**Source Code**

**User.cs**

namespace BankWebApp.Models

{

public class User

{

public int Id { get; set; }

public string Username { get; set; }

public string Password { get; set; }

}

}

**AccountController.cs**

using BankWebApp.Models;

using Microsoft.AspNetCore.Mvc;

namespace BankWebApp.Controllers

{

public class AccountController : Controller

{

// Simulated user data (replace with your actual authentication logic)

private readonly User \_user = new User { Id = 1, Username = "user", Password = "password" };

[HttpGet]

public IActionResult Login()

{

return View();

}

[HttpPost]

public IActionResult Login(User model)

{

if (ModelState.IsValid && model.Username == \_user.Username && model.Password == \_user.Password)

{

// Successful login, redirect to dashboard

return RedirectToAction("Index", "Dashboard");

}

// Invalid login, show error message

ModelState.AddModelError("", "Invalid username or password");

return View(model);

}

}

}

**DashboardController.cs**

using Microsoft.AspNetCore.Authorization;

using Microsoft.AspNetCore.Mvc;

namespace BankWebApp.Controllers

{

public class DashboardController : Controller

{

[Authorize]

public IActionResult Index()

{

return View();

}

}

}

**Login.cshtml**

@model BankWebApp.Models.User

@{

ViewData["Title"] = "Login";

}

<h1>Login</h1>

<**form** **asp-controller**="Account" **asp-action**="Login" method="post">

<div class="form-group">

<**label** **asp-for**="Username">Username:</**label**>

<**input** **asp-for**="Username" class="form-control" />

<**span** **asp-validation-for**="Username" class="text-danger"></**span**>

</div>

<div class="form-group">

<**label** **asp-for**="Password">Password:</**label**>

<**input** **asp-for**="Password" **type**="password" class="form-control" />

<**span** **asp-validation-for**="Password" class="text-danger"></**span**>

</div>

<button type="submit" class="btn btn-primary">Login</button>

</**form**>

**Index.cshtml**

@{

ViewData["Title"] = "Index";

}

<h1 style="text-align:center">Welcome to Dashboard</h1>

**Program.cs**

using Microsoft.AspNetCore.Authentication.Cookies;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.

builder.Services.AddControllersWithViews();

builder.Services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme)

.AddCookie(options =>

{

options.LoginPath = "/Account/Login"; // Redirect to the login page for unauthenticated users

});

var app = builder.Build();

// Configure the HTTP request pipeline.

if (!app.Environment.IsDevelopment())

{

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

}

app.UseStaticFiles();

app.UseRouting();

app.UseAuthentication();

app.UseAuthorization();

app.UseEndpoints(endpoints =>

{

endpoints.MapControllerRoute(

name: "dashboard",

pattern: "dashboard",

defaults: new { controller = "Dashboard", action = "Index" });

endpoints.MapControllerRoute(

name: "default",

pattern: "{controller=Account}/{action=Login}/{id?}");

});

app.Run();