

[Home](#)[Temperatures](#)[Weights](#)[About](#)

Temperature converter and Planetary Weights

Created by Dr. Paravastu

This lab gives you practice about:

- Chapter 7 Views
- ViewModel and BindingModel
- Submitting results to a different view (Weights and WeightsResults Views)
- Submitting results to the same View/page: Temperature View
- _ViewStart, _Layout, Shared Folder, @RenderBody() method
- Controls: Radio, Textboxes, and Select DropDownList in Views and their implementation in Models
- Validation with Span and Validation-for Tag Helpers

Should practice for the midterm!!!!

[Home](#)[Temperatures](#)[Weights](#)[About](#)

About Temperature Converter

— Dr. Paravastu

This lab has several components. The structure for setting up this project is given below:

This project has:

- One controller: HomeController.cs
- Two models: TemperatureModel.cs and WeightsModel.cs
- Five Views: (1) Index.cshtml, (2) Temperature.cshtml, (3) Weights.cshtml, (4) WeightsResults.cshtml, and (5) About.cshtml
- one ViewImports file, one ViewStart page for layout, and CSS.

In addition, this lab has several important other aspects:

Views Folder:

- The css for the pages are provided in the CSS folder.
- The project uses a shared folder, a uniform layout for navigation with navigation menus for all the views listed above. The shared folder has the _Layout.cshtml that we care about.
- The _ViewImports.cshtml file: This is at the root of the Views folder. This file has the directives for validation and the @using statements that we are familiar with already.
- The _ViewStart.cshtml file: This is at the root of the Views folder. This is a special view page containing the statement declaration to include the Layout page. This page enables us to use a common and uniform layout through out the pages for navigation (and for so many other things of course) instead of declaring the Layout page in every view page.
- The _Layout.cshtml file: The Head section, body section and everything inside head section such as meta tags, link tag for css etc. are provided inside the _Layout.cshtml file inside the shared folder.

A Word on Displaying the Results

The Index page and About page are static pages. There is no much action there, and there are no forms or results being displayed.

On the Temperature and Weights pages, each page has a form that submits and displays the results as detailed below. Both pages have accompanying ViewModels: TemperatureModel and WeightsModel as you can see from the @Model declarations at the top of each of these views.

The Temperature Page

The temperature page uses the same page to display the results. This page uses radio buttons. How many ever radio buttons you may have, there will only be one underlying Property in the model. Important thing to observe, The form submits the form data to the same page: (asp-action = "Temperature"). Observe that when the form posts back, the form values are retained (sticky form! -- ViewModel automatically handles this.) When you are ready to display the results, check for the postback using the ViewBag's dynamic property that is being returned from the HomeController's [HttpPost]Temperature() method.

Weights Page

The Weights page displays the results on a different view (WeightsResults.cshtml). The page has a select dropdown, and observe one underlying property for storing the value posted by the form, and the underlying variable of type LIST<SelectListItem> that supplies the items (asp-items) to the select dropdown. Important Observation: please note that in the weights model you are using a dictionary object that simplifies the job of comparing several planets.

```
1  @*
2      Created by Dr. Paravastu
3      FileName: _Layout.cshtml
4      Observe that this is where the html, head and body sections are being rendered, ↵
5          as well as navigation.
6      Starts with the doctype declaration
7  *@
8 <!DOCTYPE html>
9 <html lang="en">
10    <head>
11        <meta charset="utf-8" />
12        <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
13        <title>Paravastu TemperatureConverter</title>
14        <link rel="stylesheet" href("~/css/site.css" asp-append-version="true" />
15
16    </head>
17    <body>
18        <header>
19            <nav>
20                <a class="nav" asp-controller="Home" asp-action="Index">Home</a>
21                <a class="nav" asp-controller="Home" asp-
22                    action="Temperature">Temperatures</a>
23                <a class="nav" asp-controller="Home" asp-action="Weights">Weights</
24                    a>
25                <a class="nav" asp-controller="Home" asp-action="About">About</a>
26            </nav>
27        </header>
28        <div class = "bodyDiv">
29            @RenderBody()
30        </div>
31
32    </body>
33 </html>
```

```
1 html {
2   font-size: 14px;
3 }
4
5 @media (min-width: 768px) {
6   html {
7     font-size: 16px;
8   }
9 }
10
11 html {
12   position: relative;
13   min-height: 100%;
14 }
15
16 body {
17   margin: 20px;
18 }
19
20 nav {
21   margin-left: 15px;
22   font-weight: bold;
23   color: white;
24   border: 5px solid RoyalBlue;
25   background-color: RoyalBlue;
26 }
27
28
29
30 nav a {
31   display: inline-block;
32   border-top: 5px solid RoyalBlue;
33   background-color: white;
34   color: black;
35   width: 100px;
36   margin: 0;
37   padding: 5px;
38   list-style: none;
39 }
40
41 nav a:hover {
42   background-color: skyblue;
43 }
44
45 a {
46   text-decoration: none;
47 }
48 .bodyDiv{margin: 15px;}
49 .lspace {margin-left: 15px;}
```

```
1  @*
2      Created by Dr. Paravastu
3      FileName: Index.cshtml
4  *@
5  <h1>Temperature converter and Planetary Weights</h1>
6  <h4>Created by Dr. Paravastu</h4>
7  <p>
8      <ul>This lab gives you practice about:
9          <li>Chapter 7 Views</li>
10         <li>ViewModel and BindingModel</li>
11         <li>Submitting results to a different view (Weights and WeightsResults
12             Views)</li>
13         <li>Submitting results to the same View/page: Temperature View</li>
14         <li>_ViewStart, _Layout, Shared Folder, &commat;RenderBody() method</li>
15         <li>Controls: Radio, Textboxes, and Select DropDown in Views and their
16             implementation in Models</li>
17         <li>Validation with Span and Validation-for Tag Helpers</li>
18     </ul>
19     <h5>Should practice for the midterm!!!!</h5>
```

```
1  @*
2      Created by Dr. Paravastu
3      fileName: About.cshtml
4  *@
5  <h1>About Temperature Converter</h1>
6
7  <h6>&mdash; Dr. Paravastu</h6>
8
9  <p>This lab has several components. The structure for setting up this project is
10     given below:</p>
11 <ul>This project has:
12     <li>One controller: HomeController.cs</li>
13     <li>Two models: TemperatureModel.cs and WeightsModel.cs</li>
14     <li>Five Views: (1) Index.cshtml, (2) Temperature.cshtml, (3) Weights.cshtml,
15         (4) WeightsResults.cshtml, and (5) About.cshtml</li>
16     <li>one ViewImports file, one ViewStart page for layout, and CSS.</li>
17 </ul>
18 <p>In addition, this labs has several important other aspects:
19     <ul>Views Folder:
20         <li>The css for the pages are provided in the CSS folder.</li>
21         <li>The project uses a shared folder, a uniform layout for navigation
22             with navigation menus for all the views listed above. The shared
23             folder has the _Layout.cshtml that we care about. </li>
24         <li>The _ViewImports.chstml file: This is at the root of the Views
25             folder. This file has the directives for validation and the
26             &commat;using statements that we are familiar with already.</li>
27
28         <li>The _ViewStart.cshtml file: This is at the root of the Views folder.
29             This is a spcial view page containing the statment dclaration to
30             include the Layout page. This page enables us to use a common and
31             uniform layout through out the pages for navigation (and for somany
32             other things of course) instead of declaring the Layout page in every
33             view page</li>
34
35         <li>The _Layout.cshtml file: The Head section, body section and
36             everything inside head section such as meta tags, link tag for css
37             etc. are provided inside the _Layout.cshtml file inside the shared
38             folder. </li>
39     </ul>
40
41 <h3>A Word on Displaying the Results</h3>
42 <p>The Index page and About page are static pages. There is no much action there,
43     and there are no forms or results being displayed.</p>
44 <p>On the Temperature and Weights pages, each page has a form that submits and
45     displays the results as detailed below. Both pages have accompanying ViewModels:
46     TemperatureModel and WeightsModel as you can see from the &commat;Model
47     declarations at the top of each of these views.</p>
48 <h5>The Temperature Page</h5>
49 <p>The temperature page uses the same page to display the results. This page uses
50     radio buttons. Howmany ever radio buttons you may have, there will only be one
51     underlying Property in the model. Important thing to observe, The form submits the
52     form data to the same page: (asp-action ="Temperature"). Observe that when the
53     form posts back, the form values are retained (sticky form! -- ViewModel
54     automatically handles this.) When you are ready to display the results, check for
55     the postback using the ViewBag's dynamic property that is being returned from the
56     HomeConotroller's [HttpPost]Temperature() method. </p>
```

```
30 <h5>Weights Page</h5>
31 <p>The Weights page displays the results on a different view
     (WeightsResults.cshtml). The page has a select dropdown, and observe one
     underlying property for storing the value posted by the form, and the underlying
     variable of type LIST<SelectListItem>; that supplies the items (asp-items) to
     the select dropdown. Important Observation: please note that in the weights model
     you are using a dictionary object that simplifies the job of comparing several
     planets.</p>
32 </p>
33 </p>
```

```
1  using Microsoft.AspNetCore.Mvc;
2  using System.Diagnostics;
3  using TemperatureConverter.Models;
4  /*
5   * Created by Dr. Paravastu
6   */
7  namespace TemperatureConverter.Controllers
8  {
9      public class HomeController : Controller
10     {
11         public IActionResult Index()
12         {
13             return View();
14         } //Index()
15
16         public IActionResult About()
17         {
18             return View();
19         } //About()
20         public IActionResult Temperature()
21         {
22             return View(new TemperatureModel()); //view with ViewModel
23         } //Temperature
24         public IActionResult Weights()
25         {
26             return View(new WeightsModel()); //view with ViewModel
27         } //Weights
28
29         [HttpPost]
30         public IActionResult WeightsResult(WeightsModel model) //binding Model
31         {
32             if (ModelState.IsValid) //check if the model is valid
33             {
34                 return View(model); //view with ViewModel
35             } //end if
36             return View("Weights", model);
37         } //WeightsResult()
38
39         [HttpPost]
40         public IActionResult Temperature(TemperatureModel model) //binding Model
41         {
42             if (ModelState.IsValid) //check if the model is valid
43             {
44                 ViewBag.RoundTrip = "RoundTrip";
45                 return View(model); //view with ViewModel
46             } //ModelState if
47             return View(model); //view with ViewModel
48
49         } //Temperature
50
51     } //Controller
52 } //namespace
```

```
1  using System.ComponentModel.DataAnnotations;
2
3  /*
4   * Created by Dr. Paravastu
5   * filename: TemperatureModel.cs
6   */
7
8  namespace TemperatureConverter.Models
9  {
10     public class TemperatureModel
11     {
12         /*
13          * Properties with Data Validation annotations.
14          * Data validation annotations will require using statement for
15          * dataAnnotations (above)
16          * There are two properties and one method which does the
17          * calculation.
18         */
19         //Property for textbox.
20         [Required(ErrorMessage ="Please enter Temperature")]
21         [Display(Name="Please enter Temperature as a decimal or
22             integer")]
23         public decimal Temperature { get; set; }
24
25         //property for radiobuttons
26         [Required(ErrorMessage = "Please Select Farenheit or Celcius")]
27         public String? TemperatureUnit { get; set; }
28
29         public string ConvertTemp()
30         {
31             decimal conversionResult = 0;
32             string result = "";
33             switch (TemperatureUnit)
34             {
35                 case "F2C":
36                     //°C = 5/9 x (°F-32)
37                     conversionResult = (Temperature - 32) * 5 / 9;
38                     result = $"Temperature:{N4} degrees F =
39                     {conversionResult:N4} degrees C";
40                     break;
41
42                 case "C2F":
43                     //Formula: °F = °C × (9/5) + 32
44                     conversionResult = (Temperature * 9 / 5) + 32;
45                     result = $"Temperature:{N4}&deg; C =
46                     {conversionResult:N4}&deg; F";
47                     break;
48
49             } //switch
50
51             return result;
52         }
53     }
54 }
```

...rter\TemperatureConverter\Models\TemperatureModel.cs

```
48         } //ConvertTemp()
49
50     } //TemperatureModel
51 } //namespace
52
```

```
1 @model TemperatureModel
2 /*
3     Created by Dr. Paravastu
4     FileName: Temperature.cshtml
5 */
6 <h1>Paravastu Temperature Converter</h1>
7 <form asp-action="Temperature" asp-controller="Home" method="post">
8
9     <p>
10         <label asp-for = "TemperatureUnit" > Convert Farenheit to Celcius</label>
11         <input asp-for = "TemperatureUnit" value="F2C" type = "radio"/>
12         <label asp-for = "TemperatureUnit" class="lspan">Convert Celcius to
13             Farenheit </label>
14         <input asp-for = "TemperatureUnit" value="C2F" type = "radio" />
15         <span asp-validation-for="TemperatureUnit"></span>
16     </p>
17     <p>
18         <label asp-for="Temperature"></label>
19         <input asp-for="Temperature" />
20         <span asp-validation-for="Temperature"></span>
21     </p>
22     <p>
23         <button type="submit">Submit</button>
24     </p>
25 </form>
26
27     @{
28         /*
29         * razor block
30         * The ViewBag.RoundTrip is created in [HttpPost]Temperature() method.
31         * If this has been set to RoundTrip, then it is a roundtrip --display
32             the results.
33         */
34         if (ViewBag.RoundTrip == "RoundTrip")
35         {
36             <p>@Html.Raw(Model.ConvertTemp()) </p>
37         }
38
39
40
41
```

```
1  using Microsoft.AspNetCore.Mvc.Rendering;
2  using System.ComponentModel.DataAnnotations;
3  /*
4   * Created by Dr. Paravastu
5   * FileName: WeightsModel.cs
6   */
7  namespace TemperatureConverter.Models
8  {
9      public class WeightsModel
10     {
11         //Property for textbox
12         [Display(Name = "Please Enter Weight in Pounds")]
13         [Range(30, 300, ErrorMessage = "You weight in Lbs. between 30 and 300")]
14         public int Weight { get; set; }
15
16         //property for select dropdown
17         [Display(Name = "Please select Planet:")]
18         public string? Planet { get; set; }
19
20         //List for select Dropdown.
21         public List<SelectListItem> PlanetList { get; set; } = new
22             List<SelectListItem>
23         {
24             new SelectListItem{Value="Sun", Text = "Sun"},
25             new SelectListItem{Value="Mercury", Text = "Mercury"},
26             new SelectListItem{Value="Venus", Text = "Venus"},
27             new SelectListItem{Value="Earth", Text = "Earth"},
28             new SelectListItem{Value="Moon", Text = "Moon"},
29             new SelectListItem{Value="Mars", Text = "Mars"},
30             new SelectListItem{Value="Jupiter", Text = "Jupiter"},
31             new SelectListItem{Value="Saturn", Text = "Saturn"},
32             new SelectListItem{Value="Uranus", Text = "Uranus"},
33             new SelectListItem{Value="Neptune", Text = "Neptune"},
34             new SelectListItem{Value="Pluto", Text = "Pluto"}
35         };
36
37         //dictionary object for weights:
38         private Dictionary<String, double> planetWeights = new Dictionary<String,
39             double>();
40
41         public string GetWeight()
42         {
43             //adding the dictionary items for planetWeights.
44             planetWeights.Add("Sun", 27.01);
45             planetWeights.Add("Mercury", 0.38);
46             planetWeights.Add("Venus", 0.91);
47             planetWeights.Add("Earth", 1);
48             planetWeights.Add("Moon", 0.166);
49             planetWeights.Add("Mars", 0.38);
50             planetWeights.Add("Jupiter", 2.34);
51             planetWeights.Add("Saturn", 1.06);
52             planetWeights.Add("Uranus", 0.92);
53             planetWeights.Add("Neptune", 1.19);
54             planetWeights.Add("Pluto", 0.06);
```

```
54
55     //calculation: the incoming selection in Planet Property serves as the ↵
      key for retrieving the value from the planetWeights dictionary object.
56     return $"Your Weight on of {Weight} lb. on Earth will be equivalent of ↵
      {planetWeights[Planet] * Weight} lb. on {Planet}.";
57
58 } //GetWeight()
59
60 } //weights model
61 } //namespace
62
```

```
1 @model WeightsModel
2 @{
3     Created by Dr. Paravastu
4     FileName: Weights.cshtml
5 *@
6 <h1>Paravastu Weights</h1>
7 <h4>What is my weight on another planet?</h4>
8 <form asp-action="WeightsResult" asp-controller = "Home" method = "post">
9     <p>
10         <label asp-for="Weight"></label>
11         <input asp-for="Weight" />
12         <span asp-validation-for=Weight></span>
13     </p>
14     <p>
15         <label asp-for="Planet"></label>
16         <select asp-for="Planet" asp-items = "@Model.PlanetList"></select>
17         <span asp-validation-for = "Planet"></span>
18     </p>
19     <p>
20         <button type="submit">Submit</button>
21     </p>
22
23 </form>
```

```
1  @*
2      Created by Dr. Paravastu
3      FileName: WeightsResult.cshtml
4  *@
5  @model WeightsModel
6  <h1>Paravastu Planet Weights Results</h1>
7  <p>@Model.GetWeight()</p>
8
9
10
```