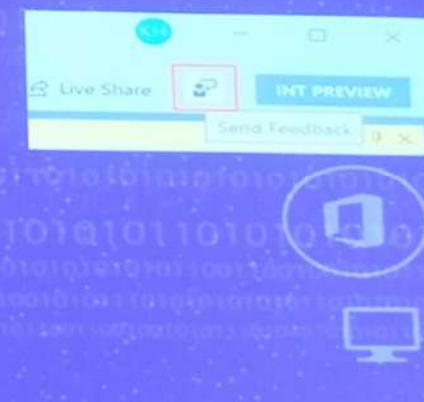


Resources

File bugs on
<https://developercommunity.visualstudio.com>
or directly from Visual Studio!

Docs: <https://aka.ms/refactor>
Our open source GitHub repo:
<https://aka.ms/roslyn>

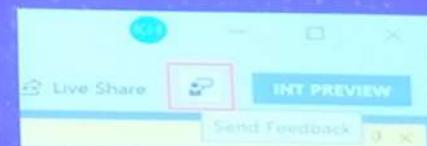


Resources

File bugs on
<https://developercommunity.visualstudio.com>
or directly from Visual Studio!

Docs: <https://aka.ms/refactor>

Our open source GitHub repo:
<https://aka.ms/roslyn>

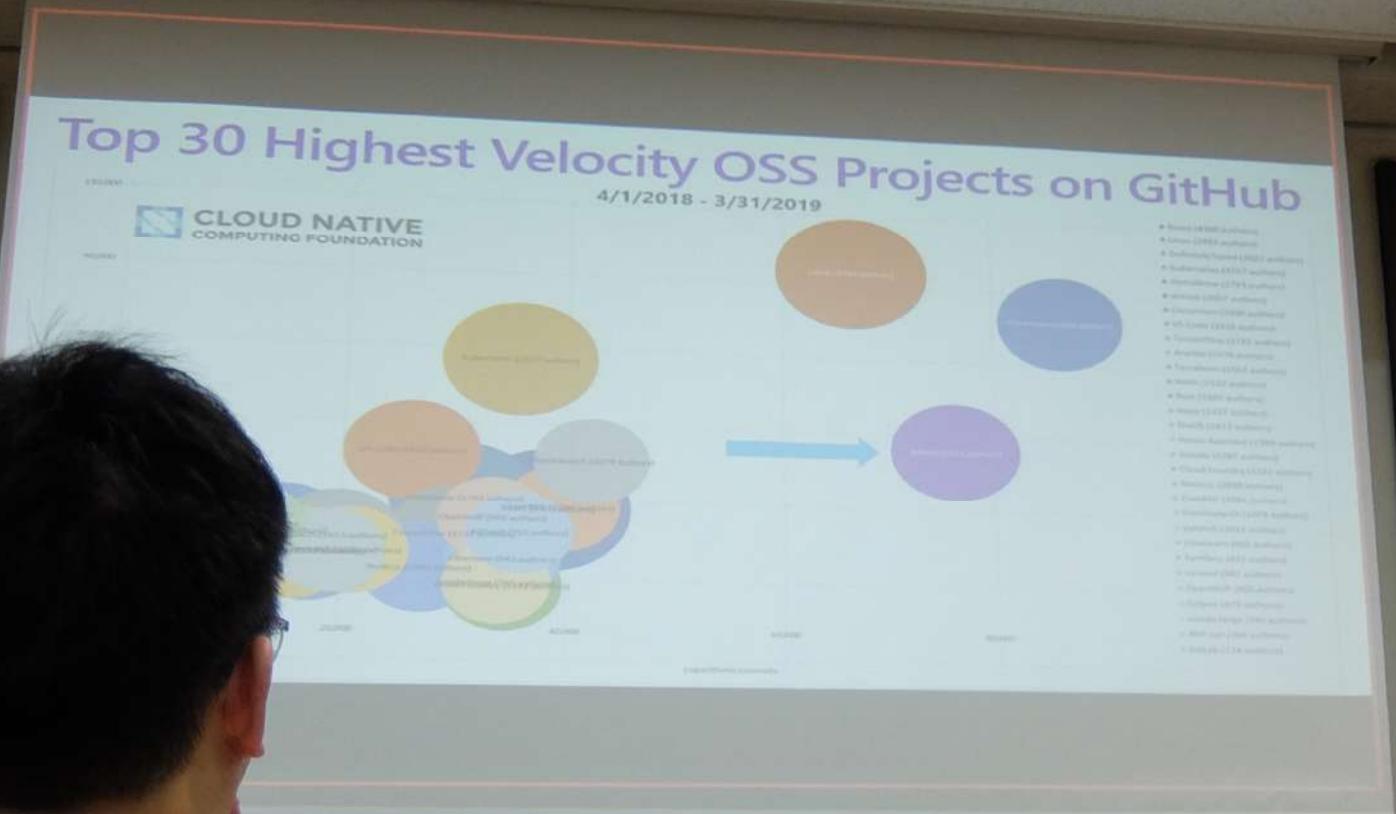


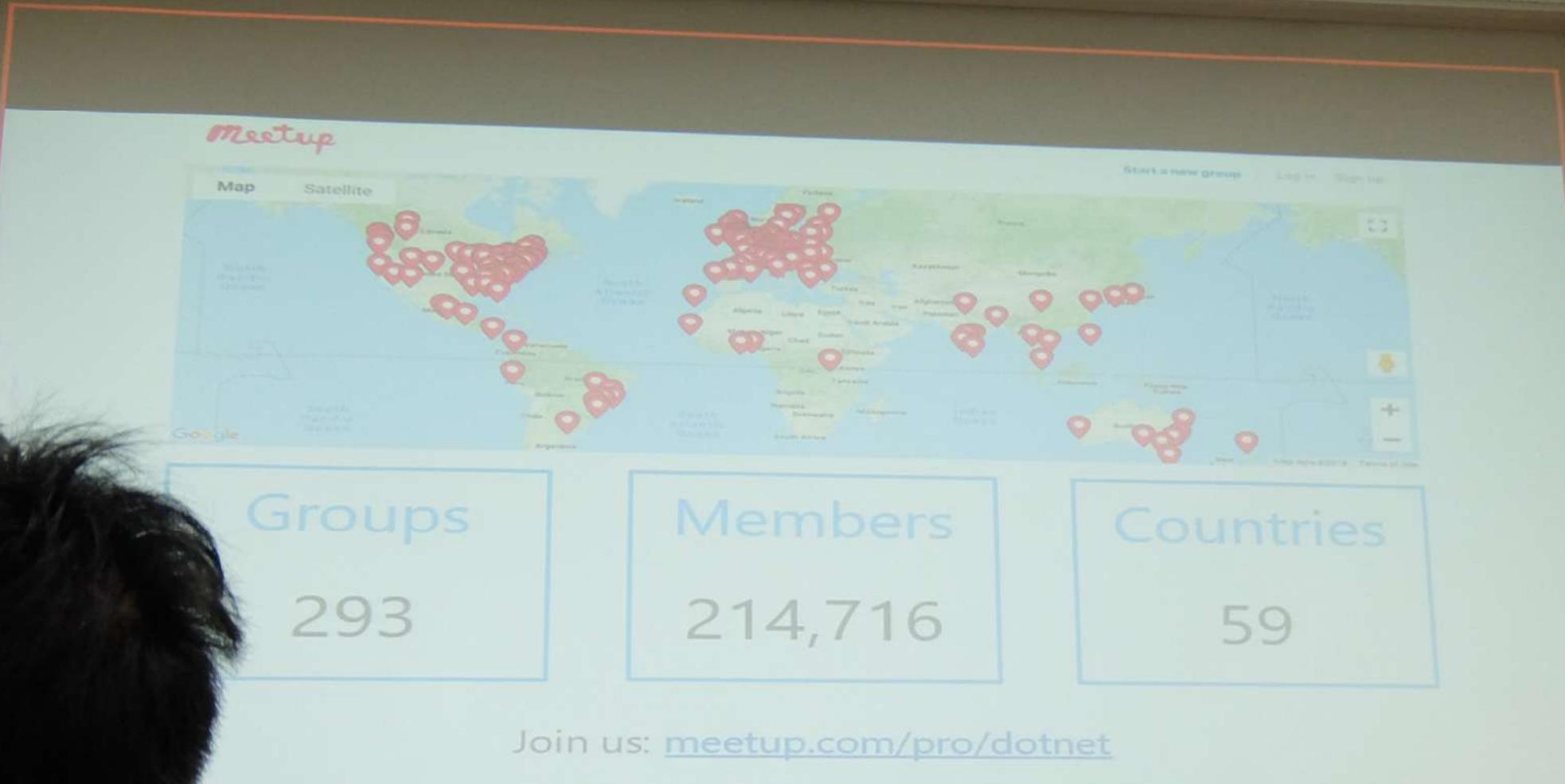
Resources

- Code
<https://aka.ms/keynote/weatherapp>
- HTTP Repl
<https://aka.ms/dotnetcore/httprepl>
- EndPoint Routing
<https://aka.ms/endpointrouting>
- HealthChecks
<https://aka.ms/healthchecks>
- Cloud Native Show – <https://aka.ms/cloudnativeshow>
- ProductivityFeatures - <https://aka.ms/productivityfeature>

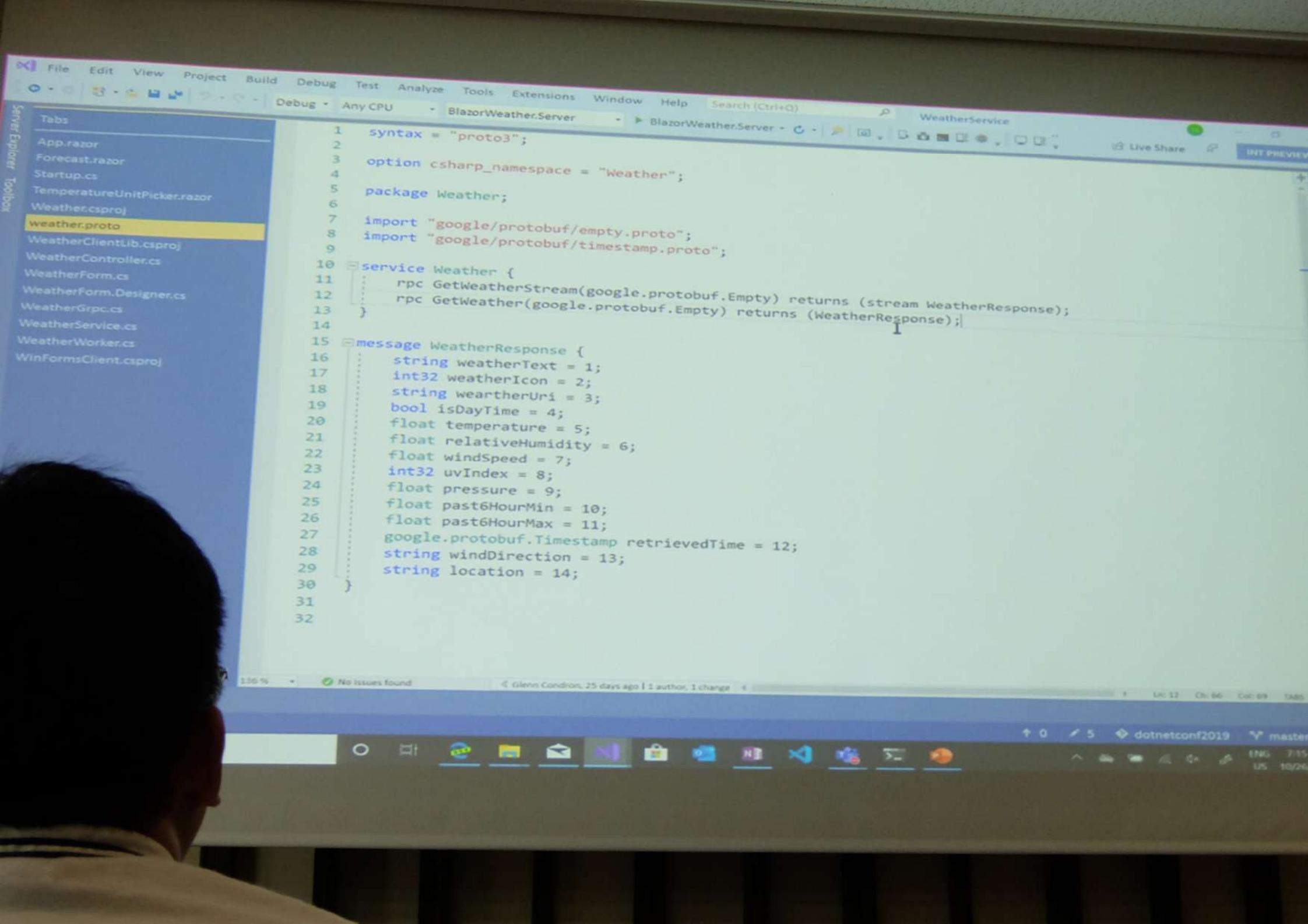
Resources

- Code
<https://aka.ms/keynote/weatherapp>
- HTTP Repl
<https://aka.ms/dotnetcore/httprepl>
- EndPoint Routing
<https://aka.ms/endpointrouting>
- HealthChecks
<https://aka.ms/healthchecks>
- Cloud Native Show – <https://aka.ms/cloudnativeshow>
ProductivityFeatures - <https://aka.ms/productivityfeatures>





```
1 <Project Sdk="Microsoft.NET.Sdk.Web">
2
3   <PropertyGroup>
4     <TargetFramework>netcoreapp3.0</TargetFramework>
5     <UserSecretsId>3ee6d0f8-011d-4eb7-8f82-c4fc8f1475b</UserSecretsId>
6     <DockerDefaultTargetOS>Linux</DockerDefaultTargetOS>
7   </PropertyGroup>
8
9   <ItemGroup>
10    <Protobuf Include="Protos\weather.proto" GrpcServices="Server" />
11  </ItemGroup>
12
13  <ItemGroup>
14    <PackageReference Include="Swashbuckle.AspNetCore" Version="5.0.0-rc2" />
15    <PackageReference Include="Grpc.AspNetCore" Version="2.23.1" />
16    <PackageReference Include="Microsoft.VisualStudio.Azure.Containers.Tools.Targets" Version="1.9.5" />
17  </ItemGroup>
18
19 </Project>
20
```

A photograph of a person's head in profile, facing a computer monitor. The monitor displays a screenshot of a Microsoft Visual Studio IDE window. The title bar shows "BlazorWeather.Server" and "WeatherService". The code editor contains a .proto file named "weather.proto". The code defines a service "Weather" with two RPC methods: "GetWeatherStream" and "GetWeather". It also defines a message "WeatherResponse" with various fields like weatherText, weatherIcon, and retrievedTime. The code is color-coded, and the "weather.proto" tab is highlighted in yellow. The bottom status bar of the IDE shows "No issues found" and other standard build statistics.

```
1 syntax = "proto3";
2
3 option csharp_namespace = "Weather";
4
5 package Weather;
6
7 import "google/protobuf/empty.proto";
8 import "google/protobuf/timestamp.proto";
9
10 service Weather {
11     rpc GetWeatherStream(google.protobuf.Empty) returns (stream WeatherResponse);
12     rpc GetWeather(google.protobuf.Empty) returns (WeatherResponse);
13 }
14
15 message WeatherResponse {
16     string weatherText = 1;
17     int32 weatherIcon = 2;
18     string weartherUri = 3;
19     bool isDayTime = 4;
20     float temperature = 5;
21     float relativeHumidity = 6;
22     float windSpeed = 7;
23     int32 uvIndex = 8;
24     float pressure = 9;
25     float past6HourMin = 10;
26     float past6HourMax = 11;
27     google.protobuf.Timestamp retrievedTime = 12;
28     string windDirection = 13;
29     string location = 14;
30 }
31
32
```

Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) BlazorWeatherServer BlazorWeatherServer WeatherService Live Share INT PREVIEW

Weather

```
23 // For more information on how to configure your application, visit https://go.microsoft.com/fwlink/?linkid=868599
24 - references | Glenn Condon, 25 days ago | 1 author, 1 change | 0 exceptions
25 public void ConfigureServices(IServiceCollection services)
26 {
27     services.AddCors(c => c.AddDefaultPolicy(builder =>
28         builder.AllowAnyOrigin());
29 });
30 services.AddGrpc();
31 services.AddHttpClient("AccuWeather")
32     .ConfigurePrimaryHttpMessageHandler(_ => new HttpClientHandler()
33         {
34             AutomaticDecompression = System.Net.DecompressionMethods.GZip | System.Net.Decompr
35         });
36 services.AddMemoryCache();
37 services.AddControllers();
38 services.AddHostedService<WeatherWorker>();
39
40 services.AddSwaggerGen(c =>
41 {
42     c.SwaggerDoc("v1", new OpenApiInfo { Title = "WeatherApi", Version = "v1" });
43 });
44 }
45
46 // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
47 - references | Glenn Condon, 25 days ago | 1 author, 1 change | 0 exceptions
48 public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
49 {
50     if (env.IsDevelopment())
51     {
52         app.UseDeveloperExceptionPage();
53     }
54 }
```

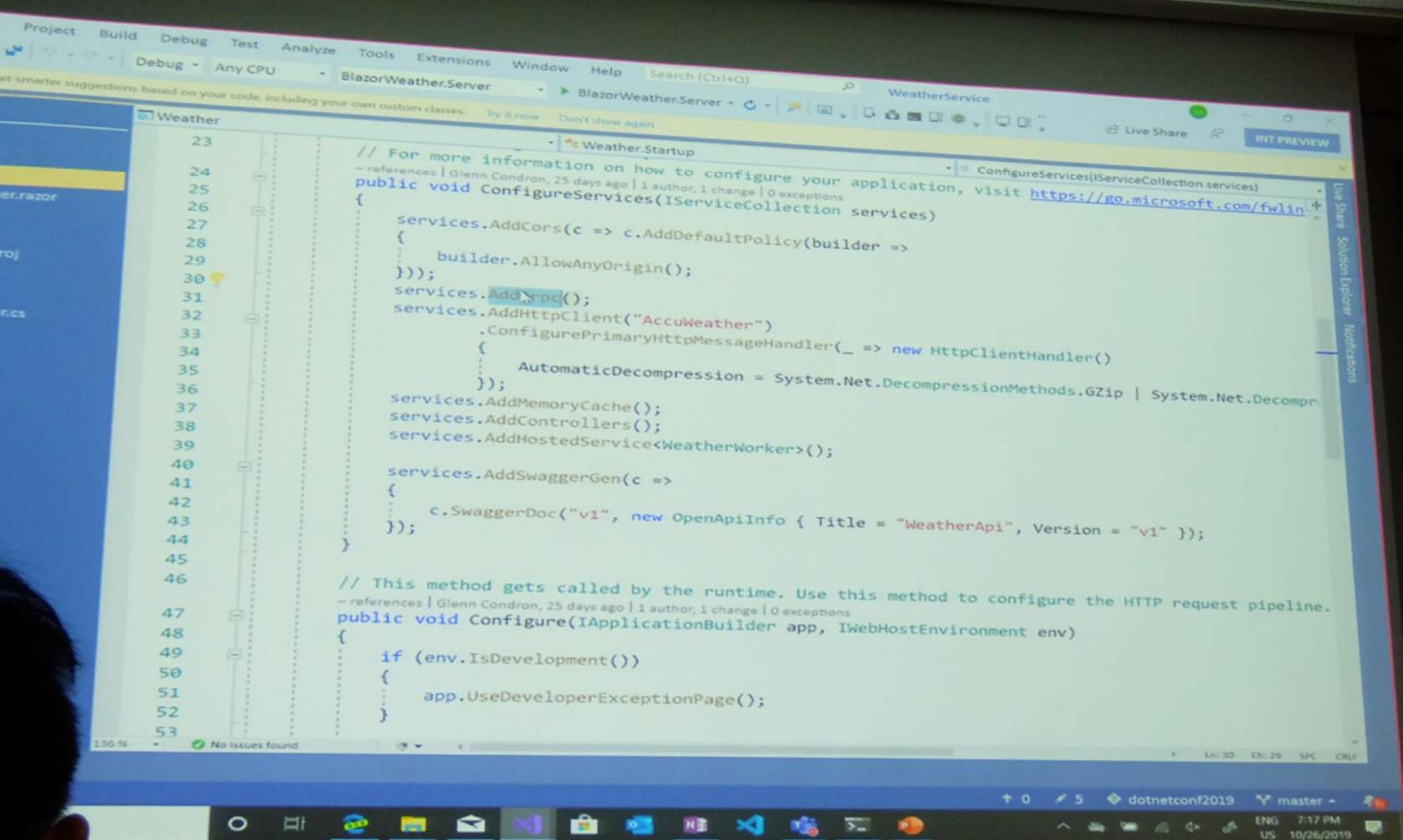
No issues found

Live Share Solution Explorer Notifications



↑ 0 ↻ 5 dotnetconf2019 master

ENG 7:17 PM
US 10/26/2019



Build Debug Test
Debug Any CPU
Presenting...
suggestions based on your code, including your own custom classes. Try it now Don't show again

Give control Stop presenting

Live Share INT PREVIEW

Weather

```
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
```

{
 services.AddCors(c => c.AddDefaultPolicy(builder =>
 builder.AllowAnyOrigin());
 services.AddGrpc();
 services.AddHttpClient("AccuWeather")
 .ConfigurePrimaryHttpMessageHandler(_ => new HttpClientHandler()
 {
 AutomaticDecompression = System.Net.DecompressionMethods.GZip | System.Net.Decompr
 });
 services.AddMemoryCache();
 services.AddControllers();
 services.AddHostedService<WeatherWorker>();
 services.AddSwaggerGen(c =>
 {
 c.SwaggerDoc("v1", new OpenApiInfo { Title = "WeatherApi", Version = "v1" });
 });

 // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
 public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
 {
 if (env.IsDevelopment())
 {
 app.UseDeveloperExceptionPage();
 }

 app.UseRouting();

 app.UseSwagger();
 }
}

No issues found

136 %

Un: 38 Ch: 43 SPC CRLF

↑ 0 ↗ 5 ⚡ dotnetconf2019 ⚡ master ↘ ENG 7:20 PM
US 10/26/2019

Live Share Solution Explorer Notifications

Project Build Debug Test
Debug - Any CPU Presenting... Give control Stop presenting Live Share INT PREVIEW

```
12  using System.Threading.Tasks;
13  using Weather.Model;
14  using Weather.Services;
15
16  namespace Weather.Workers
17  {
18      public class WeatherWorker : BackgroundService
19      {
20          private ILogger<WeatherWorker> _logger;
21          private IConfiguration _configuration;
22          private IHttpClientFactory _httpClientFactory;
23          private IMemoryCache _cache;
24
25          public WeatherWorker(ILogger<WeatherWorker> logger,
26                               IConfiguration configuration,
27                               IHttpClientFactory httpClientFactory,
28                               IMemoryCache cache)
29          {
30              _logger = logger;
31
32              _configuration = configuration;
33              _httpClientFactory = httpClientFactory;
34              _cache = cache;
35          }
36
37          protected override async Task ExecuteAsync(CancellationToken stoppingToken)
38          {
39              while (!stoppingToken.IsCancellationRequested)
40              {
41                  try
42                  {
43                      await DoWork();
44                  }
45                  catch (Exception ex)
46                  {
47                      _logger.LogError(ex, "An error occurred while executing the background worker.");
48                  }
49
50                  await Task.Delay(_configuration.GetValue<TimeSpan>("BackgroundWorkerInterval"));
51              }
52          }
53
54          private async Task DoWork()
55          {
56              var weatherForecast = await _httpClientFactory.CreateClient().GetWeatherForecast();
57
58              foreach (var forecast in weatherForecast)
59              {
60                  var entry = new CacheEntry(forecast);
61                  _cache.Set(forecast.Key, entry);
62              }
63          }
64
65          private class CacheEntry
66          {
67              public string Key { get; set; }
68              public WeatherForecast Forecast { get; set; }
69          }
70      }
71  }
```

No issues found

F 10:18 CH: 31 SPC CRLF

0 5 dotnetconf2019 master ENG 7:20 PM US 10/26/2019

Debug Test
bug - Any CPU
Presenting...
BlazorWeatherServer
based on your code, including your own custom classes.
Try it now Don't show again

Give control Stop presenting

INT PREVIEW

Live Share

Weather

```
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
```

```
{  
    try  
    {  
        DateTimeOffset? expiresHeader = null;  
        var token = _configuration["accuweathertoken"];  
        if (!string.IsNullOrWhiteSpace(token))  
        {  
            var client = _httpClientFactory.CreateClient("AccuWeather");  
            var response = await client.GetAsync($"{_configuration["weather:uri"]}/340247?apikey={token}");  
            var model = await JsonSerializer.DeserializeAsync<Forecast>(response);  
            _cache.Set(Constants.LATEST_FORECAST_CACHE_KEY, model.First());  
            expiresHeader = response.Content.Headers.Expires;  
        }  
        else  
        {  
            Console.WriteLine("No accuweather key set, returning mock data.");  
  
            //Honoring the expires time  
            if (expiresHeader.HasValue)  
            {  
                await Task.Delay(expiresHeader.Value.UtcDateTime.Subtract(DateTime.UtcNow));  
            }  
            else  
            {  
                await Task.Delay(TimeSpan.FromMinutes(60));  
            }  
        }  
    }  
}
```

No issues found

Ln: 47 Ch: 56 SPC CRLF

↑ 0 ↵ 5 ⌘ dotnetconf2019 ⌘ master ⌘

ENG 7:21 PM
US 10/26/2019

```
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
```

No issues found

```
try
{
    DateTimeOffset? expiresHeader = null;
    var token = _configuration["accuweathertoken"];
    if (!string.IsNullOrWhiteSpace(token))
    {
        var client = _httpClientFactory.CreateClient("AccuWeather");
        var response = await client.GetAsync($"{_configuration["weather:uri"]}/340247?apikey={token}");
        var model = await JsonSerializer.DeserializeAsync<Forecast>(response.Content);
        _cache.Set(Constants.LATEST_FORECAST_CACHE_KEY, model.First());
        expiresHeader = response.Content.Headers.Expires;
    }
    else
    {
        Console.WriteLine("No accuweather key set, returning mock data.");
    }

    //Honoring the expires time
    if (expiresHeader.HasValue)
    {
        await Task.Delay(expiresHeader.Value.UtcDateTime.Subtract(DateTime.UtcNow));
    }
    else
    {
        await Task.Delay(TimeSpan.FromMinutes(60));
    }
}
```

↑ 0 ↗ 5 ⚡ dotnetconf2019 ⚡ master ↘

ENG 7:21 PM
US 10/26/2019

Debug Test Presenting...
Debug Any CPU BlazorWeatherServer Give control Stop presenting
INT PREVIEW

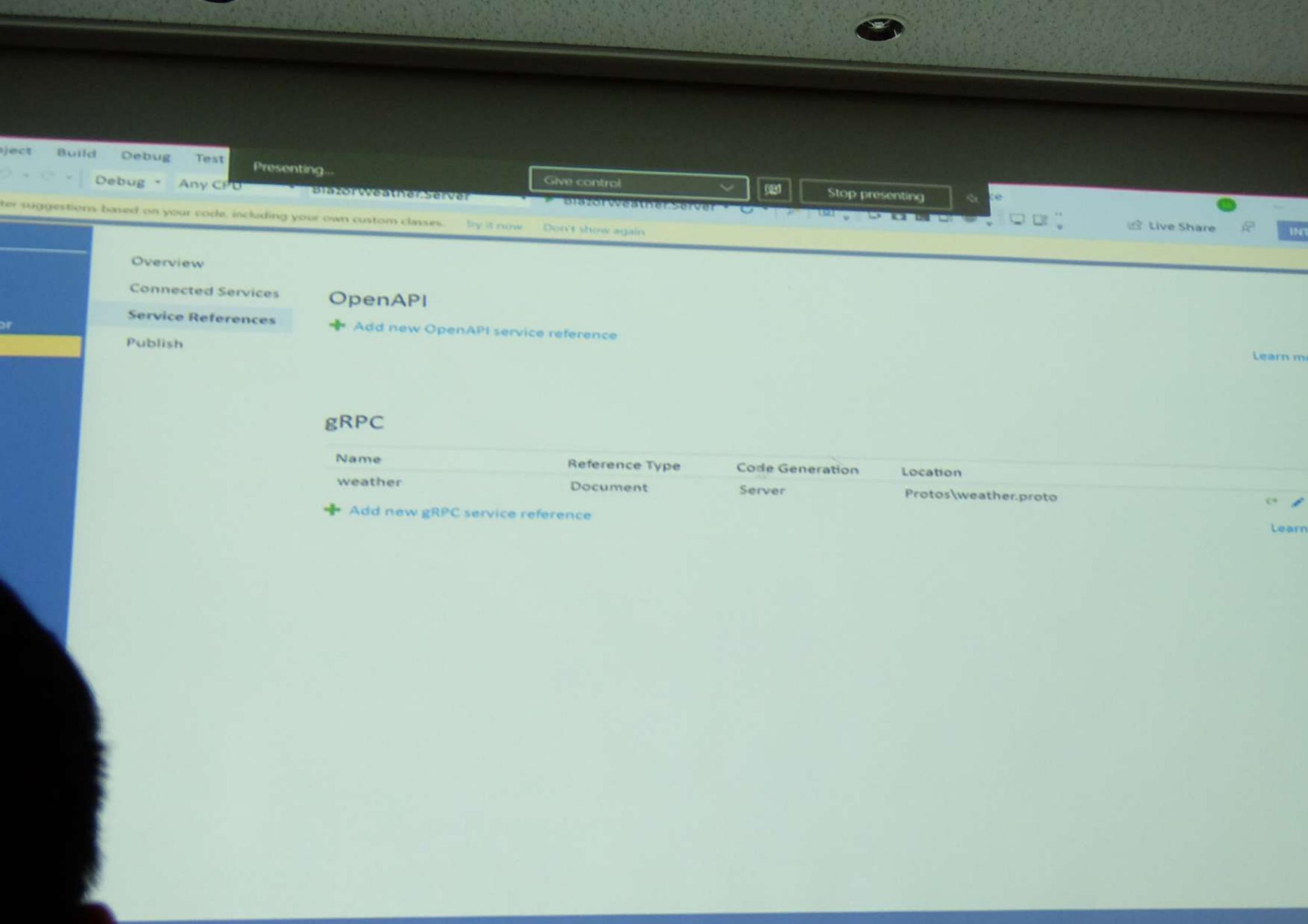
```
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5 using Google.Protobuf.WellKnownTypes;
6 using Grpc.Core;
7 using Microsoft.Extensions.Caching.Memory;
8 using Weather.Model;
9
10 namespace Weather.Services
11 {
12     public class WeatherService : Weather.WeatherBase
13     {
14         private IMemoryCache _cache;
15
16         public WeatherService(IMemoryCache cache)
17         {
18             _cache = cache;
19         }
20
21         public override async Task GetWeatherStream(Empty request, IServerStreamWriter<WeatherResponse> responseStream)
22         {
23             while (!context.CancellationToken.IsCancellationRequested)
24             {
25                 var cachedForecast = _cache.Get<Forecast>(Constants.LATEST_FORECAST_CACHE_KEY);
26                 await responseStream.WriteAsync(GetCurrentWeatherResponse(cachedForecast));
27                 await Task.Delay(TimeSpan.FromSeconds(2));
28             }
29         }
30     }

```

No issues found

Output

dotnetconf2019 master ENG 7:23 PM US 10/26/2019



Overview

Connected Services

Service References

Publish

Learn m...

OpenAPI

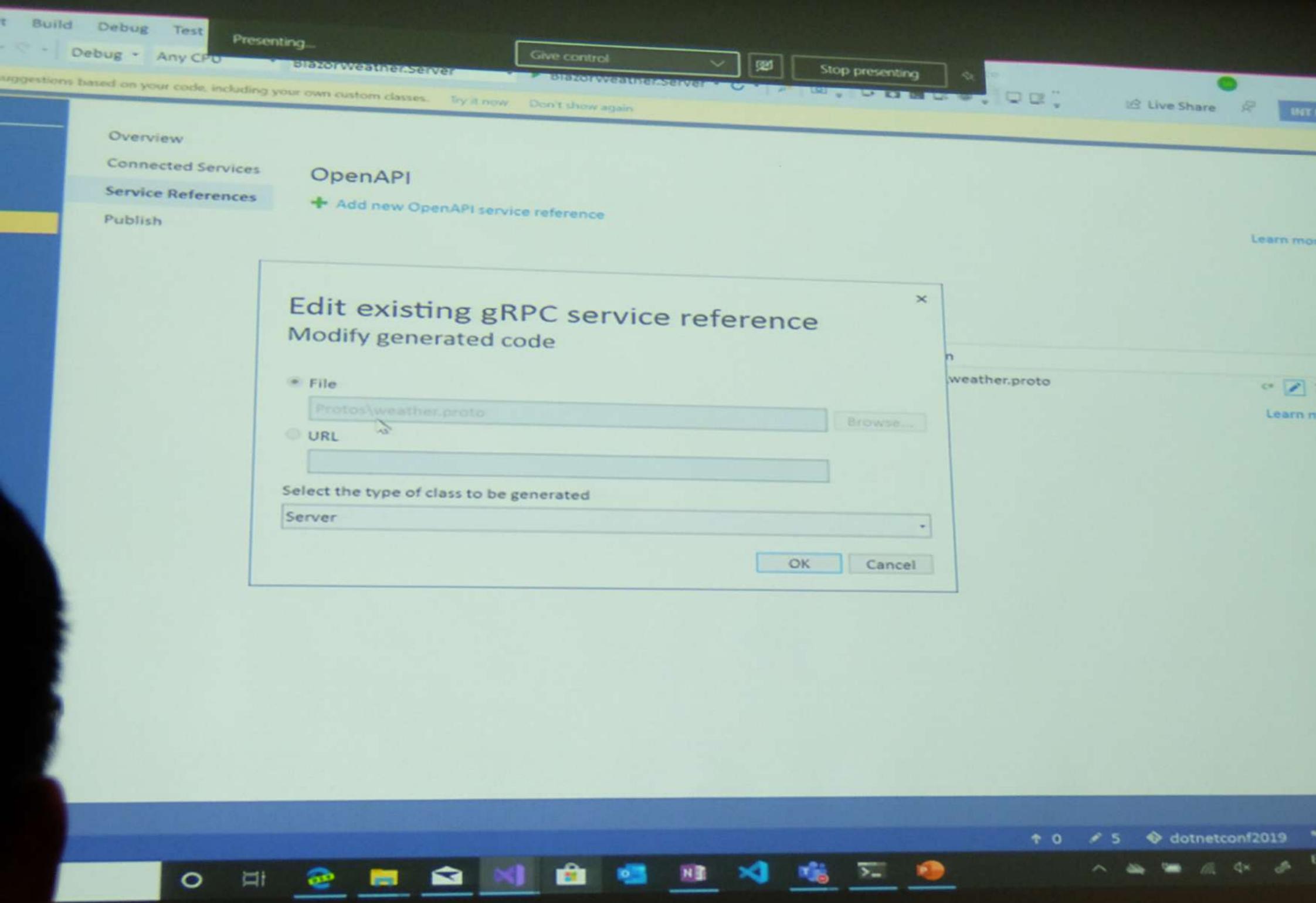
+ Add new OpenAPI service reference

gRPC

Name	Reference Type	Code Generation	Location
weather	Document	Server	Protos\weather.proto

+ Add new gRPC service reference

C Learn m...



Build Debug Test Presenting... Give control Stop presenting Live Share INT PREVIEW

Weather

```
8  using Weather.Model;
9
10 namespace Weather.Services
11 {
12     public class WeatherService : Weather.WeatherBase
13     {
14         private IMemoryCache _cache;
15
16         public WeatherService(IMemoryCache cache)
17         {
18             _cache = cache;
19         }
20
21         public override async Task GetWeatherStream(Empty request, IServerStreamWriter<WeatherResponse> responseStream, ServerCallContext context)
22         {
23             while (!context.CancellationToken.IsCancellationRequested)
24             {
25                 var cachedForecast = _cache.Get<Forecast>(Constants.LATEST_FORECAST_CACHE_KEY);
26                 await responseStream.Write(cachedForecast);
27                 await Task.Delay(TimeSpan.FromSeconds(2));
28             }
29         }
30
31         public override Task<WeatherResponse> GetWeather(Empty request, ServerCallContext context)
32         {
33             var forecast = _cache.Get<Forecast>(Constants.LATEST_FORECAST_CACHE_KEY);
34             return Task.FromResult(GetCurrentWeatherResponse(forecast));
35         }
36     }
}
```

No issues found

136% 100% 120% 140% 160% 180% 200% 220% 240% 260% 280% 300% 320% 340% 360% 380% 400% 420% 440% 460% 480% 500% 520% 540% 560% 580% 600% 620% 640% 660% 680% 700% 720% 740% 760% 780% 800% 820% 840% 860% 880% 900% 920% 940% 960% 980% 1000%

project.

dotnetconf2019 master ENG 7:24 PM US 10/26/2019

Project Build Debug Test Presenting... Give control Stop presenting Live Share INT PREVIEW

```
6  using System.Collections.Generic;
7  using System.Linq;
8  using System.Threading.Tasks;
9  using Weather.Model;
10
11 namespace Weather.Services
12 {
13     [ApiController]
14     public class WeatherController : Controller
15     {
16         private IMemoryCache _cache;
17
18         public WeatherController(IMemoryCache cache)
19         {
20             _cache = cache;
21         }
22
23         [EnableCors()]
24         [HttpGet("/json")]
25         public ActionResult<WeatherResponse> GetWeatherForecast()
26         {
27             var forecast = _cache.Get<Forecast>(Constants.LATEST_FORECAST_CACHE_KEY);
28             return WeatherService.GetCurrentWeatherResponse(forecast);
29         }
30     }
31 }
32
```

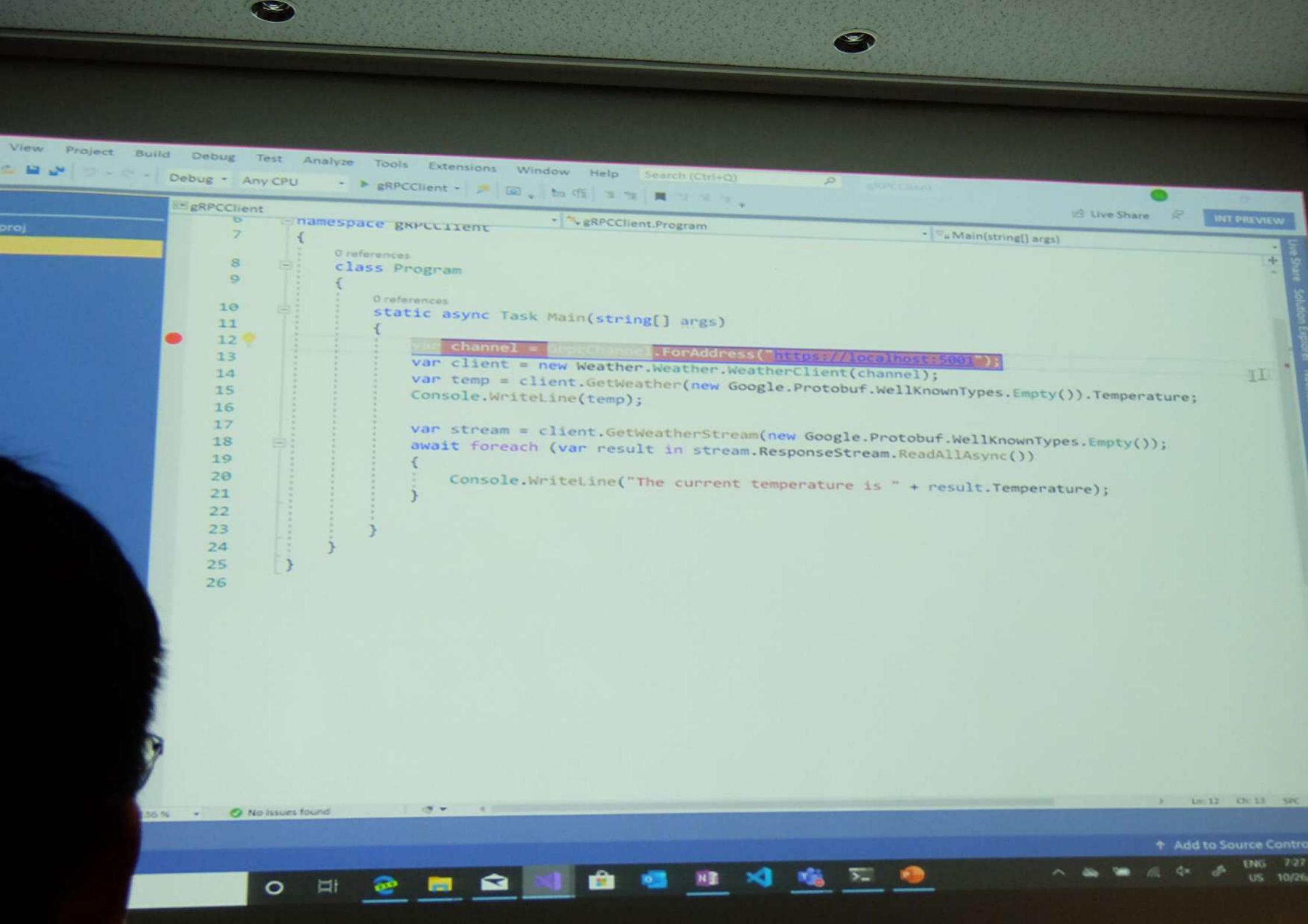
No issues found

LN: 28 CH: 14 SPC: CRLF

dotnetconf2019 master

```
{"weatherText": "Sunny", "weatherIcon": 1, "weartherUri": "https://developer.accuweather.com/sites/default/files/100x100/01-Sun.png", "isDayTime": true, "temperature": 57, "relativeHumidity": 45, "windSpeed": 5.8, "uvIndex": 2, "pressure": 1013, "seconds": 1572140858, "nanos": 722049000}, "windDirection": "SSE", "location": "Seattle"}
```





Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) gRPCClient

gRPCClient

```
6  namespace gRPCClient
7  {
8      0 references
9      class Program
10     {
11         0 references
12         static async Task Main(string[] args)
13         {
14             var channel = GrpcChannel.ForAddress("https://localhost:5001");
15             var client = new Weather.Weather.WeatherClient(channel);
16             var temp = client.GetWeather(new Google.Protobuf.WellKnownTypes.Empty());
17             Console.WriteLine(temp);
18
19             var stream = client.GetWeatherStream(new Google.Protobuf.WellKnownTypes.Empty());
20             await foreach (var result in stream.ResponseStream.ReadAllAsync())
21             {
22                 Console.WriteLine("The current temperature is " + result.Temperature);
23             }
24         }
25     }
26 }
```

No Issues found

Add to Source Control

INT PREVIEW

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'gRPCClient' (1 of 1 project)

- gRPCClient
- Dependencies
- Protos
 - proto.proto
- Program.cs

Live Share

Live Share

Notifications

ENG 7:27 PM
US 10/26/2019

Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) gRPCClient

gRPCClient

Live Share INT PREVIEW

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'gRPCClient' (1 of 1 project)

Dependencies

g.proto

Program.cs

Live Share Solution Explorer Notifications

gRPCClient

gRPCClient.Program

Main(string)

Build Rebuild Clean Analyze and Code Cleanup Pack Publish... Scope to This New Solution Explorer View Show on Code Map Edit Project File Add Manage NuGet Packages... Manage User Secrets Set as StartUp Project Debug Cut Ctrl+X Remove Del Rename Unload Project Load Project Dependencies Open Folder in File Explorer Properties Alt+Enter

No issues found

Add to Source Control

136% ENG 7:27 PM 10/26/2019

```
6  namespace gRPCClient
7  {
8      class Program
9      {
10         static async Task Main(string[] args)
11         {
12             var channel = GrpcChannel.ForAddress("https://");
13             var client = new Weather.Weather.WeatherClient();
14             var temp = client.GetWeather(new Google.Protobuf.ByteString());
15             Console.WriteLine(temp);
16
17             var stream = client.GetWeatherStream(new Google.Protobuf.ByteString());
18             await foreach (var result in stream.Response)
19             {
20                 Console.WriteLine("The current temperature is " + result);
21             }
22         }
23     }
24 }
25 }
```

Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q)

gRPCClient

gRPCClient

gRPCClient.Program

Main(string)

Live Share INT PREVIEW

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'gRPCClient' (1 of 1 project)

Dependencies

.proto

.cs

Notifications

Live Share

Build

Rebuild

Clean

Analyze and Code Cleanup

Pack

Publish...

Scope to This

New Solution Explorer View

Show on Code Map

Edit Project File

Add

New Item... Ctrl+Shift+A

Existing Item... Shift+Alt+A

New Folder

REST API Client...

Container Orchestrator Support...

Docker Support...

Machine Learning

Reference...

Service Reference...

Connected Service

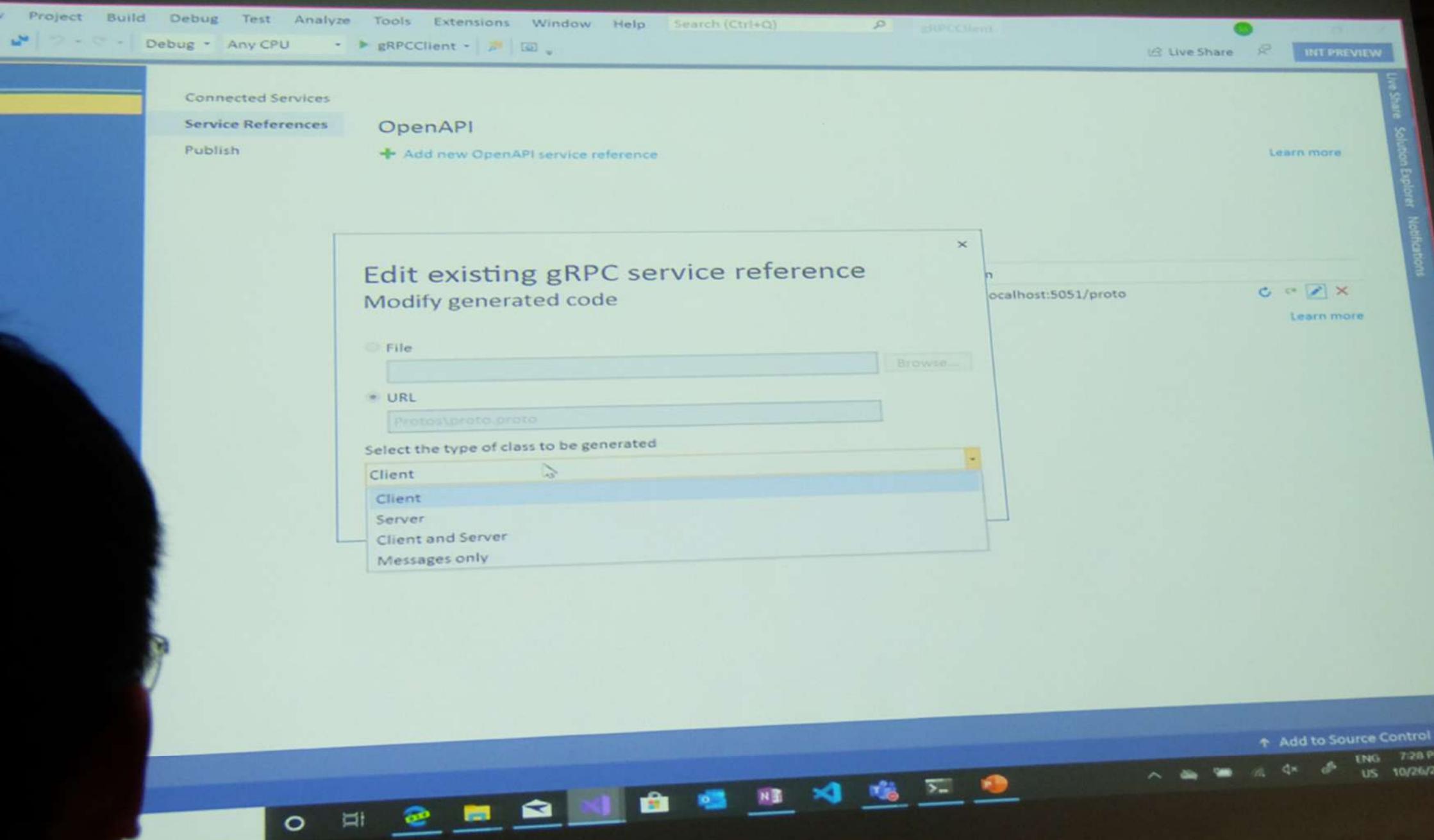
Class...

New EditorConfig (IntelliCode)

No issues found

100% Add to Source Control ENG 7:27 PM US 10/26/2019

```
6  namespace gRPCClient
7  {
8      0 references
9      class Program
10     {
11         0 references
12         static async Task Main(string[] args)
13         {
14             var channel = GrpcChannel.ForAddress("https://")
15             var client = new Weather.Weather.WeatherClient()
16             var temp = client.GetWeather(new Google.Protobuf
17             Console.WriteLine(temp);
18
19             var stream = client.GetWeatherStream(new Go
20             await foreach (var result in stream.Response
21             {
22                 ...
23             }
24         }
25     }
26 }
```



Test Analyze Tools Extensions Window Help

Search (Ctrl+Q)

gRPCClient

Live Share

INT PREVIEW

Any CPU gRPCClient gRPCClient.Program Main(string[] args)

```
namespace gRPCClient
{
    class Program
    {
        static async Task Main(string[] args)
        {
            var channel = GrpcChannel.Create(new Uri("https://localhost:5001"));
            var client = new Weather.Weather.WeatherClient(channel);
            var temp = client.GetWeather(new Google.Protobuf.WellKnownTypes.Empty());
            Console.WriteLine(temp);

            var stream = client.GetWeatherStream(new Google.Protobuf.WellKnownTypes.Empty());
            await foreach (var result in stream.ResponseStream.ReadAllAsync())
            {
                Console.WriteLine("The current temperature is " + result.Temperature);
            }
        }
    }
}
```

No issues found

Add to Source Control

ENG 7:28 PM
US 10/26/2019

The screenshot shows a Windows desktop environment with a .NET IDE open. The title bar of the IDE reads "RESTClient". The menu bar includes "Build", "Debug", "Test", "Analyze", "Tools", "Extensions", "Window", and "Help". The search bar says "Search (Ctrl+Q)". The code editor displays the following C# code:

```
7 0 references
8 class Program
{
9     0 references
10    static async Task Main(string[] args)
11    {
12        var http = new HttpClient();
13        var client = new RESTClient.swaggerClient("https://localhost:5001");
14        var result = await client.JsonAsync();
15        Console.WriteLine(result.Temperature);
16    }
17}
18
```

The Solution Explorer on the right shows the project "RESTClient" with its contents: "Dependencies", "OpenAPIs", "swagger.json", and "Program.cs". The status bar at the bottom left says "No issues found".

Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) RESTClient RESTClient

Debug Any CPU RESTClient RESTClient.Program Main(string)

```
7 0 references
8 class Program
{
9     0 references
10    static async Task Main(string[] args)
11    {
12        var http = new HttpClient();
13        var client = new RESTClient.swaggerClient("https://localhost:5001");
14        var result = await client.JsonAsync();
15        Console.WriteLine(result.Temperature);
16    }
17}
18
```

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'RESTClient' (1 of 1 project)

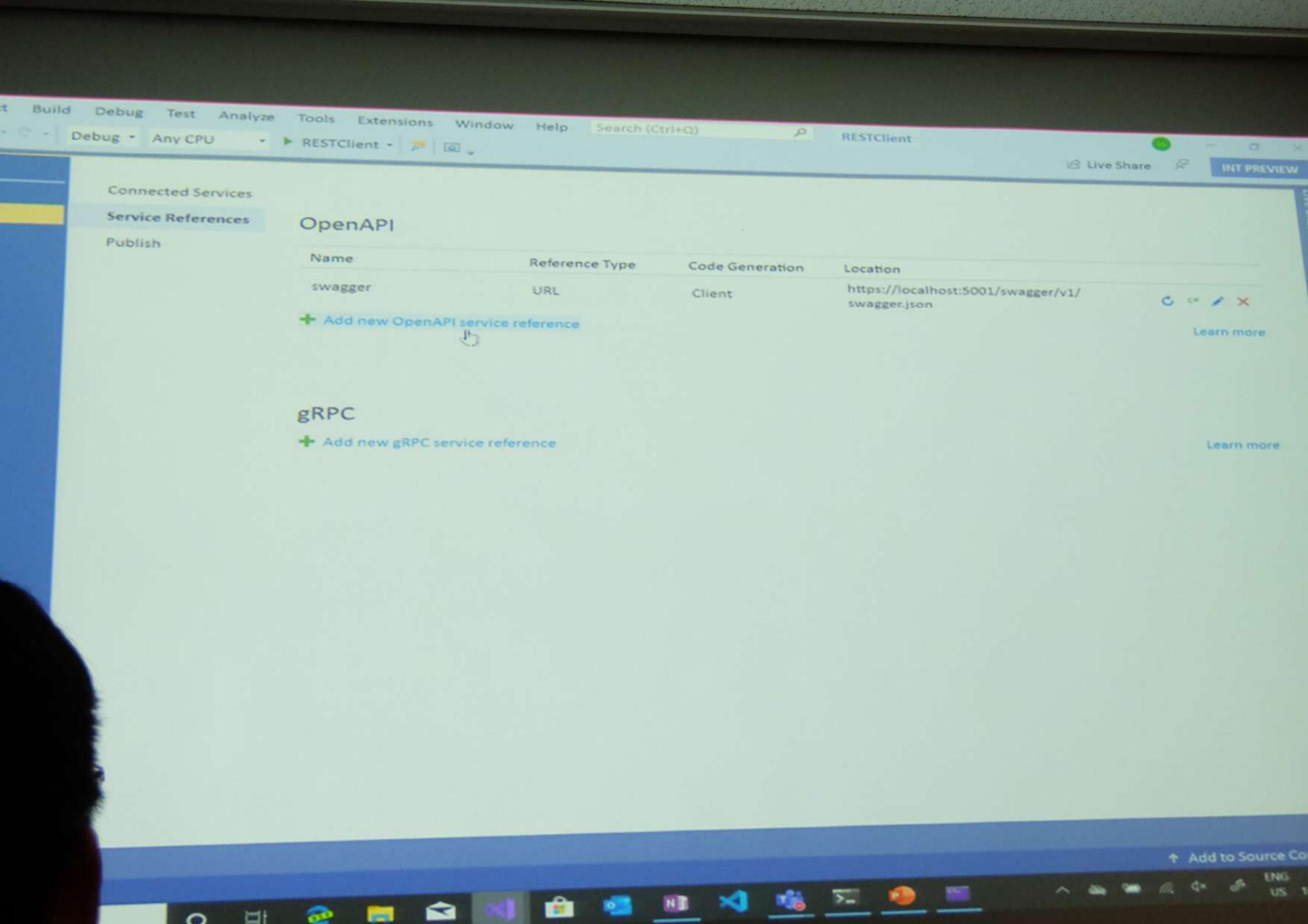
- RESTClient
 - Dependencies
 - OpenAPIs
 - swagger.json
- Program.cs

INT PREVIEW

Live Share

Live Share

Solution Explorer



C# 8.0



Safe

Nullable and non-nullable reference types help you write safer code

Declare your intent more clearly



Modern

Async streams for modern workloads like cloud & IoT communication

Easily work with cloud scale datasets using indexes and ranges



Productive

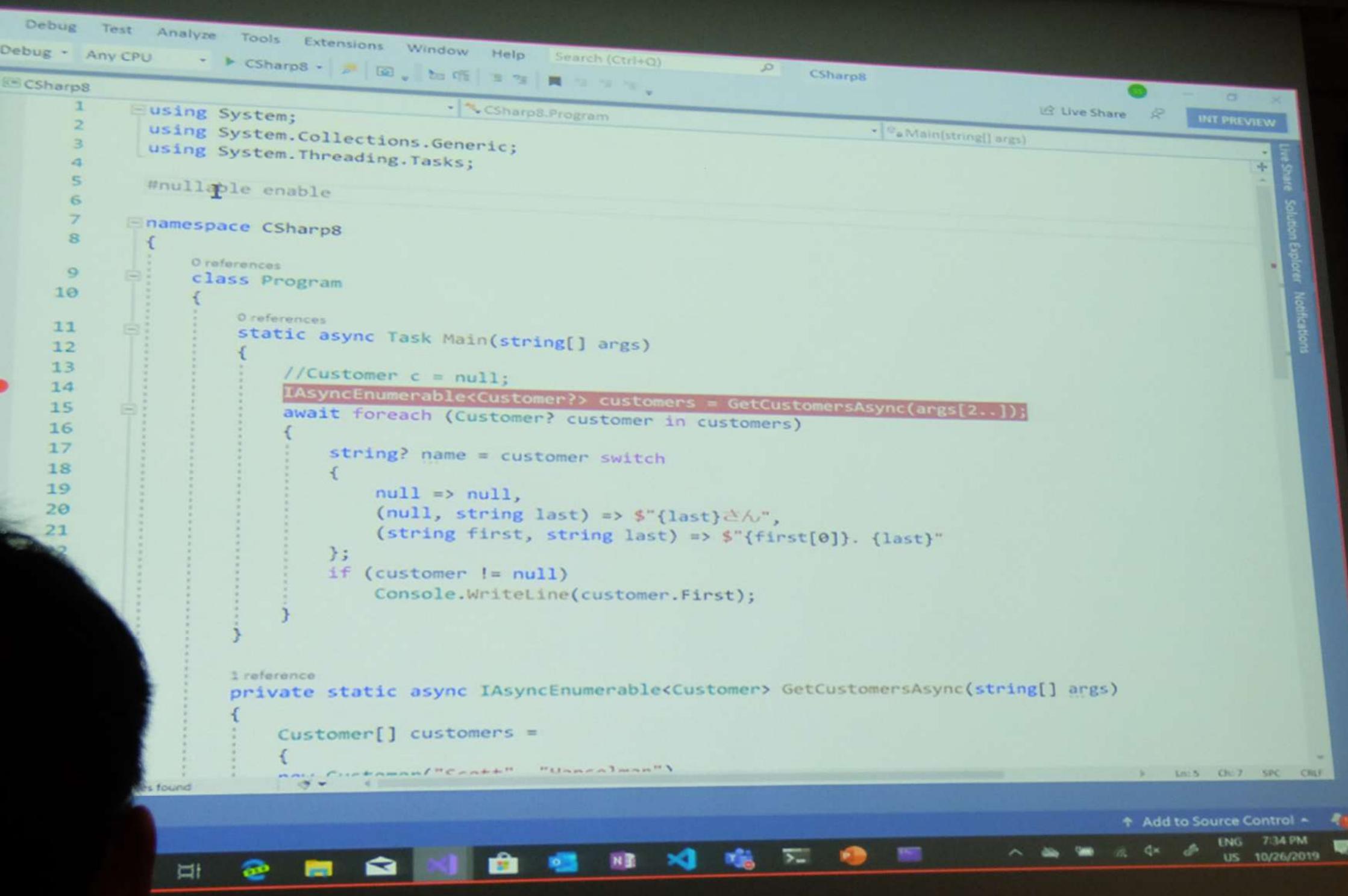
Write less code using patterns
Protect data with readonly members

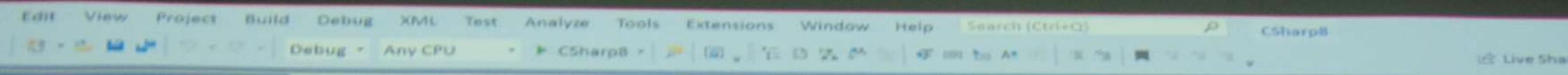
Improved using statements for resource management

A screenshot of the Microsoft Visual Studio IDE interface. The menu bar includes 'Build', 'Debug', 'Test', 'Analyze', 'Tools', 'Extensions', 'Window', and 'Help'. The top status bar shows 'Search (Ctrl+Q)'. The title bar says 'CSharp8' with a 'CSharp8' tab. The main code editor window displays the following C# program:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Threading.Tasks;
4
5  #nullable enable
6
7  namespace CSharp8
8  {
9      0 references
10     class Program
11     {
12         0 references
13         static async Task Main(string[] args)
14         {
15             //Customer c = null;
16             IAsyncEnumerable<Customer?> customers = GetCustomersAsync(args[2..]);
17             await foreach (Customer? customer in customers)
18             {
19                 string? name = customer switch
20                 {
21                     null => null,
22                     (null, string last) => $"{last}\n",
23                     (string first, string last) => $"{first[0]}. {last}"
24                 };
25                 if (customer != null)
26                     Console.WriteLine(customer.First);
27             }
28
29             1 reference
30             private static async IAsyncEnumerable<Customer> GetCustomersAsync(string[] args)
31             {
32                 Customer[] customers =
33                 {
34                     new Customer("Carrie", "Hansen"),
35                     new Customer("Derek", "Hanson"),
36                     new Customer("Leanne", "Kemp"),
37                     new Customer("Loreen", "Kemp"),
38                     new Customer("Mark", "Kemp"),
39                     new Customer("Natalie", "Kemp"),
40                     new Customer("Peter", "Kemp"),
41                     new Customer("Samuel", "Kemp"),
42                     new Customer("Samantha", "Kemp"),
43                     new Customer("Trevor", "Kemp"),
44                     new Customer("Travis", "Kemp"),
45                     new Customer("Vivian", "Kemp"),
46                     new Customer("Wendy", "Kemp"),
47                     new Customer("Zach", "Kemp")
48                 };
49             }
50         }
51     }
52 }
```

ENG 7:34 PM
US 10/26/2011





File Explorer window open showing a project structure:

```
1 <Project Sdk="Microsoft.NET.Sdk">
2
3     <PropertyGroup>
4         <OutputType>Exe</OutputType>
5         <TargetFramework>netcoreapp3.0</TargetFramework>
6         <!--<NullableContextOptions>enable</NullableContextOptions>
7             <LangVersion>8.0</LangVersion>-->
8     </PropertyGroup>
9 </Project>
10
```

Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q)

Debug Any CPU CSharp8.csproj CSharp8.cs

CSharp8

Main(string[] args)

```
35     new Customer(null, "Jones")
36 };
37     foreach (var c in customers)
38     {
39         await Task.Delay(1000);
40         yield return c;
41     }
42 }
43 }
44
45 class Customer
46 {
47     public string? First { get; }
48     public string Last { get; }
49     public Customer(string? first, string last) => (First, Last) = (first, last);
50     public void Deconstruct(out string first, out string last) => (first, last) = (First, Last);
51 }
52 }
53 }
54 }
```

No issues found

Add to Source Control

ENG US 10

View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) CSharp8

CSharp8.cs

```
27
28     1 reference
29     private static async IAsyncEnumerable<Customer> GetCustomersAsync(string[] args)
30     {
31         1 reference
32         Customer[] customers =
33         {
34             new Customer("Scott", "Hanselman"),
35             new Customer("Scott", "Hunter"),
36             new Customer(null, "Smith"),
37             new Customer(null, "Jones")
38         };
39         foreach (var c in customers)
40         {
41             await Task.Delay(1000);
42             yield return c;
43         }
44     }
45
46     9 references
47     class Customer
48     {
49         3 references
50         public string? First { get; }
51         2 references
52         public string Last { get; }
53         4 references
54         public Customer(string? first, string last) => (First, Last) = (first, last);
55         0 references
56         public void Deconstruct(out string? first, out string last) => (first, last) = (First, Last);
57     }
58 }
```

No issues found

Add to Source Control

Live Share

INT PREVIEW

Live Share Solution Explorer Notifications

LN: 28 CH: 42 SPC: ORLF

ENG 7:35 PM
US 10/26/2019

Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) CSharp8

CSharp8

Live Share INT PREVIEW

Main(string[] args)

```
7  namespace CSharp8
8  {
9      // References
10     class Program
11    {
12        // References
13        static async Task Main(string[] args)
14        {
15            //Customer c = null;
16            IAsyncEnumerable<Customer?> customers = GetCustomersAsync(args[2..]);
17            await foreach (Customer? customer in customers)
18            {
19                string? name = customer switch
20                {
21                    null => null,
22                    (null, string last) => $"{last}さん",
23                    (string first, string last) => $"{first[0]}. {last}"
24                };
25                if (customer != null)
26                    Console.WriteLine(customer.First);
27            }
28
29        private static async IAsyncEnumerable<Customer> GetCustomersAsync(string[] args)
30        {
31            Customer[] customers =
32            {
33                new Customer("Scott", "Hanselman"),
34                new Customer("Scott", "Hunter"),
35                new Customer(null, "Smith"),
36                new Customer(null, "Jones")
37            };
38            foreach (var c in customers)
39            {
40            }
41        }
42    }
43}
```

No issues found

Add to Source Control

100% 136% 136% 136%

UNO 7:36 PM US 10/26/2019

.NET Core 3.0 for Windows Desktop



Deployment Flexibility

Side-by-side deployment, self-contained EXEs

Install machine global or app local framework



Windows 10

Access modern Windows 10 APIs from WPF and WinForms

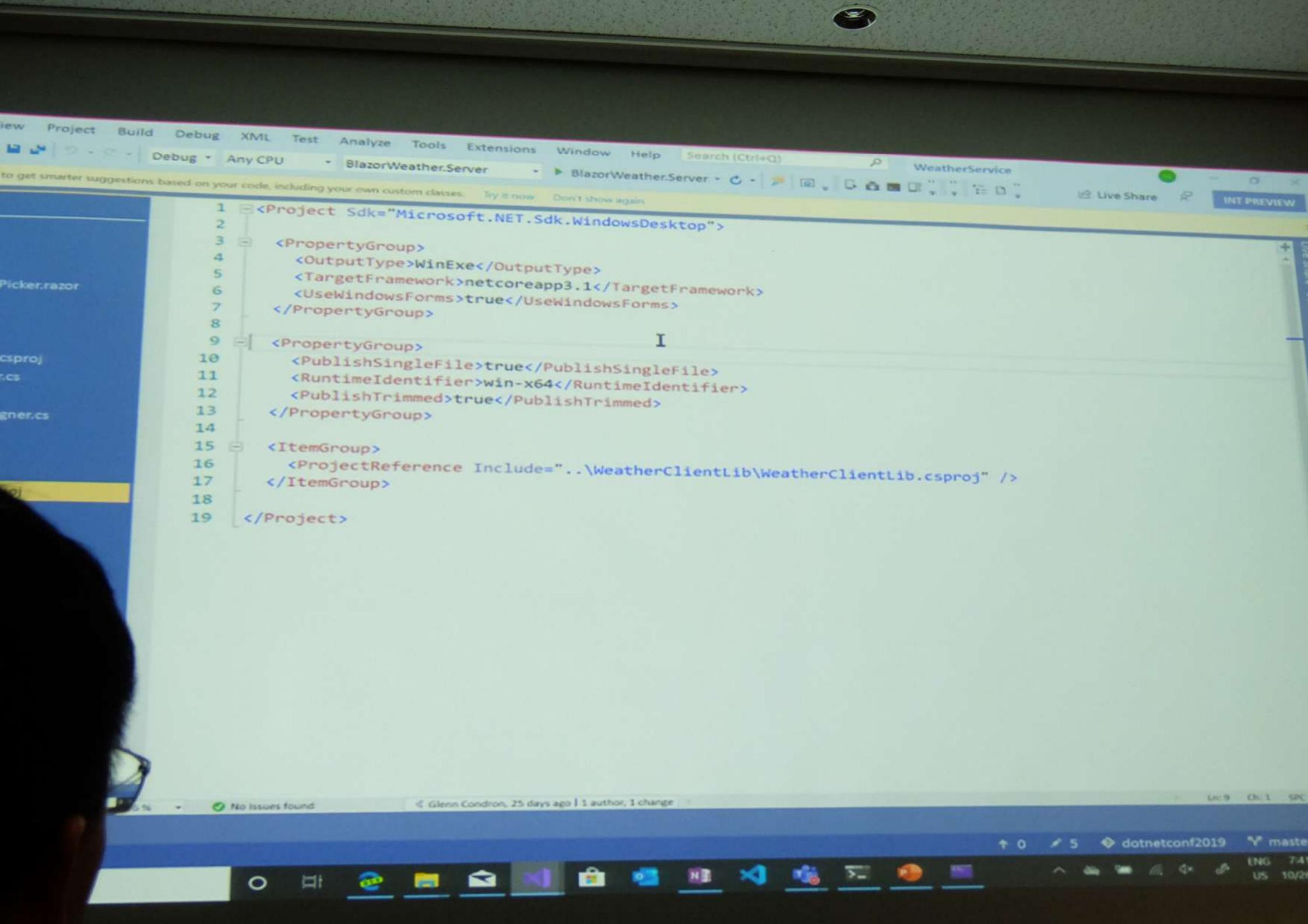
Use native Windows 10 controls via XAML islands



Open Source

WPF and WinForms projects also open source on GitHub

Take advantage of performance, runtime and API improvements happening in .NET Core



```
1 <Project Sdk="Microsoft.NET.Sdk.WindowsDesktop">
2
3   <PropertyGroup>
4     <OutputType>WinExe</OutputType>
5     <TargetFramework>netcoreapp3.1</TargetFramework>
6     <UseWindowsForms>true</UseWindowsForms>
7   </PropertyGroup>
8
9   <PropertyGroup>
10    <PublishSingleFile>true</PublishSingleFile>
11    <RuntimeIdentifier>win-x64</RuntimeIdentifier>
12    <PublishTrimmed>true</PublishTrimmed>
13  </PropertyGroup>
14
15  <ItemGroup>
16    <ProjectReference Include=".\\WeatherClientLib\\WeatherClientLib.csproj" />
17  </ItemGroup>
18
19 </Project>
```



A screenshot of a Microsoft Visual Studio IDE window showing the configuration file for a Blazor application.

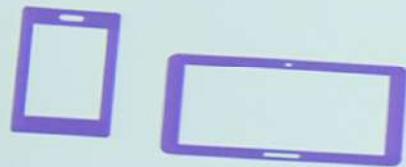
The title bar shows "BlazorWeather.Server" and the tab bar shows "BlazorWeatherServer".

The code editor displays the following XML configuration:

```
1 <Project Sdk="Microsoft.NET.Sdk.WindowsDesktop">
2
3   <PropertyGroup>
4     <OutputType>WinExe</OutputType>
5     <TargetFramework>netcoreapp3.1</TargetFramework>
6     <UseWindowsForms>true</UseWindowsForms>
7   </PropertyGroup>
8
9   <PropertyGroup>
10    <PublishSingleFile>true</PublishSingleFile>
11    <RuntimeIdentifier>win-x64</RuntimeIdentifier>
12    <PublishTrimmed>true</PublishTrimmed>
13  </PropertyGroup>
14
15  <ItemGroup>
16    <ProjectReference Include="..\WeatherClientLib\WeatherClientLib.csproj" />
17  </ItemGroup>
18
19 </Project>
```

The status bar at the bottom indicates "No issues found", "Glenn Condon, 25 days ago | 1 author, 1 change", and "dotnetconf2019 master".

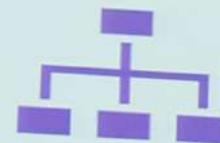
Xamarin



iOS & Android with C#

Native performance, native APIs,
native UI

Build native mobile apps with
Visual Studio



Cross-Platform

Build cross-platform UI with
Xamarin.Forms

Access native APIs from shared
code with Xamarin.Essentials

Share code with .NET Standard



Free & Open Source

Xamarin is part of .NET and
open source on GitHub

ASP.NET Core 3.0 Blazor



Full stack web development with C#

You don't need to know AngularJS,
React, Vue, etc.

Take advantage of stability and
consistency of .NET



Runs in all browsers

Strongly typed on the client
and server
Share C# code with the client
and server



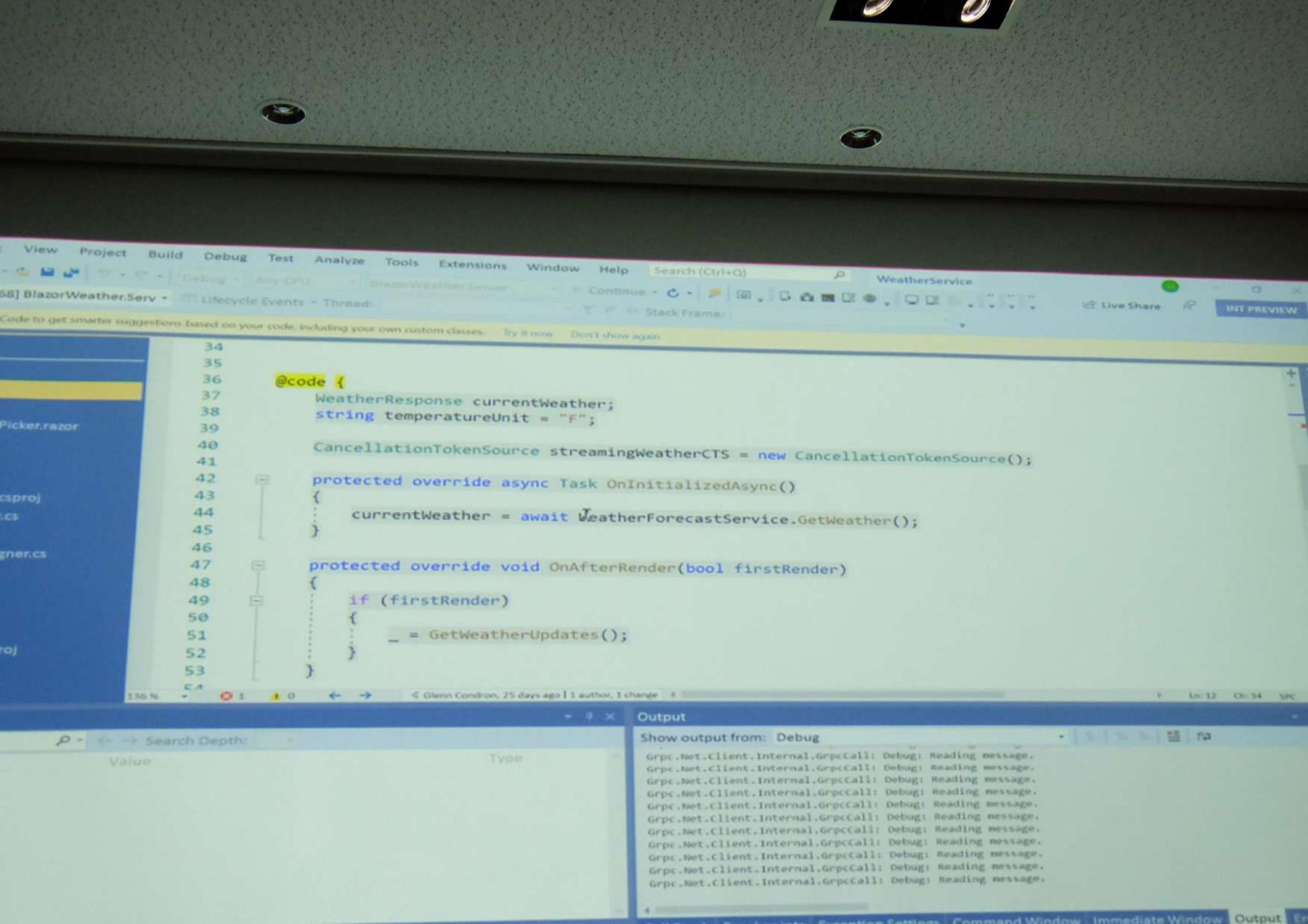
Web Assembly (In Preview, Release in May 2020)

Native performance
Requires no plugin or code
transpilation

```
10
11    <div id="main" class="@GetBackgroundClass()
12      <div class="weather-now">
13        <h1>SEATTLE</h1>
14        <div class="location-img"></div>
15        <div class="temperature">
16          @GetTemperature()
17          <TemperatureUnitPicker @bind-Te
18        </div>
19        <div class="summary">
20          Updated @G
24        <div class="metrics">
25          <div>UV index <p>@currentWeathe
26          <div>Barometer <p>@currentWeath
27          <div>Humidity <p>@currentWeathe
28          <div>Wind <p>@currentWeather.Wi
29        </div>
```

```
Show output from Grpc.Net.Client
```







```
1 <div class="unit-switch" @onclick="SwitchTemperatureUnit">
2   <span class="temp-unit">@TemperatureUnit</span>
3   <span class="temp-unit-option"/> @({TemperatureUnit == "F" ? "C" : "F"})</span>
4 </div>
5
6 @code {
7   [Parameter]
8   public string TemperatureUnit { get; set; } = "F";
9
10  [Parameter]
11  public EventCallback<string> TemperatureUnitChanged { get; set; }
12
13  Task SwitchTemperatureUnit()
14  {
15    TemperatureUnit = TemperatureUnit == "F" ? "C" : "F";
16    return TemperatureUnitChanged.InvokeAsync(TemperatureUnit);
17 }
```

No issues found

Glenn Condron, 25 days ago | 1 author, 1 change

Output

Show output from: Debug

```
Grpc.Net.Client.Internal.GrpcCall: Debug: Reading message.
```

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output Error

ML.NET



Built for .NET developers

Create custom ML models using C#
or F# without having to leave the
.NET ecosystem



Custom ML made easy with AutoML

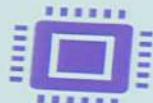
Visual Studio Model Builder
and CLI make it super easy to
build custom ML Models



Extended with TensorFlow & more

Leverage other popular ML
frameworks (TensorFlow,
ONNX, and more)

.NET Core 3.0 IoT Support



Supports Raspberry Pi and other devices

You can now run .NET Core apps in small places, including ASP.NET



Read sensor data & write to displays

New APIs for GPIO pins that enable using millions of IoT peripherals



Works with containers

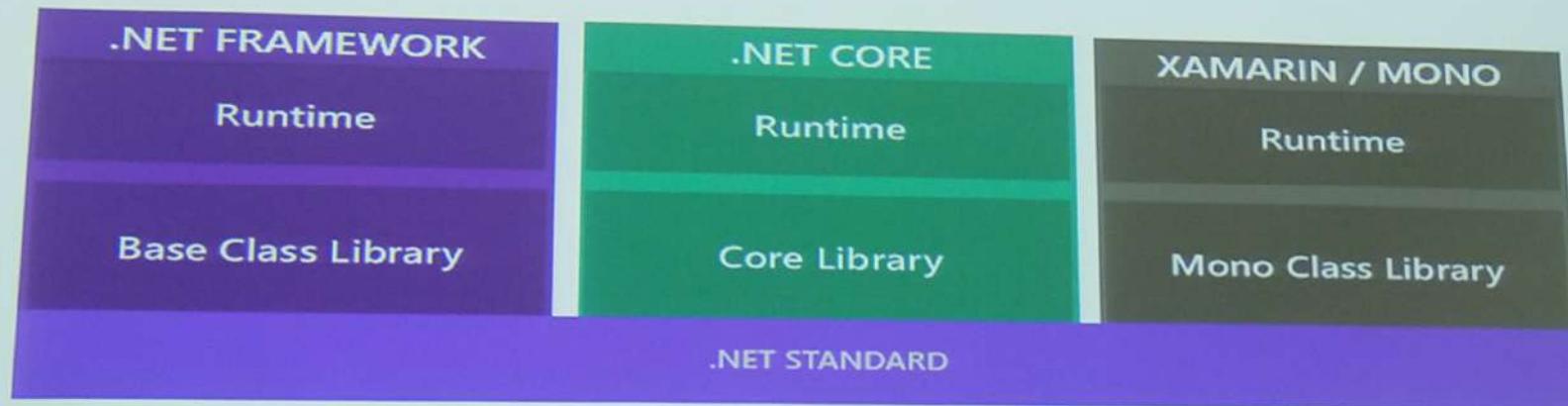
Deploy apps directly onto devices or with containers

.NET 5

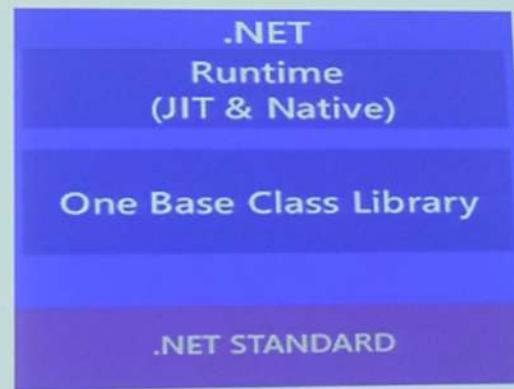


.NET

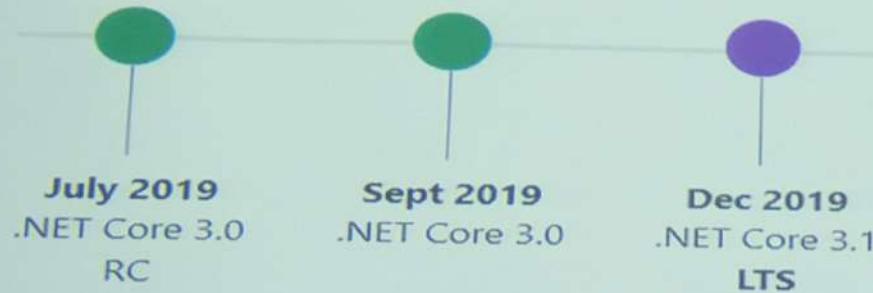
.NET



.NET 5



.NET Schedule



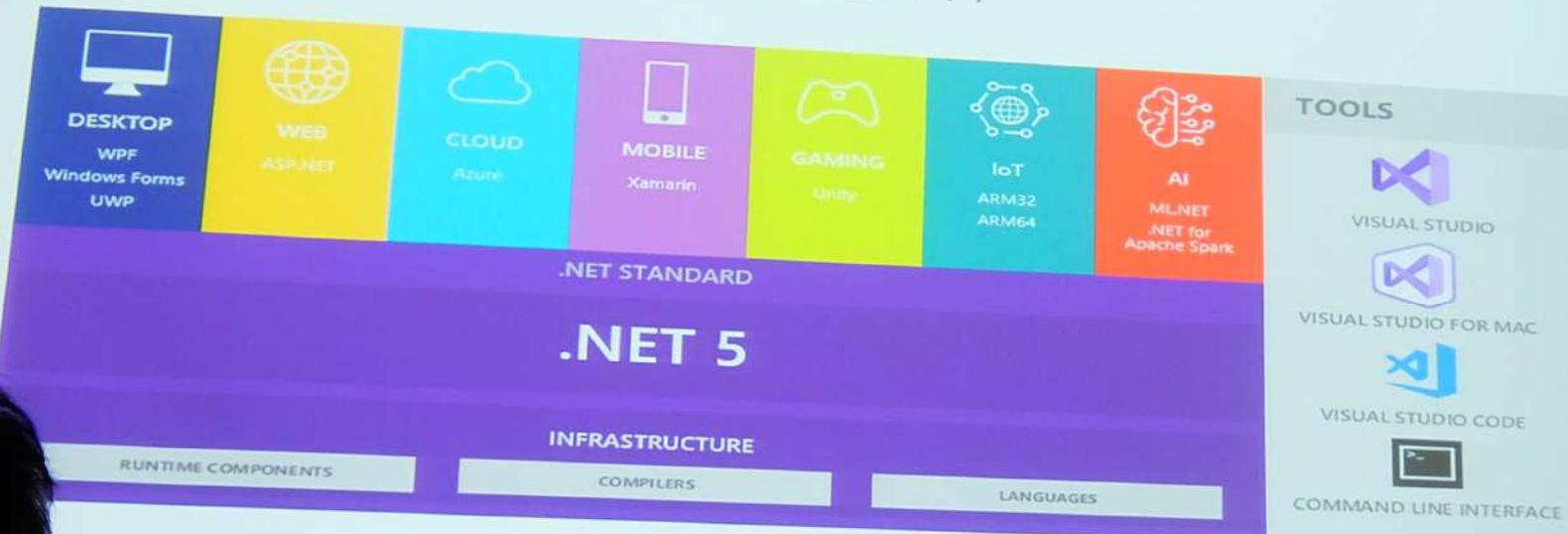
- .NET Core 3.0 released in late September!
- .NET Core 3.1 = Long Term Support (LTS)

.NET Schedule



- .NET Core 3.0 released in late September!
- .NET Core 3.1 = Long Term Support (LTS)
- .NET 5.0 release in November 2020
- Major releases every year, LTS for even numbered releases
- Predictable schedule, minor releases if needed

.NET – A unified platform



Download .NET Core 3.0 Today!

dot.net/get-core3

visualstudio.com/downloads



DESKTOP



WEB



CLOUD



MOBILE



GAMING



IoT



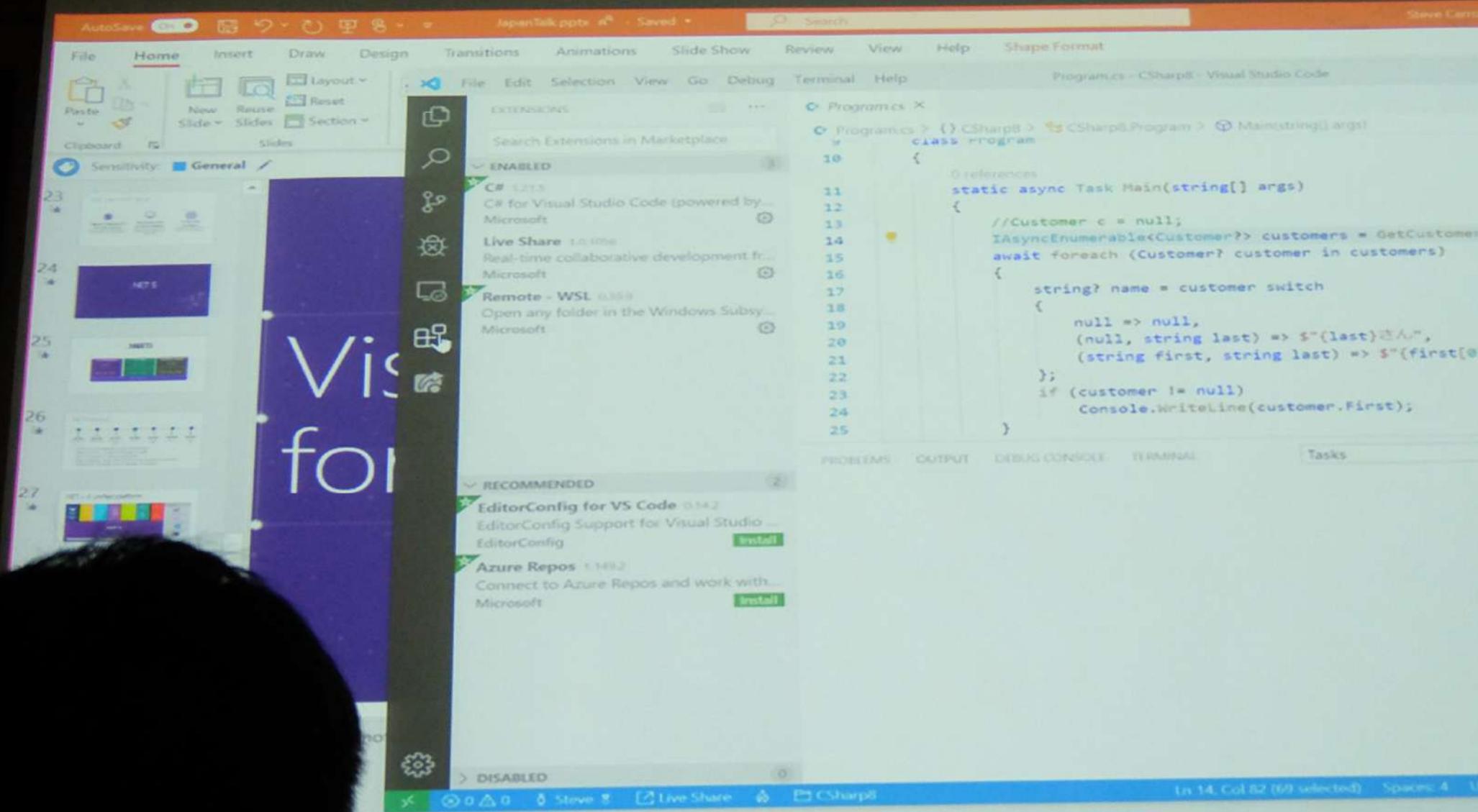
AI

.NET



Visual Studio 2019 for .NET devs





AutoSave 10m

X JapanTalk.pptx - Saved

Steve Carroll (N)

File Edit Selection View Go Debug Terminal Help

Programs - CSharp8 - Visual Studio Code

LIVE SHARE

SESSION DETAILS

Starting collaboration session...

CONTACTS

Recent contacts (1)

Steve Carroll (VISUAL STUDIO)

Suggested contacts (0)

invite participants...

HELP

Tweet us your feedback

Report a problem

Configure Settings

More Info

Program.cs X

Program.cs > ()> CSharp8> CSharp8.Program > Main(string[] args)

```
10    <
11    <
12    <
13    <
14    < 
15    <
16    <
17    <
18    <
19    <
20    <
21    <
22    <
23    <
24    <
25    <
26    <
27    <
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Tasks

Starting collaboration session

Source: Live Share (Extension)

16:14, 10:48 80 selected Spaces: 4 Unit: B

```
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
```

```
static async Task Main(string[] args)
{
    //Customer c = null;
    IAsyncEnumerable<Customer?> customers = GetCustomersAsync(args[2]);
    await foreach (Customer? customer in customers)
    {
        string? name = customer switch
        {
            null => null,
            (null, string last) => $"{last}\n",
            (string first, string last) => $"{first[0]}. {last}"
        };
        if (customer != null)
            Console.WriteLine(customer.First);
    }
}
```

COMMENTS

PROBLEMS

OUTPUT

Tasks



Invitation link copied to clipboard! Send it to anyone you trust. [More info](#) [Copy again](#)

Source: Live Share (Extensi...

Make read-only

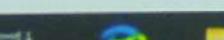
More info

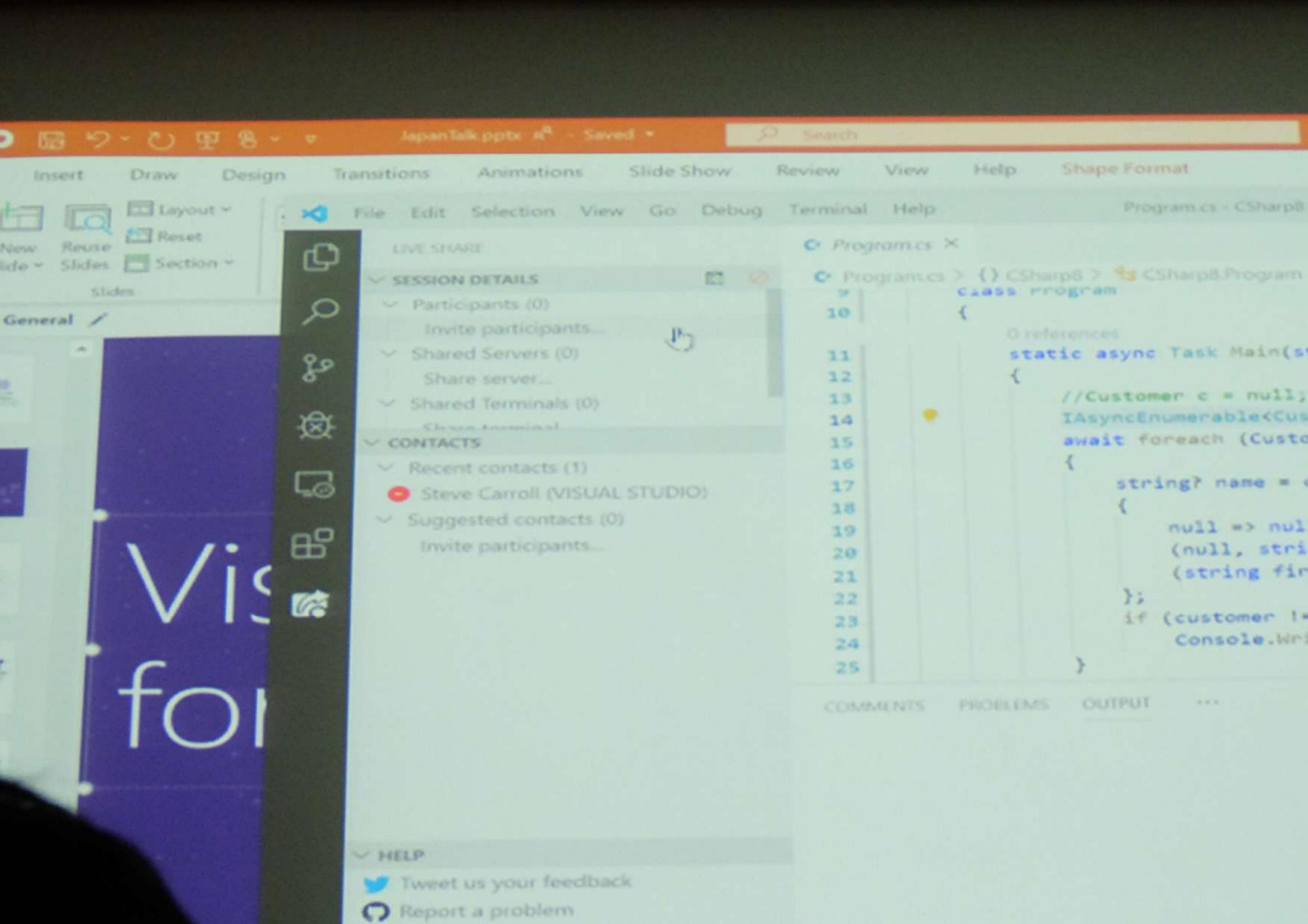
Copy again

CSharp8

Ln 14, Col 82 (69 selected) Spaces: 4 UTF-8 with BOM CRLF ⌂ ⌂ 1

Notes ENG 7:58 PM
US 10/26/2019





Live Share

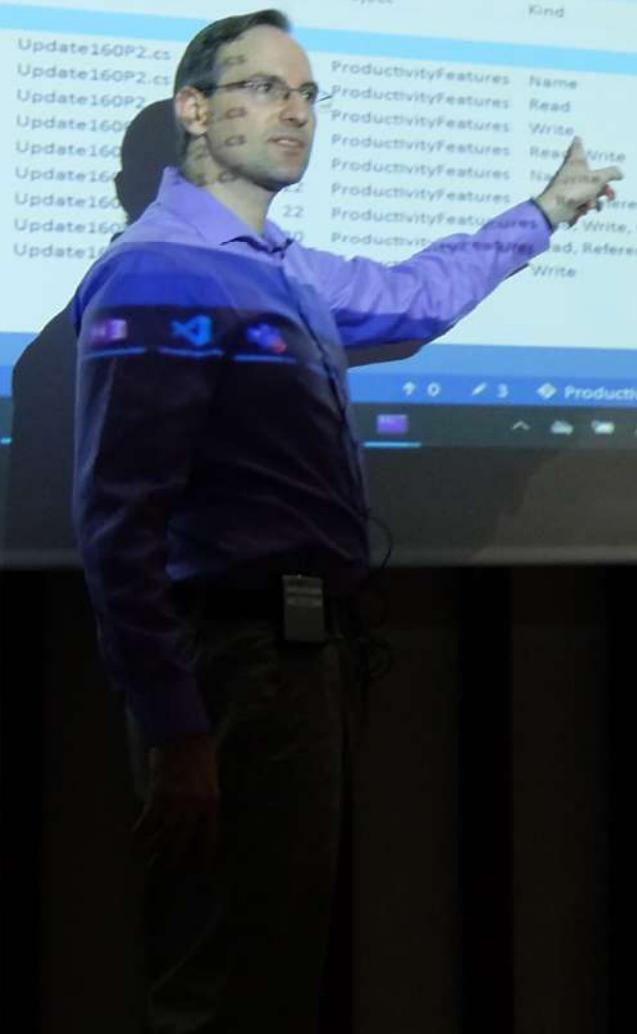
- Collaborative co-editing
- Co-debugging
- No environment set-up for the guest
- Up to 5 guests
- Read/write access
- Shared terminal
- Shared servers



```
int x = 7;
string myInterpolatedString = $@"This is my favorite number: {x}";
// Categorize references by Read/Write
// Select foo and type (Shift + F12) to open Find All References window
// Notice the new "Kind" column that lists different categories of references
public int foo;
/// <summary>
/// <see cref="foo">
/// </summary>
public void WriteToFoo()
{
    Console.WriteLine(foo);
    foo = 1;
}
```

0 references | Mika Dumont, 10 days ago [2 authors, 2 changes | 1 work item]

File	Li...	Col	Project	Kind	Search Find All References... ↗
Update160P2.cs	22	22	ProductivityFeatures	Name	U...
Update160P2.cs	22	22	ProductivityFeatures	Read	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Write	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Read/Write	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Name	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Reference	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Write, Reference	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Read, Reference	WriteToFoo U...
Update160P2.cs	22	22	ProductivityFeatures	Write	AdditionsTaken... U...



The screenshot shows a Microsoft Visual Studio IDE window with the following details:

- Title Bar:** Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q)
- Solution Explorer:** ProductivityFeatures
- Code Editor:** The file ProductivityFeatures.cs is open. The code demonstrates various productivity features of the Regex class, including classification, brace/reference matching, parser recovery, and XML tag matching.
- Status Bar:** Shows the current file (ProductivityFeatures.cs), line 19, character 42, and other standard status information.

```
using System.Linq;
using System.Runtime.Serialization.Json;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;

namespace ProductivityFeatures
{
    public class Update160P1
    {
        public void RegexParsingSupport()
        {
            // Classification and Brace/Reference Matching
            // Place cursor in a parenthesis or bracket below to see the highlighted matching
            Match match = new Regex(@"(A|[B]|C)Test\1").Match("ATestC");

            // Parser recovery enables multiple diagnostics and classification despite errors.
            // Delete a parenthesis in the Regex below to view the parenthesis warning.
            Regex removeDuplicates =
                new Regex(@"(?<=,|^)(?'word'[^,]*)(,\?'word')+(?=,|$)");
            I

            // Enabled for specifically commented strings
            // Delete the extra "r" in "rregex" in the comment below this line to get regex support.
            // language=rregex,ExplicitCapture
            const string xmlTagMatcher = @"<([A-Z][A-Z0-9]*)\b[^>]*>(.*)?</>";
            UseMatcher(xmlTagMatcher);
        }
    }
}

public void UseMatcher(string xmlTagMatcher) => throw new NotImplementedException();

// Place cursor in foreach keyword and type (Ctrl + .)
// Select 'Convert to LINQ' to convert to a query
```

Solution Explorer Notifications

```
ProductivityFeatures
4  using System.Linq;
5  using System.Runtime.Serialization.Json;
6  using System.Text;
7  using System.Text.RegularExpressions;
8  using System.Threading.Tasks;
9
10 namespace ProductivityFeatures
11 {
12     O references | kendrahavens, 177 days ago | 1 author, 2 changes
13     public class Update160P1
14     {
15         O references | kendrahavens, 248 days ago | 1 author, 1 change
16         public void RegexParsingsupport()
17         {
18             // Classification and Brace/Reference Matching
19             // Place cursor in a parenthesis or bracket below to see the highlighted matching
20             Match match = new Regex(@"(A|[B]|C)Test\1").Match("ATestC");
21
22             // Parser recovery enables multiple diagnostics and classification despite errors.
23             // Delete a parenthesis in the Regex below to view the parenthesis warning.
24             Regex removeDuplicates =
25                 new Regex(@"(?<=,|^)(?'word'[^,]*)(,\'word\')+(?=,|$)");
26
27             // Enabled for specifically commented strings
28             // Delete the extra "r" in "rregex" in the comment below
29             // language=rregex,ExplicitCapture
30             const string xmlTagMatcher = @"<([A-Z][A-Z0-9]*)\b[^>]*>";
31             UseMatcher(xmlTagMatcher);
32         }
33         1 reference | kendrahavens, 248 days ago | 1 author, 1 change
34         public void UseMatcher(string xmlTagMatcher) => throw new NotImplementedException();
35
36         // Place cursor in foreach keyword and type (Ctrl + .)
37         // Select 'Convert to LINQ' to convert to a query
38     }
39 }
```

Live Share INT PREVIEW

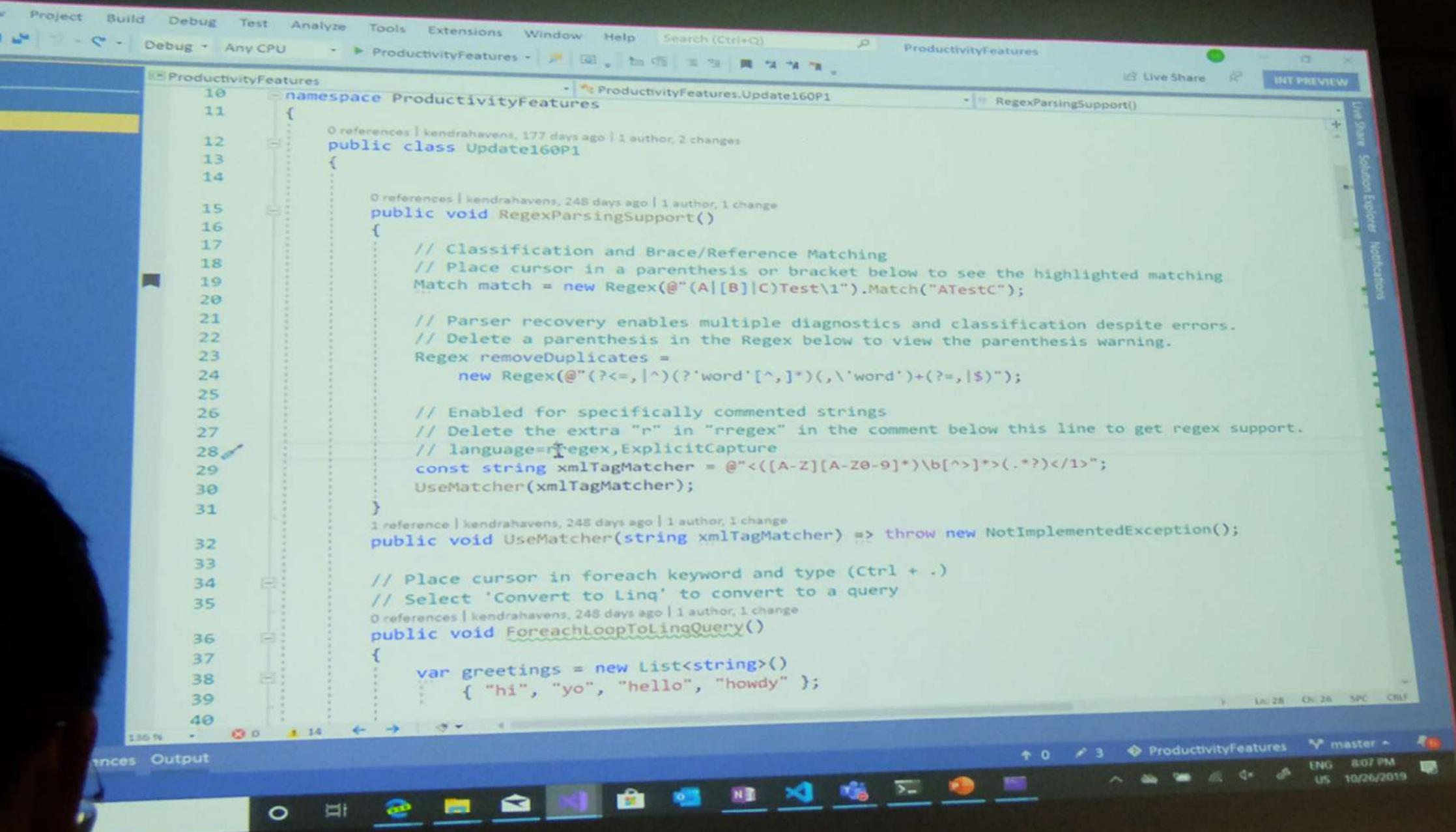
class System.String
Represents text as a sequence of UTF-16 code units.

Regex issue: Not enough ')'s

Show potential fixes (Alt+Enter or Ctrl+.)

Line 25 Ch 13 SPC CR LF

ProductivityFeatures master ENG 8:07 PM US 10/26/2019



A screenshot of a Microsoft Visual Studio IDE window showing a C# code editor. The code is part of a class named `Update160P2` within the `ProductivityFeatures` namespace. The code demonstrates various Roslyn features, including new colors and codefixes.

```
2  using System.Collections.Generic;
3  using System.Drawing;
4
5  namespace ProductivityFeatures
6  {
7      // First thing you may notice are new colors according to Roslyn classification.
8      // These colors are similar to the very popular Visual Studio Code colors.
9      // You can opt in/out of these new colors in Tools > Options > Environment > Preview Features
10     // with the checkbox "Use enhanced colors for C#".
11     // As always, any feedback is appreciated!
12
13     public void NewColors(Update160P2 parameterNewColor)
14     {
15         int localVariableNewColor = 100;
16
17         // for keyword has new color
18         for (int i = 0; i < 10; i++)
19         {
20             localVariableNewColor--; I
21             parameterNewColor--; // Overloaded operators get slightly different coloring tool!
22         }
23     }
24
25     public static Update160P2 operator --(Update160P2 x)
26     {
27         return x;
28     }
29
30     // This codefix will wrap/indent/align lists of parameters/arguments
31     // Place cursor in a parameter and type (Ctrl + .)
32     // There are several wrapping options provided

```

Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) ProductivityFeatures

Live Share INT PREVIEW

ProductivityFeatures

PullUp.MyClass PullUpMethod()

```
164
165 // Pull members up refactoring with dialog options
166 // A dialog will appear to pull the Method up into the interface
167 // Place your cursor in PullUpMethod and type (Ctrl + .)
168 // Select 'Pull members up to base type'
169 namespace PullUp
170 {
171     public interface IMyInterface
172     {
173     }
174
175     public class MyClass : IMyInterface
176     {
177
178         public void PullUpMethod()
179         {
180             Make static
181             Use expression body for methods
182
183             ...
184             void PullUpMethod();
185             ...
186             Preview changes
187             ...
188
189         }
190
191 }
```

1 reference | kendrahavens, 177 days ago | 1 author, 1 change

0 references | kendrahavens, 177 days ago | 1 author, 3 changes

0 references | kendrahavens, 248 days ago | 1 author, 1 change

as well.

JPString and type (Ctrl + .)

base type'

author, 1 change

get; set; }

Line Status Solution Explorer Notifications

od' up to 'IMyInterface' to base type...

Preview changes

Output ProductivityFeatures master ENG 8:09 PM US 10/26/2019

A screenshot of a Microsoft Visual Studio IDE window. The title bar reads "ProductivityFeatures". The menu bar includes File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, and Search (Ctrl+Q). The tabs pane shows several files: ".editorconfig", "Update160P1.cs", "Update160P2.cs" (which is selected), "Update161P1.cs", "Update161P2.cs", "Update161P3.cs", "Update162P2.cs", "Update163P1.cs", "Update163P2.cs", "Update164P1.cs", and "Update164P2.cs". The main code editor window displays the following C# code:

```
164 // Pull members up refactoring with dialog options
165 // A dialog will appear to pull the Method up into the interface
166 // Place your cursor in PullUpMethod and type (Ctrl + .)
167 // Select 'Pull members up to base type'
168 namespace PullUp
169 {
170     public interface IMyInterface
171     {
172     }
173 }
174 public class MyClass : IMyInterface
175 {
176     public void PullUpMethod()
177     {
178         ...
179         ...
180         ...
181         ...
182         ...
183         ...
184         ...
185         ...
186         ...
187         ...
188         ...
189         ...
190         ...
191 }
```

A context menu is open over the line "public void PullUpMethod()", showing options: "Make static", "Use expression body for methods", "Preview changes", and "as well. JpString and type (Ctrl + .) > base type' author, 1 change get; set; }". The status bar at the bottom shows "Output" and various icons.

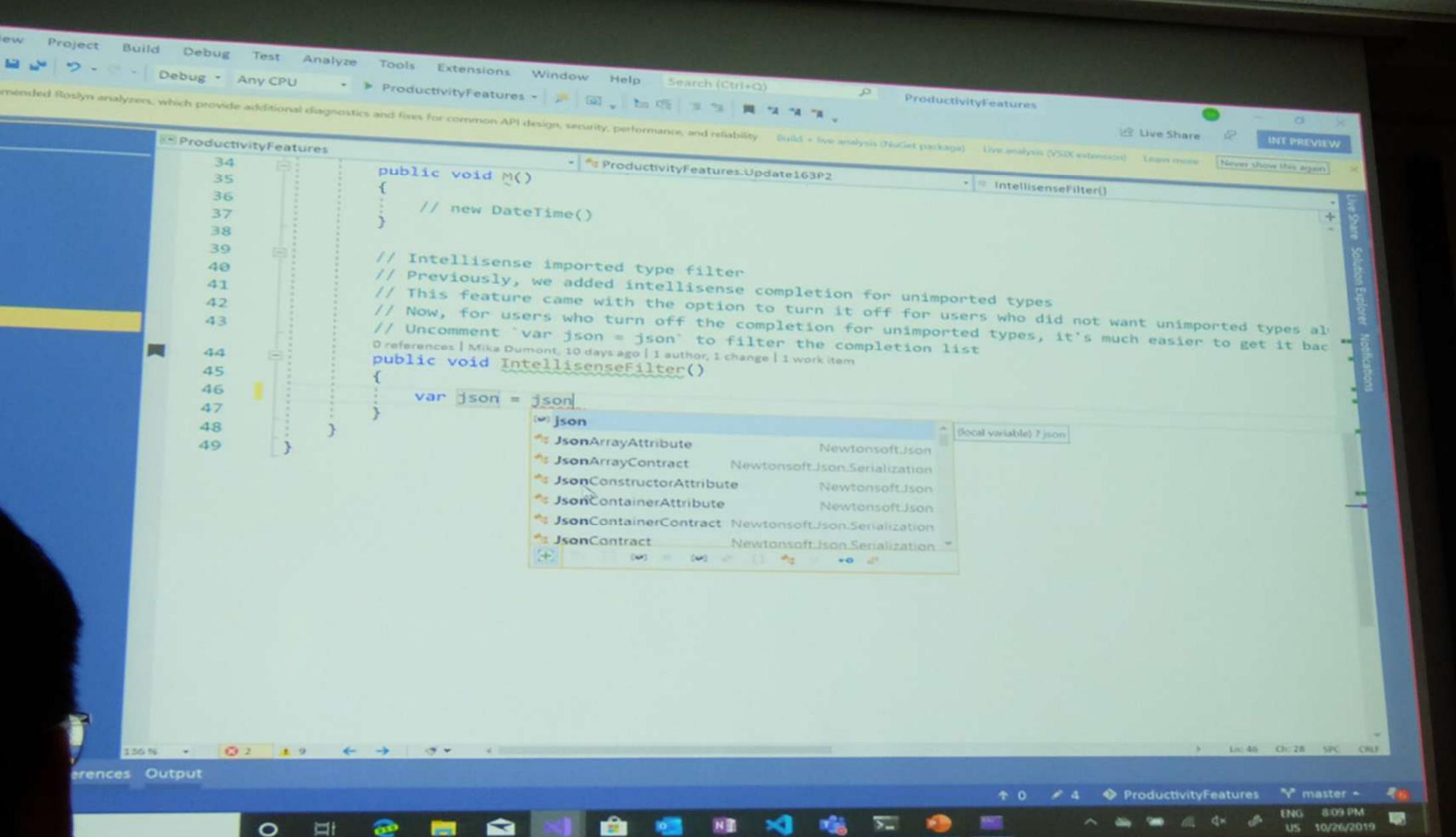
A screenshot of the Microsoft Visual Studio IDE interface. The main window shows a code editor with the following C# code:

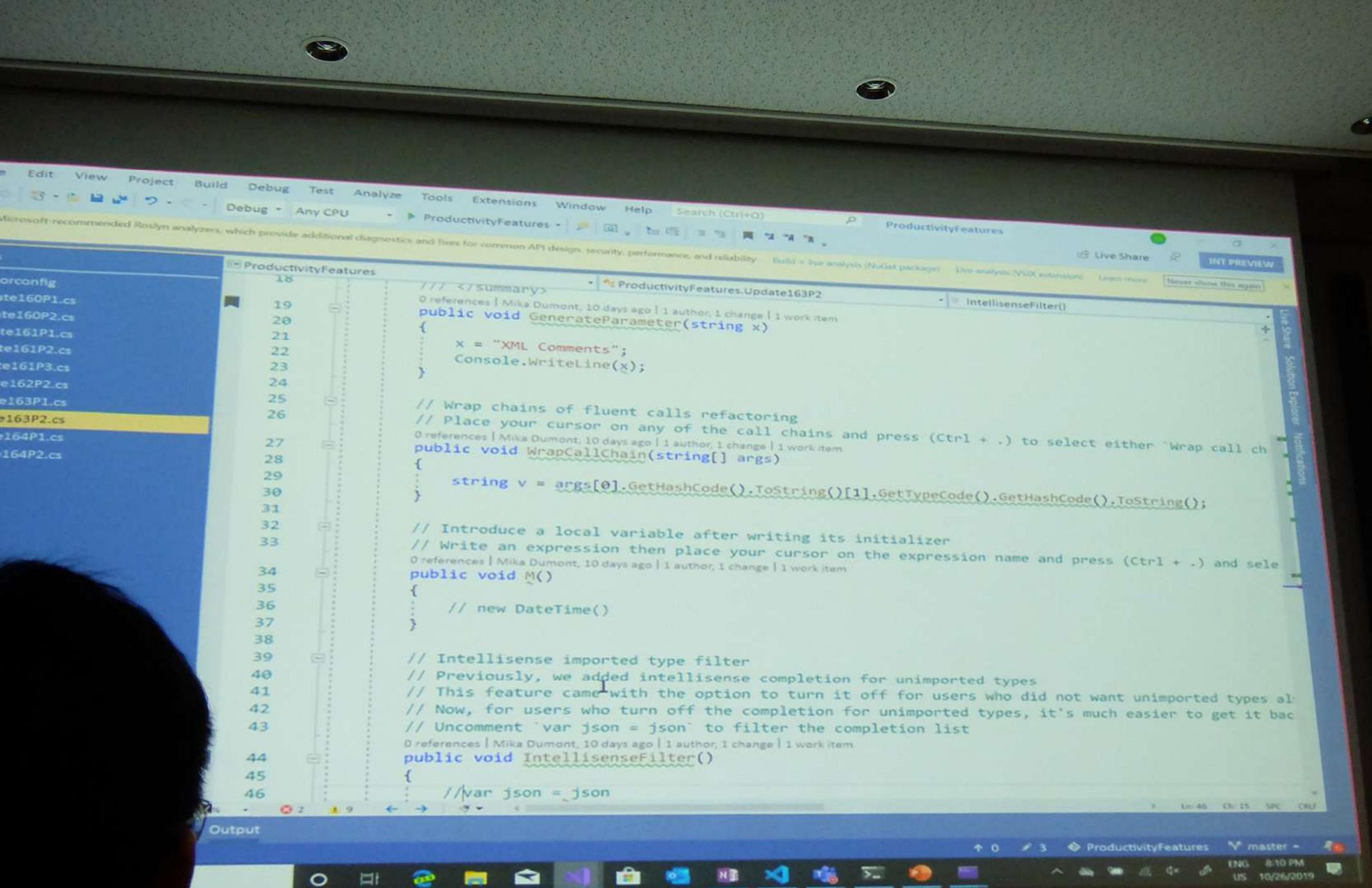
```
30 }  
31 }  
32 }  
33 }  
34 // Introduce a local variable after writing its initializer  
35 // Write an expression then place your cursor on the expression name and press (Ctrl + .) and sele  
36 public void M()  
37 {  
38     // new DateTime()  
39  
40     // Intellisense imported type filter  
41     // Previously, we added intellisense completion for unimported types  
42     // This feature came with the option to turn it off for users who did not want unimported types al  
43     // Uncomment `var json = json` to filter the completion list  
44     // 0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item  
45     public void IntellisenseFilter()  
46     {  
47         var json = _ I  
48     }  
49 }
```

The code editor has several annotations and tool tips displayed above the code lines:

- Line 33: *// Introduce a local variable after writing its initializer*
- Line 34: *// Write an expression then place your cursor on the expression name and press (Ctrl + .) and sele*
- Line 43: *0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item*
- Line 44: *// Intellisense imported type filter*
- Line 45: *// Previously, we added intellisense completion for unimported types*
- Line 46: *// This feature came with the option to turn it off for users who did not want unimported types al*
- Line 47: *// Now, for users who turn off the completion for unimported types, it's much easier to get it bac*
- Line 48: *// Uncomment `var json = json` to filter the completion list*
- Line 49: *0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item*

The status bar at the bottom shows the following information: 236%, 2, 9, ←, →, ⇤, ⇥, ↵, Line: 46, Ch: 24, SPC, CRLF.





File Edit View Project Build Debug Test Analyze Tools

Debug Any CPU

1 Install Microsoft-recommended Roslyn analyzers, which provide additional diagnostics and issues.

Server Explorer Toolbox

Tabs

- .editorconfig
- Update160P1.cs
- Update160P2.cs
- Update161P1.cs
- Update161P2.cs
- Update161P3.cs
- Update162P2.cs
- Update163P1.cs
- Update163P2.cs
- Update164P1.cs
- Update164P2.cs

ProductivityFeatures

```
5
6     namespace
7     {
8         public
9         {
10            ...
11            ...
12            ...
13            ...
14            ...
15            ...
16            ...
17            ...
18            ...
19            ...
20            ...
21            ...
22            ...
}
```

```
8 {  
9     0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item  
10    public class Update163P2  
11    {  
12        // Quick Info style support for XML comments  
13        // Place your cursor over the method name. Quick Info will then di...  
14        /// <summary>  
15        /// <see href="https://bing.com">Bing</see>  
16        /// <list type="bullet">  
17        /// <item><description><para>Line</para></description></item>  
18        /// <item><description><para>Line</para></description></item>  
19        /// </list>  
20        /// </summary>  
21        0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item  
22        public void GenerateParameter(string x)  
23        {  
24            x = "XML Comment";  
25            Console.WriteLine(x);  
26        }  
27  
28        // Wrap chains of f...  
29        // Place your cursor...  
30        0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item  
31        public void WrapCallChain(string[] args)  
32        {  
33            string v = args[0].GetHashCode().ToString()[1].GetTypeCode();  
34        }  
35  
36        // Introduce a local variable after writing its initializer  
37        // Write an expression then place your cursor on the expression  
38        0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item
```

Quick Info style support for XML comments
Place your cursor over the method name. Quick Info will then display the supported st

```
<summary>
<see href="https://bing.com">Bing</see>
<list type="bullet">
<item><description><para>Line</para></description></item>
<item><description><para>Line</para></description></item>
</list>
</summary>
```

Sources | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item

```
ic void GenerateParameter(string x)
```

I = "XML Comment"
Console.WriteLine("XML Comment")

- void Update163P2.GenerateParameter(string x)
- Bing
- Line
- Line

Member GenerateParameter does not access instance data and can be marked as static (Shared in VisualBasic)
Show potential fixes (Alt+Enter or Ctrl+.)

Sources | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item

```
ic void WrapCallChain(string[] args)
```

String v = args[0].GetHashCode().ToString()[1].GetTypeCode().GetHashCode().ToStrin

Introduce a local variable after writing its initializer
Write an expression then place your cursor on the expression name and press (Ctrl

Sources | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item

↑ 0 ↗ 3 ◆ ProductivityFeature

The screenshot shows a Visual Studio code editor with the following code:

```
ProductivityFeatures.cs
4  using System.Security.Cryptography.X509Certificates;
5
6  namespace ProductivityFeatures
7  {
8      public class Update164P1
9      {
10         // Configure severity level of a code style rule directly through the editor
11         // Place your cursor on the error, warning, or suggestion and type (Ctrl + .) and select 'Configure'
12         // Then select the rule and choose the severity level you would like to configure for that rule
13         // This will update your existing EditorConfig with the rule's new severity
14         // This also works for third party analyzers
15
16         public void M()
17         {
18             var v = string.Empty;
19         }
20
21         // Add null checks for all parameters
22         // Place your cursor on any parameter within the method
23         // Press (Ctrl + .)
24         public void Addt
25             Make static
26             Add null check
27             Add 'string.IsNullOrEmpty' check
28             Add 'string.IsNullOrWhiteSpace' check
29             Add null checks for all p
30             Add null checks for all parameters
31             Change signature...
32             Wrap every parameter
33             Unwrap and indent all parameters
34             Configure or Suppress issues
```

A context menu is open at line 29, with "Add null checks for all parameters" selected. A tooltip for this option is visible, stating: "Add null checks for all parameters". The status bar at the bottom right shows: "Ln: 24 Ch: 42 SPC CRLF".

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q)

Debug Any CPU ProductivityFeatures ProductivityFeatures.Update164P1

Install Microsoft-recommended Roslyn analyzers, which provide additional diagnostics and fixes for common API design, security, performance, and reliability Build - live analysis

Server Explorer Toolbox

Tabs .editorconfig Update160P1.cs Update160P2.cs Update161P1.cs Update161P2.cs Update161P3.cs Update162P2.cs Update163P1.cs

None Silent Suggestion Warning Error

CA1822 Member M does not access instance data and can be marked as static (Shared in VisualBasic)

csharp_style_unused_value_assignment_preference = discard_variable:none
CA1822: Mark members as static
dotnet_diagnostic.CA1822.severity = none

Preview changes

```
10 // Configure severity level of a code style rule
11 // Place your cursor on the error, warning, or s
12 // Then select the rule and choose the severity
13 // This will update your existing EditorConfig v
14 // This also works for third party analyzers
15 0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item
16 public void M()
17     = string.Empty;
18     Make static
19
20     for all parameters
21     r on any parameter wihtin t
22     and select the option to
23     0 days ago | 1 author, 1 change | 1 work item
24     Checks(string a, string b,
25
26     ...
27     ...
28     }
29
30     // Go To Base command
31     // You can now use the Go To Base command to navi
32     // Go To Base command is available on the context
33     // Place your cursor on class D and press (Alt +
34     // If there is more than one result a tool window
35     2 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item
36     internal interface I
37     {
38 }
```

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q)

Debug Any CPU ProductivityFeatures

Install Microsoft-recommended Roslyn analyzers, which provide additional diagnostics and fixes for common API design, security, performance, and reliability.

ProductivityFeatures.cs

ProductivityFeatures.Update164P1

10 // Configure severity level of a code style rule di
11 // Place your cursor on the error, warning, or sug
12 // Then select the rule and choose the severity le
13 // This will update your existing EditorConfig wit
14 // This also works for third party analyzers
15 0 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item
16 public void M()
17 = string.Empty;

18 Make static

19 CA1822 Member M does not access instance data and can be marked as
20 static (Shared in VisualBasic)
21 ...
22 csharp_style_unused_value_assignment_preference = discard_variable:none
23 # CA1822: Mark members as static
24 dotnet_diagnostic.CA1822.severity = none

for all parameters
on any parameter within the
and select the option to 'Add'
0 days ago | 1 author, 1 change | 1 work item
Checks(string a, string b, str

Preview changes

26
27
28 }
29
30 // Go To Base command
31 // You can now use the Go To Base command to navigate
32 // Go To Base command is available on the context menu
33 // Place your cursor on class D and press (Alt + Home)
34 // If there is more than one result a tool window will appear
2 references | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item
35 internal interface I
36 {
37 }
38 1 reference | Mika Dumont, 10 days ago | 1 author, 1 change | 1 work item

```
// Pass a variable explicitly into a local static function
// Place your cursor on the variable name 'x' in the static local function
// Press (Ctrl + .) and select 'Pass variable explicitly in local static function'
static int AddLocal()
{
    return x += 1;
}

return s;
}
0 references | 0 changes | 0 authors, 0 changes
private void NoOneCalledMeThough()
{
}
.
.
.
}
0 references | 0 changes | 0 authors, 0 changes
private void FxCopAnalyzers(ImmutableArray<string> array)
{
    // now while you type!
    // now with fixes! (gotta fix 'em all)
    var count = array.Count();
}
```

Use Length/Count property instead of Count() when available >
Use explicit type instead of 'var'
Configure or Suppress issues

CA1829 Use the "Length" property instead of
Enumerable.Count().

```
...
// now with fixes! (gotta fix 'em all)
var count = array.Count();
var count = array.Length;
```

Preview changes

Fix all occurrences in: Document | Project | Solution

Line: 45 Ch: 35 SPC CRLF

ProductivityFeatures master ENG 8:13 PM
US 10/26/2019

```
27 // Pass a variable explicitly into a local static function
28 // Place your cursor on the variable name 'x' in the static local function
29 // Press (Ctrl + .) and select 'Pass variable explicitly in local static function'
30 static int AddLocal()
31 {
32     return x += 1;
33 }
34
35     return s;
36 }

37 0 references | 0 changes | 0 authors, 0 changes
38 private void NoOneCalledMeThough()
39 {
40 }
41 0 references | 0 changes | 0 authors, 0 changes
42 private void FxCopAnalyzers(ImmutableArray<string> array)
43 {
44     // now while you type!
45     // now with fixes! (gotta fix 'em all)
46     var count = array.Count();
```

Use Length/Count property instead of Count() when available
Use explicit type instead of 'var'
Configure or Suppress issues

CA1829 Use the "Length" property instead of
Enumerable.Count().

```
... // now with fixes! (gotta fix 'em all)
var count = array.Count();
var count = array.Length;
```

Preview changes

Fix all occurrences in: Document | Project | Solution

Output

Ln: 45 Ch: 35 SPC: CBL

ENG 8:13 PM
US 10/26/2019

A screenshot of the Microsoft Visual Studio 2019 IDE interface. The main window shows a C# code editor with several test methods defined in a file named `TwitterDataModelTests.cs`. The code includes annotations like `[TestMethod]` and `[TestMethod]`, and contains references to historical tweet data. A tooltip is visible in the bottom right corner of the code editor area. The status bar at the bottom right shows the date and time as "TUE 8:14 PM US 10/26/2019".

```
15 }
16 
17 [TestMethod]
18 // References |历史文档, 27 天前 | 2 作者, 0 变更
19 public void HistoricalTweetTest()
20 {
21     Alert a = new Alert();
22 }
23 
24 [TestMethod]
25 // References | 历史更改, 263 天前 | 1 作者, 1 变更
26 public void GetTweetSentimentTest()
27 {
28     I
29 }
30 
31 [TestMethod]
32 // References | 历史更改, 263 天前 | 1 作者, 1 变更
33 public void AverageTweetTest()
34 {
35 }
36 
37 [TestMethod]
38 // References | 历史更改, 263 天前 | 1 作者, 1 变更
39 public void MultipleAverageTweetTest()
40 {
41 }
42 
43 [TestMethod]
44 // References | 历史更改, 263 天前 | 1 作者, 1 变更
45 public void ActiveCognitiveServiceTest()
46 {
```

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) UserSentiment

UserSentimentAnalysis

IIS Express (Microsoft Edge Dev)

TwitterDataModelTests.cs

```
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
```

No issues found

[TestMethod]
0 references | Mika Dumont, 37 days ago | 2 authors, 3 changes
public void HistoricalTweetTest()
{
.. Alert m = new Alert();
}

[TestMethod]
0 references | kendrahavens, 263 days ago | 1 author, 1 change
public void GetTweetSentimentTest()
{
.. I
}

[TestMethod]
0 references | kendrahavens, 263 days ago | 1 author, 1 change
public void AverageTweetTest()
{
..
}

[TestMethod]
0 references | kendrahavens, 263 days ago | 1 author, 1 change
public void MultipleAverageTweetTest()
{
..
}

[TestMethod]
0 references | kendrahavens, 263 days ago | 1 author, 1 change
public void ActiveCognitiveServiceTest()
{

Visual Studio 2019 update
Version 16.4.0 Preview 3.0 [29] is downloaded and ready to in

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) UserSentiment

UserSentimentAnalysis

IIS Express (Microsoft Edge Dev)

UserSentimentAnalysis.Tests.TwitterDataModelTests

UserSentimentAnalysis.Tests

TwitterDataModelTests.cs*

```
15 }  
16  
17 [TestMethod]  
18 0 references | Mika Dumont, 37 days ago | 2 authors, 3 changes  
19 public void HistoricalTweetTest()  
20 {  
21     Alert m = new Alert();  
22 }  
23  
24 [TestMethod]  
25 0 references | kendrahavens, 263 days ago | 1 author, 1 change  
26 public void GetTweetSentimentTest()  
27 {  
28     string s = "steve";  
29     if (s.  
30         [TestMethod]  
31 0 references | ke  
32 public void  
33 {  
34 }  
35  
36 [TestMethod]  
37 0 references | kendrahavens, 263 days ago | 1 author, 1 change  
38 public void MultipleAverageTweetTest()  
39 {  
40 }  
41  
42 [TestMethod]  
43 0 references | kendrahavens, 263 days ago | 1 author, 1 change  
44 public void ActiveCognitiveServiceTest()
```

Length Equals StartsWith Contains Clone CompareTo Contains

int string.Length { get; } Gets the number of characters in the current string object.

★ IntelliCode suggestion based on this context

ago | 1 author, 1 change :Test()

Visual Studio Version is down

```
29 [TestMethod]  
30 0 references | kendrahavens, 263 days ago | 1 author, 1 change  
31     public void AverageTweetTest()  
32     {  
33         Assert.Fail();  
34     }  
35 [TestMethod]  
36 0 references | kendrahavens, 263 days ago | 1 author, 1 change  
37     public void MultipleAverageTweetTest()  
38     {  
39     }  
40 ★ IntelliCode refactorings suggested edit:  
41     ( Assert.Fail(); )  
42     Show potential fixes (Alt+Enter or Ctrl+. ) days ago | 1 author, 1 change  
put
```



```
26     Assert.Fail();
27 }
28
29 [TestMethod]
30 0 references | kendrahavens, 263 days ago | 1 author, 1 change
31 public void AverageTweetTest()
32 {
33     Assert.Fail();
34 }
35
36 [TestMethod]
37 0 references | kendrahavens, 263 days ago | 1 author, 1 change
38 public void MultipleAverageTweetTest()
39 {
```

★ IntelliCode refactorings suggested edit: { Assert.Fail(); }

Ignore Suggestion

136%

42

No issues found

```
...
public void Multipl
{
}
{ Assert.Fail(); }
...
```



```
[TestMethod]
```

```
0 references | kendrahavens, 263 days ago | 1 author, 1 change  
public void AverageTweetTest()  
{  
    ...  
    Assert.Fail();  
}
```

```
[TestMethod]
```

```
0 references | kendrahavens, 263 days ago | 1 author, 1 change  
public void MultipleAverageTweetTest()  
{  
    ...  
}
```



★ IntelliCode refactorings suggested edit: { Assert.Fail(); }

Ignore Suggestion

public void MultipleAverageTweetTest()

issues found

```
...  
public void MultipleAverageTweetTest()  
{  
    ...  
    Assert.Fail();  
    ...
```



File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) UserSentiment

Debug Any CPU UserSentimentAnalysis IIS Express (Microsoft Edge Dev) Live Share INT PREVIEW

diCode found more locations where your edits may apply - see error list messages category More info Don't show again

Tabs .editorconfig TwitterDataModelTests.cs*

```
7 ##### Core EditorConfig Options #####
8
9 # Indentation and spacing
10 indent_size = 4
11 indent_style = space
12 tab_width = 4
13
14 # New line preferences
15 end_of_line = crlf
16 insert_final_newline = false
17
18 ##### .NET Coding Conventions #####
19
20 # Organize usings
21 dotnet_separate_import_directive_groups = false
22 dotnet_sort_system_directives_first = false
23
24 # this. and Me. preferences
25 dotnet_style_qualification_for_event = false:silent
26 dotnet_style_qualification_for_field = false:silent
27 dotnet_style_qualification_for_method = false:silent
28 dotnet_style_qualification_for_property = false:silent
29
30 # Language keywords vs BCL types preferences
31 dotnet_style_predefined_type_for_locals_parameters_members = true:silent
32 dotnet_style_predefined_type_for_member_access = true:silent
33
34 # Parentheses preferences
35 dotnet_style_parentheses_in_arithmetic_binary_operators = always_for_clarity:silent
36 dotnet_style_parentheses_in_other_binary_operators = always_for_clarity:silent
37 dotnet_style_parentheses_in_other_operators = never_if_unnecessary:silent
38 dotnet_style_parentheses_in_relational_binary_operators = always_for_clarity:silent
39
40 # Modifier preferences
41 dotnet_style_modifier_order = none:silent
```

136% 0 changes | authors | changes | 4

Visual Studio 2019 update
Version 16.4.0 Preview 3.0 [29425.166.master] is downloaded and ready to install.

UserSentimentAnalysis master 8:18 PM 10/26/2023

