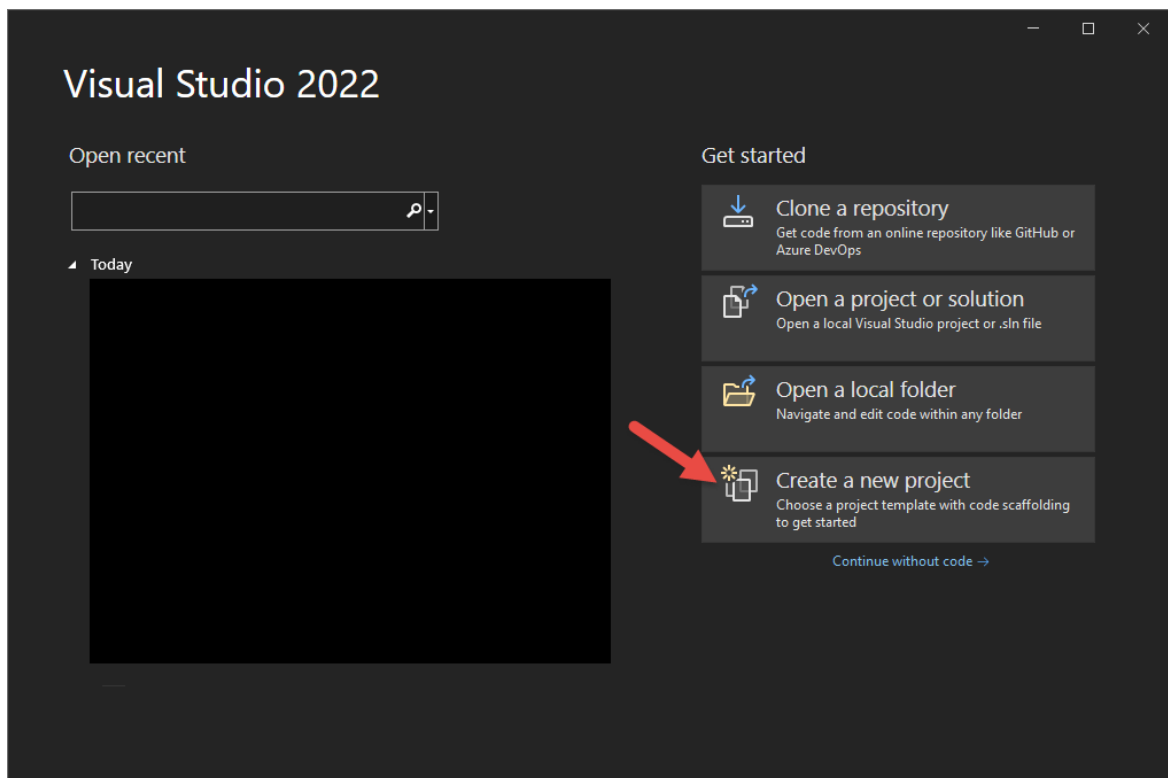


MVC Basics Lab Using VS 2022

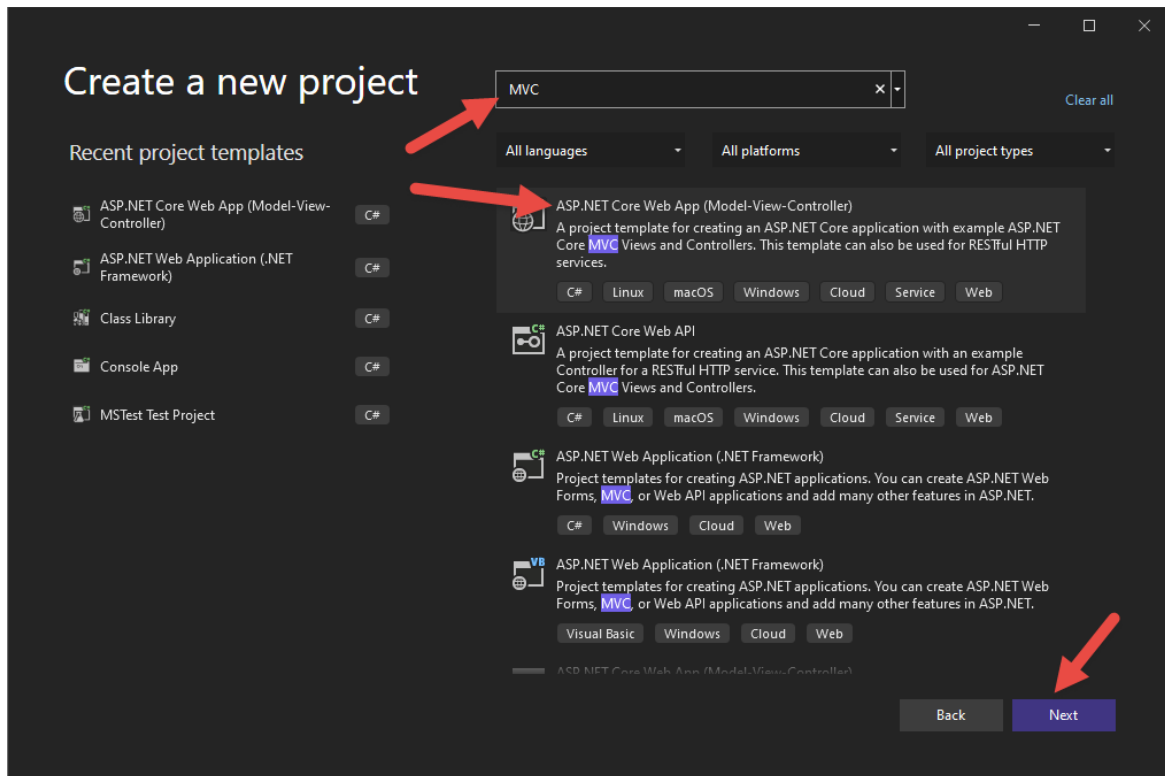
Lab 1: Create MVC Project

Open Visual Studio 2022 and select **Create a New Project**



In the search box at the top right, type in **MVC**

In the list below the box locate **ASP.NET Core Web App (Model-View-Controller)** and click on it.



Click the **Next** button

Fill in the **Project Name** with **AdvWorks**

Fill in the **Location** with your project folder

Configure your new project

ASP.NET Core Web App (Model-View-Controller) C# Linux macOS Windows Cloud Service Web

Project name
AdvWorks

Location
D:\Samples

Solution name ⓘ
AdvWorks

☒ Place solution and project in the same directory

Back Next

Click the **Next** button

Change the Target Framework to **.NET 6.0 (Long-term support)**

Set the Authentication Type to **None**

Uncheck **Configure for HTTPS**

Click the **Create** button

Additional information

ASP.NET Core Web App (Model-View-Controller) C# Linux macOS Windows Cloud Service Web

Framework ⓘ
[.NET 6.0 (Long-term support)]

Authentication type ⓘ
[None]

☐ Configure for HTTPS ⓘ
☐ Enable Docker ⓘ

Docker OS ⓘ
[Linux]

Back Create

Modify Program Class

Open the **Program.cs** file and modify it to look like the following:

```
//*****
// Configure Services
//*****
var builder = WebApplication.CreateBuilder(args);

// Add MVC services to the Web Application
builder.Services.AddControllersWithViews();

//*****
// Configure Application
// For a list of what services need to be started, in what order,
// see the following link
// https://docs.microsoft.com/en-
// us/aspnet/core/fundamentals/middleware/?view=aspnetcore-
// 6.0#middleware-order
//*****
var app = builder.Build();

// Configure the HTTP request pipeline.
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
}
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");

app.Run();
```

Try it Out

Select **Debug** | **Start Debugging** to run the application

Lab 2: Modify _Layout

Open the `\Views\Shared_Layout.cshtml` file

Remove the ' - AdvWorks' from the `<title>` element on line 6

```
<title>@ViewData["Title"]</title>
```

Go to Line 15 and add change the hyperlink to:

```
<a class="navbar-brand"
  asp-area=""
  asp-controller="Home"
  asp-action="Index">
  Adventure Works
</a>
```

Locate the "Home" and "Privacy" menus in the `_Layout.cshtml` file

```
<ul class="navbar-nav flex-grow-1">
  <li class="nav-item">
    <a class="nav-link text-dark" asp-area="" asp-controller="Home"
    asp-action="Index">Home</a>
  </li>
  <li class="nav-item">
    <a class="nav-link text-dark" asp-area="" asp-controller="Home"
    asp-action="Privacy">Privacy</a>
  </li>
</ul>
```

Remove the "Home" ` ... ` element but don't delete the `<ul class="navbar-nav flex-grow-1">`

Remove the "Privacy" menu also

Add the following menu items after the `` element where the Home and Privacy menus were.

```
<li class="nav-item">
  <a class="nav-link text-dark" asp-area=""
    asp-controller="ProductMaint" asp-action="ProductMaintenance">
    Products
  </a>
</li>
<li class="nav-item">
  <a class="nav-link text-dark" asp-area=""
    asp-controller="ProductMaint"
    asp-action="ColorMaintenance">Colors</a>
</li>
<li class="nav-item">
  <a class="nav-link text-dark" asp-area=""
    asp-controller="ProductMaint" asp-action="CategoryMaintenance">
    Categories
  </a>
</li>
```

Locate the `<footer>` element

Within the `<div class="container">` element, remove the text shown below. Leave the `<div>` and `</div>` tags.

```
&copy; 2022 - AdvWorks - <a asp-area="" asp-controller="Home"
    asp-action="Privacy">Privacy</a>
```

Delete the **\Views\Home\Privacy.cshtml** page

Open the **HomeController.cs** file

Remove the **Privacy()** method

Try it Out

Select **Debug | Start Debugging** to run the application

Lab 3: Change the Home Page

Open **\Views\Home\Index.cshtml**

Make the HTML look like the following:

```
@{
    ViewData["Title"] = "Home Page";
}
<div class="text-center">
    <h1 class="display-4">Welcome to Adventure Works</h1>
    <p>This site helps you work with product data.</p>
</div>

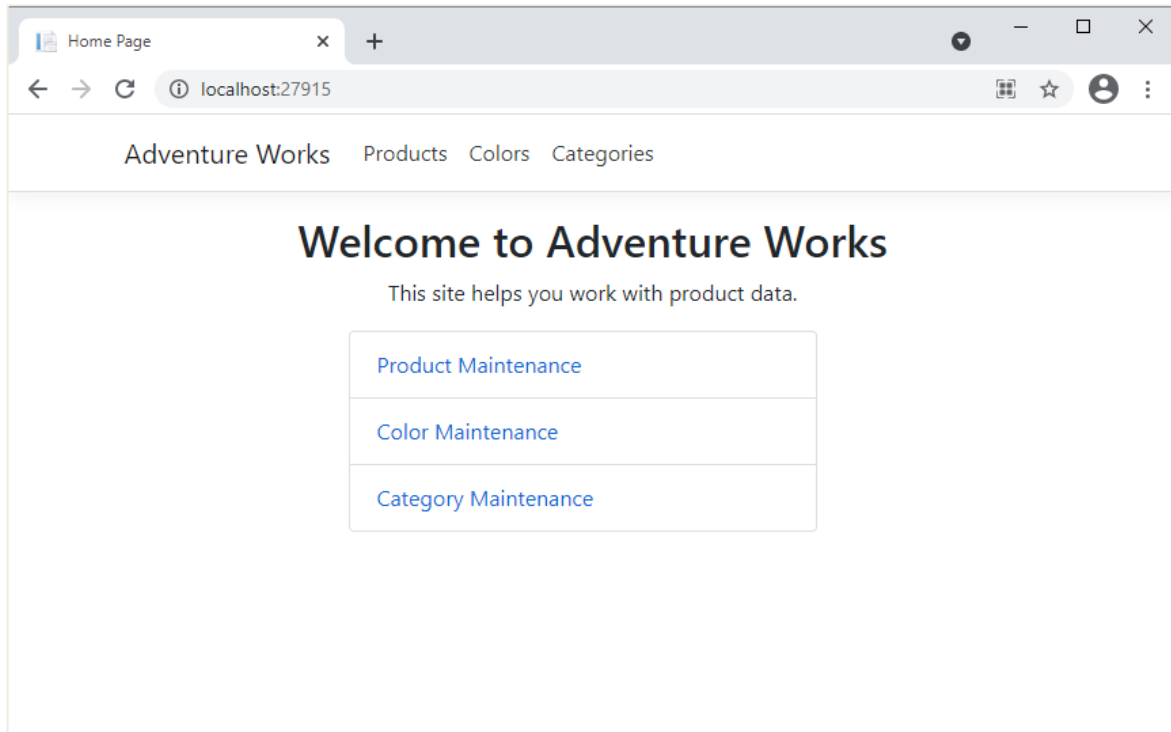
<div class="row justify-content-center">
    <div class="list-group col-6">
        <a asp-action="ProductMaintenance"
            asp-controller="ProductMaint" class="list-group-item">
            Product Maintenance
        </a>
        <a asp-action="ColorMaintenance"
            asp-controller="ProductMaint" class="list-group-item">
            Color Maintenance
        </a>
        <a asp-action="CategoryMaintenance"
            asp-controller="ProductMaint" class="list-group-item">
            Category Maintenance
        </a>
    </div>
</div>
```

Save the changes

Try it Out

If you left the application running, tab over to your browser and in just a couple of seconds you should see your changes appear. If you did not leave it running, select **Debug | Start Debugging** to run the application.

Your page should now look similar to the following:



Lab 4: Add Product Pages

Right mouse-click on the \Views folder and add a new folder named **ProductMaint**

Right mouse-click on the \ProductMaint folder and select **Add | View ... | Razor View - Empty**.

Set the name to **ProductMaintenance.cshtml**.

Replace all the code in the page with the following:

```
<h2>Product Maintenance</h2>
```

Right mouse-click on the \ProductMaint folder and select **Add | View ... | Razor View - Empty**.

Set the name to **ColorMaintenance.cshtml**.

Add the following code into this page:

```
<h2>Color Maintenance</h2>
```

Right mouse-click on the \ProductMaint folder and select **Add | View ... | Razor View - Empty**.

Set the name to **CategoryMaintenance.cshtml**.

Add the following code into this page:

```
<h2>Category Maintenance</h2>
```

Right mouse-click on the \Controllers folder and select **Add | Controller... | MVC Controller - Empty**.

Set the name to **ProductMaintController.cs**.

Replace the code in the file with the following:

```
using Microsoft.AspNetCore.Mvc;

namespace AdvWorks.Controllers
{
    public class ProductMaintController : Controller
    {
        [HttpGet]
        public IActionResult ProductMaintenance()
        {
            return View();
        }

        [HttpGet]
        public IActionResult ColorMaintenance()
        {
            return View();
        }

        [HttpGet]
        public IActionResult CategoryMaintenance()
        {
            return View();
        }
    }
}
```

Try it Out

Select **Debug | Start Debugging** to run the application

Click on each of the menus and links to make sure you can get to each page

Lab 5: Partial Pages

Right mouse-click on the \ProductMaint folder and select **Add | View ... | Razor View - Empty**.

Set the name to **_ProductList.cshtml**.

Replace all the code in this file with the following HTML:

```
<table class="table table-bordered table-hover">
  <thead>
    <tr>
      <th>Product Name</th>
      <th>Product Number</th>
      <th class="text-right">Cost</th>
      <th class="text-right">Price</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>HL Road Frame - Red, 58</td>
      <td>FR-R92R-58</td>
      <td class="text-right">$1,509</td>
      <td class="text-right">$1,431</td>
    </tr>
    <tr>
      <td>Sport-100 Helmet, Red</td>
      <td>HL-U509-R</td>
      <td class="text-right">$13</td>
      <td class="text-right">$34</td>
    </tr>
  </tbody>
</table>
```

Open the **ProductMaintenance.cshtml** file and add a `<partial>` tag below the header:

```
<h2>Product Maintenance</h2>

<partial name="_ProductList" />
```

Try it Out

Select **Debug | Start Debugging** to run the application

Click on the Product Maintenance link to see a page that looks like the following:

Adventure Works Products Colors Categories			
Product Maintenance			
Product Name	Product Number	Cost	Price
HL Road Frame - Red, 58	FR-R92R-58	\$1,509	\$,1431
Sport-100 Helmet, Red	HL-U509-R	\$13	\$34

Demo 6: Page Head Styles

Open the `\Views\Shared_Layout.cshtml` file and in the `<head>` area, add the following:

```
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-
scale=1.0" />
  <title>@ViewData["Title"]</title>
  <link rel="stylesheet"
href="~/lib/bootstrap/dist/css/bootstrap.min.css" />
  <link rel="stylesheet" href="~/css/site.css" />

  @RenderSection("HeadStyles", required: false)
</head>
```

Open the `ProductMaintenance.cshtml` file and add a `<div class="list-area">` around the `<partial>` tag

```
<div class="row list-area">
  <partial name="_ProductList" />
</div>
```

Add the following before the `<h2>` tag

```
@section HeadStyles {  
  <style>  
    .list-area {  
      border: solid black 0.25em;  
      padding: 1em;  
      margin-top: 1em;  
    }  
  </style>  
}
```

Try it Out

Run the application and click on the Products link

View the page source in your browser to ensure that the location of the <style> tag is in the <head> section

Now that you have seen the style in action, go ahead and remove the following rule:

```
border: solid black 0.25em;
```

Demo 7: Add JavaScript

Open the **ProductMaintenance.cshtml** file and at the bottom of the page, add the following:

```
@section Scripts  
{  
  <script>  
    function displayProduct(name) {  
      alert('You clicked on ' + name);  
    }  
  </script>  
}
```

Open the **_ProductList.cshtml** file

Change the first <td> in the first <tr> to call the displayProduct() function

```
<td>  
  <a href="#" onclick="displayProduct(this.text);">  
    HL Road Frame - Red, 58  
  </a>  
</td>
```

Change the first <td> in the second <tr> to call the displayProduct() function

```
<td>
  <a href="#" onclick="displayProduct(this.text);">
    Sport-100 Helmet, Red
  </a>
</td>
```

Try it Out

Run the application and click on the Products link

Click on each of the hyperlinks in the table to display the current product