Prof. Krishnan

IT 2030 – Spring 2021

Robert Pratt

Assignment02

HomeController.cs

using Microsoft.AspNetCore.Mvc;

using FutureValue.Models;

namespace FutureValue.Controllers

{

public class HomeController : Controller

{

[HttpGet]

public IActionResult Index()

{

ViewBag.Name = "Mary";

ViewBag.FV = 99999.99;

return View();

}

[HttpPost]

public IActionResult Index(FutureValueModel model)

{

if(ModelState.IsValid)

{

ViewBag.FV = model.CalculateFutureValue();

}

else

{

ViewBag.FV = 0;

}

return View(model);

}

}

}

FutureValueModel.cs

using System.ComponentModel.DataAnnotations;

namespace FutureValue.Models

{

public class FutureValueModel

{

[Required(ErrorMessage = "Please enter a monthly investment.")]

[Range(1, 500, ErrorMessage =

"Monthly investment amount must be between 1 and 500.")]

public decimal? MonthlyInvestment { get; set; }

[Required(ErrorMessage = "Please enter a yearly interest rate.")]

[Range(0.1, 10.0, ErrorMessage =

"Yearly interest rate must be between 0.1 and 10.0.")]

public decimal? YearlyInterestRate { get; set; }

[Required(ErrorMessage = "Please enter a number of years.")]

[Range(1, 50, ErrorMessage =

"Number of years must be between 1 and 50.")]

public int? Years { get; set; }

public decimal? CalculateFutureValue()

{

int? months = Years \* 12;

decimal? monthlyInterestRate = YearlyInterestRate / 12 / 100;

decimal? futureValue = 0;

for(int i = 0; i < months; i++)

{

futureValue = (futureValue + MonthlyInvestment) \*

(1 + monthlyInterestRate);

}

return futureValue;

}

}

}

Index.html

@model FutureValueModel

@{

ViewBag.Title = "Future Value Calculator";

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

<title>Home Page</title>

</head>

<body>

<h1>Future Value Calculator</h1>

<form **asp-action**="Index" method="post">

<div **asp-validation-summary**="All"></div>

<div>

<label **asp-for**="MonthlyInvestment">Monthly Investment:</label>

<input **asp-for**="MonthlyInvestment"/>

</div>

<div>

<label **asp-for**="YearlyInterestRate">Yearly Interest Rate:</label>

<input **asp-for**="YearlyInterestRate" />

</div>

<div>

<label **asp-for**="Years">Number of Years:</label>

<input **asp-for**="Years" />

</div>

<div>

<label>Future Value:</label>

<input value="@ViewBag.FV.ToString("C2")" readonly />

</div>

<button type="submit">Calculate</button>

<a **asp-action**="Index">Clear</a>

</form>

</body>

</html>

\_Layout.cshtml

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="="width="device-width" />

<title>@ViewBag.Title</title>

<link rel="stylesheet" href="I~/css/custom.css" />

</head>

<body>

<div>

@RenderBody()

</div>

</body>

</html>

\_ViewImports.cshtml

@using FutureValue.Models

@addTagHelper \*, Microsoft.AspNetCore.Mvc.TagHelpers

\_ViewStart.cshtml

@{

Layout = "\_Layout";

}

Startup.cs

using Microsoft.AspNetCore.Builder;

using Microsoft.AspNetCore.Hosting;

using Microsoft.Extensions.DependencyInjection;

using Microsoft.Extensions.Hosting;

namespace FutureValue

{

public class Startup

{

//Use this method to add services to the container

public void ConfigureServices(IServiceCollection services)

{

services.AddControllersWithViews();

}

//Use this method to configure the HTP request pipeline

public void Configure(IApplicationBuilder app,

IWebHostEnvironment env)

{

if(env.IsDevelopment())

{

app.UseDeveloperExceptionPage();

}

else

{

app.UseExceptionHandler("/Home/Error");

app.UseHsts();

}

app.UseHttpsRedirection();

app.UseStaticFiles();

app.UseRouting();

app.UseEndpoints(endpoints =>

{

endpoints.MapControllerRoute(

name: "default",

pattern: "{controller=Home}/{action=Index}/{id?}");

});

}

}

}

custom.css

body {

padding: 1em;

font-family: Arial, Arial, Helvetica, sans-serif;

}

h1 {

margin-top: 0;

color: navy;

}

label {

display: inline-block;

width: 10em;

padding-right: 1em;

}

div {

margin-bottom: .5em;

}

Graphical user interface, text

Description automatically generated