MVVM és komponensek

[Introduction to the Blazor WebAssembly .NET 6 in Blazor WebAssembly .NET 6 - Blazor School](https://blazorschool.com/tutorial/blazor-wasm/dotnet6/introduction-500443)

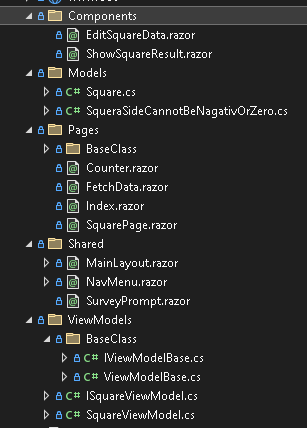
[Blazor - Getting Started | blazor Tutorial (riptutorial.com)](https://riptutorial.com/blazor/learn/100000/getting-started)

[MVVM Pattern in Blazor For State Management – A Complete Guide | Syncfusion Blogs](https://www.syncfusion.com/blogs/post/mvvm-pattern-in-blazor-for-state-management.aspx)

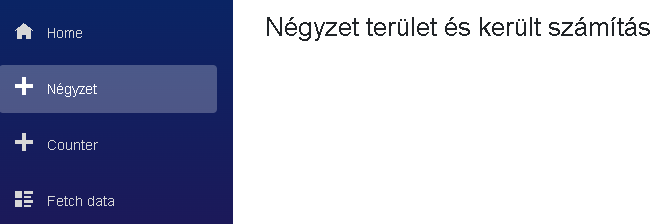
# Projekt létrehozása és kezdeti lépések

Készítsen projektet SquareProject néven.

A megfelelő mappákba helyezze el a megadott kódokat:



* Készítsük el a SuqarePage oldalt és kössük be a menübe.



* A SquarePage a BasePage osztálytól öröklődik.

## Interface

* [Interfészek | C# Tutorial.hu (csharptutorial.hu)](https://csharptutorial.hu/docs/hellovilag-hellocsharp/5-osztalyok-objektumok/interfeszek/)

public interface ISquareViewModel : IViewModelBase

{

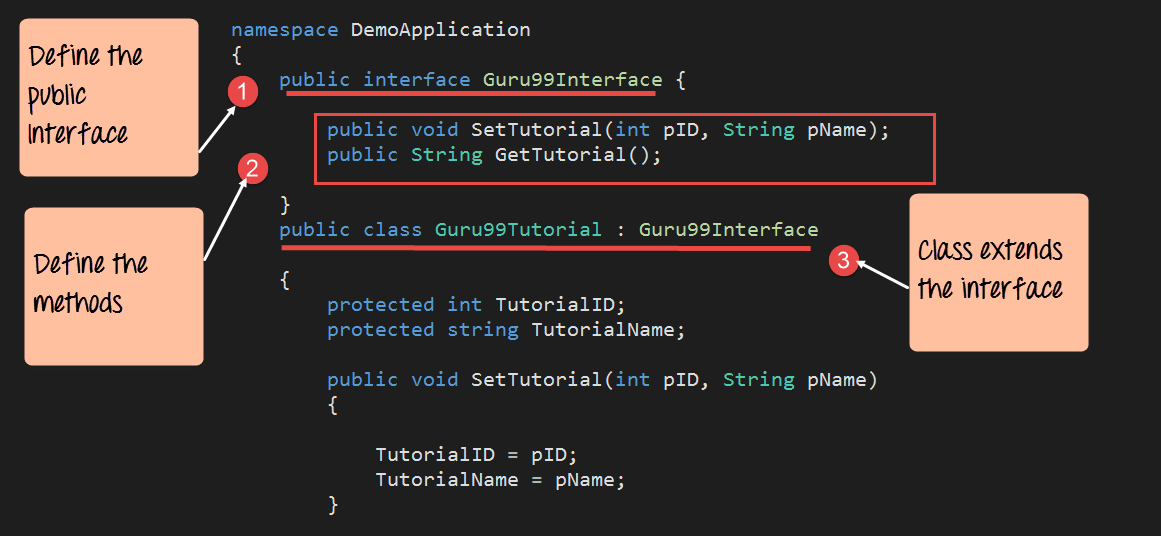
public string Side { get; set; }

public string Perimeter { get; }

public string Area { get; }

public void Compute();

}



A SquareViewModel osztály az ISquareViewModel interfactól is öröklődik.

public class SquareViewModel : ViewModelBase, ISquareViewModel

{

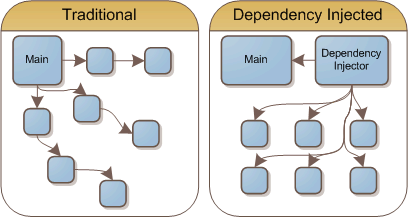
private Square \_square;

# Dependency Injection

[Függőséginjektálás - .NET | Microsoft Learn](https://learn.microsoft.com/hu-hu/dotnet/core/extensions/dependency-injection)

[Dependency Injection in C# (2023) (c-sharpcorner.com)](https://www.c-sharpcorner.com/UploadFile/85ed7a/dependency-injection-in-C-Sharp/)

[Dependency Injection (devopedia.org)](https://devopedia.org/dependency-injection)



ViewModel dependency injection:

Program.cs

builder.Services.AddSingleton<ISquareViewModel, SquareViewModel>();

SquarePage.razor:

@code {

[**Inject**]

public ISquareViewModel? ViewModel { get; set; }

}

Form és elágazás:

@if (ViewModel is not null)

{

<**EditForm** **Model**="@ViewModel">

<div class="col-4">

<div class="form-group">

<label for="Oldal">Név</label>

<**InputText** class="form-control" id="nev" @bind-Value="@ViewModel.Side" />

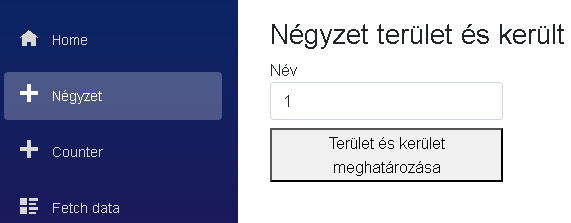
</div>

<button class="mt-2" type="submit" disabled="@ViewModel.IsBusy">Terület és kerület meghatározása</button>

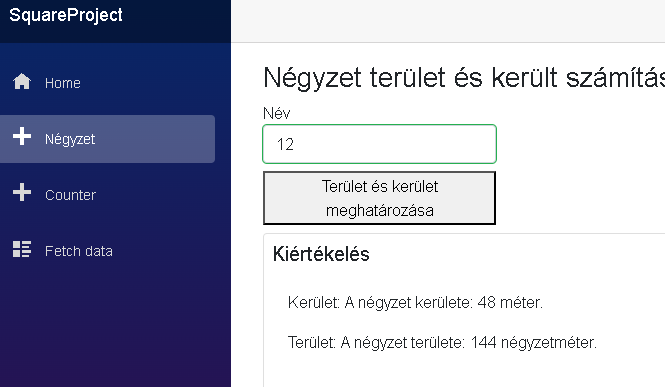
</div>

</**EditForm**>

}



Eredmény megjelenítés:



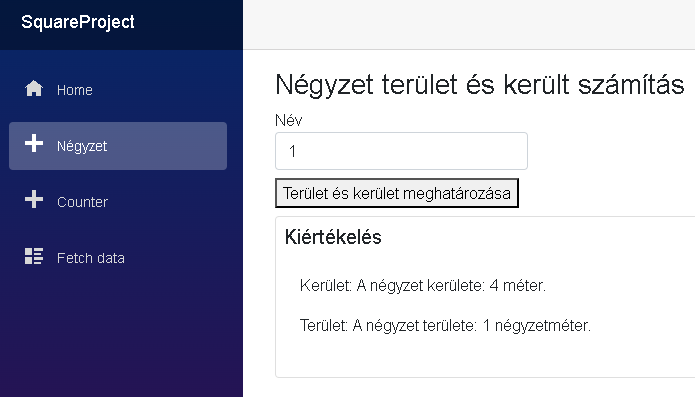
A lap lényegében két komponensből áll:

<**CascadingValue** **Value**="@ViewModel" **Name**="ViewModel">

<**EditSquareData** />

<**ShowSquareResult** />

</**CascadingValue**>



EditSquareData:

@using SquareProject.Client.ViewModels;

@if (ViewModel is not null)

{

<**EditForm** **Model**="@ViewModel">

<div class="col-4">

<div class="form-group">

<label for="Oldal">Név</label>

<**InputText** class="form-control" id="nev" @bind-Value="@ViewModel.Side" />

</div>

<button class="mt-2" type="submit" disabled="@ViewModel.IsBusy">Terület és kerület meghatározása</button>

</div>

</**EditForm**>

}

@code {

[CascadingParameter(Name = "ViewModel")]

private ISquareViewModel? ViewModel { get; set; }

}

ShowSqaureData:

@using SquareProject.Client.ViewModels;

@if (ViewModel != null)

{

<div class="card mt-2 p-2">

<h5 class="card-title">Kiértékelés</h5>

<div class="card-body">

<p>Kerület: @ViewModel.Perimeter</p>

<p>Terület: @ViewModel.Area</p>

</div>

</div>

}

@code {

[CascadingParameter(Name = "ViewModel")]

private ISquareViewModel? ViewModel { get; set; }

}

Apró javítások után:

A BasePage-t működésre bírjuk és az adatbeírásnál a Side property frissül

