

# 2023.02

# BLAZOR 밋업




## Sessions

- WinForm 프로젝트에서 Blazor 프로젝트로 이동하기
- Sample 앱으로 친해지기





## Speaker

-  김 진석
-  (주)케이에이치시스템즈 대표이사
-  [facebook.com/iamjinseok.kr](https://facebook.com/iamjinseok.kr)

# WinForm 프로젝트에서 Blazor 프로젝트로 이동하기

## WinForm이 내 마음의 고향인데...

### WinForm

- 폼?
- 이벤트?
- 상용컴포넌트 사용법?

### WPF

- XAML?
- Data Binding?
- Prism?

### ASP.NET

- 웹서버?
- JavaScript?
- HTML?
- CSS?

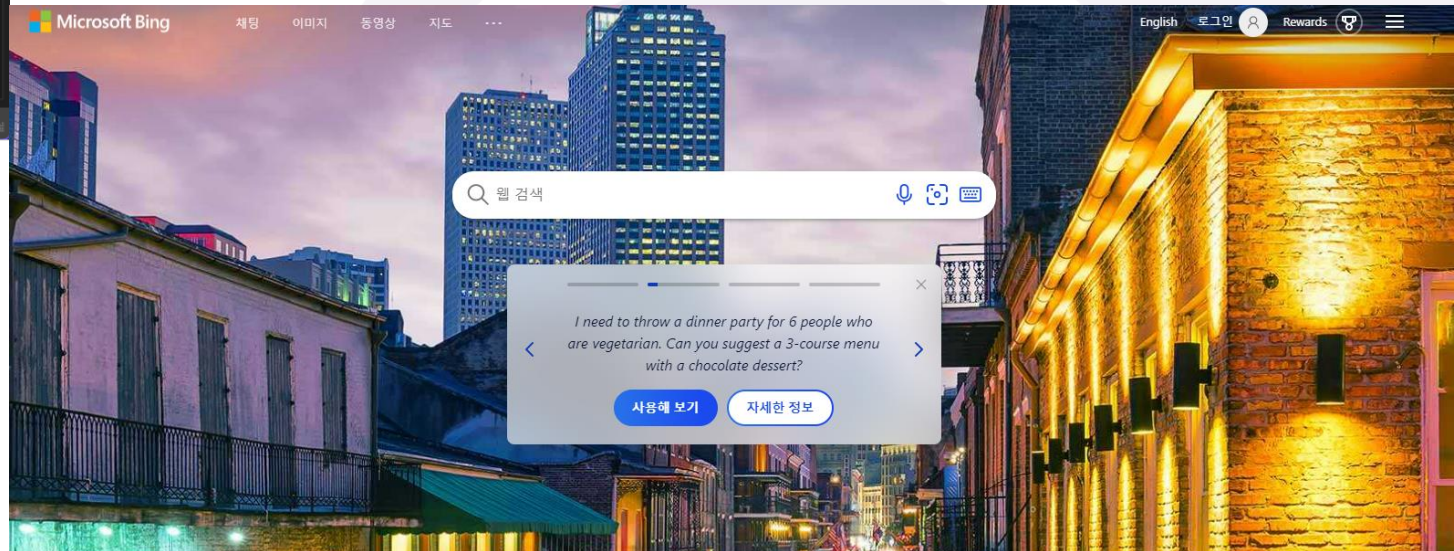
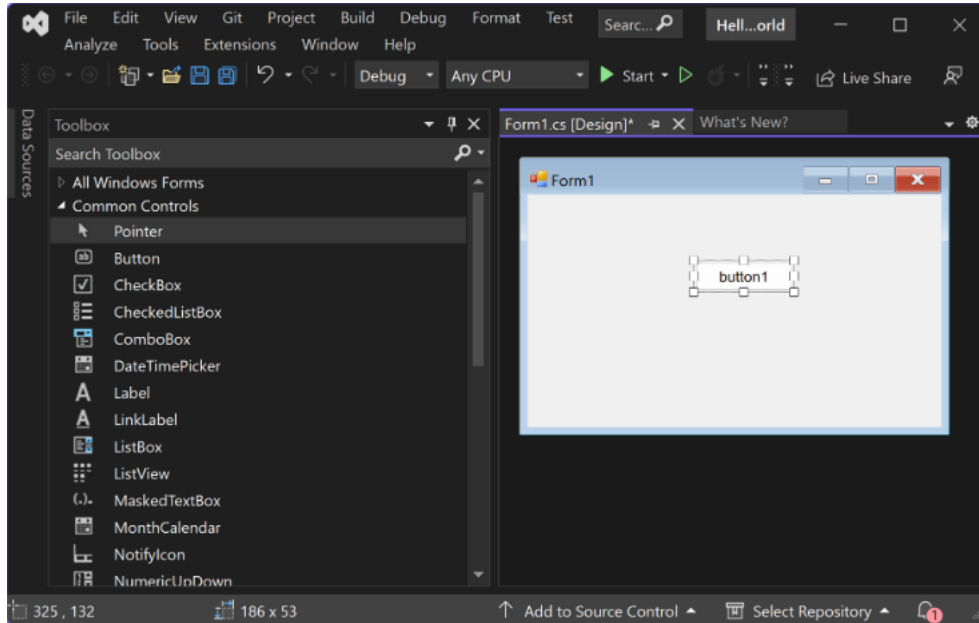
### React

- MS가 아니라고?
- TypeScript?
- FrontEnd?
- Node.js?

### Blazor

- ....또?

## WinForm에서는 나혼자 다 했었는데...?

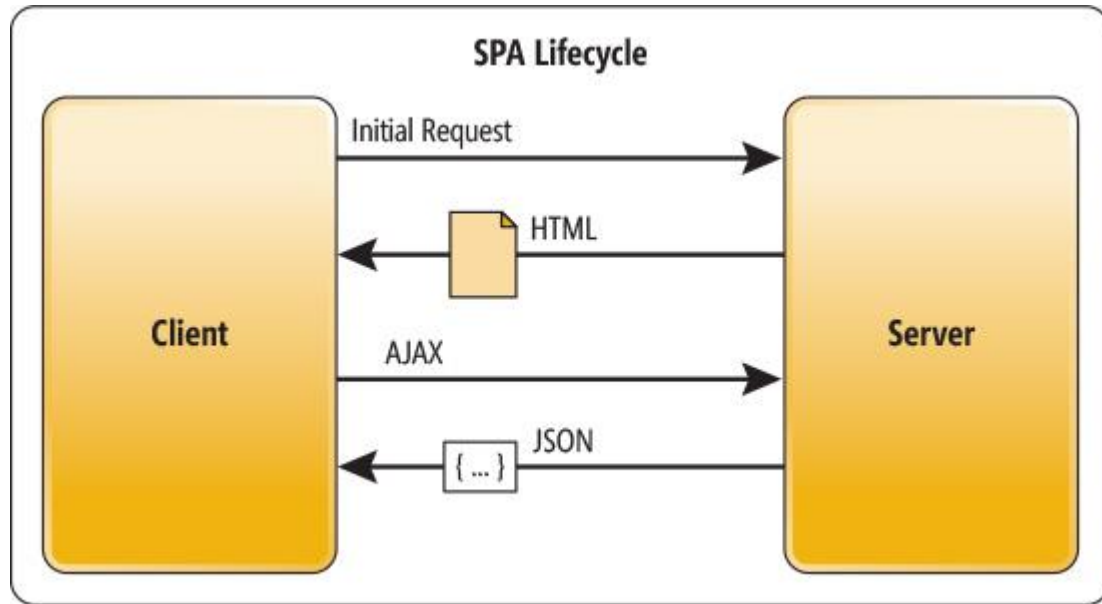


## Blazor 프로젝트가 되면...?

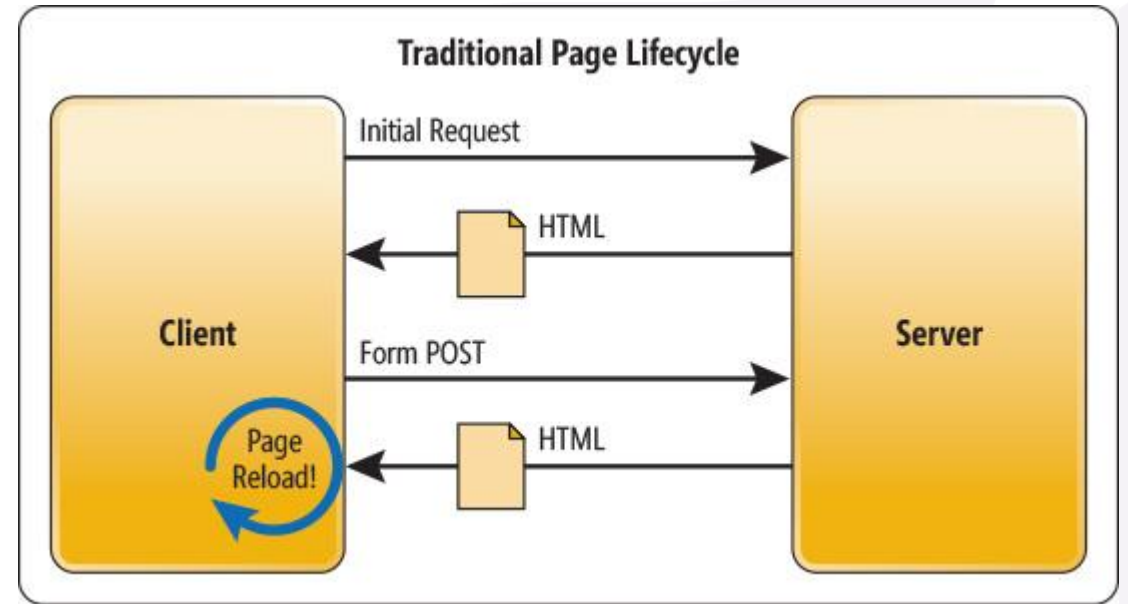
~~여긴 어디? 나는 누구?~~

## Blazor Project란?

# C#으로 SPA를 하고 싶다!!!



SPA



MPA

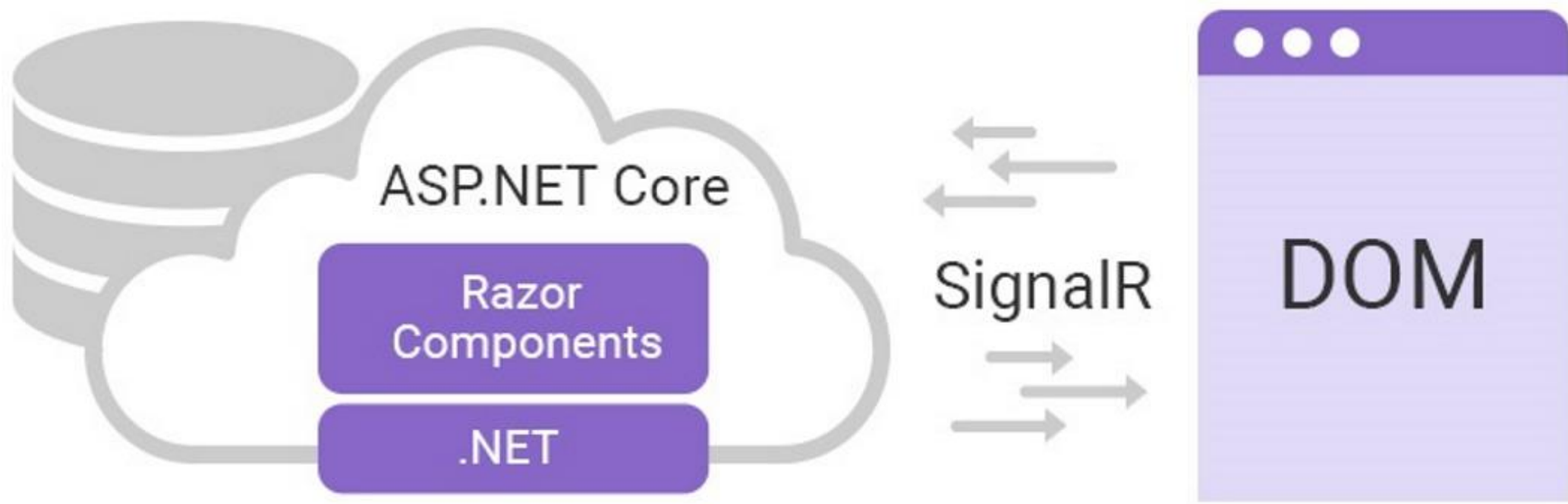
<https://velog.io/@gwanuuoo/>

## Blazor?

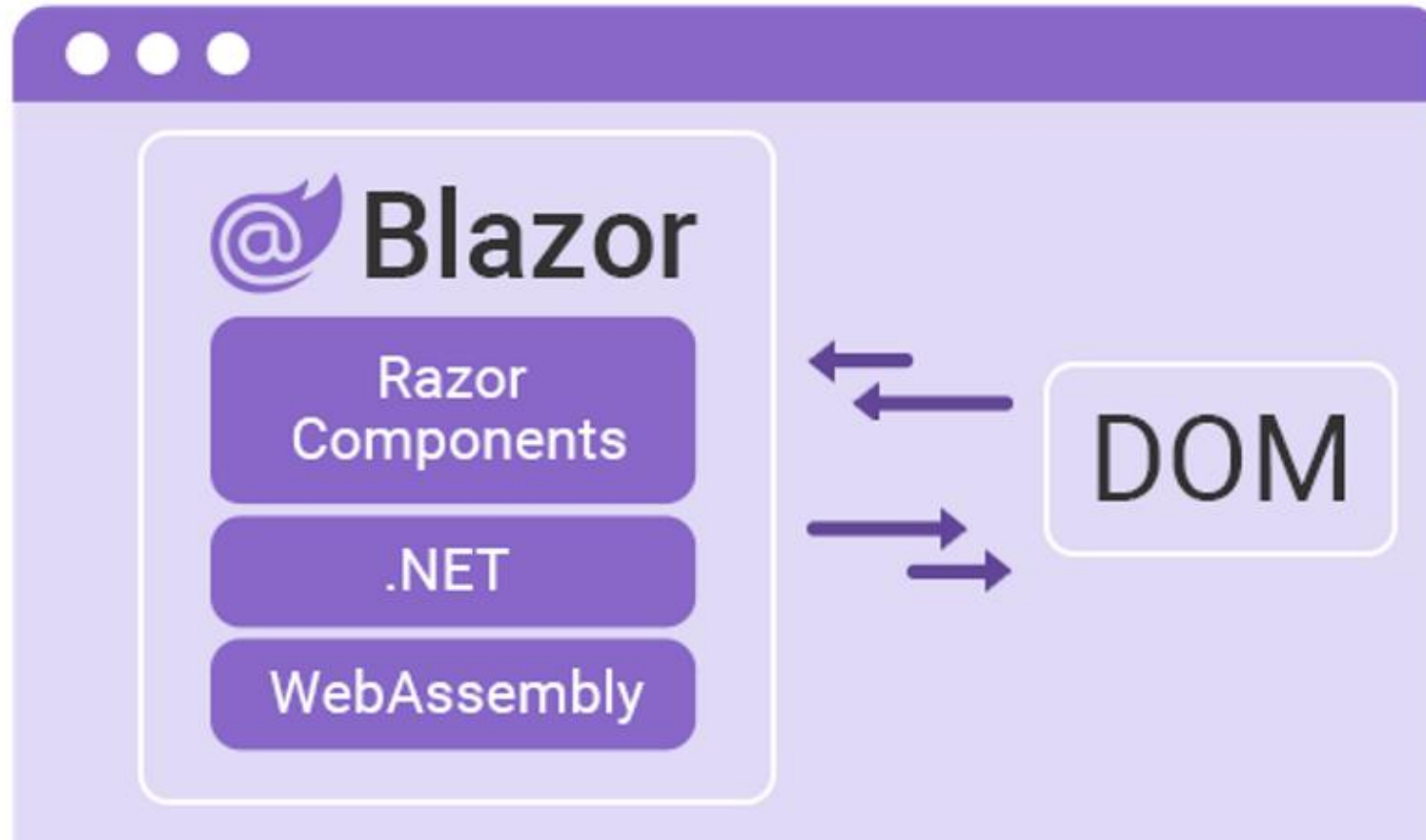
- Blazor Server
- Blazor WebAssembly
- Blazor PWA(Progressive Web Apps)
- Blazor Hybrid
- Blazor Native



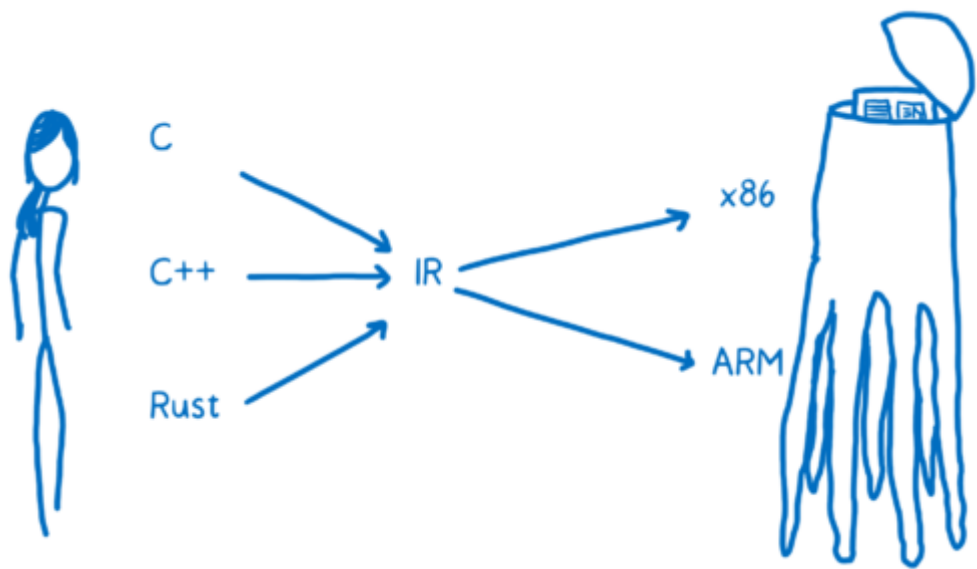
## Blazor Server



# Blazor Server



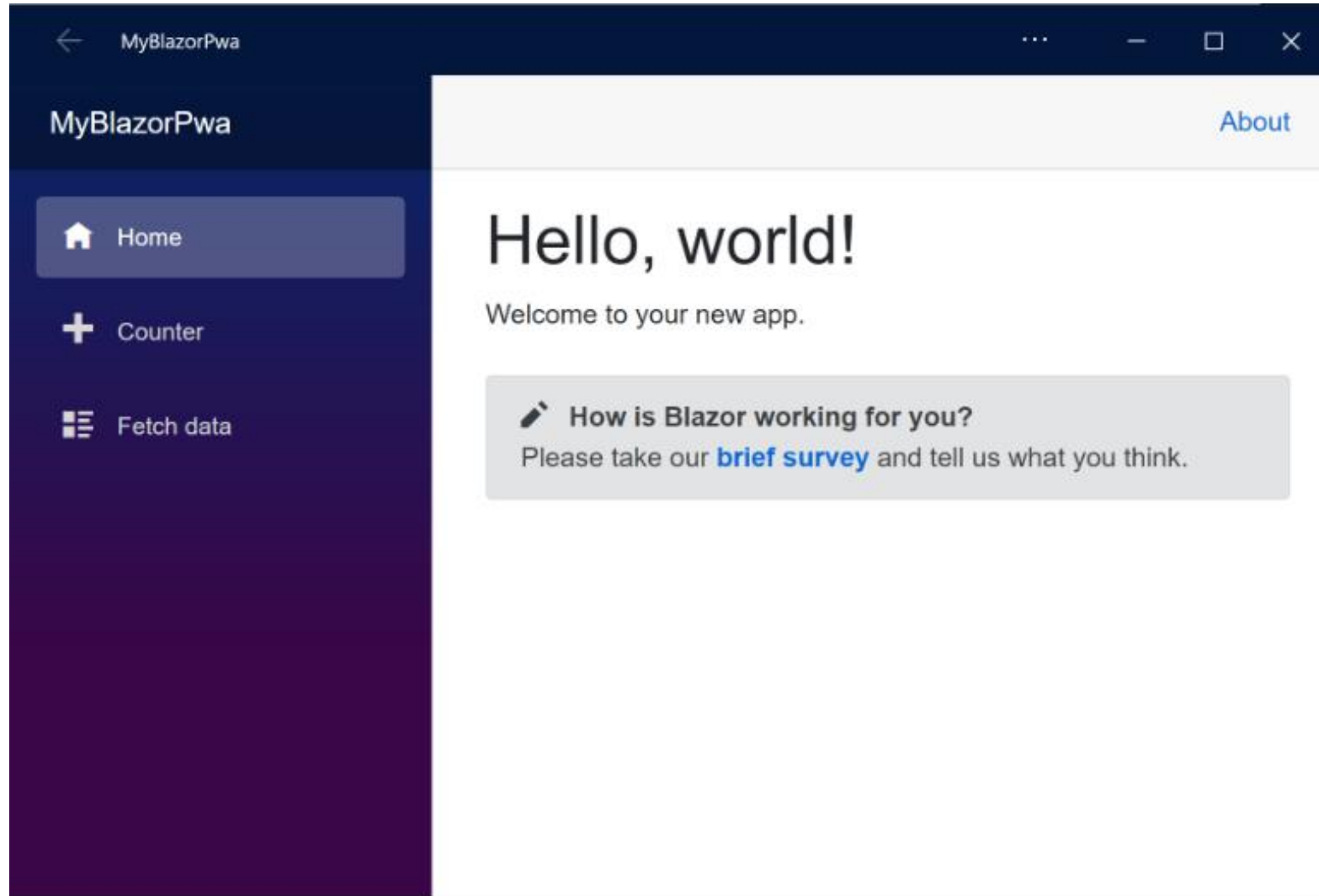
# WebAssembly?



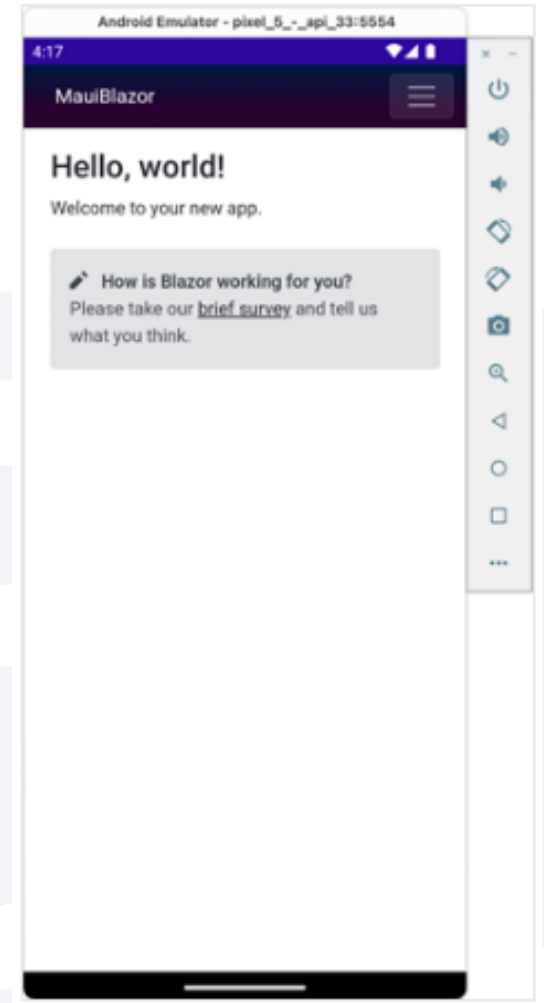
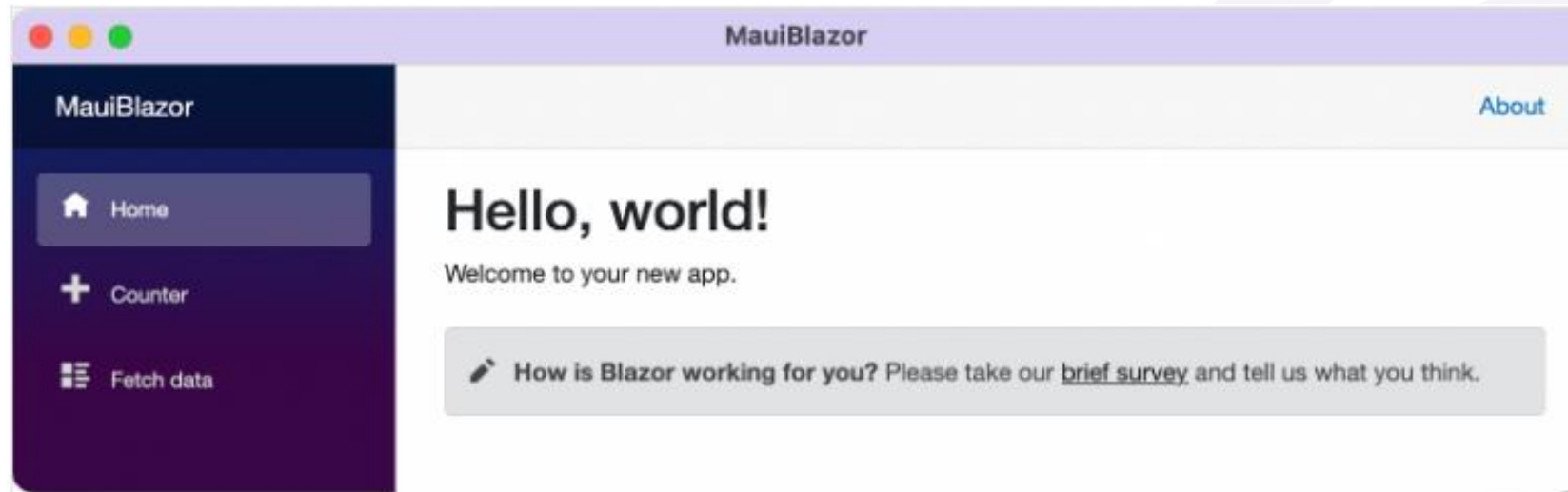
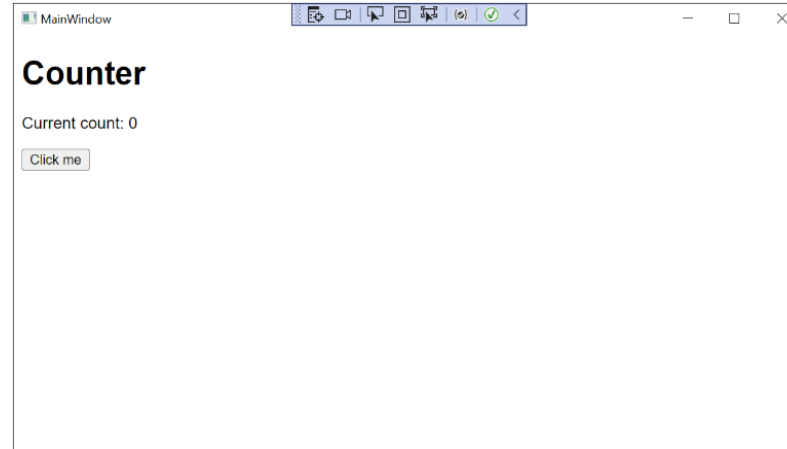
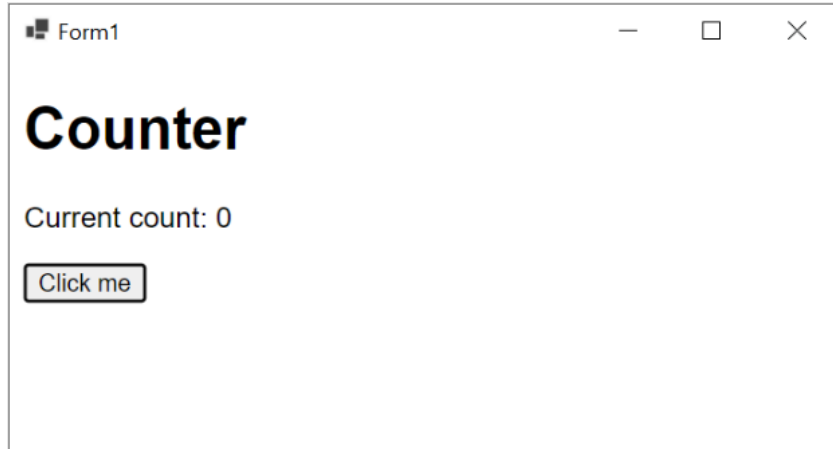
- 2017년 처음 발표
- 브라우저에서 실행하는 바이트 코드
- 최신 브라우저가 대응

<https://dongwoo.blog/>

# Blazor PWA



# Blazor Hybrid



## Blazor supported platforms

### Browser

- Apple Safari
- Google Chrome
- Microsoft Edge
- Mozilla Firefox

### Hybrid

- Microsoft Edge  
WebView2 on Windows
- Chrome on Android
- Safari on iOS and macOS

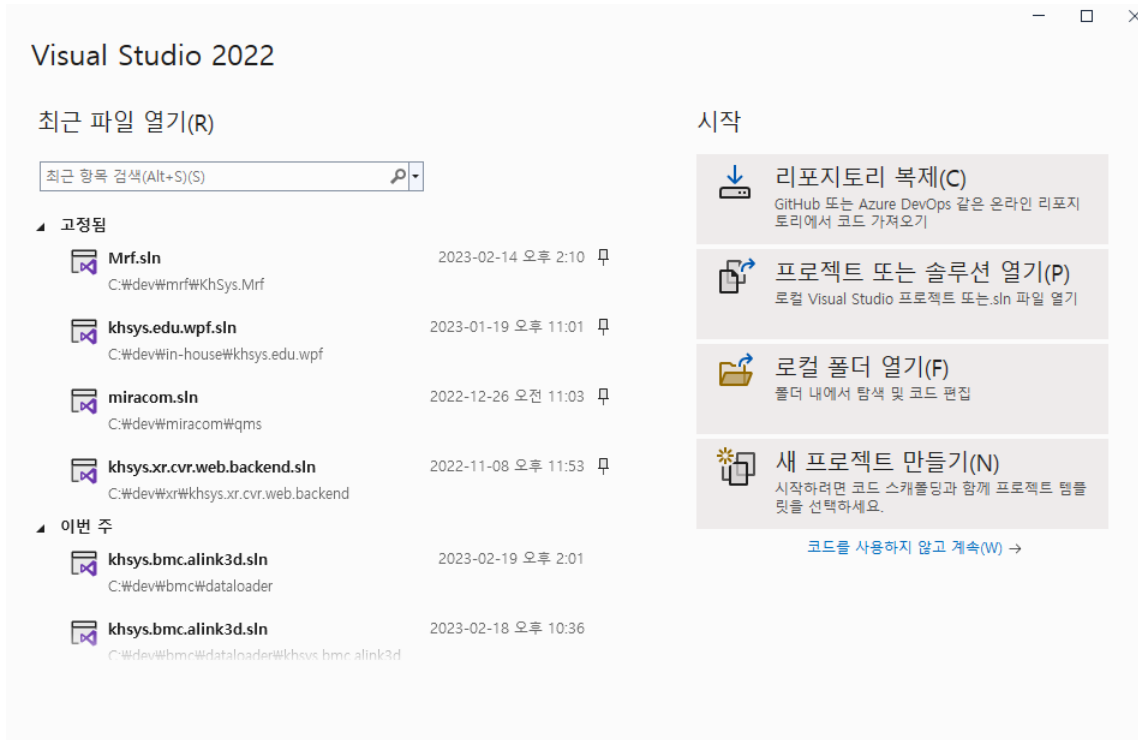


*break  
time*

# Sample 앱으로 친해지기 (Blazor WebAssembly 앱)



# Blazor project 만들기 – Visual Studio



## 새 프로젝트 만들기 > Blazor WebAssembly 앱

# Blazor project 만들기 – Visual Studio

## 새 프로젝트 구성

Blazor WebAssembly 앱

C# Linux macOS Windows Blazor 클라우드 웹

프로젝트 이름(I)

BlazorApp1

위치(L)

C:\dev\blazor\

솔루션 이름(M) ⓘ

BlazorApp1

☐ 솔루션 및 프로젝트를 같은 디렉터리에 배치(D)

뒤로(B)

다음(N)

## 추가 정보

Blazor WebAssembly 앱

C# Linux macOS Windows Blazor 클라우드 웹

프레임워크(F) ⓘ

.NET 7.0 (표준 용어 지원)

인증 유형(A) ⓘ

없음

☐ HTTPS에 대한 구성(H) ⓘ

☐ ASP.NET Core 호스팅(H) ⓘ

☐ 프로그레시브 웹 애플리케이션(P) ⓘ

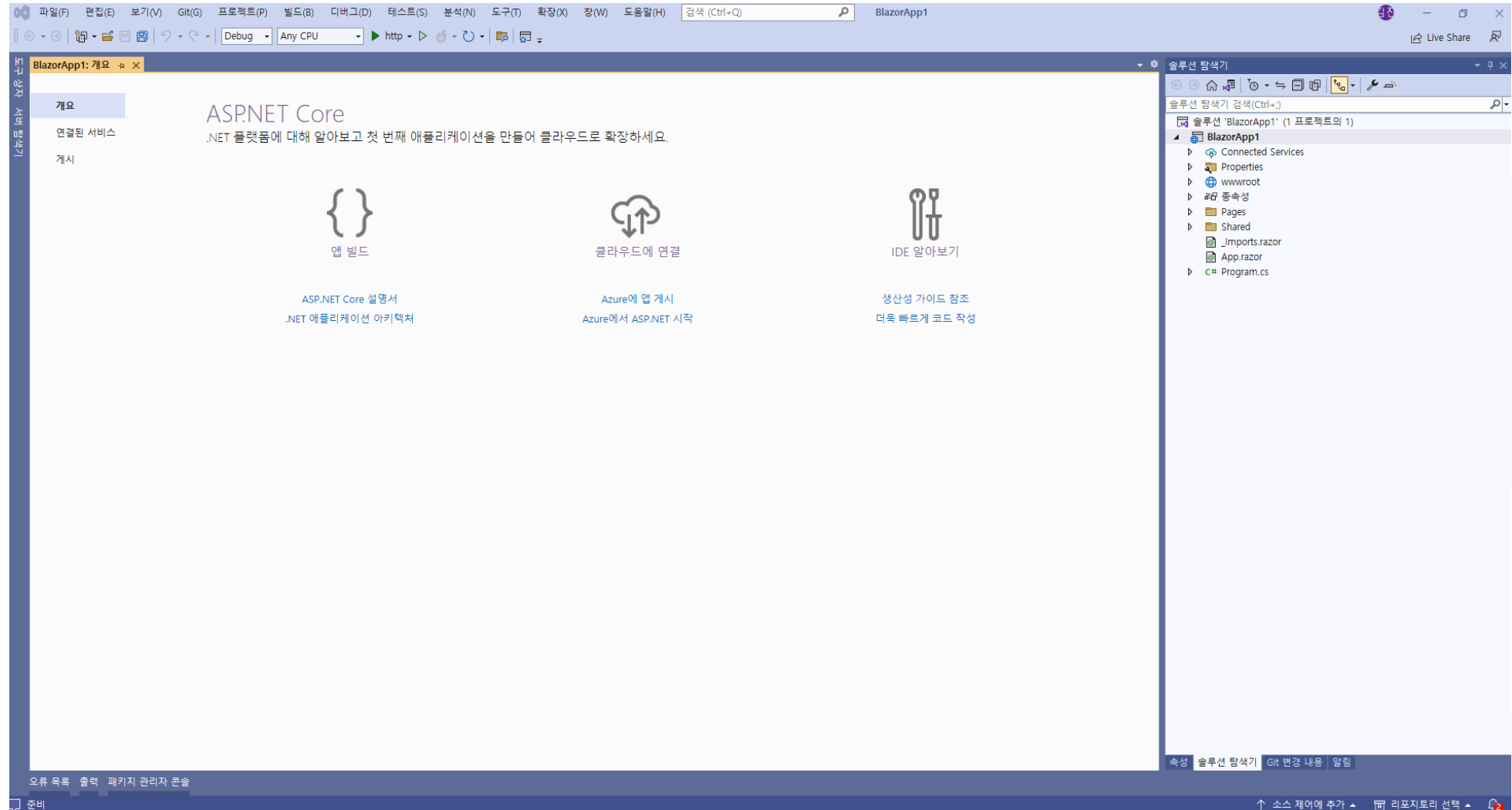
☐ 최상위 문 사용 안 함(T) ⓘ

뒤로(B)

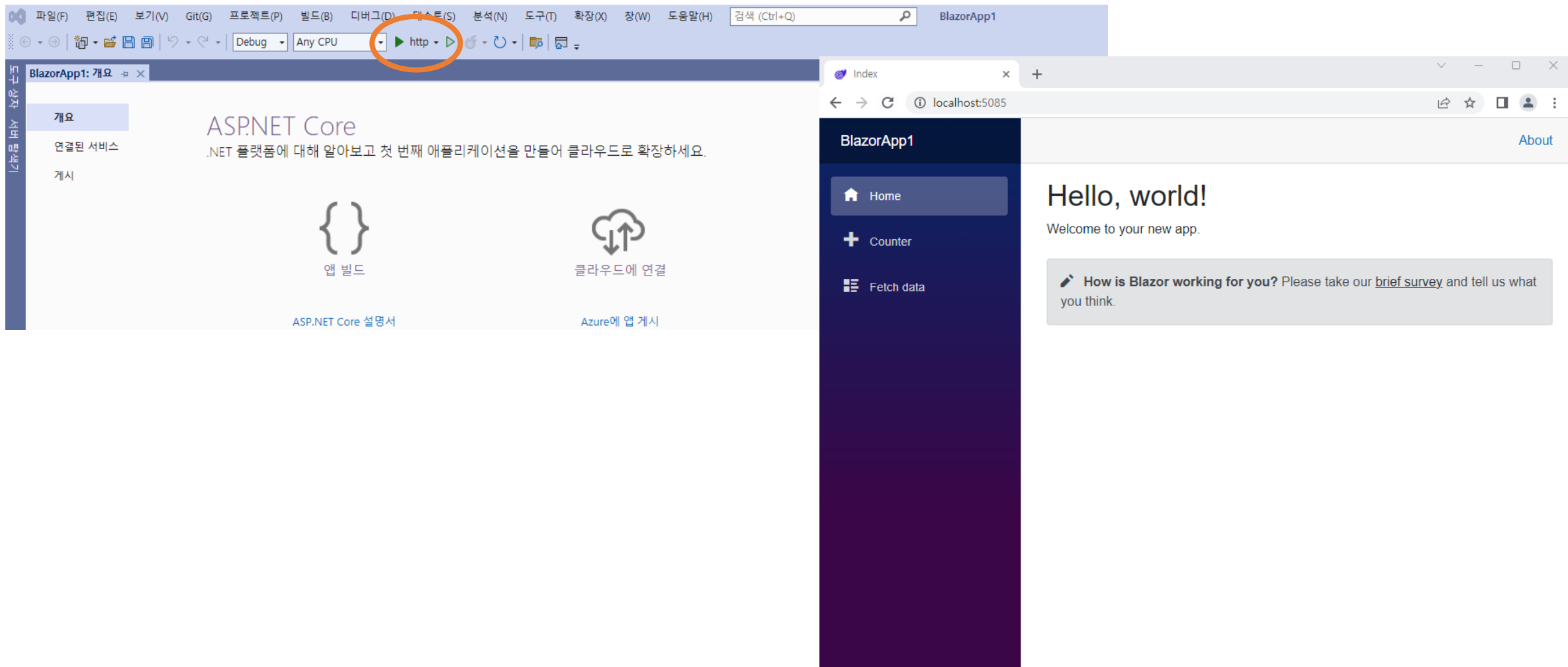
만들기(C)

# Blazor PWA로 진행시 체크

# Blazor project 만들기 – Visual Studio



# Blazor 기동 – Visual Studio



## Blazor project 만들기 – Visual Studio Code

```
Windows PowerShell
PS C:\dev\blazor> dotnet new blazorwasm -o BlazorApp2 -f net7.0 --no-https
```

```
Windows PowerShell
PS C:\dev\blazor> dotnet new blazorwasm -o BlazorApp2 -f net7.0 --no-https
"Blazor WebAssembly 앱" 템플릿이 성공적으로 생성되었습니다.
이 템플릿에는 Microsoft 이외의 타사 기술이 포함되어 있습니다. 자세한 내용은 https://aka.ms/aspnetcore/7.0-third-party-notices을(를) 참조하세요.

생성 후 작업 처리 중...
C:\dev\blazor\BlazorApp2\BlazorApp2.csproj 복원 중:
  복원할 프로젝트를 확인하는 중...
  C:\dev\blazor\BlazorApp2\BlazorApp2.csproj을(를) 222 ms 동안 복원했습니다.
  복원에 성공했습니다.

PS C:\dev\blazor>
```

dotnet new blazorwasm -o BlazorApp2 -f net7.0 --no-https

# Blazor project 만들기 – Visual Studio Code

```
Windows PowerShell
PS C:\dev\blazor> dotnet new blazorwasm -o BlazorApp2 -f net7.0 --no-https
"Blazor WebAssembly 앱" 템플릿이 성공적으로 생성되었습니다.
이 템플릿에는 Microsoft 이외의 타사 기술이 포함되어 있습니다. 자세한 내용은 https://aka.ms/aspnetcore/7.0-third-party-not
ices을(를) 참조하세요.

생성 후 작업 처리 중...
C:\dev\blazor\BlazorApp2\BlazorApp2.csproj 복원 중:
  복원할 프로젝트를 확인하는 중...
  C:\dev\blazor\BlazorApp2\BlazorApp2.csproj을(를) 222 ms 동안 복원했습니다.
  복원에 성공했습니다.

PS C:\dev\blazor> ls

디렉터리: C:\dev\blazor

Mode                LastWriteTime         Length Name
----                -
d-----         2023-02-22 오후 9:20           BlazorApp1
d-----         2023-02-22 오후 9:25           BlazorApp2

PS C:\dev\blazor>
```

```
Windows PowerShell
PS C:\dev\blazor\BlazorApp2> dir

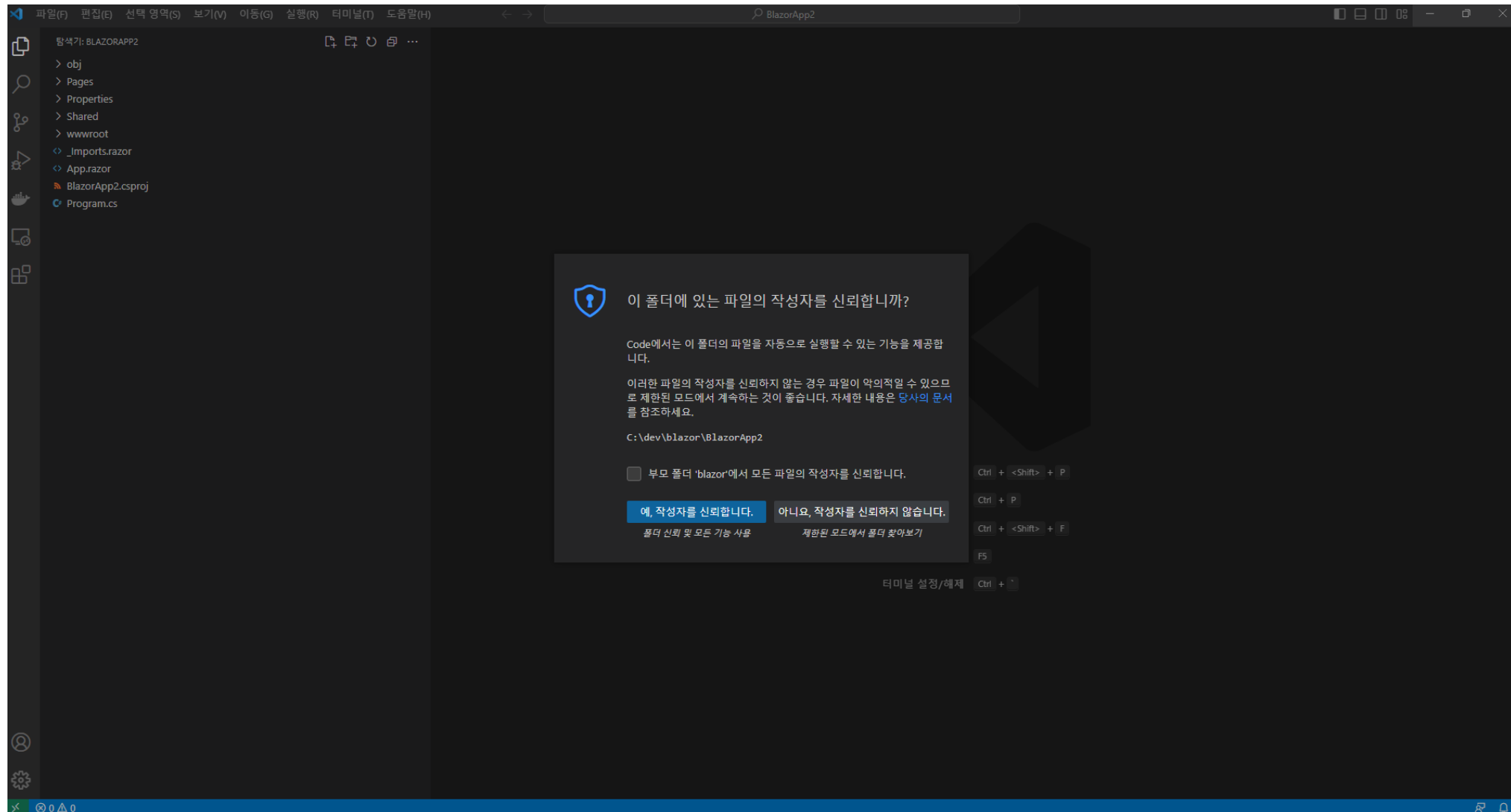
디렉터리: C:\dev\blazor\BlazorApp2

Mode                LastWriteTime         Length Name
----                -
d-----         2023-02-22 오후 9:25           obj
d-----         2023-02-22 오후 9:25           Pages
d-----         2023-02-22 오후 9:25           Properties
d-----         2023-02-22 오후 9:25           Shared
d-----         2023-02-22 오후 9:25           wwwroot
-a-----         2023-02-22 오후 9:25          477 App.razor
-a-----         2023-02-22 오후 9:25          488 BlazorApp2.csproj
-a-----         2023-02-22 오후 9:25          435 Program.cs
-a-----         2023-02-22 오후 9:25          383 _Imports.razor

PS C:\dev\blazor\BlazorApp2> code .|
```

code .

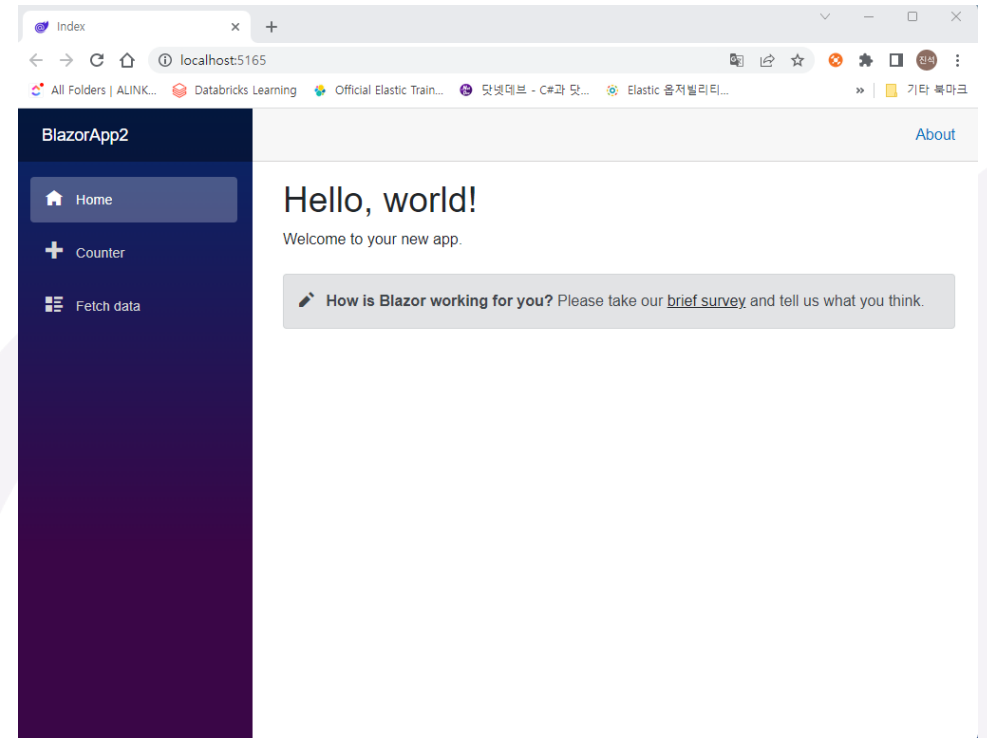
# Blazor project 만들기 – Visual Studio Code



# Blazor 기동 – Visual Studio Code

```
문제  출력  디버그 콘솔  터미널  AZURE

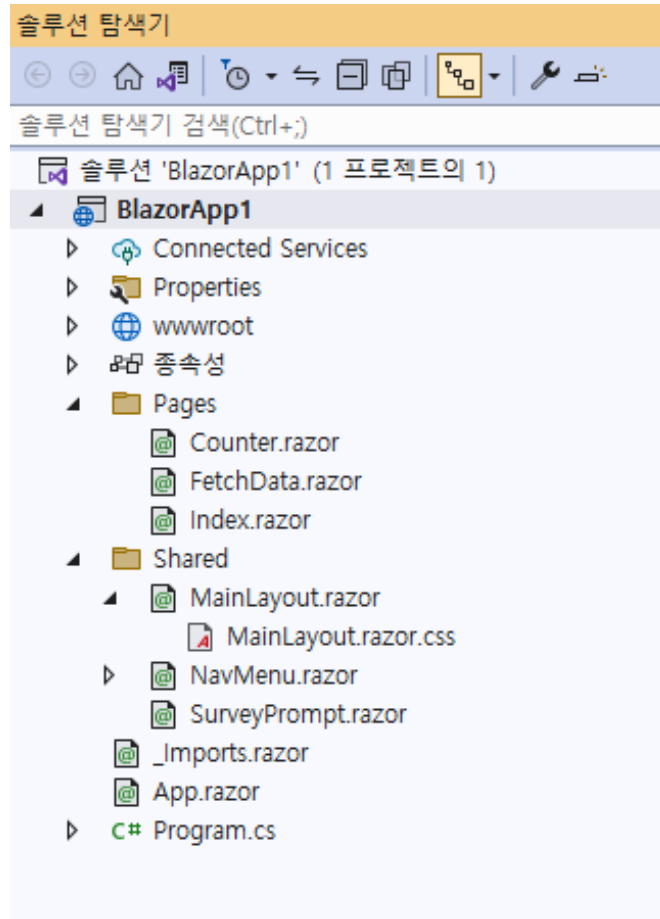
PS C:\dev\blazor\BlazorApp2> dotnet watch
dotnet watch 🔥 Hot reload enabled. For a list of supported edits, see https://aka.ms/dotnet/hot-reload.
💡 Press "Ctrl + R" to restart.
dotnet watch 🛠 Building...
복원할 프로젝트를 확인하는 중...
복원할 모든 프로젝트가 최신 상태입니다.
BlazorApp2 -> C:\dev\blazor\BlazorApp2\bin\Debug\net7.0\BlazorApp2.dll
BlazorApp2 (Blazor output) -> C:\dev\blazor\BlazorApp2\bin\Debug\net7.0\wwwroot
dotnet watch 🚀 Started
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5165
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\dev\blazor\BlazorApp2
[]
```



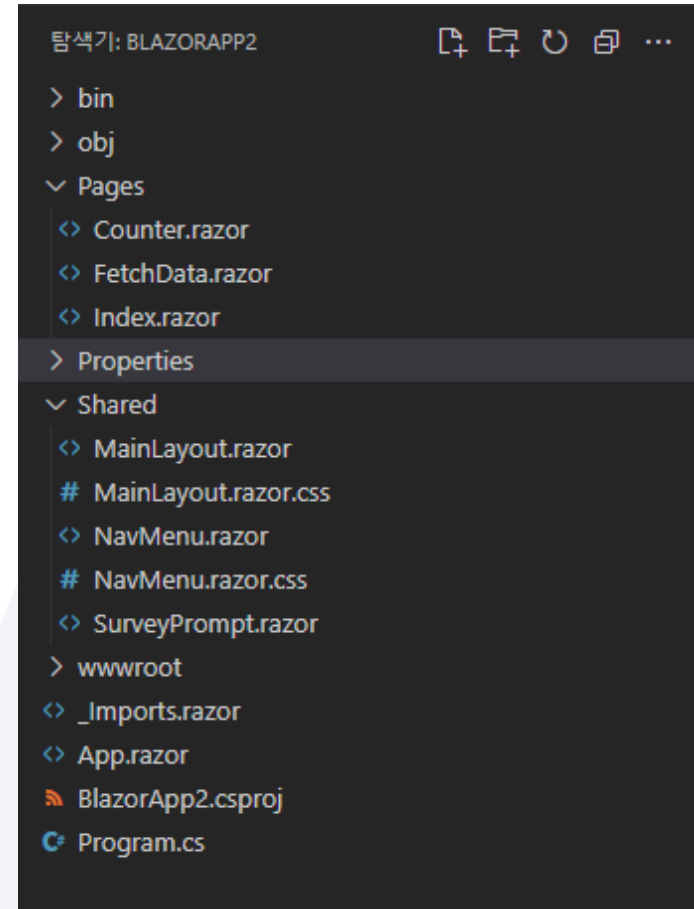
dotnet watch



## Project 구성



Visual Studio



Visual Studio Code

## Index.html – 앱이 동작하는 메인 페이지

index.html BlazorApp1: 개요

```

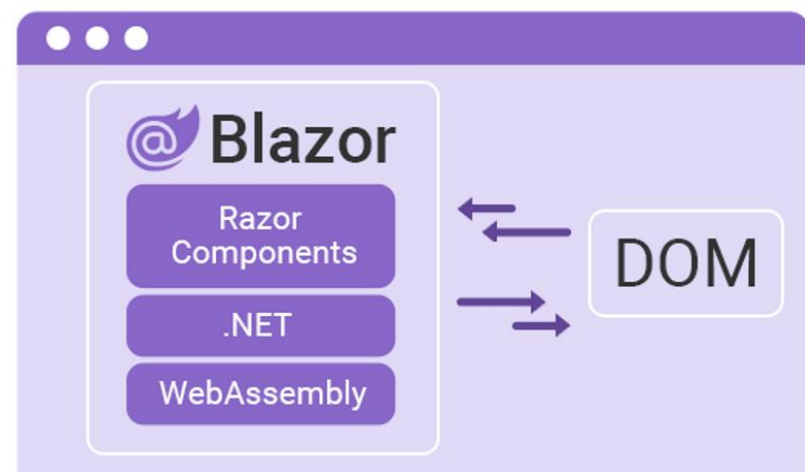
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="utf-8" />
6      <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no" />
7      <title>BlazorApp1</title>
8      <base href="/" />
9      <link href="css/bootstrap/bootstrap.min.css" rel="stylesheet" />
10     <link href="css/app.css" rel="stylesheet" />
11     <link rel="icon" type="image/png" href="favicon.png" />
12     <link href="BlazorApp1.styles.css" rel="stylesheet" />
13 </head>
14
15 <body>
16     <div id="app">
17         <svg class="loading-progress">
18             <circle r="40%" cx="50%" cy="50%" />
19             <circle r="40%" cx="50%" cy="50%" />
20         </svg>
21         <div class="loading-progress-text"></div>
22     </div>
23
24     <div id="blazor-error-ui">
25         An unhandled error has occurred.
26         <a href="" class="reload">Reload</a>
27         <a class="dismiss">✕</a>
28     </div>
29     <script src="_framework/blazor.webassembly.js"></script>
30 </body>
31 </html>
32
33

```

앱이 동작하는 위치

에러일 때 표시

Blazor 기동을 위한 라이브러리



## Program.cs – blazor.webassembly.js로부터 처음 로딩되는 파일

Program.cs x BlazorApp1: 개요

BlazorApp1

```
1 using BlazorApp1;
2 using Microsoft.AspNetCore.Components.Web;
3 using Microsoft.AspNetCore.Components.WebAssembly.Hosting;
4
5 var builder = WebAssemblyHostBuilder.CreateDefault(args);
6 builder.RootComponents.Add<App>("#app");
7 builder.RootComponents.Add<HeadOutlet>("head::after");
8
9 builder.Services.AddScoped(sp => new HttpClient { BaseAddress = new Uri(builder.HostEnvironment.BaseAddress) });
10
11 await builder.Build().RunAsync();
12
```

Root 컴포넌트 준비

호스트 준비 및 실행

# App.razor – 어플리케이션의 시작점이 되는 Root Component

App.razor

```
1 <Router AppAssembly="@typeof(App).Assembly">
```

Routing 관련 처리

```
2   <Found Context="routeData">
```

```
3     <RouteView RouteData="@routeData" DefaultLayout="@typeof(MainLayout)" />
```

페이지를 찾음

```
4     <FocusOnNavigate RouteData="@routeData" Selector="h1" />
```

```
5   </Found>
```

```
6   <NotFound>
```

```
7     <PageTitle>Not found</PageTitle>
```

```
8     <LayoutView Layout="@typeof(MainLayout)">
```

페이지를 찾지 못함

```
9       <p role="alert">Sorry, there's nothing at this address.</p>
```

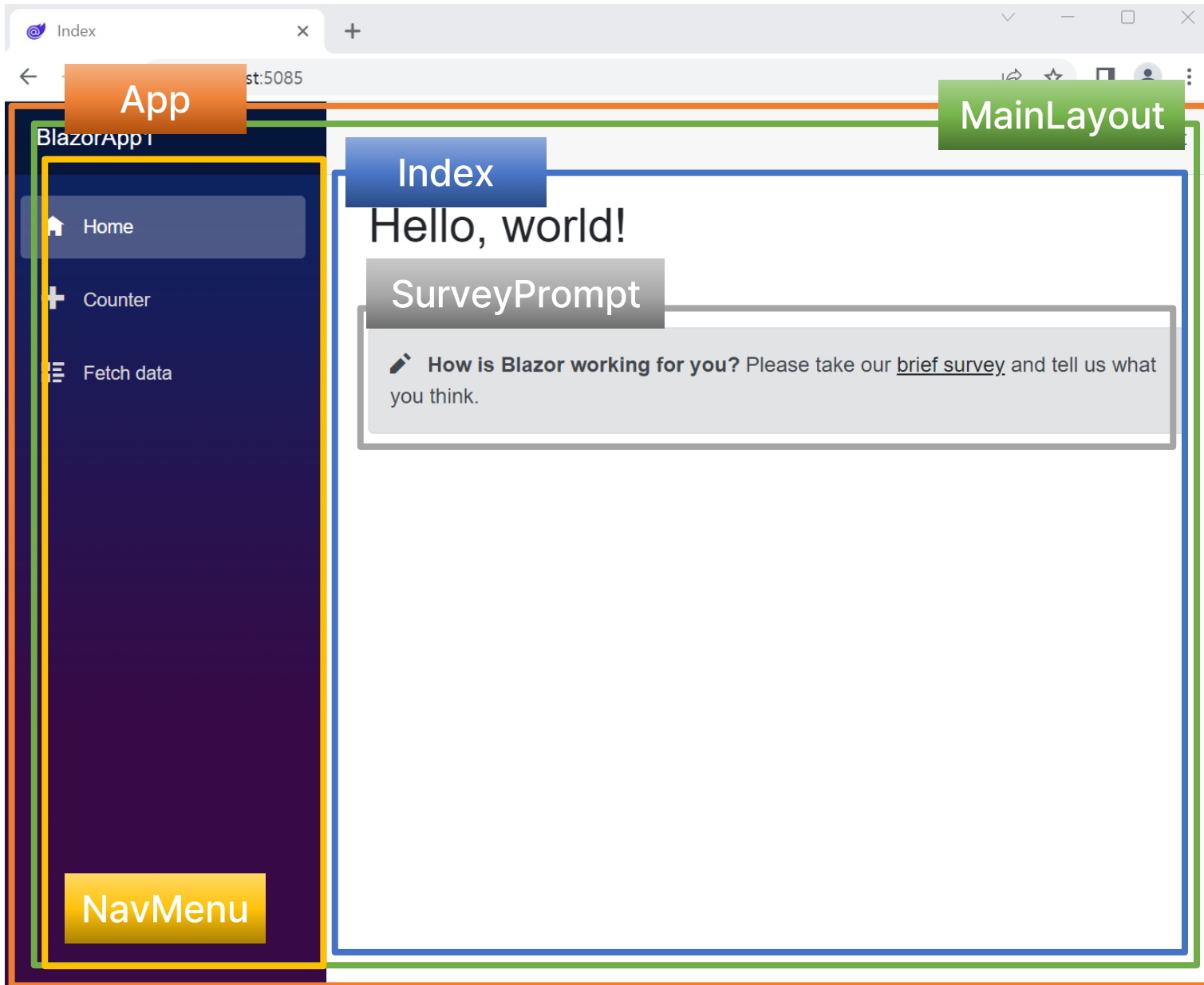
```
10    </LayoutView>
```

```
11  </NotFound>
```

```
12 </Router>
```

```
13
```

## Sample 앱의 Component 구조



## Pages/Index.razor

Index.razor

```
1 @page "/"
2
3 <PageTitle>Index</PageTitle>
4
5 <h1>Hello, world!</h1>
6
7 Welcome to your new app.
8
9 <SurveyPrompt Title="How is Blazor working for you?" />
10
```

Routing 경로

## Shared/MainLayout.razor

```
MainLayout.razor X Index.razor
1  @inherits LayoutComponentBase
2
3  <div class="page">
4      <div class="sidebar">
5          <NavMenu />
6      </div>
7
8      <main>
9          <div class="top-row px-4">
10             <a href="https://docs.microsoft.com/aspnet/" target="_blank">About</a>
11          </div>
12
13          <article class="content px-4">
14              @Body
15          </article>
16      </main>
17  </div>
18
```

Layout은 LayoutComponentBase를 상속 받아야 합니다.

## Pages/Counter.razor – 이벤트 핸들러, @코드

Counter.razor

```
1  @page "/counter"
2
3  <PageTitle>Counter</PageTitle>
4
5  <h1>Counter</h1>
6
7  <p role="status">Current count: @currentCount</p>
8
9  <button class="btn btn-primary" @onclick="IncrementCount">Click me</button>
10
11  @code {
12      private int currentCount = 0;
13
14      private void IncrementCount()
15      {
16          currentCount++;
17      }
18  }
```

- @code 블록 안에 코드 작성
- @변수명으로 사용
- Count.razor.cs로 클래스로 생성가능
  - partial 로 선언
  - 동일 위치 저장

이벤트를 중지하거나 기본동작을 무시하는 경우 별도로 지정합니다.  
예) @onclick:stopPropagation, @onclick:preventDefault

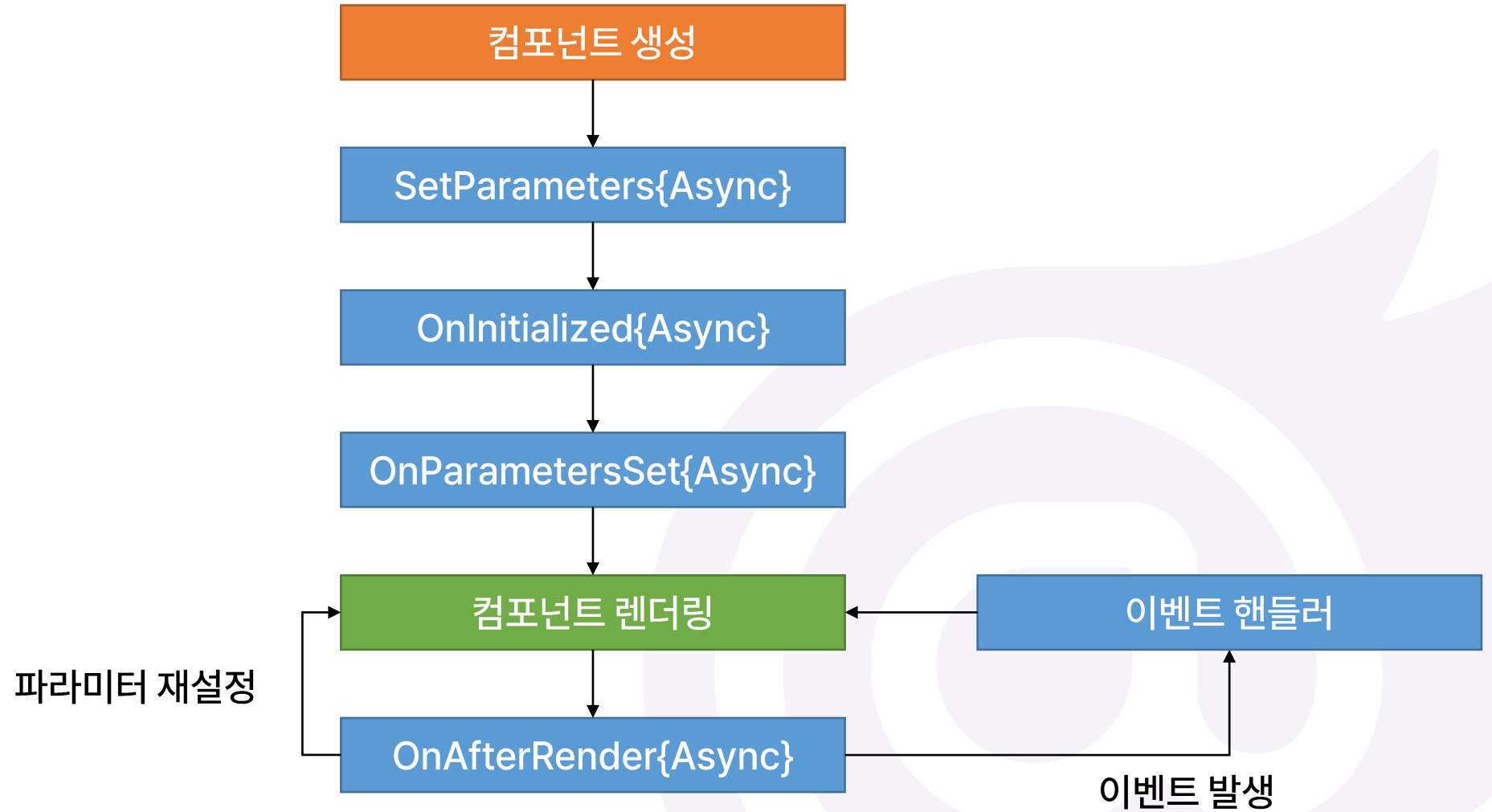


## 기본 제공 이벤트 인수

- 클립보드 : ClipboardEventArgs
- 끌기 : DragEventArgs
- Error : EventArgs
- 이벤트 : EventArgs
- 포커스 : FocusEventArgs
- 입력 : ChangeEventArgs
- Keyboard : KeyboardEventArgs
- 마우스 : ClipboardEventArgs
- 마우스 포인터 : DragEventArgs
- 마우스 휠 : EventArgs
- 진행률 : EventArgs
- 터치 : FocusEventArgs

[https://learn.microsoft.com/ko-kr/aspnet/core/blazor/components/event-handling?  
view=aspnetcore-7.0#built-in-event-arguments](https://learn.microsoft.com/ko-kr/aspnet/core/blazor/components/event-handling?view=aspnetcore-7.0#built-in-event-arguments)

# Component LifeCycle



<https://learn.microsoft.com/ko-kr/aspnet/core/blazor/components/lifecycle?view=aspnetcore-7.0>

# Pages/FetchData.razor

FetchData.razor

```

1  @page "/fetchdata"
2  @inject HttpClient Http
3
4  <PageTitle>Weather forecast</PageTitle>
5
6  <h1>Weather forecast</h1>
7
8  <p>This component demonstrates fetching data from the server.</p>
9
10 @if (forecasts == null)
11 {
12     <p><em>Loading...</em></p>
13 }
14 else
15 {

```

injection service 주입

```

39 @code {
40     private WeatherForecast[]? forecasts;
41
42     protected override async Task OnInitializedAsync()
43     {
44         forecasts = await Http.GetFromJsonAsync<WeatherForecast[]>("sample-data/weather.json");
45     }
46
47     public class WeatherForecast
48     {
49         public DateOnly Date { get; set; }
50
51         public int TemperatureC { get; set; }
52
53         public string? Summary { get; set; }
54
55         public int TemperatureF => 32 + (int)(TemperatureC / 0.5556);
56     }
57 }
58

```

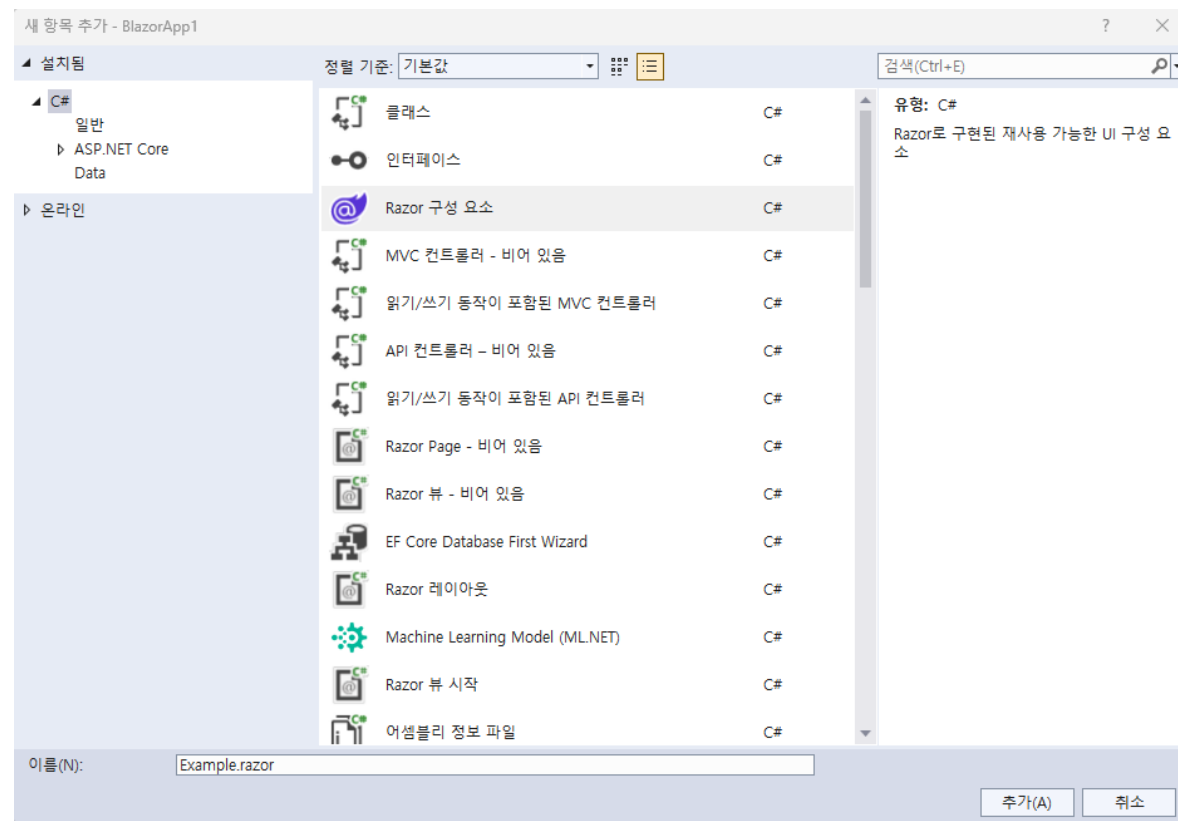
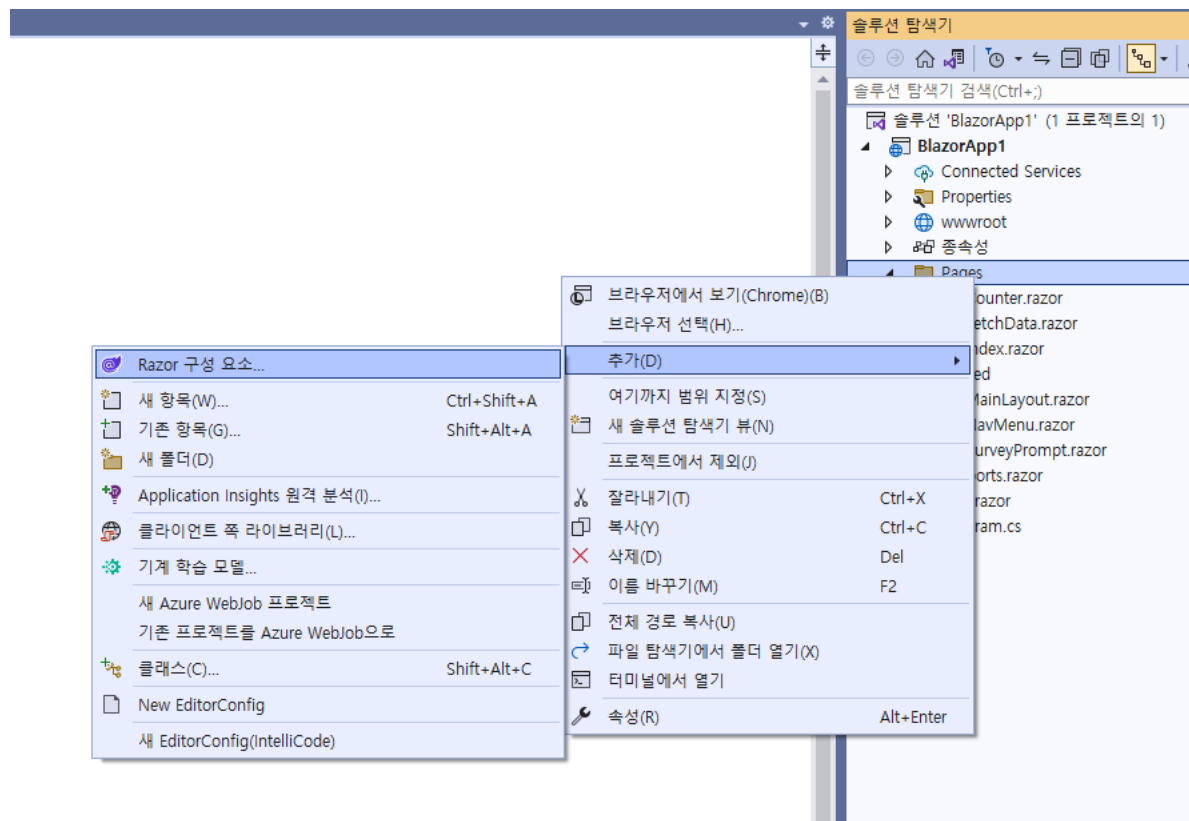
injection service 사용

injection service는 Program.cs에서 등록합니다.

## Session 2

뭐 좀 추가해 볼까요?

# 새로 추가 : Example.razor, SubText.razor



## 페이지 및 컴포넌트 추가

# NavLink

```

NavMenu.razor
1 <div class="top-row ps-3 navbar navbar-dark">
2   <div class="container-fluid">
3     <a class="navbar-brand" href="">BlazorApp1</a>
4     <button title="Navigation menu" class="navbar-toggler" @onclick="ToggleNavMenu">
5       <span class="navbar-toggler-icon"></span>
6     </button>
7   </div>
8 </div>
9
10 <div class="@NavMenuCssClass nav-scrollable" @onclick="ToggleNavMenu">
11   <nav class="flex-column">
12     <div class="nav-item px-3">
13       <NavLink class="nav-link" href="" Match="NavLinkMatch.All">
14         <span class="oi oi-home" aria-hidden="true"></span> Home
15       </NavLink>
16     </div>
17     <div class="nav-item px-3">
18       <NavLink class="nav-link" href="counter">
19         <span class="oi oi-plus" aria-hidden="true"></span> Counter
20       </NavLink>
21     </div>
22     <div class="nav-item px-3">
23       <NavLink class="nav-link" href="fetchdata">
24         <span class="oi oi-list-rich" aria-hidden="true"></span> Fetch data
25       </NavLink>
26     </div>
27     <div class="nav-item px-3">
28       <NavLink class="nav-link" href="example">
29         <span class="oi oi-list-rich" aria-hidden="true"></span> Example
30       </NavLink>
31     </div>
32   </nav>
33 </div>
34
35 @code {
36   private bool collapseNavMenu = true;
37
38   private string? NavMenuCssClass => collapseNavMenu ? "collapse" : null;
39
40   private void ToggleNavMenu()
41   {
42     collapseNavMenu = !collapseNavMenu;
43   }
44 }
45

```

## ■ NavLink : 내부 경로 추가

# Parameters

```
Example.razor  ▸ X NavMenu.razor
1  @page "/example"
2
3  <h3>Example</h3>
4
5  <button @onclick="clickBtn1">버튼 1</button>
6  <button @onclick="clickBtn2">버튼 2</button>
7
8  <SubText Text="@args" />
9
10 @code {
11     private string args = "클릭전 ";
12     private void clickBtn1()
13     {
14         args = "버튼1";
15         Console.WriteLine("버튼1클릭");
16     }
17
18     private void clickBtn2()
19     {
20         args = "버튼2";
21         Console.WriteLine("버튼2클릭");
22     }
23 }
24
```

```
SubText.razor  ▸ X Example.razor  NavMenu.razor
1  <h3>@Text</h3>
2
3  @code {
4      [Parameter]
5      public string? Text{ get; set; } = "기본값";
6  }
7
```

[Parameter]로 컴포넌트에 파라미터를 추가

# State Management

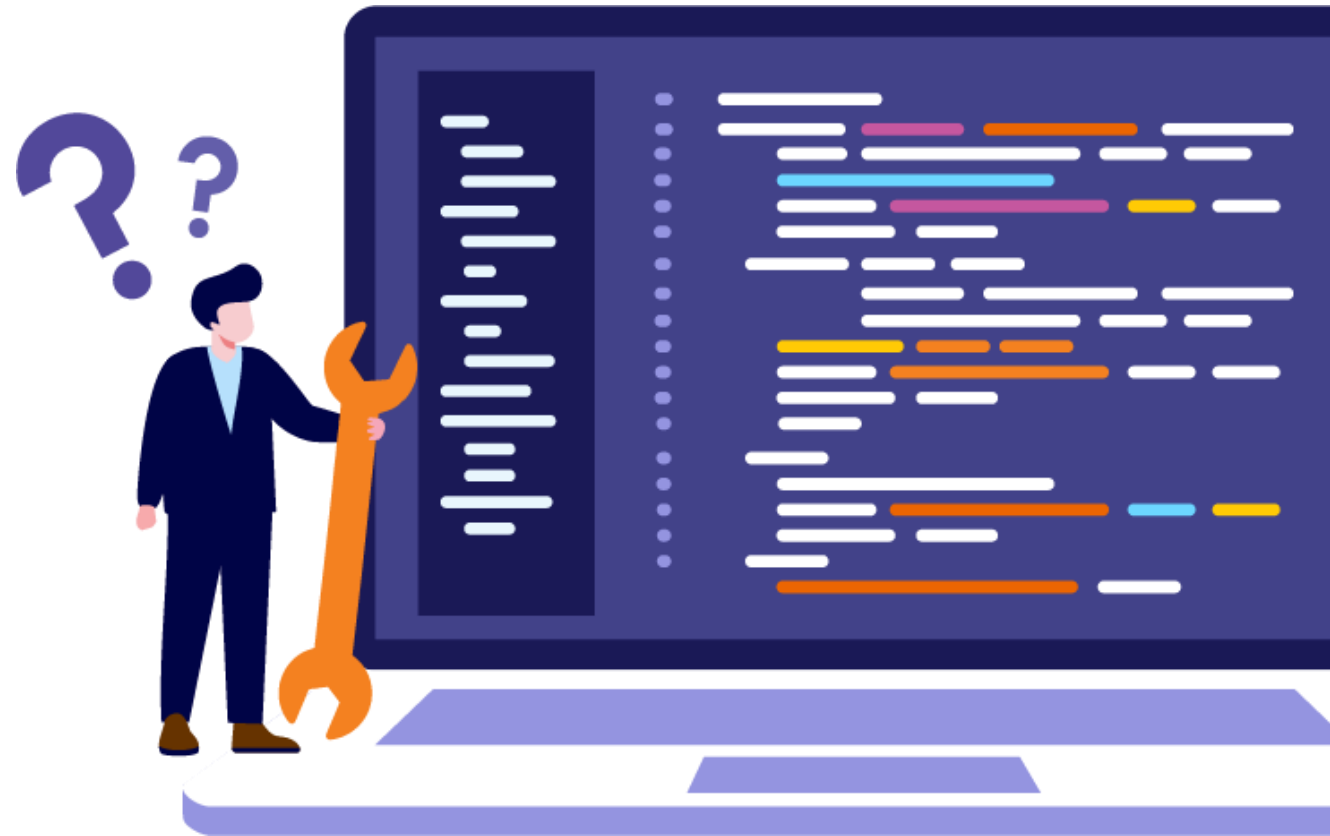
- Storing state in components  
member value로 저장  
event callback으로 parent component에 전달
- AppState class  
singleton class를 주입받아 사용
- State Management Lib 사용  
Fluxor library ... etc  
<https://github.com/mrppmorris/fluxor>



## Q&A

### Blazor Korea User Group

[facebook.com/groups/blazor.korea](https://facebook.com/groups/blazor.korea)



**THANK YOU**