2853/ Home Page - My ASP.NET ...

## Online BookStore

Home About Contact

### Home Page.

**Modify this template to jump-start your ASP.NET MVC application.**

To learn more about ASP.NET MVC visit <http://asp.net/mvc>. The page features **videos, tutorials, and samples** to help you get the most from ASP.NET MVC. If you have any questions about ASP.NET MVC visit [our forums](#).

**We suggest the following:**

- 1. Getting Started**

ASP.NET MVC gives you a powerful, patterns-based way to build dynamic websites that enables a clean separation of concerns and that gives you full control over markup for enjoyable, agile development. ASP.NET MVC includes many features that enable fast, TDD-friendly development for creating sophisticated applications that use the latest web standards. [Learn more...](#)
- 2. Add NuGet packages and jump-start your coding**

NuGet makes it easy to install and update free libraries and tools. [Learn more...](#)
- 3. Find Web Hosting**

You can easily find a web hosting company that offers the right mix of features and price

### Search

# Conclusion

- ▶ The HTML template has now been added to the project.