Today's Agenda

- Building blocks for creating web sites
  - You can use layouts for content shared by views
  - Sections are for working with content that belongs to individual views
  - You can use partial views for fragments of layout you want to inject and reuse across views
  - You can inject dynamic content into views by using child actions
- Exercises

Break

- Dividing the application into Areas (e.g. public part / administration part)
- Packages, bundling and minification
- Exercises (continued)

Questions

WU-Backend 16a (1wu16aBACK) > Documents > Presentations/lesson_07 questions.pdf

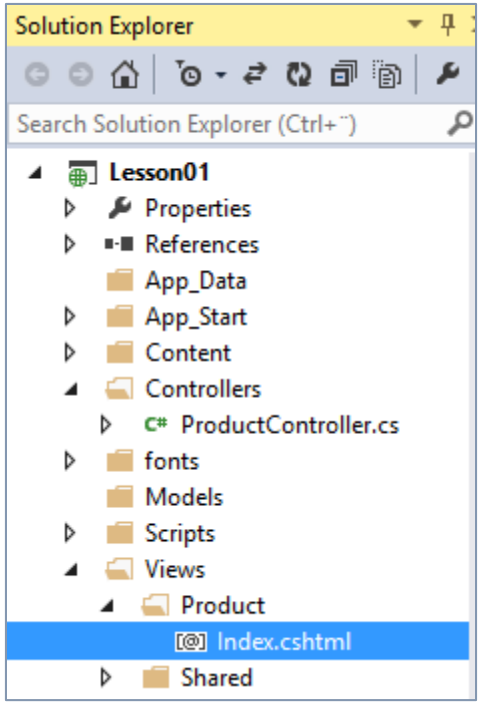
Views

Folder and file naming conventions for views (sp. 1.1)

Convention over configuration

- Directories
  - Controllers
  - Models
  - Views
- Naming conventions for Controllers
  - Each controller's class name ends with *Controller*: `ProductController`
- Naming conventions for Views
  - Views that controllers use is in a **subfolder** named after the controller and **filename** named after the action method. For example:

`/Views/Product/Index.cshtml`



Controller name

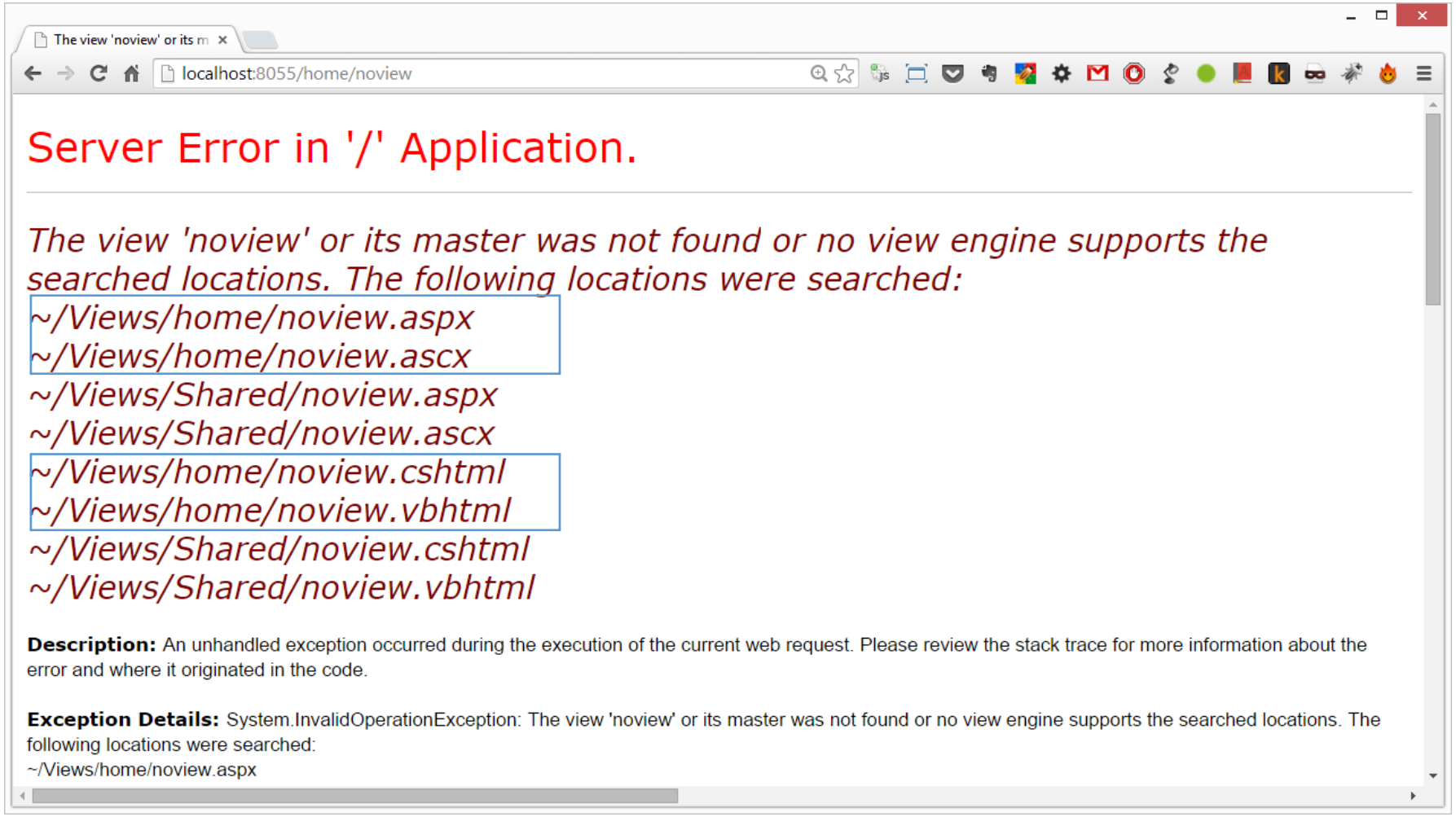
Action Method name

View Search Locations (sp. 1.2)

- `~/Views/Home/Index.cshtml`
- `~/Views/Home/Index.vbhtml`
- `~/Views/Shared/Index.cshtml`
- `~/Views/Shared/Index.vbhtml`

- You can change the locations of the views by extending the **`RazorViewEngine`** class (change the **`ViewLocationsFormats`** property that is set in the constructor) and register the extended class in the **`global.asax`** file (see Freeman pp. 570-73 for detailed instructions).

The view or its master was not found error



Layout pages

For content shared by views

Quick Overview (sp. 2.1)

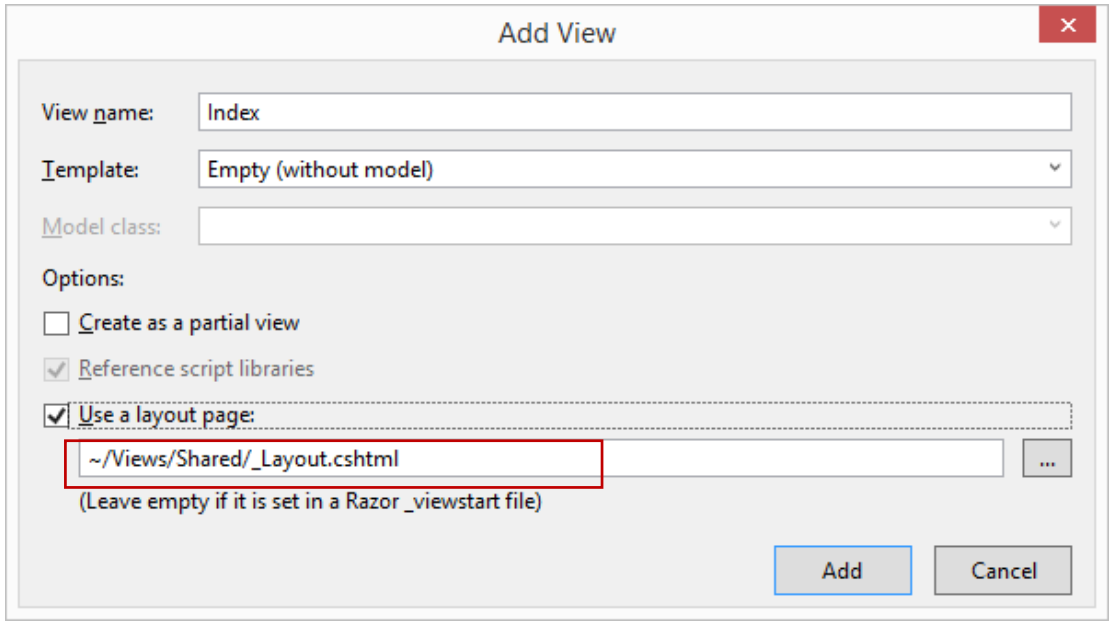
Layout files (sp. 2.2 + 2.3)

The first time you check,
Use a layout page
with no file reference, you'll get:

- `Contents\bootstrap.css`
- `Contents\bootstrap.min.css`
- `Contents\Site.css`

- `\Shared\_Layout.cshtml`
- `\_ViewStart.cshtml`

1.



2.

`Views\_ViewStart.cshtml` with content:

```
1   @{
2       Layout = "~/Views/Shared/_Layout.cshtml";
3   }
```

~/Views/Shared/_Layout.cshtml

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width" />
    <title>@ViewBag.Title</title> <!-- @ViewBag.Title from the view -->
</head>
<body>
    <div id="wrapper">
        <div class="layout">
            This is part of the layout file
        </div>

        @RenderBody() <!-- content from the view goes into this section -->

        <footer>[Insert footer information here]</footer>
    </div>
</body>
</html>
```

The View file (sp. 2.2 + 2.3)

```
@{
    ViewBag.Title = "Lesson 7, examples";
    // Layout file reference
    // Layout = "~/Views/Shared/_Layout.cshtml
    // Layout = null
    // No layout file reference
<!- Goes into the Body-section of the layout file -->
<p>This it part of the view</p>
```

~/Views/Shared/_Layout.cshtml

- a more elaborated example with sections (sp. 2.4)

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width" />
    <title>@ViewBag.Title</title>
</head>
<body>
    <div id="wrapper">
        <header>
            // false means that the section is optional; true is default
            @RenderSection("Header", true) // Content for views
        </header>
        <div class="main-content">
            @RenderBody() // Mandatory for all views
        </div>
        <footer>
            @RenderSection("Footer", true) // Content for views
        </footer>
    </div>
    @RenderSection("Scripts", false) </body> // Optional content for views
</html>
```

The view file
- with a `string[]` model sent in from controller

Index.cshtml

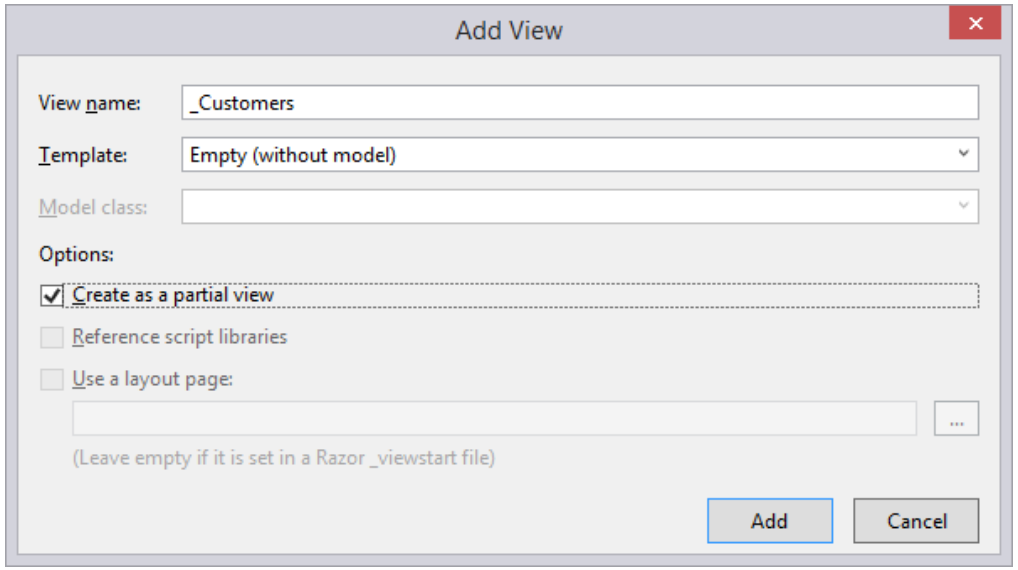
```
@model string[]
@{
    ViewBag.Title = "Lesson 7, examples";

}
@section Header {
    <div class="view">
        @foreach (string str in new[] { "Index", "List", "Edit" }) {
            @Html.ActionLink(str, str, null, new { style = "margin: 5px" })
        }
    </div>
}
<div class="view">
    This is a list of fruit names:
    @foreach (string name in Model) {
        <span><b>@name</b></span>
    }
</div>
@section Footer {
    <div class="view">
        This is the footer from the view
    </div>
}
```

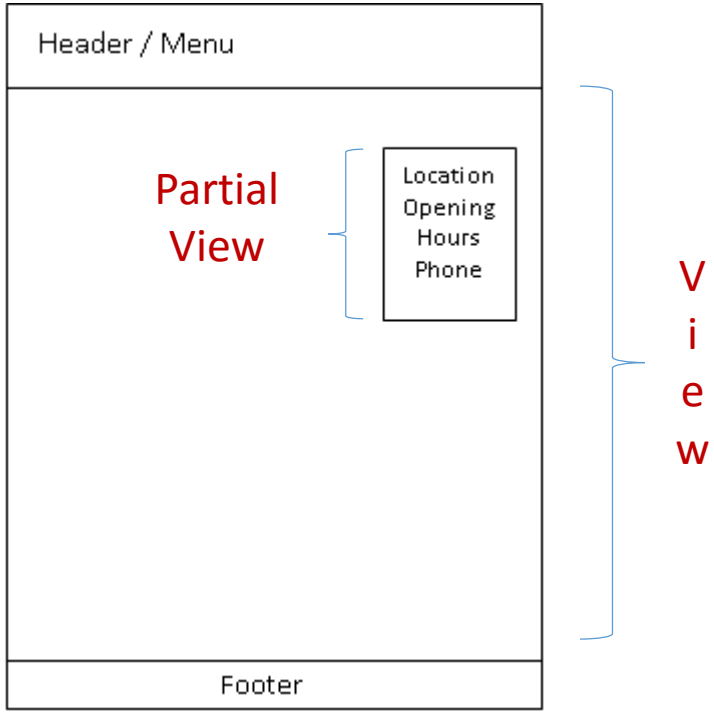
Partial views

Fragments of layout to inject and reuse across views

Partial Views (sp. 3.1+3.2)

- Fragments of markup and data to be **included in other views** (for **re-use**)
- Add a partial view





- Partial views should be located in the **`views\[controllernamefolder]`** or – if it's used from more than one view – in the **`views\shared`** folder

To Include Partial Views in a View (sp. 3.3)

- You can use html and razor in partial view files as in ordinary view files
- You can include partial view in views and layout files:

```
@Html.Partial("_PartialPage")
```

- You can also send a model to a partial view

```
@Html.Partial("_PartialPage", Model)
```

The strongly typed partial view

```
@model IEnumerable<string>
<div>
    Delicious fruit:
    <ul>
        @foreach (string name in Model) {
            <li>@name</li>
        }
    </ul>
</div>
```

The partial view

```
@Html.Partial("_Fruit", Model)
```

Calling the partial view from the view

Child actions

Child Actions (sp. 4.1+4.2)

- Child actions are to actions as partial views are to views
- You can use child actions whenever you have a piece of code your want to use in multiple controller action methods –
  Like a data-drive widget (reuse of code that generates dynamic content)
- The child action returns a partial view to display the data generated by child action

Child Action Method (sp. 4.3)

```
[ChildActionOnly]
public ActionResult ChildActionExample() {
    string[] myMovies = { "One Flew Over the Cuckoo's Nest",
                          "Schindler's List",
                          "Before Sunrise",
                          "Blue Ruin" };

    return PartialView(myMovies);
}
```

The Partial View (**`ChildActionExample`**) (sp. 4.4)

```
@model IEnumerable<string>

<p>
    Movies sent from a <strong>child action</strong>:
    <ul>
        @foreach (string movie in Model) {
            <li style="margin-left:30px;">@movie</li>
        }
    </ul>
</p>
```

Call and display the Child Action (sp. 4.4)

```
// call the same controller that generated the view
@Html.Action("ChildActionExample");

// call another controller
@Html.Action("ChildActionExample", "AnotherController");

// call the same controller & send a data object as parameter
@Html.Action("ChildActionExample", new {name="Sigmund"} );
```

The Razor Engine

- Translates views into C# classes and compiles the code
  - that's why it is easy to mix HTML code with C# fragments

- The initial request to MVC web application triggers the compilation process of all views

Exercises

Continuing building the MbmStore website

Mandatory Assignment 2 deadline: Mandag den 31. oktober

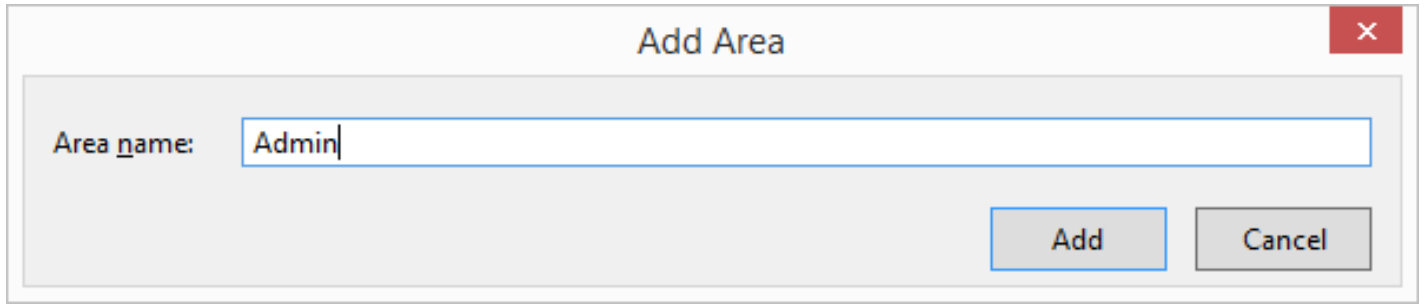
Areas

With Areas you can ….

- Divide a website into functional sub-segments (like administration, billing, customer support etc.)
- Each area can have its own folder structure with controllers, models, and views
- Accessing a specific area is supported through the routing system

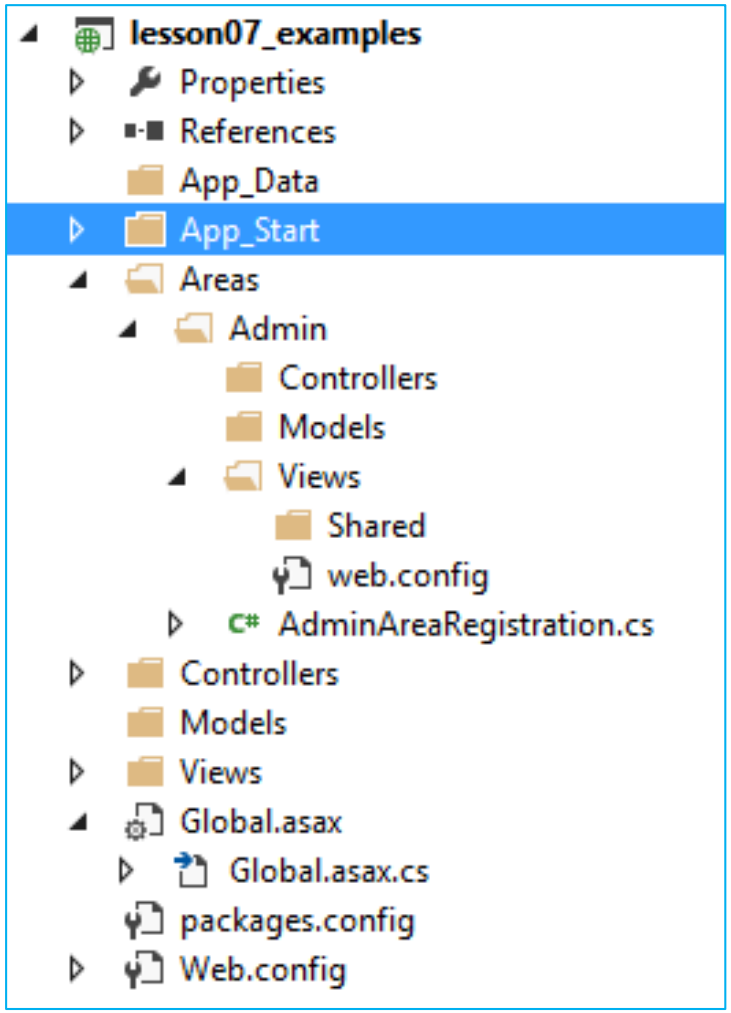
Add an Area

- Add -> Area …



You'll get:

- A new folder structure
- A **`web.config`** file in the Views folder
- A **`AdminAreaRegistration.cs`** file



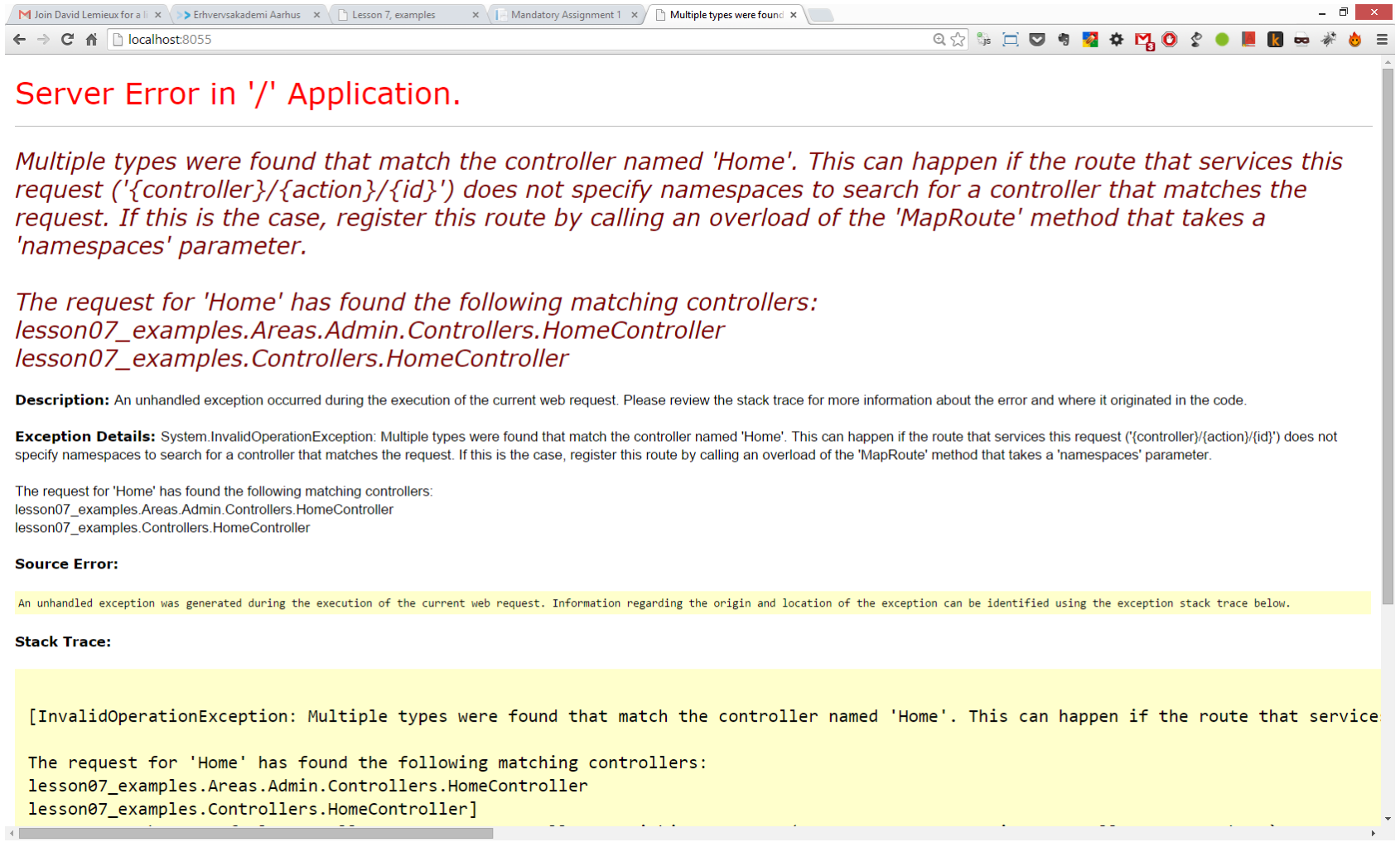
Admin/AdminAreaRegistration.cs

```
public class AdminAreaRegistration : AreaRegistration
{
    public override string AreaName
    {
        get
        {
            return "Admin";
        }
    }

    public override void RegisterArea(AreaRegistrationContext context)
    {
        context.MapRoute(
            "Admin_default",
            "Admin/{controller}/{action}/{id}",
            new { action = "Index", id = UrlParameter.Optional }
        );
    }
}
```

Global.asax – check to see that **AreaRegistration.RegisterAllAreas()** called

```
public class MvcApplication : System.Web.HttpApplication
{
    protected void Application_Start()
    {
        AreaRegistration.RegisterAllAreas();
        RouteConfig.RegisterRoutes(RouteTable.Routes);
        BundleConfig.RegisterBundles(BundleTable.Bundles);
    }
}
```

Caution: multiple controllers with the same name!



The cure

1. Use different names (RouteName and controller name) or
2. Add a namespace to the controller

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional },
        namespaces: new[] { "lesson07_examples.Controllers" }
    );
}
```

How to link to a different Area?

```
@Html.ActionLink("Admin", "Index", "Home",
new {area = "Admin" }, {@class="menu")
```

CSS, JavaScript

Bundling and minification

Install jQuery

To install jQuery, run the following command in the Package Manager Console

```
PM> Install-Package jQuery
```

Install Bootstrap

To install Bootstrap, run the following command in the Package Manager Console

```
PM> Install-Package Twitter.Bootstrap
```

You can ask for specific versions

```
Install-Package jQuery -version 1.10.2
Install-Package jQuery.Validation -version 1.11.1
Install-Package Microsoft.jQuery.Unobtrusive.Validation -version 3.0.0
Install-Package Bootstrap -version 3.0.0
Install-Package Microsoft.jQuery.Unobtrusive.Ajax -version 3.0.0
```

Include CSS and JS in the Layout file

– drag-and-drop from **Solution Explorer**

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="utf-8" />
5       <meta name="viewport" content="width=device-width" />
6       <title>@ViewBag.Title</title>
7       <link href="~/Content/bootstrap.css" rel="stylesheet" />
8       <link href="~/Content/bootstrap-theme.css" rel="stylesheet" />
9       <script src="~/Scripts/bootstrap.js"></script>
10      <script src="~/Scripts/jquery-1.9.0.js"></script>
11
12      @RenderSection("Scripts", false)
13
14
15  </head>
16  <body>
```

Script and CSS bundles

- Minifies JS and CSS files
- Gathers each type in a single file

- You'll need to install Microsoft ASP.NET Web Optimization Framework

```
PM> Install-Package Microsoft.AspNet.Web.Optimization
```

Views/web.config

- make the bundle-related classes available within views

```
11  <system.web.webPages.razor>
12    <host factoryType="System.Web.Mvc.MvcWebRazorHostFactory, System.Web.Mvc, Version=5.2.0.0,
13    <pages pageBaseType="System.Web.Mvc.WebViewPage">
14      <namespaces>
15        <add namespace="System.Web.Mvc" />
16        <add namespace="System.Web.Mvc.Ajax" />
17        <add namespace="System.Web.Mvc.Html" />
18        <add namespace="System.Web.Routing" />
19        <add namespace="System.Web.Optimization"/>
20        <add namespace="lesson07_examples" />
21      </namespaces>
22    </pages>
23  </system.web.webPages.razor>
```

App_Start.BundleConfig.cs

– is included in MVC project templates

```
 7    namespace lesson07_examples {
 8        public class BundleConfig {
 9            public static void RegisterBundles(BundleCollection bundles) {
10
11                bundles.Add(new StyleBundle("~/Content/css").Include(
12                "~/Content/*.css"));
13
14                bundles.Add(new ScriptBundle("~/Scripts/jquery-and-validation")
15                .Include("~/Scripts/jquery-{version}.js",
16                "~/Scripts/jquery.validate.js",
17                "~/Scripts/jquery.validate.unobtrusive.js",
18                "~/Scripts/jquery.unobtrusive-ajax.js"));
19            }
20        }
21    }
22
```

Bundle name

Scripts to bundle

global.asax

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using System.Web.Optimization;
7  using System.Web.Routing;
8
9  namespace lesson07_examples
10 {
11     public class MvcApplication : System.Web.HttpApplication
12     {
13         protected void Application_Start()
14         {
15             AreaRegistration.RegisterAllAreas();
16             RouteConfig.RegisterRoutes(RouteTable.Routes);
17             BundleConfig.RegisterBundles(BundleTable.Bundles);
18         }
19     }
20 }
21
```

Apply bundles to the layout file

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="utf-8" />
5       <meta name="viewport" content="width=device-width" />
6       <title>@ViewBag.Title</title>
7
8       @Styles.Render("~/Content/css")
9       @Scripts.Render("~/Scripts/jquery-and-validation")
10
11      @RenderSection("Scripts", false)
12
13  </head>
```

References the bundled and minified versions – HTML source view

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8" />
5      <meta name="viewport" content="width=device-width" />
6      <title>Lesson 7, examples</title>
7
8      <link href="/Content/css?v=zrPVifNAvSrIr6vahkfVwlBKAalk61woClq8fPY-6t41" rel="stylesheet"/>
9
10     <script src="/bundles/clientfeaturesscripts?v=R7cnfLpHk-OE7xj5C8NNZX6YQSZDGJ1-2PI2YaVvGk41"></script>
11
12
```

These files are saved in the browser cache one year or until bundles are updated

CSS (bundled & minified)

```
.btn-default,.btn-primary,.btn-success,.btn-info,.btn-
warning,.btn-danger{text-shadow:0 -1px 0 rgba(0,0,0,.2);-
webkit-box-shadow:inset 0 1px 0 rgba(255,255,255,.15),0 1px
1px rgba(0,0,0,.075);box-shadow:inset 0 1px 0
rgba(255,255,255,.15),0 1px 1px rgba(0,0,0,.075)}.btn-
default:active,.btn-primary:active,.btn-success:active,
.btn-info:active,.btn-warning:active,.btn-danger:active,
.btn-default.active,.btn-primary.active,.btn-success.
active,.btn-info.active,.btn-warning.active,.btn-
danger.active{-webkit-box-shadow:inset 0 3px 5px
rgba(0,0,0,.125);box-shadow:inset 0 3px 5px
rgba(0,0,0,.125)}.btn:active,.btn.active{background-
image:none}.btn-default{text-shadow:0 1px 0
#fff;background-image:-webkit-gradient(linear,left 0,left
100%,from(#fff),to(#e0e0e0));background-image:-webkit-
linear-gradient(top,#fff 0,#e0e0e0 100%);background-image:-
moz-linear-gradient(top,#fff 0,#e0e0e0 100%);background-
image:linear-gradient(to bottom,#fff 0,#e0e0e0
100%);background-repeat:repeat-x;border-color:#dbdbdb;
border-color:#ccc;filter:progid: DXImageTransform.
Microsoft.gradient(startColorstr='#ffffffff',endColorstr='#
ffe0e0e0',GradientType=0);filter:progid:DXImageTransform.Mi
crosoft.gradient(enabled=false)}.btn-default:hover,.btn-
default:focus{background-color:#e0e0e0;background-
position:0 -15px}.btn-default:active,.btn-default.active
{background-color:#e0e0e0;border-color:#dbdbdb}.btn-
primary{background-image:-webkit-gradient(linear,left
0,left 100%,from(#428bca),to(#2d6ca2));background-image:-
webkit-linear-gradient(top,#428bca 0,#2d6ca2 100%);
```

JavaScript (bundled & minified)

```
(function(n,t){"use strict";function yt(n){var t=n.length,
r=i.type(n);return i.isWindow(n)?!1:1=== n.nodeType&&t?!0:
"array"===r||"function"!==r&&(0===t||"number"==typeof
t&&t>0&&t-1 in n)}function rf(n){var t=ii[n]={};return
i.each(n.match(o)||[],function(n,i){t[i]=!0}),t}function
pi(n,r,u,f){if(i.acceptData(n)){var s,h,c=i.expando,
a="string"==typeof r,l=n.nodeType,o=l?i.cache:n,
e=l?n[c]:n[c]&&c;if(e&&o[e]&&(f||o[e].data)||!a||u!==t)retu
rn e||(l?n[c]=e=w.pop()||i.guid++:e=c),o[e]||(o[e]=
{},l||(o[e].toJSON=i.noop)),("object"==typeof
r||"function"==typeof r)&&(f?o[e]=i.extend(o[e],r):
o[e].data=i.extend(o[e].data,r)),s=o[e],f||(s.data||(s.data
={}),s=s.data),u!==t&&(s[i.camelCase(r)]=u),a?(h=s[r],null=
=h&&(h=s[i.camelCase(r)])):h=s,h}}function
wi(n,t,r){if(i.acceptData(n)){var e,o,h,s=n.nodeType,
u=s?i.cache:n,f=s?n[i.expando]:i.expando;if(u[f]){if(t&&(e=
r?u[f]:u[f].data)){for(i.isArray(t)?t=t.concat(i.map(t,i.ca
melCase)):(t in e)?t=[t]:(t=i.camelCase(t),t=(t in
e)?[t]:t.split(" ")),o=0,h=t.length;h>o;o++)delete
e[t[o]];if(!(r?pt:i.isEmptyObject)(e))return}(r||(delete
u[f].data,pt(u[f])))&&(s?i.cleanData([n],!0):i.support.dele
teExpando||u!=u.window?delete u[f]:u[f]=null)}}} function
bi(n,r,u){if(u===t&&1===n.nodeType){var f="data-
"+r.replace(nu,"-$1").toLowerCase();if(u=n.
getAttribute(f),"string"==typeof u){try{u="true"===u?!0
:"false"===u?!1:"null"===u?null:+u+""===u?+u:gr.test(u)?i.p
arseJSON(u):u}catch(e){}i.data(n,r,u)}else u=t}return
u}function pt(n){var t;for(t in n)if(("data"!==t|| !i.
isEmptyObject(n[t]))&&"toJSON"!==t)return!1;return!0}
```

… and of course,
JavaScript should be loaded after HTML/CSS content

```
50
51              @RenderSection("Footer", false)
52          </div>
53
54
55          @Scripts.Render("~/Scripts/jquery-and-validation")
56          @RenderSection("Scripts", false)
57
58      </body>
59      </html>
60
```

Exercises (continued)

… mandatory assignment 2