

# 2023.05

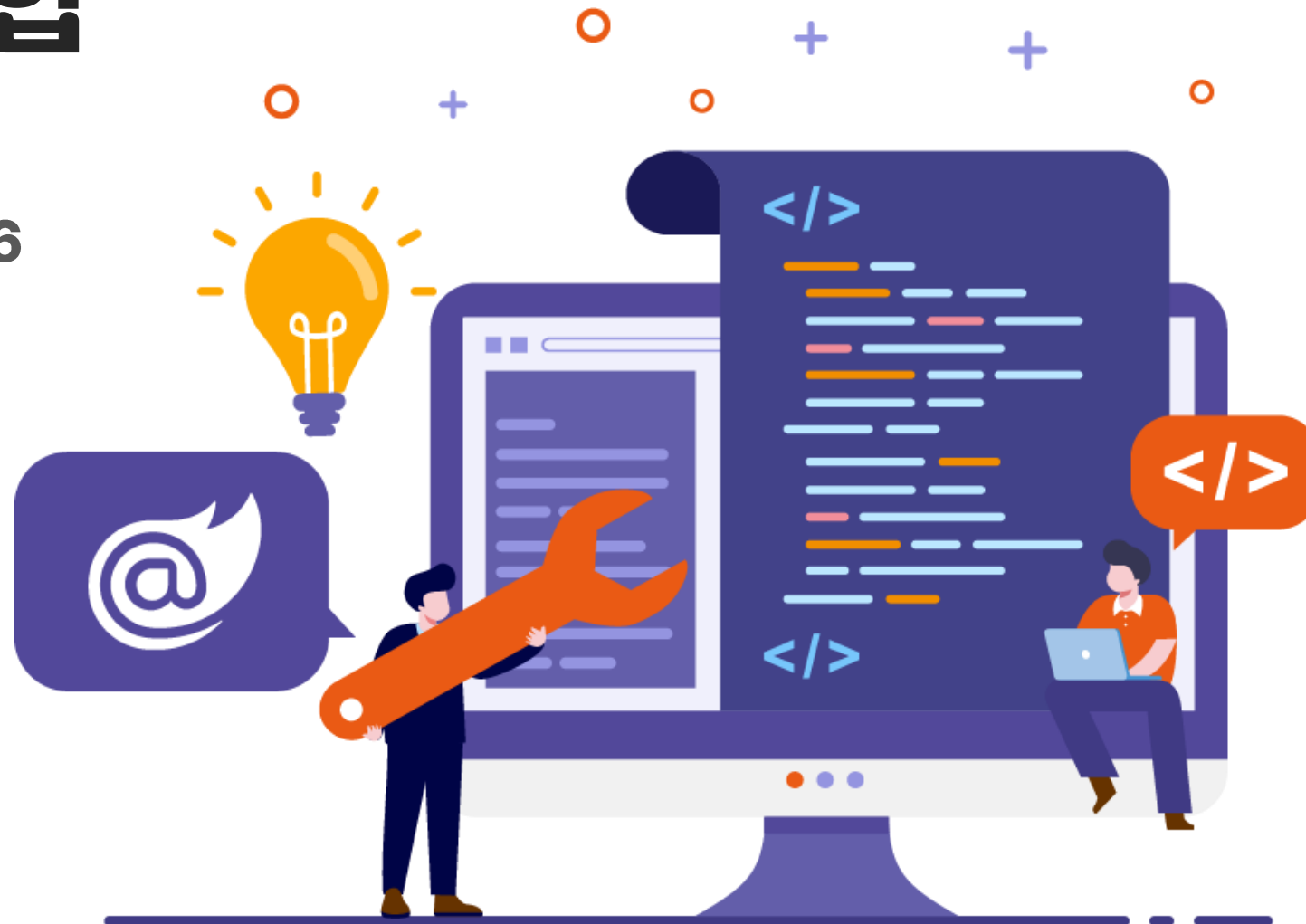
## BLAZOR 밋업

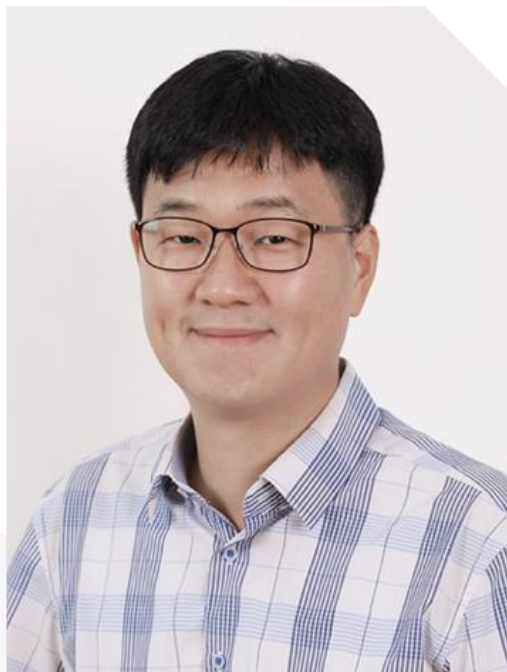
Wi-Fi : Blueport\_Edu

PWD : BLUEport#\$56



### Sessions

- Follow-up
- PWA랑 친해져 보자





## Speaker

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

# Follow-up

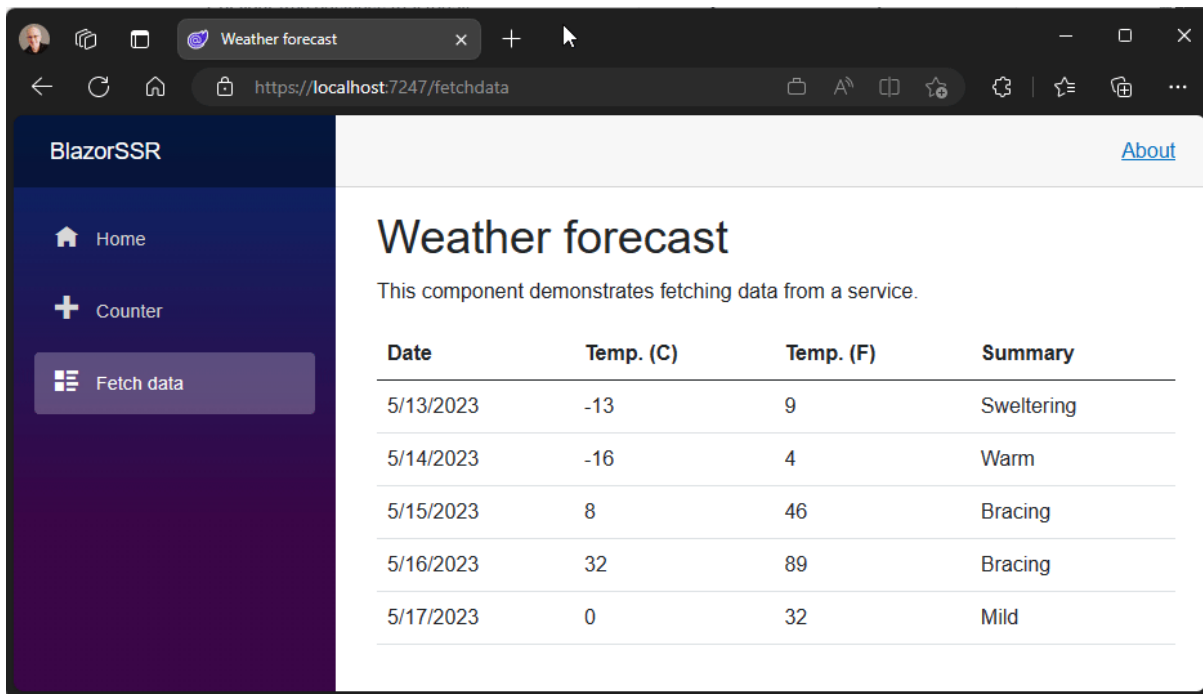
## .NET 8 Preview 4

### .NET8 preview 4



- Streaming rendering with Blazor components
- Handling form posts with Blazor SSR
- Route to named elements in Blazor
- Webcil packaging for Blazor WebAssembly apps

## Streaming rendering with Blazor components



```
@page "/fetchdata"
@using BlazorSSR.Data
@inject WeatherForecastService ForecastService
@attribute [StreamRendering(true)]

<PageTitle>Weather forecast</PageTitle>

<h1>Weather forecast</h1>

@if (forecasts is null)
{
    <p><em>Loading...</em></p>
}
else
{
    // Render weather forecasts
}

@code {
    private string message;

    protected override async Task OnInitializedAsync()
    {
        forecasts = await ForecastService.GetForecastAsync(DateOnly.FromDateTime(DateTime.Now));
    }
}
```

## .NET 8 Preview 4

### Handling form posts with Blazor SSR

```
@inject FormDataProvider FormData

<EditForm method="POST" Model="exampleModel" OnValidSubmit="HandleSubmit">
    <InputText @bind-Value="exampleModel.Name" />
    <button type="submit">Submit</button>
</EditForm>

@code {
    ExampleModel exampleModel = new();

    protected override void OnInitialized()
    {
        // Manually model bind the form data using the FormDataProvider service
        if (FormData.Entries.TryGetValue("Name", out var nameValues))
        {
            exampleModel.Name = nameValues.First();
        }
    }

    void HandleSubmit()
    {
        // Handle the submitted form data
    }

    public class ExampleModel
    {
        public string? Name { get; set; }
    }
}
```

- CascadingModelBinder 사용
- 폼 데이터의 POST 처리

### Route to named elements in Blazor

<https://devblogs.microsoft.com/dotnet/asp-net-core-updates-in-dotnet-8-preview-4/#blazor>

### Webcil packaging for Blazor WebAssembly apps

- WebAssembly format, EMA-355 공통언어 기반
- 백신 등의 소프트웨어가 WebAssembly의 다운로드를 규제하지 않도록



2023.05 Blazor 밋업

# .NET LIVE 2023 Spring

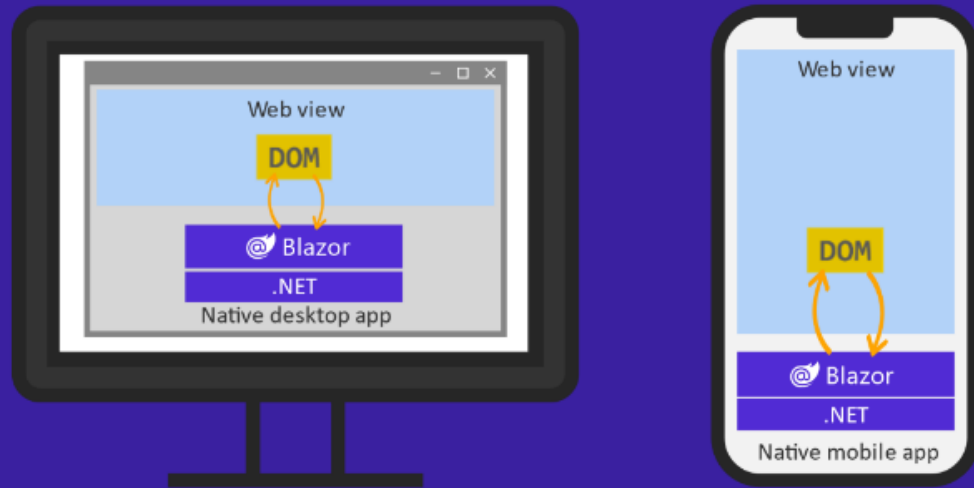


- 5/18 13:00 ~ 18:00 @ MARU180
- Everywhere .NET



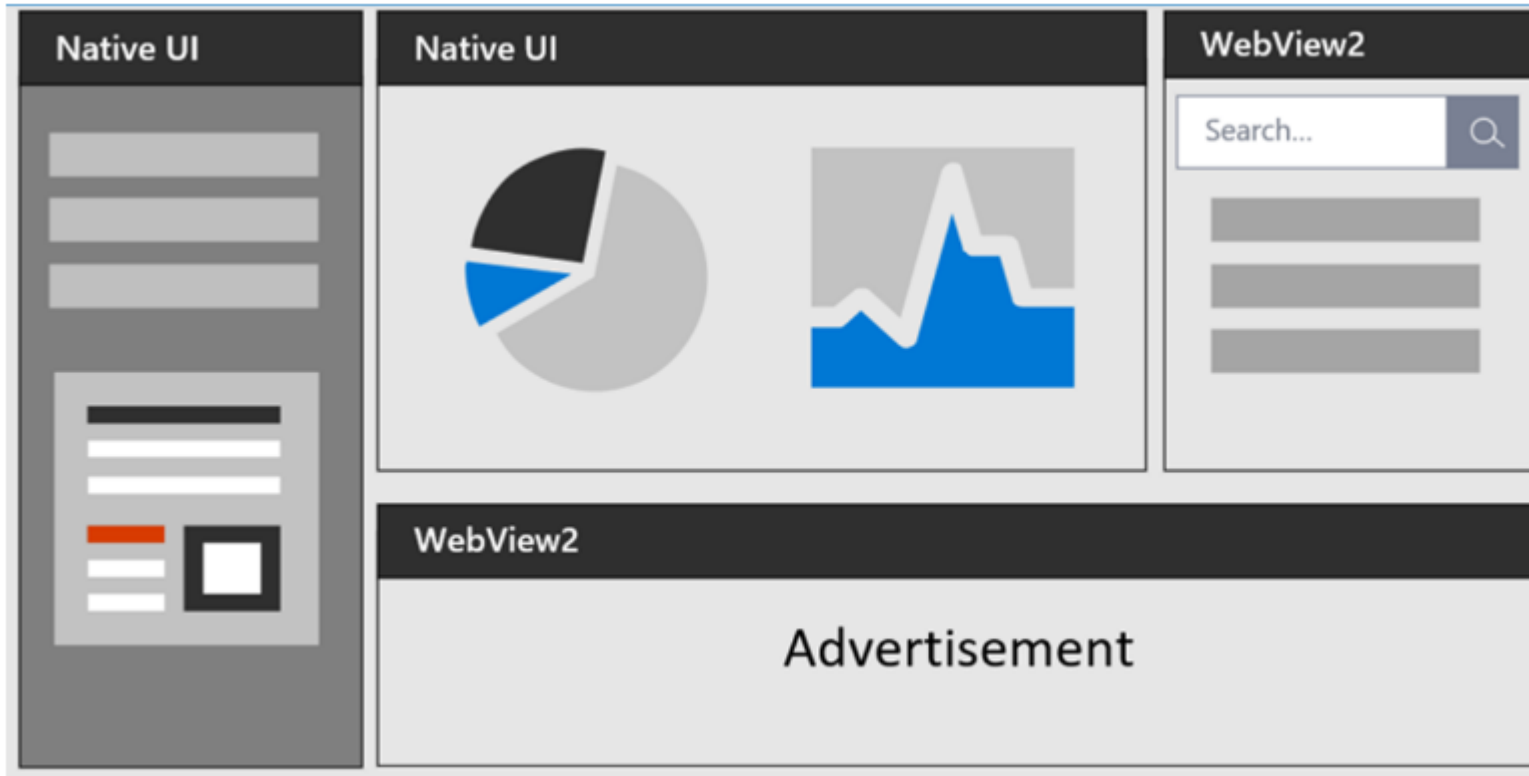
# Blazor Hybrid

## Hybrid apps with .NET & Blazor



- .NET MAUI
- WPF
- Windows Forms

# Blazor Hybrid



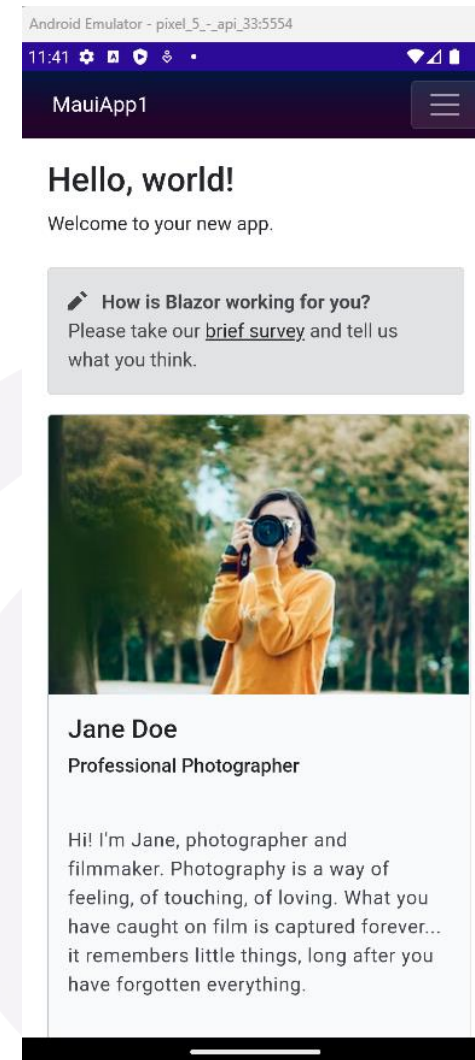
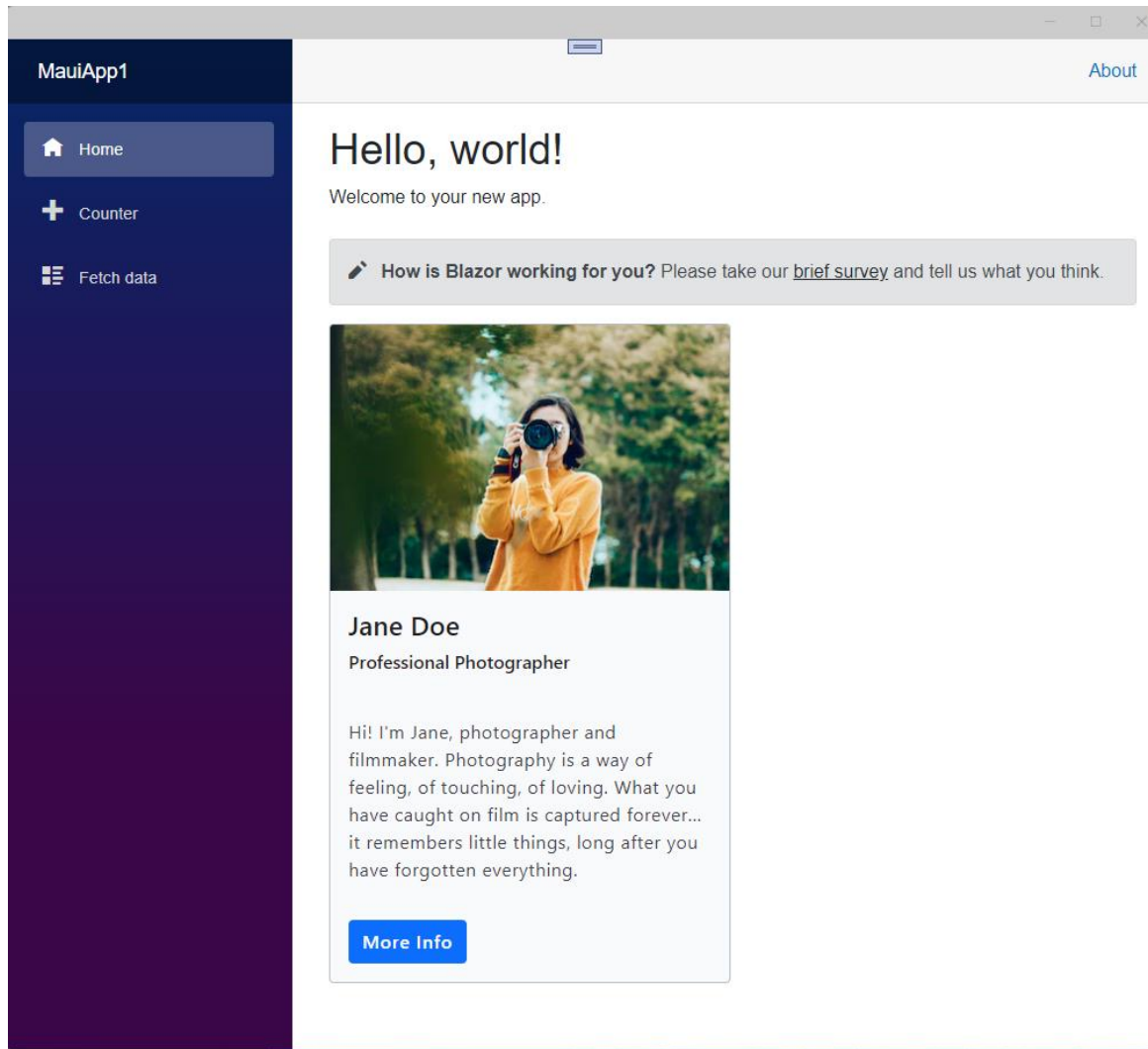
- WebView2
- BlazorWebView

In a Blazor Hybrid app, [Razor components](#) run natively on the device. Components render to an embedded Web View control through a local interop channel. Components don't run in the browser, and WebAssembly isn't involved. Razor components load and execute code quickly, and components have full access to the native capabilities of the device through the .NET platform. Component styles rendered in a Web View are platform dependent and may require you to account for rendering differences across platforms using custom stylesheets.

<https://learn.microsoft.com/en-us/aspnet/core/blazor/hybrid/?view=aspnetcore-7.0>

<https://learn.microsoft.com/en-us/microsoft-edge/webview2/>

# Blazor Hybrid



# Azure AD B2C

## Azure AD B2C



- 카카오로 로그인? 네이버로 로그인?
- 오늘은 메일 계정으로 로그인

<https://learn.microsoft.com/ko-kr/azure/active-directory-b2c/overview>

# Azure AD B2C

readme.md

## Azure Active Directory B2C: Custom CIAM User Journeys

In this repo, you will find samples for several enhanced Azure AD B2C Custom CIAM User Journeys.

### Getting started

- See our [Custom Policy overview](#).
- See our [Azure AD B2C Wiki articles](#) to help walkthrough the custom policy components.
- See our [Custom Policy Schema reference](#).

### Prerequisites

- You will require to [create an Azure AD B2C directory](#).
- You can automate the prerequisites (where applicable) by using our using automated tool called [Deploy AAD B2C Custom Policies](#) if you already have an Azure AD B2C tenant.

### Sample scenarios

Samples are available for the following categories

- [Password Management](#)
- [General Security](#)
- [User Experience](#)
- [Terms of Use / Consent](#)
- [Passwordless](#)
- [Multi Factor](#)
- [Account Linking](#)
- [Identity Providers](#)
- [User Interface](#)
- [Data Residency](#)
- [User Migration](#)
- [Endpoints](#)
- [Web Test](#)
- [CI / CD](#)

Report repository

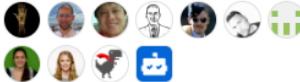
Releases

No releases published

Packages


No packages published

Contributors 54




+ 43 contributors

Environments 1

 github-pages Active

Languages



|            |       |            |       |
|------------|-------|------------|-------|
| HTML       | 44.8% | C#         | 41.8% |
| JavaScript | 5.7%  | PowerShell | 4.3%  |
| CSS        | 3.3%  | Batchfile  | 0.1%  |

# Azure AD B2C

[samples / policies / sign-in-with-kakao / README.md](#)

chadhasbrook Accessibility updates b6b27ec · 2 years ago History

Preview Code Blame 101 lines (72 loc) · 5.79 KB

## Setting up Sign in with Kakao as a Custom OAuth2 provider

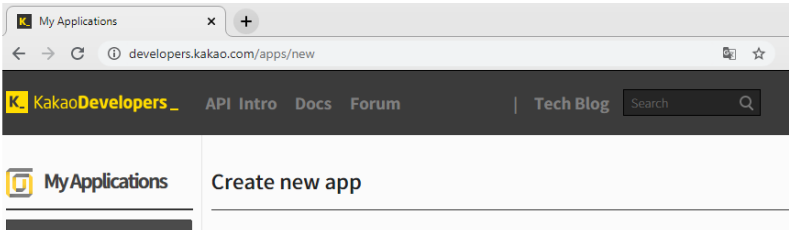
This sample shows how to setup **Kakao** as an identity provider in Azure AD B2C. **Kakao** is a South Korean Internet company that provides a diverse set of services.

There is no Kakao page in the official documentation, but the documentation for [setting up sign-in with Google using custom policies](#) shows you the important steps in creating a custom policy that integrates with an OAuth2 provider. The following documentation will help you registering your app with Kakao and do the custom policy modifications that are unique for it.

### Kakao Developer Platform steps

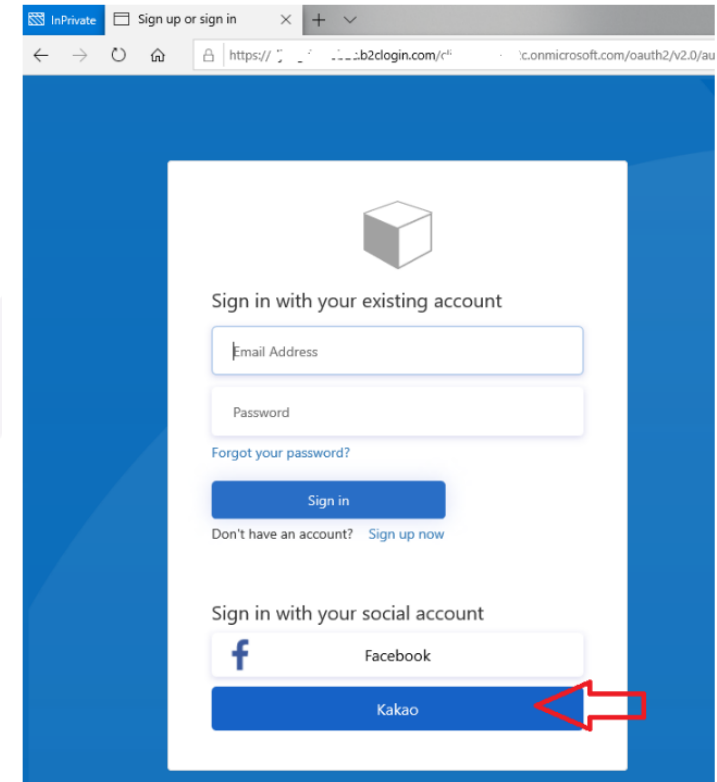
You need to [create your account at KakaoDevelopers\\_](#) first and then create your app registration:

1. Select **+Create new app** in the drop down menu and give your app a name.



## Signing in with Kakao

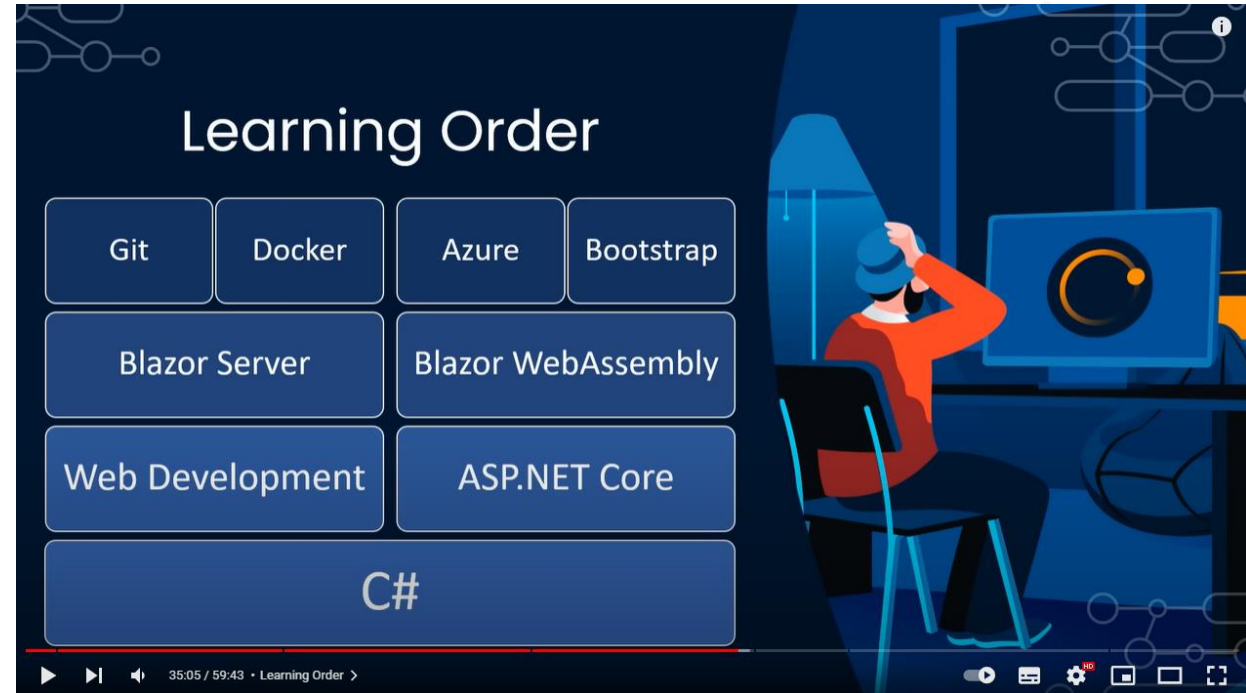
When you test run your policy, you should see a button labeled **Kakao**. You can do some UX customization to get the real icon for the button.



- 구독 필요
- 추가 확인 필요



# HOW TO LEARN BLAZOR IN 2023



<https://www.youtube.com/watch?v=GlcAADkInRo>

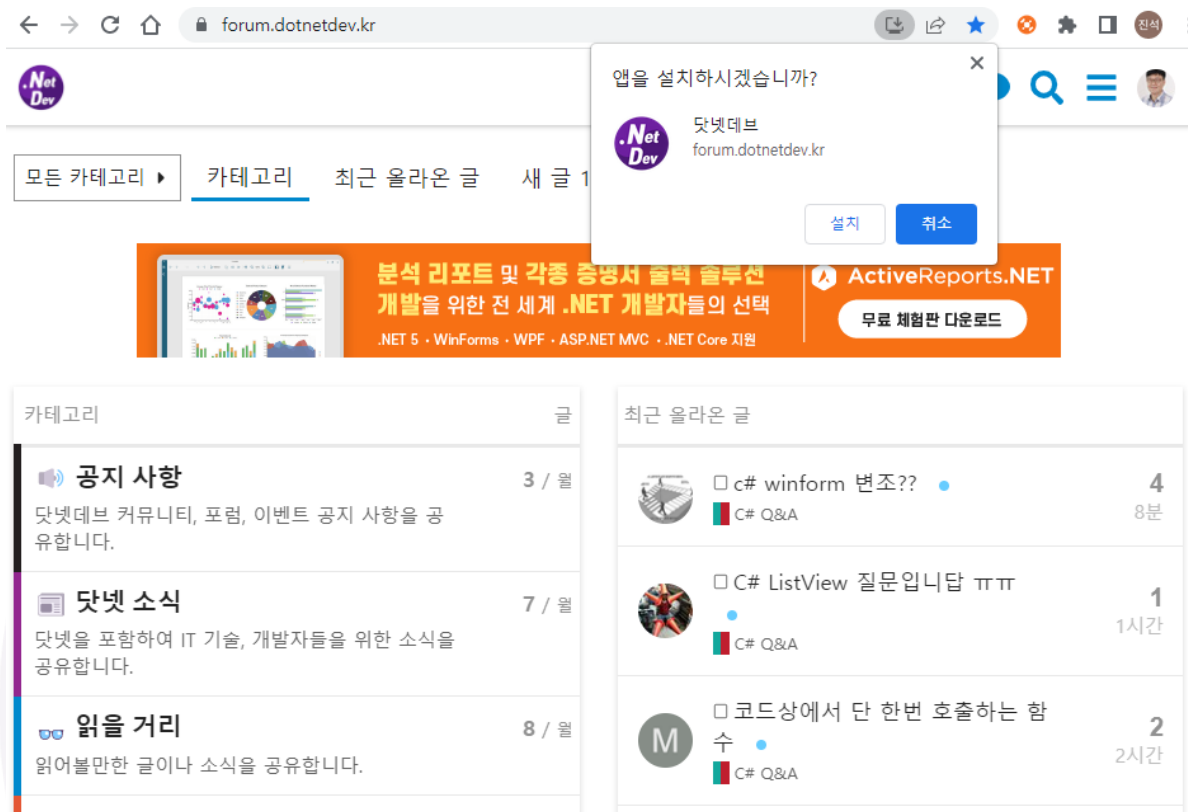
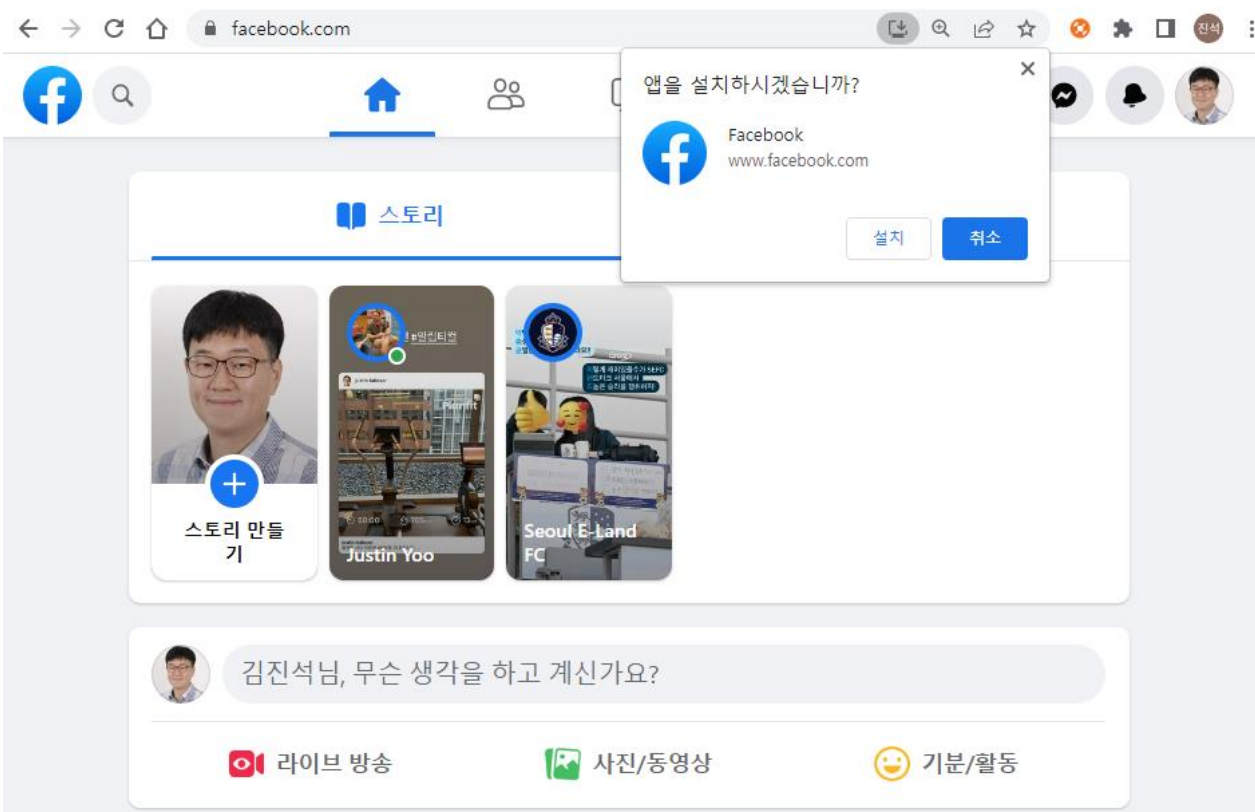


*break  
time*

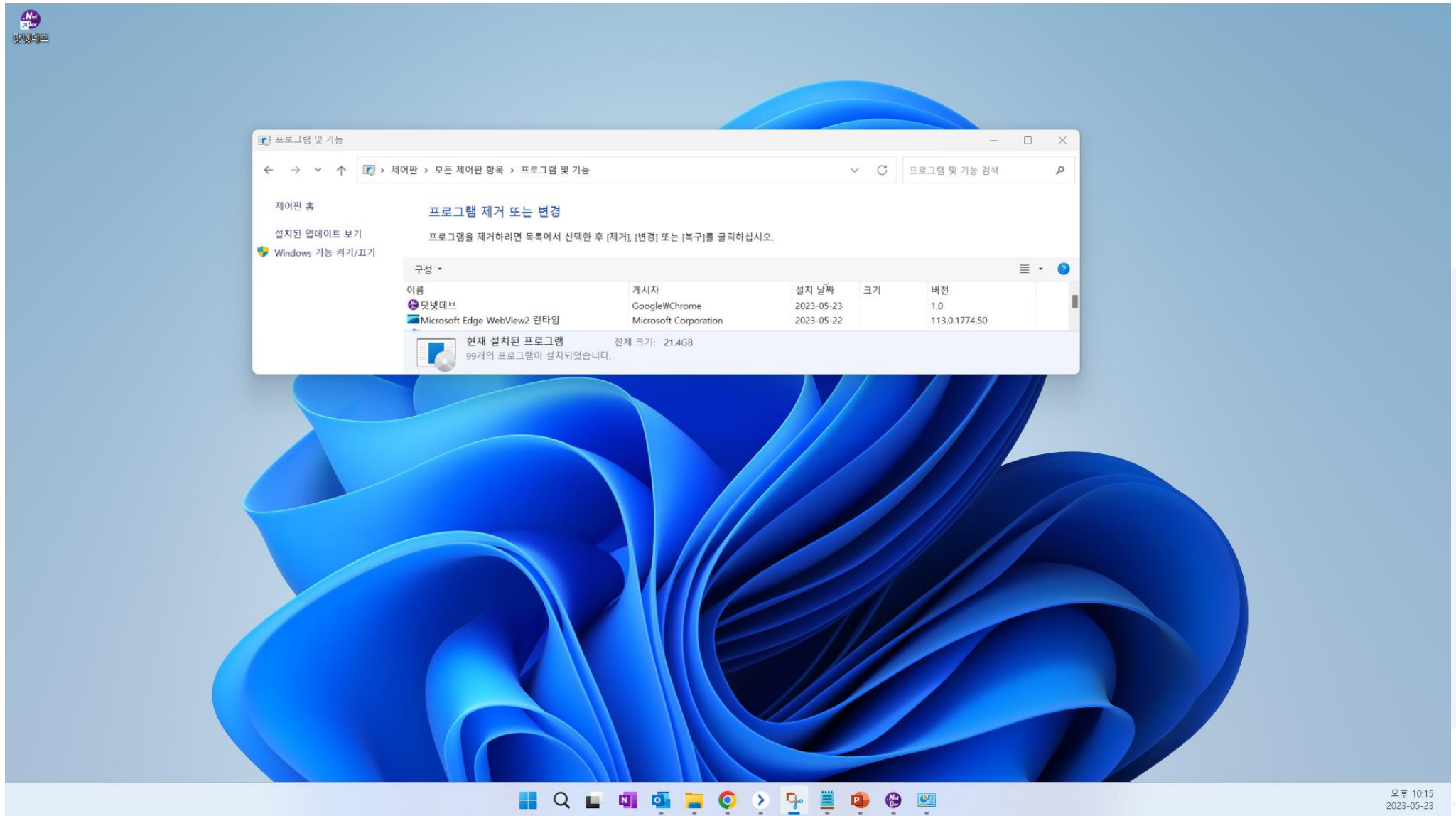
오늘의 도전!

PWA랑 친해져 보자

# PWA랑 친해져 보자



# PWA랑 친해져 보자



## PWA랑 친해져 보자

# Progressive Web Apps

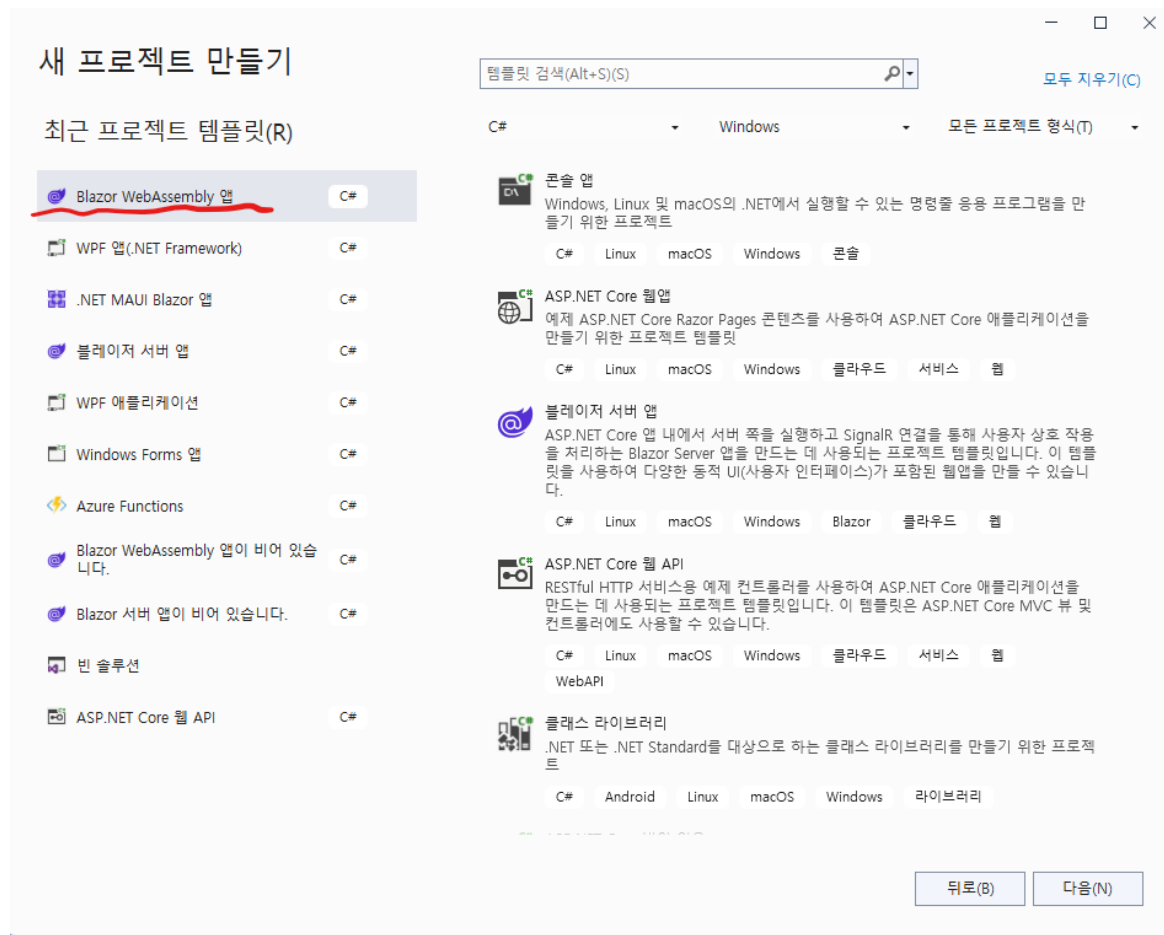
- 프로그레시브 웹 앱스
  - 구글 크롬 엔지니어 알렉스 러셀이 2015년 고안한 개념
  - 앱 수준으로 점진적으로 발전해 나가는 웹을 지향
  - 궁극적으로 앱 수준과 같은 사용자 경험을 제공
- 특징
  - 진화 : 브라우저에 상관없이
  - 반응형 : 디바이스에 상관없이
  - 연결 독립적 : 오프라인 지원
  - 앱과 유사
  - 링크 연결 : URL로 공유
  - 최신상태
  - 안전 : HTTPS로 제공
  - 검색 가능
  - 재참여 가능 : 푸시 알림
  - 설치 가능 : 앱스토어 등록이 필요 없음



# PWA랑 친해져 보자

일단 만들어 보자

# PWA랑 친해져 보자



- Blazor WebAssembly 프로젝트 선택

# PWA랑 친해져 보자

새 프로젝트 구성

Blazor WebAssembly 앱 C# Linux macOS Windows Blazor 클라우드 웹

프로젝트 이름(I)  
BlazorApp5

위치(L)  
C:\dev\test

솔루션 이름(M) ⓘ  
BlazorApp5

☐ 솔루션 및 프로젝트를 같은 디렉터리에 배치(D)

"C:\dev\test\BlazorApp5\BlazorApp5\"에 프로젝트이(가) 만들어집니다.

뒤로(B) 다음(N)

- 프로젝트 이름 설정

# PWA랑 친해져 보자

추가 정보

Blazor WebAssembly 앱 C# Linux macOS Windows Blazor 클라우드 웹

프레임워크(F) ⓘ  
.NET 7.0 (표준 용어 지원)

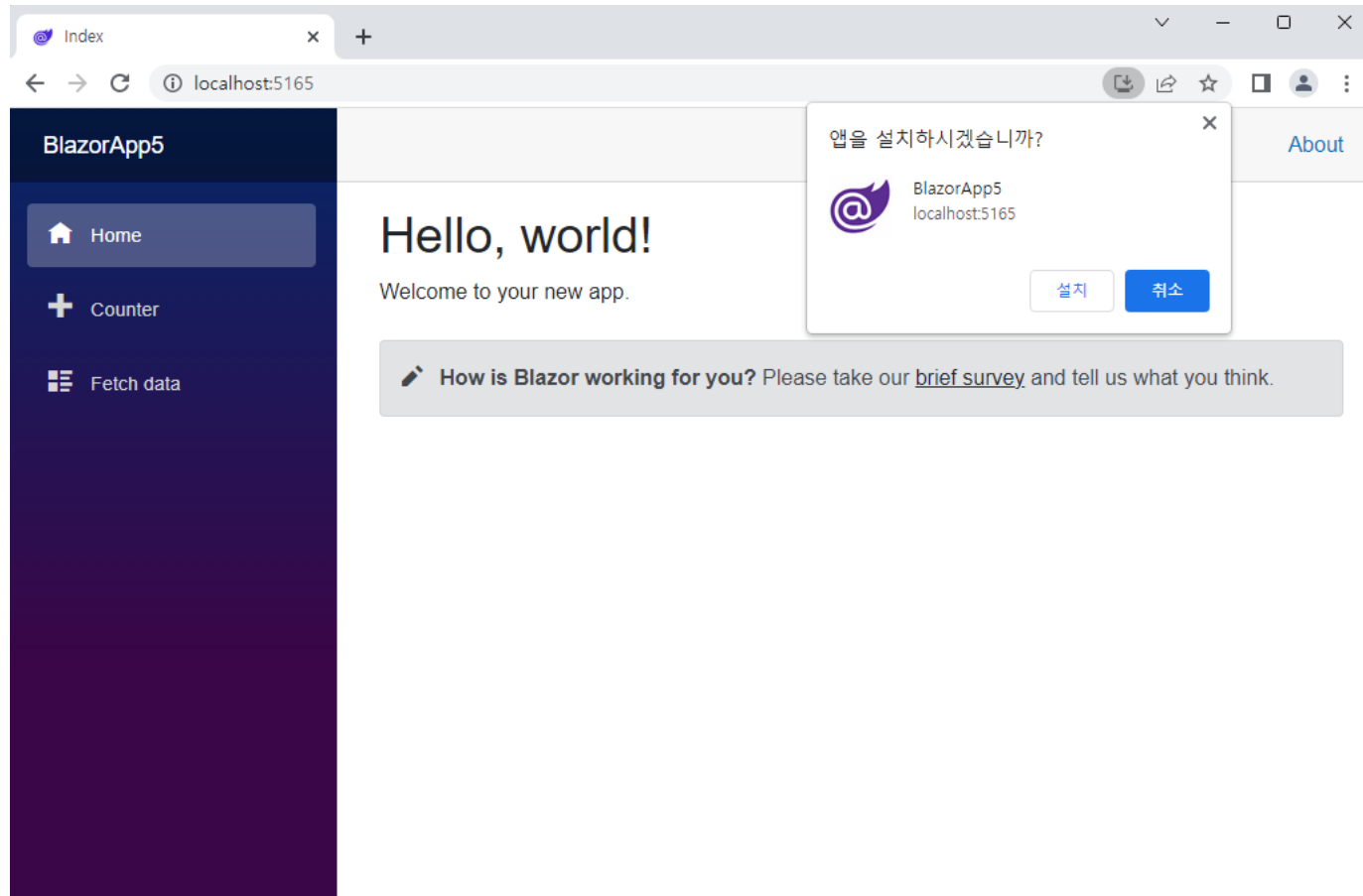
인증 유형(A) ⓘ  
없음

☐ HTTPS에 대한 구성(H) ⓘ  
☐ ASP.NET Core 호스팅(H) ⓘ  
☒ 프로그레시브 웹 애플리케이션(P) ⓘ  
☐ 최상위 문 사용 안 함(T) ⓘ

뒤로(B) 만들기(C)

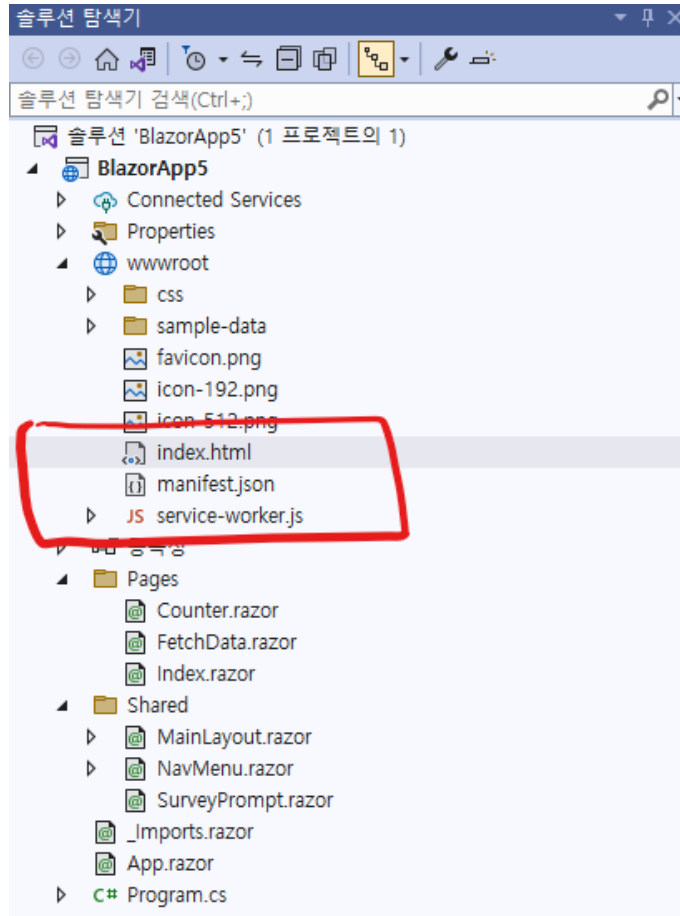
- 프로그레시브 웹 애플리케이션 선택

# PWA랑 친해져 보자



- 끝? 참 쉽죠~잉.

## PWA랑 친해져 보자



- index.html
- manifest.json
- service-worker.js



# PWA랑 친해져 보자

```
index.html  X
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>BlazorApp5</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="BlazorApp5.styles.css" rel="stylesheet" />
13     <link href="manifest.json" rel="manifest" />
14     <link rel="apple-touch-icon" sizes="512x512" href="icon-512.png" />
15     <link rel="apple-touch-icon" sizes="192x192" href="icon-192.png" />
16 </head>
17
18 <body>
19     <div id="app">
20         <svg class="loading-progress">
21             <circle r="40%" cx="50%" cy="50%" />
22             <circle r="40%" cx="50%" cy="50%" />
23         </svg>
24         <div class="loading-progress-text"></div>
25     </div>
26
27     <div id="blazor-error-ui">
28         An unhandled error has occurred.
29         <a href="" class="reload">Reload</a>
30         <a class="dismiss">✕</a>
31     </div>
32     <script src="_framework/blazor.webassembly.js"></script>
33     <script>navigator.serviceWorker.register('service-worker.js');</script>
34 </body>
35
36 </html>
37
```

- manifest.json
- navigator.serviceWorker.register('service-worker.js');

## PWA랑 친해져 보자

```
manifest.json ✕
스키마: https://json.schemastore.org/foxxx-manifest.json
1  {
2    "name": "BlazorApp5",
3    "short_name": "BlazorApp5",
4    "start_url": "./",
5    "display": "standalone",
6    "background_color": "#ffffff",
7    "theme_color": "#03173d",
8    "prefer_related_applications": false,
9    "icons": [
10     {
11       "src": "icon-512.png",
12       "type": "image/png",
13       "sizes": "512x512"
14     },
15     {
16       "src": "icon-192.png",
17       "type": "image/png",
18       "sizes": "192x192"
19     }
20   ]
21 }
22
```

### - PWA의 설치와 앱 구성정보

- name : 설치 배너에 표시되는 이름
- short\_name : 아이콘의 이름으로 표시
- icons : 아이콘
- start\_url : 시작 화면

## PWA랑 친해져 보자

service-worker.js

기타

<global>

```
1 // In development, always fetch from the network and do not enable offline support.  
2 // This is because caching would make development more difficult (changes would not  
3 // be reflected on the first load after each change).  
4 self.addEventListener('fetch', () => { });  
5
```

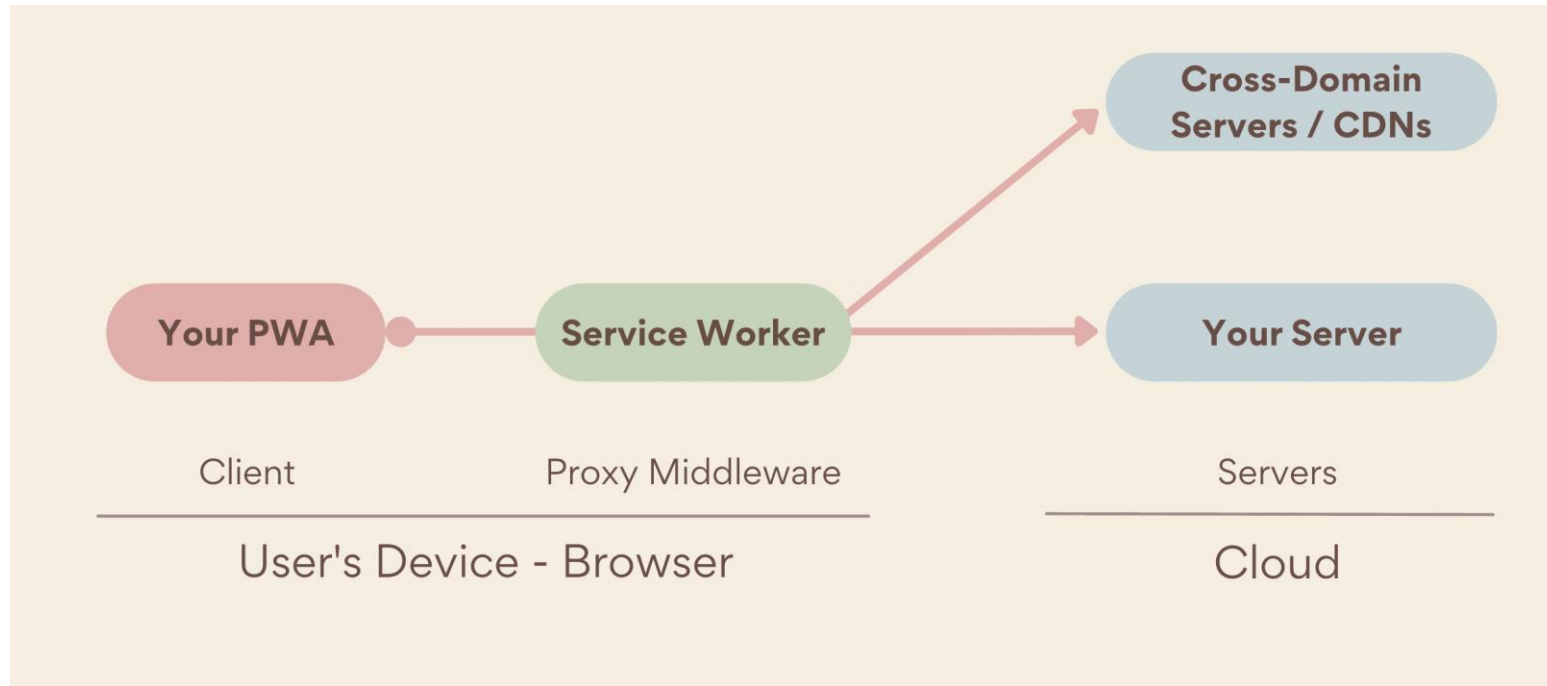
- service-worker.js
  - fetch event
  - 빈 함수?

## PWA랑 친해져 보자

정말 쉽죠잉??? 진짜???

## PWA랑 친해져 보자

### Service Worker



- 웹페이지와는 완전히 별개
- 프록시 서버의 역할
- 오프라인에서도 서비스 사용할 수 있게

## Service Worker, what are you ?



**Mariko Kosaka**

3.1K Followers

Knitting some Javascript at Scripto : Co-organizer @brooklyn\_js

me : “Service Worker, what are you?”

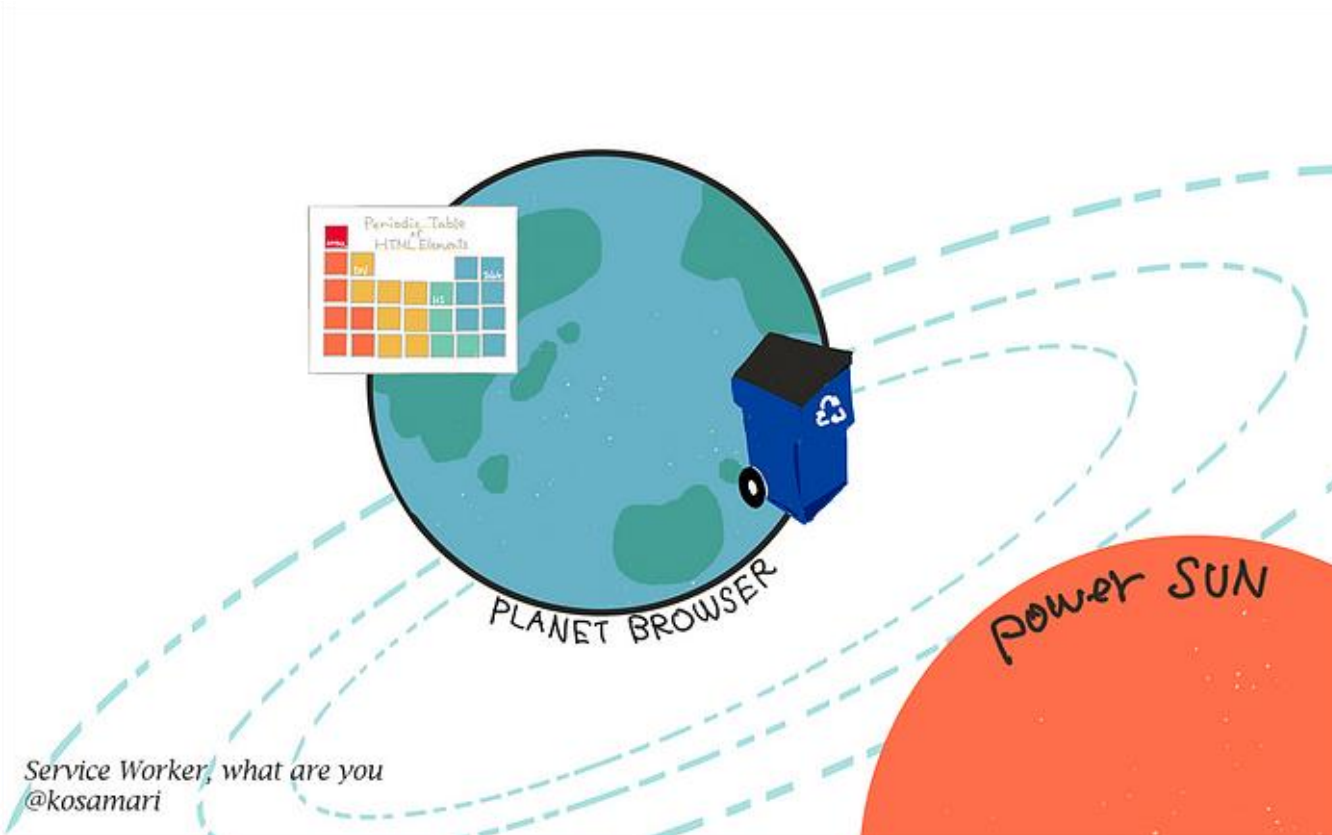
Service Worker: “I’m a programmable network proxy.”

me : “... what?”



## PWA랑 친해져 보자

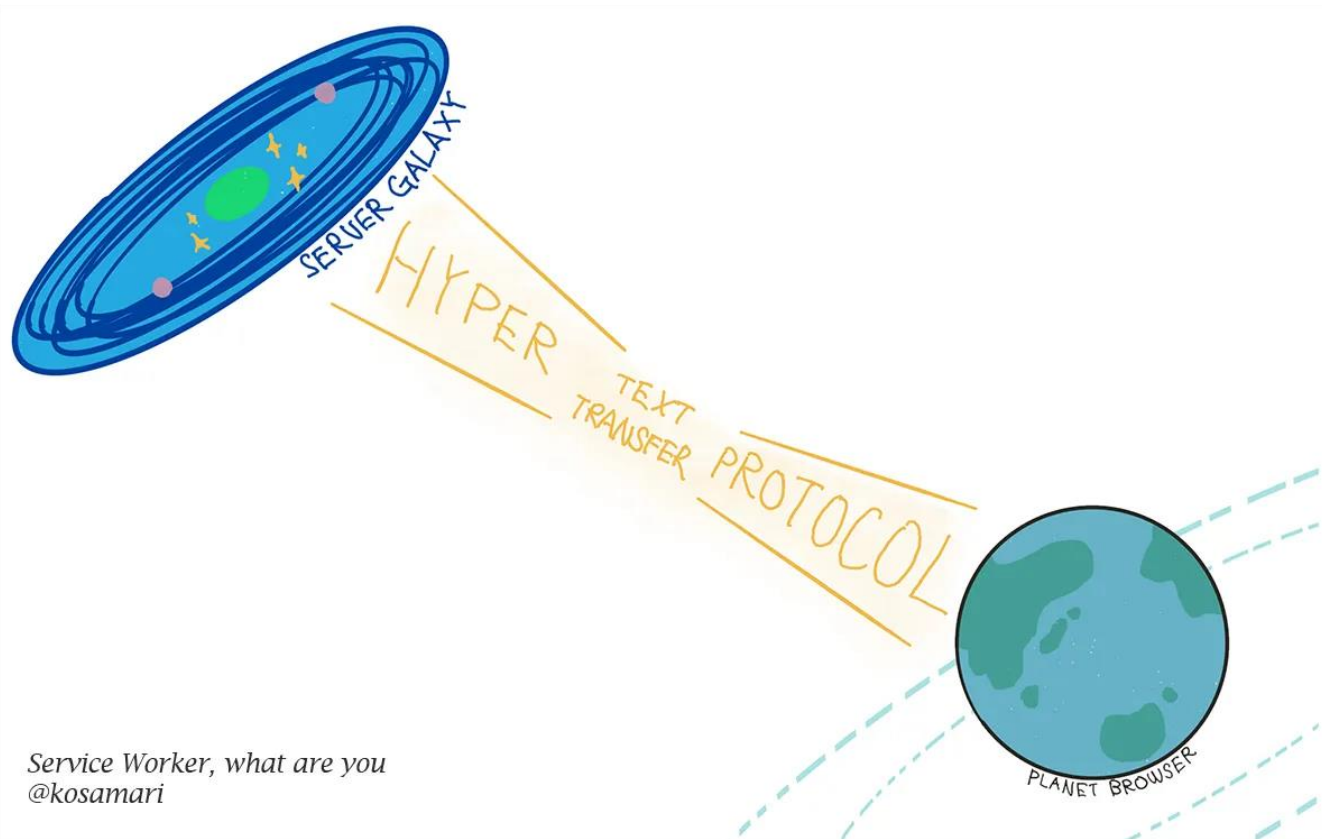
### Service Worker, what are you ?



- 태초에...
- HTML, CSS, JavaScript
- DOM, garbage collection

## PWA랑 친해져 보자

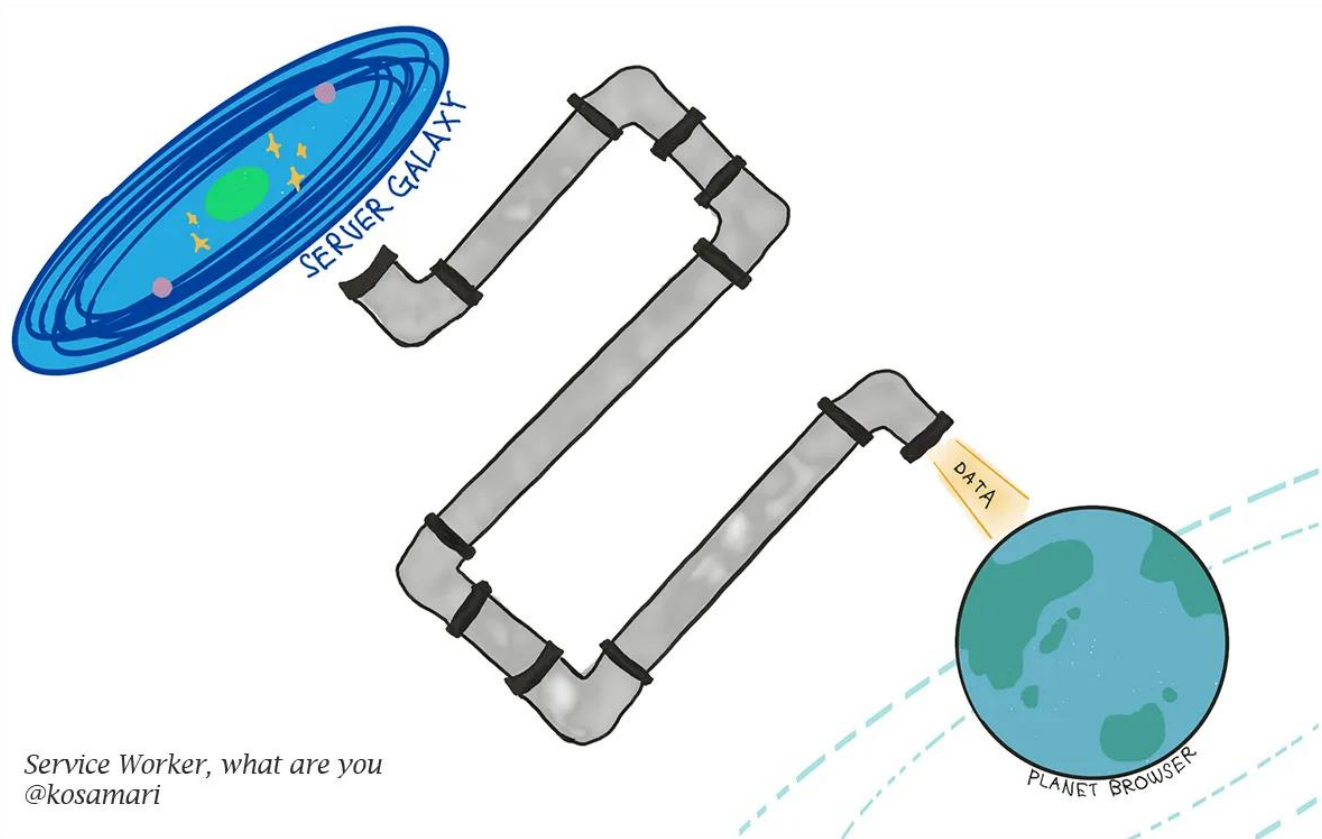
### Service Worker, what are you ?



- 이세계(Server)와의 연결은...
- Hyper Text Transfer Protocol(HTTP)

## PWA랑 친해져 보자

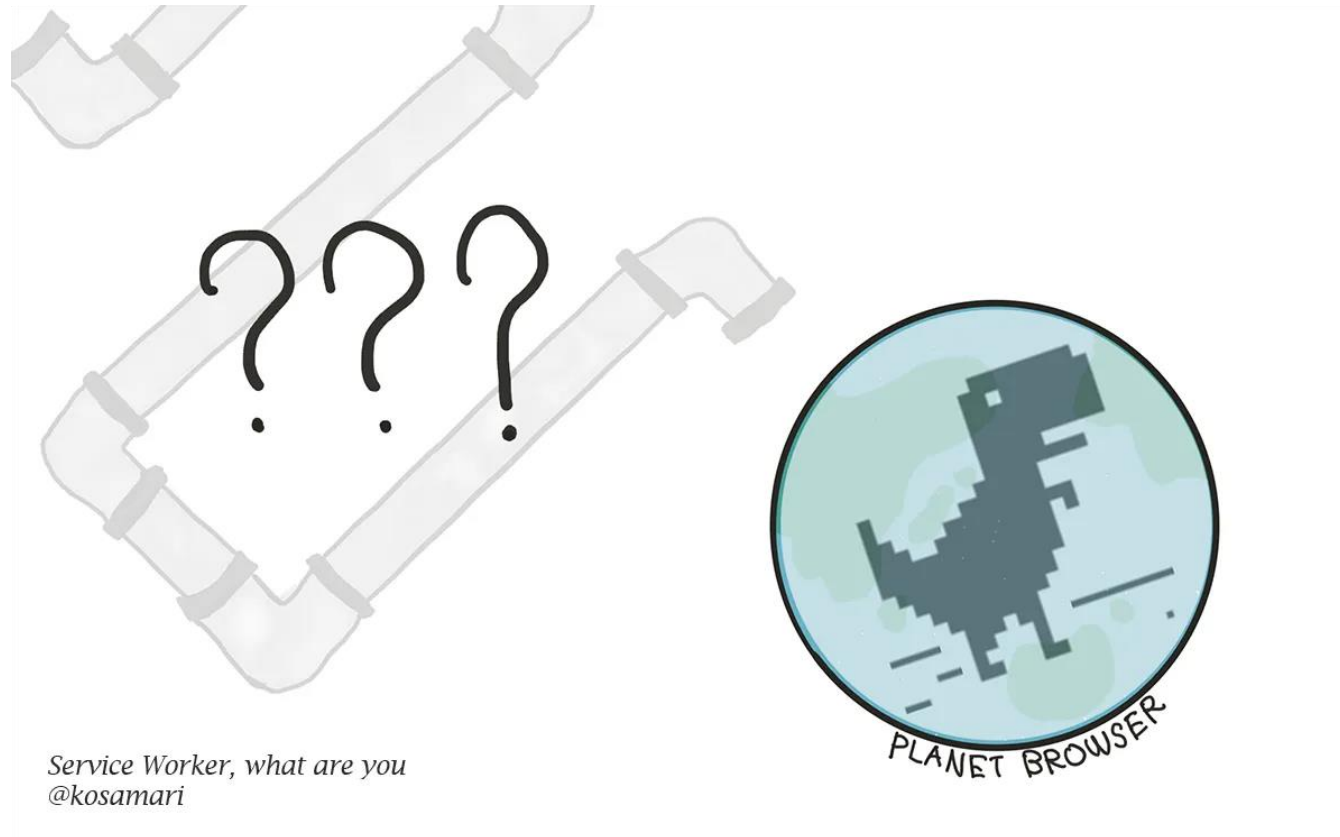
### Service Worker, what are you ?



- 이세계에서 오는 정보의 속도는...
  - 데이터 파이프라인(ISP)의 크기나 거리에 따라
- ~~\* 개인적으로 콧, 사이판 좋아하지 않습니다.~~

## PWA랑 친해져 보자

### Service Worker, what are you ?



Service Worker, what are you  
@kosamari

- 데이터 파이프라인이 망가지면..?
- 우리 행성의 운명은...?

## PWA랑 친해져 보자

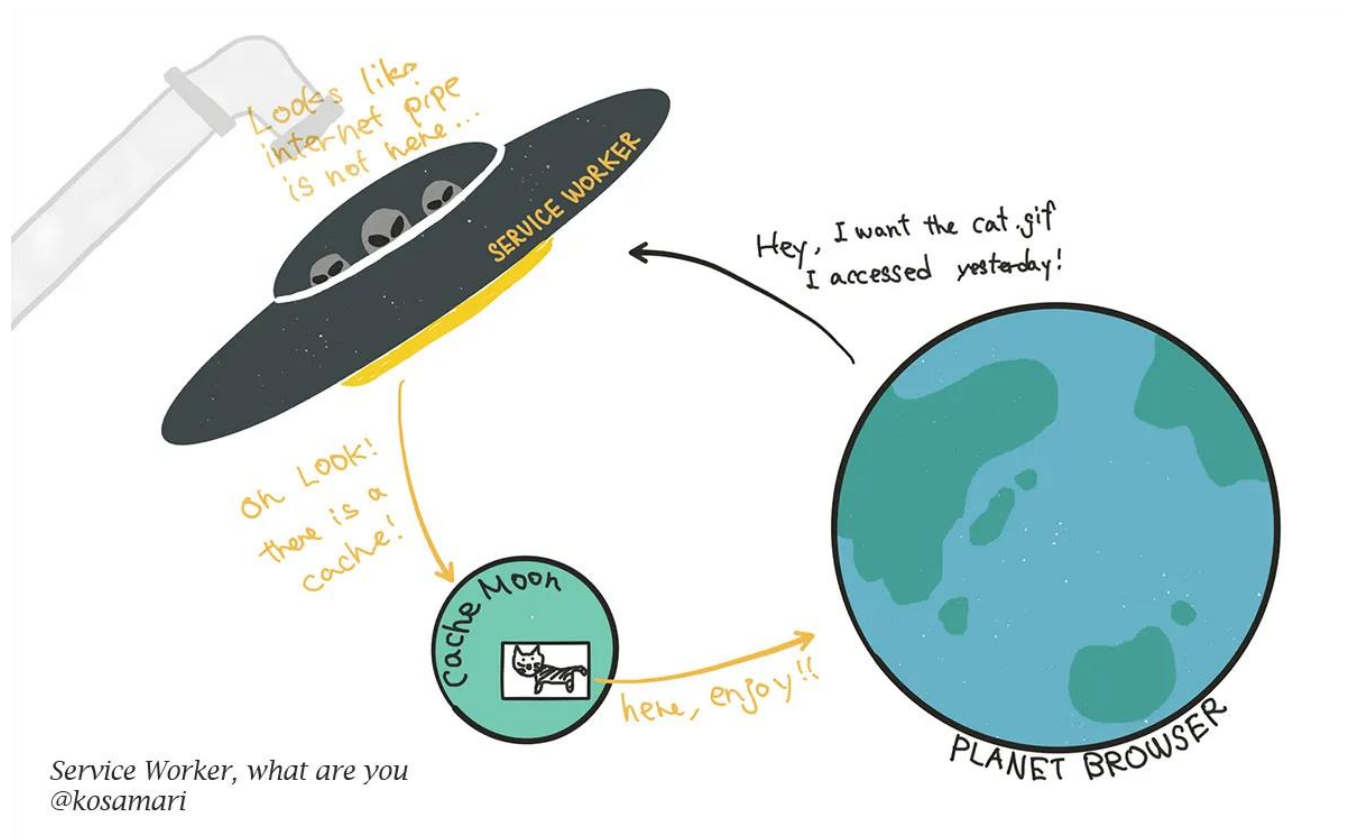
### Service Worker, what are you ?



- 외계인(Service Worker)에게 도움 요청!
  - 직접 불러야 합니다(서비스 등록)
  - 지구인이 아닙니다(웹페이지와 수명이 다름)
  - 지구에 간섭할 수 없습니다(DOM제어불가)
  - 이세계(Server) 통신은 외계인을 통해서 합니다

## PWA랑 친해져 보자

### Service Worker, what are you ?

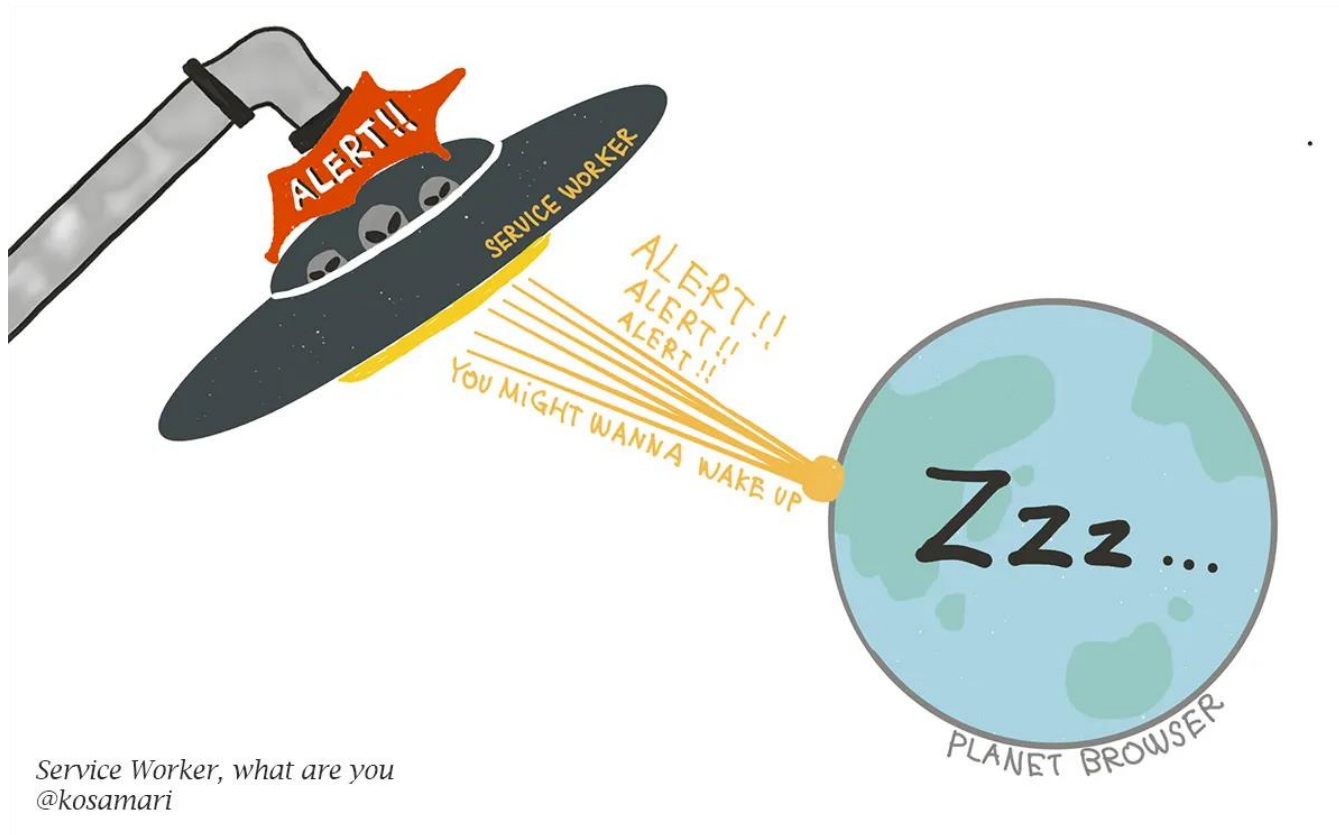


- Interact with Cache



## PWA랑 친해져 보자

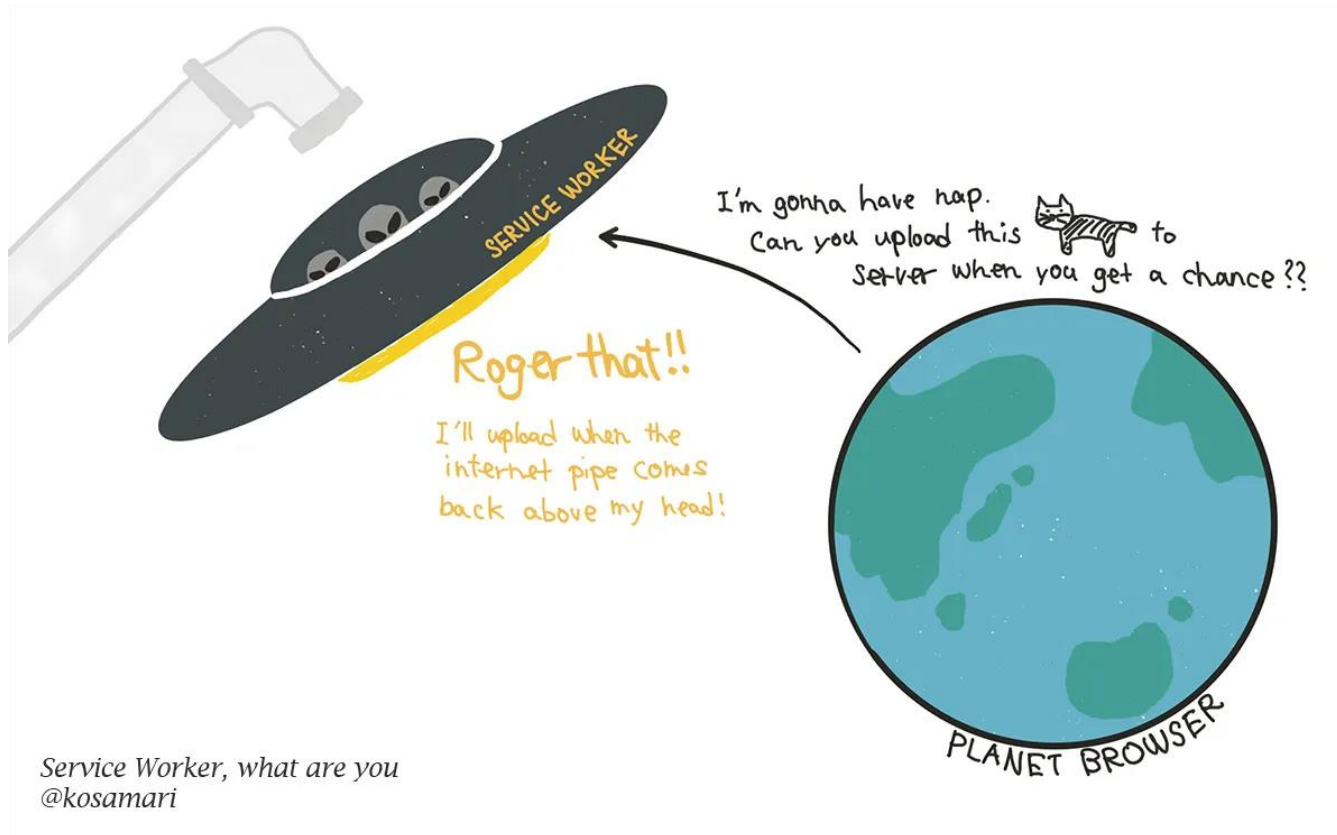
### Service Worker, what are you ?



- Send Push Notification

## PWA랑 친해져 보자

### Service Worker, what are you ?



- Run Background Sync



## PWA랑 친해져 보자

### Service Worker, what are you ?



**Mariko Kosaka**

3.1K Followers

Knitting some Javascript at Scripto : Co-organizer @brooklyn\_js

If you were confused about Service Worker, I hope this shed some light :)  
Want to start dive into code ? I recommencement checking out Jake Archibald's [offline-cookbook](https://jakearchibald.com/2014/offline-cookbook/) for more code driven samples.

<https://jakearchibald.com/2014/offline-cookbook/>

<https://medium.com/@kosamari/service-worker-what-are-you-ca0f8df92b65>

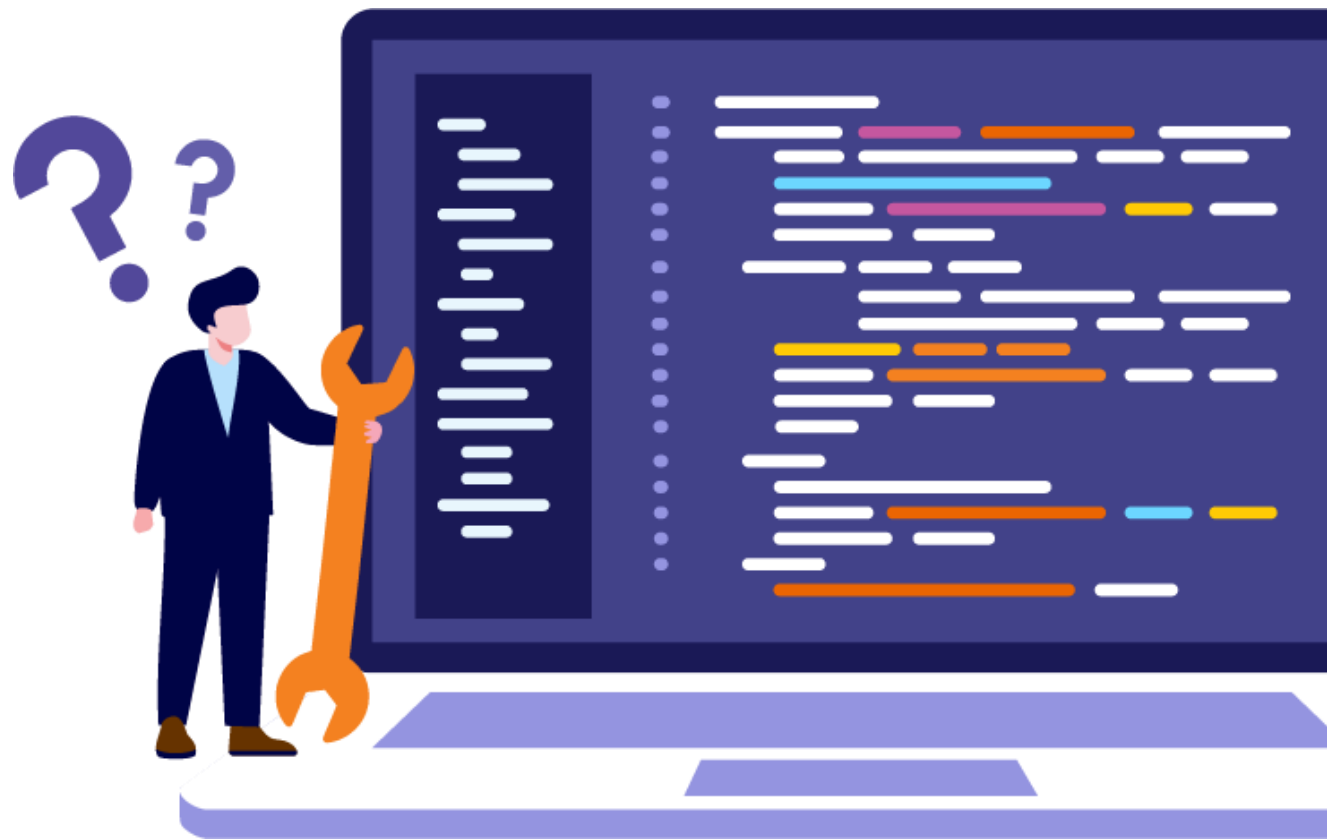
## Q&A

### Blazor Korea User Group

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

### GitHub

<https://github.com/blazor-meetup>



**다음 달(6월) 일정**

**6월 22일(목)**

**THANK YOU**