

Entity Framework / Validation Model



Razor View

```
@using MyApp.Models
@model Employee

@{
    Layout = "~/Views/_Base.cshtml";
    ViewBag.Title = "Employee Information";
}

<h2>Employee Information</h2>
<p>These are the details of @Model.FullName:</p>
@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.ValidationSummary()
    <div>
        First Name: @Html.TextBoxFor(x => x.FirstName)
    </div>
    <div>
        Family Name: @Html.TextBoxFor(x => x.FamilyName)
    </div>
    <div>
        Birth Month: @Html.TextBoxFor(x => x.BirthMonth)
    </div>
    <div>
        IP Address: @Html.TextBoxFor(x => x.IpAddress)
    </div>
    <div>
        Full Time:
        @Html.DropDownListFor(
            x => x.IsFullTime,
            new [] {
                new SelectListItem {
                    Text = "Full Time",
                    Value = "true",
                    Selected = Model.IsFullTime
                },
                new SelectListItem {
                    Text = "Part Time",
                    Value = "false",
                    Selected = !Model.IsFullTime
                }
            }
        )
    </div>
    <input type="submit" value="Save Changes" />
}
```

Import namespace into Razor view

Declare a typed View

Use a master layout / template

Set a property of the ViewBag dynamic object

Write out a value

Create a HTML <form> tag

Generate a token to prevent Cross-Site-Forgery-Requests (CSFRs)

Show any validation errors

Create Text-boxes for Properties of the Model

Create a Drop-down list for IsFullTime property of the model

```
}
```

Razor comments

```
@* Don't show the messages section if there are no messages *@
```

```
@if (Model.Messages != null && Model.Messages.Count() > 0)
```

```
{
```

```
    <p>Messages for @Model.FullName:</p>
```

```
    <ul>
```

```
    @foreach (var message in Model.Messages)
```

```
    {
```

```
        <li>@message.MessageId</li>
```

```
    }
```

```
    </ul>
```

```
}
```

If Statement

Loop

Manually create a URL to use in a link

```
<a href="@Url.Action("Index")">List All Employees</a>
```

```
@Html.ActionLink("List All Employees", "Index")
```

```
@Html.ActionLink("Go to Home", "Index", "Home")
```

```
@Html.ActionLink("View Employee 1", "Details", new { id = 1 }, null)
```

```
@Html.Raw("&nbsp;")
```

Create a link referring to the Index action of the same controller

Create a link referring to the Index action of the HomeController

Create a link referring to the Details action with route/model parameters

Output a string directly as raw HTML, without any escaping

```
@section Author {
```

```
    This page created by Ben.
```

```
}
```

Define a section (to use in a template)

Template

```
<html>
```

```
    <title>@ViewBag.Title</title>
```

```
    <body>
```

```
        <h1>My App</h1>
```

```
        <div>
```

```
            @RenderBody()
```

```
        </div>
```

```
        <div>
```

```
            @if (IsSectionDefined("Author"))
```

```
            {
```

```
                @: Author Information:
```

```
                @RenderSection("Author")
```

```
            }
```

```
        </div>
```

```
    </body>
```

```
</html>
```

Render the main content of the file that is using this template

Switch back to plain HTML text when in a section of C# code

Render a section that was defined using @section

Controller



