builder.Services.AddSession(options =>

{

options.IdleTimeout = TimeSpan.FromMinutes(30); // Session timeout

options.Cookie.HttpOnly = true; // Prevent JS access

options.Cookie.IsEssential = true; // Required for GDPR

});

app.UseSession();

--------------------------------

// Store session data (only string & int natively)

HttpContext.Session.SetString("UserName", "Ramesh");

HttpContext.Session.SetInt32("UserId", 101);

public IActionResult Dashboard()

{

// Retrieve data

string? userName = HttpContext.Session.GetString("UserName");

int? userId = HttpContext.Session.GetInt32("UserId");

ViewBag.UserName = userName;

ViewBag.UserId = userId;

return View();

}

===================================================

Complex object in session

using System.Text.Json;

public static class SessionExtensions

{

public static void SetObject<T>(this ISession session, string key, T value)

{

session.SetString(key, JsonSerializer.Serialize(value));

}

public static T? GetObject<T>(this ISession session, string key)

{

var value = session.GetString(key);

return value == null ? default : JsonSerializer.Deserialize<T>(value);

}

}

-----------------------------------------------------------------------------------

How to use

var user = new { Id = 1, Name = "Ramesh" };

HttpContext.Session.SetObject("CurrentUser", user);

var currentUser = HttpContext.Session.GetObject<dynamic>("CurrentUser");

=====================================

Accessing session in page

@inject IHttpContextAccessor HttpContextAccessor

<p>Hello, @HttpContextAccessor.HttpContext.Session.GetString("UserName")</p>