Lab: Developing ASP.NET MVC Core Views And View Components

# Scenario

You have been asked to add the following views to the photo sharing application:

* Views for the Photo model objects. These views will enable users to upload a new photo to the gallery, view the details of existing photos and deleting photos.
* A Photo Gallery view component. This view component will display many photos in thumbnail sizes, with the title, owner, and created date properties. This view component will be used on the All Photos webpage to display all the photos in the application. In addition, this view component will also be used on the home page to display the three most recent photos.

In this phase you’re not concerned with the UI style, yet. You just want some working functionality. You will review the style at a later time, in a following iteration.

After adding these views to the photo sharing application, you will also test the working of the web application.

# Objectives

After completing this lab, you will be able to:

* Add Razor views to an MVC application.
* Write both HTML markup and C# code in a view by using Razor syntax.
* Create a view component and use it to display re-usable markup.

# Exercise 1: Add the Views for Photos

## Scenario

In this exercise, you will

* Add the Create View to upload new photos.
* Add a Display View to show the properties of a photo, such as title, description, and created date.
* Add a Delete View to show the properties of a photo and ask for confirmation.

The main tasks for this exercise are as follows:

1. Add the Create View
2. Add the Display View
3. Add the Delete View

**Results**: After completing this exercise, you will be able to create a web application with a Razor view to create new photos.

# Exercise 2: Creating and Using a View Component

## Scenario

In this exercise, you will:

* Add a Photo Gallery View Component.
* Add a Default View for the Photo Gallery View Component.
* Complete the Default View.
* Use the Photo Gallery View Component.

The main tasks for this exercise are as follows:

1. Add a Photo Gallery View Component.
2. Add a Default View for the Photo Gallery View Component.
3. Complete the Default View.
4. Use the Photo Gallery View Component from a new Index View of the Photo Controller (which should display all the photos) and from the Home View of the Home Controller (which should display only the latest three photos).

**Results**: After completing this exercise, you will be able to create a web application with a partial view to display multiple photos.

**Question**: How can you improve the accessibility of the HTML that your photo views render?

**Question**: In the lab, how did you ensure that the Create view for Photo model objects could upload photo files when the user clicked the Create button?