

Logging Lab

Lab 1: Log Messages on Home Page

Open the **HomeController.cs** file and modify the **Index()** method with the lines in bold.

```
[HttpGet]
public IActionResult Index()
{
    // throw new ApplicationException("This is an error");

    _logger.LogInformation("Entering the home page");

    // Has the user been here before?
    ViewData["Returning"] = (Request.Cookies["Returning"] != null);
    // If the user has not been there, send a cookie out
    if (!Convert.ToBoolean(ViewData["Returning"])) {
        _logger.LogInformation("First time entering home page");
        CookieOptions options = new()
        {
            Expires = DateTime.Now.AddMinutes(2)
        };
        Response.Cookies.Append("Returning", "true", options);
    }
    else {
        _logger.LogInformation("Have been to the home page before");
        _session.LastColor = "Blue";
        _session.LastCost = 10;
        _session.LastPrice = 100;
    }

    return View();
}
```

Try it Out

Run the application

Click on the **View | Output** menu in Visual Studio

Run the application and see the messages you entered appear

Lab 2: Log Errors

Open the **ErrorController.cs** file and modify the **StatusCodeReExecute()** method with the line in bold.

```
public IActionResult StatusCodeReExecute(int code)
{
    // Set status code property
    StatusCode = code;

    // Get some path information
    var feature =
HttpContext.Features.Get<IStatusCodeReExecuteFeature>();
    if (feature != null) {
        StatusPath =
            feature.OriginalPathBase
            + feature.OriginalPath
            + feature.OriginalQueryString;
    }

    _logger.LogError($"Status Code='{vm.StatusCode}',
Path='{vm.StatusPath}'");

    // Build page name from status code number
    return View("StatusCode" + StatusCode.ToString(), this);
}
```

Try it Out

Run the application and enter a bad path to see the error in the output window

Lab 3: Log to a File using SeriLog

Right mouse-click on the **AdvWorks** project and select **Manage NuGet Packages...** from the menu

Click on the Browse tab and type in **Serilog.AspNetCore**

Click the Install button

If you are using VS Code, type the following into a terminal window

```
dotnet add package Serilog.AspNetCore
```

Next, add a SeriLog Sink that logs to a file

```
Serilog.Sinks.File
```

Open **Program.cs** and add a using statement:

```
using Serilog;
```

Add the following code immediately after the `builder.Services.AddControllersWithViews();` line of code:

```
// Configure logging using Serilog
builder.Host.UseSerilog((ctx, lc) =>
    lc.WriteTo.File("Logs/log-.txt",
        rollingInterval: RollingInterval.Day));
```

Right mouse-click on the **AdvWorks** Project folder and add a new folder named **\Logs**

Try it Out

Run the application

Cause a bad page

Stop the application and open the new log file in the **\Logs** folder.