



ASP.NET CORE WEB API

Skapa ett Web API helt från grunden



Visual Studio 2019

Open recent

As you use Visual Studio, any projects, folders, or files that you open will show up here for quick access.

You can pin anything that you open frequently so that it's always at the top of the list.

Get started



Clone or check out code

Get code from an online repository like GitHub or Azure DevOps



Open a project or solution

Open a local Visual Studio project or .sln file



Open a local folder

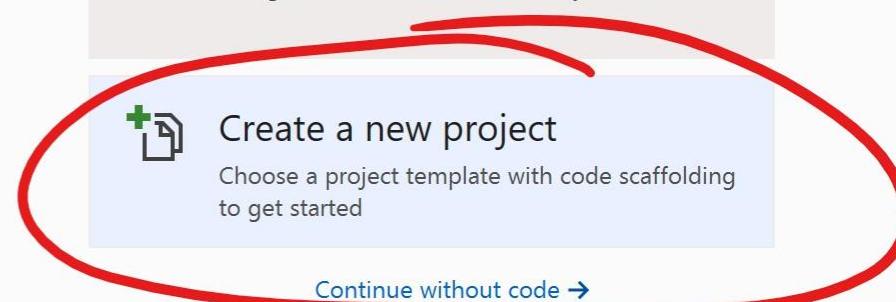
Navigate and edit code within any folder



Create a new project

Choose a project template with code scaffolding to get started

[Continue without code →](#)



X

Create a new project

Recent project templates



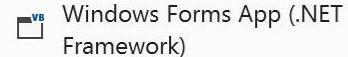
Azure Functions

C#



Console App (.NET Core)

C#



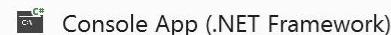
Windows Forms App (.NET Framework)

Visual Basic



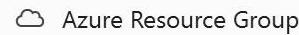
ASP.NET Core Web Application

C#



Console App (.NET Framework)

C#



Azure Resource Group

C#



ASP.NET Web Application (.NET Framework)

C#



ASP.NET Web Application (.NET Framework)

Visual Basic

Search for templates (Alt+S)



Clear all

C#

All platforms

All project types



Console App (.NET Core)

A project for creating a command-line application that can run on .NET Core on Windows, Linux and MacOS.

C#

Linux

macOS

Windows

Console



ASP.NET Core Web Application

Project templates for creating ASP.NET Core web apps and web APIs for Windows, Linux and macOS using .NET Core or .NET Framework. Create web apps with Razor Pages, MVC, or Single Page Apps (SPA) using Angular, React, or React + Redux.

C#

Linux

macOS

Windows

Cloud

Service

Web



Blazor App

Project templates for creating Blazor apps that run on the server in an ASP.NET Core app or in the browser on WebAssembly. These templates can be used to build web apps with rich dynamic user interfaces (UIs).

C#

Linux

macOS

Windows

Cloud

Web



Class Library (.NET Standard)

A project for creating a class library that targets .NET Standard.

C#

Android

iOS

Linux

macOS

Windows

Library

Back

Next



Configure your new project

ASP.NET Core Web Application

C#

Linux

macOS

Windows

Cloud

Service

Web

Project name

WebApi

Location

C:\Users\HansMattiin-Lassei\source\repos



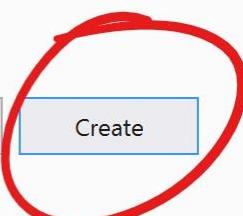
Solution name 

WebApi

Place solution and project in the same directory

Back

Create



Create a new ASP.NET Core web application

.NET Core ▾



Empty

An empty project template for creating an ASP.NET Core application. This template does not have any content in it.



API

A project template for creating an ASP.NET Core application with an example Controller for a RESTful HTTP service. This template can also be used for ASP.NET Core MVC Views and Controllers.



Web Application

A project template for creating an ASP.NET Core application with example ASP.NET Razor Pages content.



Web Application (Model-View-Controller)

A project template for creating an ASP.NET Core application with example ASP.NET Core MVC Views and Controllers. This template can also be used for RESTful HTTP services.



Angular

A project template for creating an ASP.NET Core application with Angular



React.js

[Get additional project templates](#)

Authentication

No Authentication

[Change](#)

Advanced

Configure for HTTPS

Enable Docker Support

(Requires [Docker Desktop](#))

Linux

Author: Microsoft

Source: .NET Core 3.1.1

Back

Create



Startup.cs WebApi

WebApi.WebApi

WebApi.Startup

Configure(IApplicationBuilder app, IWebHostEnvironment env)

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using Microsoft.AspNetCore.Builder;
6  using Microsoft.AspNetCore.Hosting;
7  using Microsoft.AspNetCore.Http;
8  using Microsoft.Extensions.DependencyInjection;
9  using Microsoft.Extensions.Hosting;
10 
11 namespace WebApi
12 {
13     public class Startup
14     {
15         // This method gets called by the runtime. Use this method to add services to the container.
16         // For more information on how to configure your application, visit https://go.microsoft.com/fwlink/?LinkID=398940
17         public void ConfigureServices(IServiceCollection services)
18         {
19         }
20 
21         // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
22         public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
23         {
24             if (env.IsDevelopment())
25             {
26                 app.UseDeveloperExceptionPage();
27             }
28 
29             app.UseRouting();
30 
31             app.UseEndpoints(endpoints =>
32             {
33                 endpoints.MapGet("/", async context =>
34                 {
35                     await context.Response.WriteAsync("Hello World");
36                 });
37             });
38         }
39     }
40 }
41 
```

90 % No issues found | Ln: 31 Ch: 42 SPC CRLF

Live Share

Solution Explorer

Properties

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties
- appsettings.json
- Program.cs
- Startup.cs

Toolbox

Server Explorer

SQL Server Object Explorer

Package Manager Console Error List ... Output Web Publish Activity

Add to Source Control

Ready

5



Startup.cs* WebApi

WebApi WebApi

WebApi.Startup

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5 using Microsoft.AspNetCore.Builder;
6 using Microsoft.AspNetCore.Hosting;
7 using Microsoft.AspNetCore.Http;
8 using Microsoft.Extensions.DependencyInjection;
9 using Microsoft.Extensions.Hosting;
10
11 namespace WebApi
12 {
13     public class Startup
14     {
15         public void ConfigureServices(IServiceCollection services)
16         {
17             // Configure services here
18         }
19     }
20 }
```

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

- WebApi
 - Connected Services
 - Dependencies
 - Properties
 - appsettings.json
 - Program.cs
 - Startup.cs

Toolbox

Server Explorer

SQL Server Object Explorer

90 % No issues found

Ln: 16 Ch: 9 SPC CRLF

Solution Explorer Team Explorer

Package Manager Console Error List ... Output Web Publish Activity

Ready Add to Source Control

Live Share

Properties

A screenshot of the Visual Studio IDE interface. The main window displays the `Startup.cs` file for a project named `WebApi`. The code implements the `IConfiguration` interface and its constructor, setting the `Configuration` property to the provided `configuration`.

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using Microsoft.AspNetCore.Builder;
6  using Microsoft.AspNetCore.Hosting;
7  using Microsoft.AspNetCore.Http;
8  using Microsoft.Extensions.Configuration;
9  using Microsoft.Extensions.DependencyInjection;
10 using Microsoft.Extensions.Hosting;
11
12 namespace WebApi
13 {
14     public class Startup
15     {
16         public IConfiguration Configuration { get; }
17
18         public Startup(IConfiguration configuration)
19         {
20             Configuration = configuration;
21         }
22
23     }
24 }
```

Annotations in red ink have been added to the code:

- A red arrow points from the `Configuration` field declaration at line 16 to the `configuration` parameter in the constructor at line 18.
- A red arrow points from the `configuration` parameter in the constructor at line 18 to the assignment statement at line 20.
- A large curly brace on the right side of the code block groups the `Configuration` field and the constructor, with the handwritten text "lags f. l." written above it.
- A small handwritten note "do" is written near the bottom of the brace.

The Solution Explorer on the right shows the project structure with files like `appsettings.json`, `Program.cs`, and `Startup.cs`.



Startup.cs* X WebApi

WebApi

WebApi.Startup

ConfigureServices(IServiceCollection services)

```
16     1 reference
17     public IConfiguration Configuration { get; }
18
19     0 references
20     public Startup(IConfiguration configuration)
21     {
22         Configuration = configuration;
23     }
24
25     0 references
26     public void ConfigureServices(IServiceCollection services)
27     {
28         services.AddControllers();
29     }
30
31 }
32
```

Search Solution Explorer (Ctrl+Shift+F) X

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties
- appsettings.json
- Program.cs
- Startup.cs

Properties

132 % X No issues found X

Ln: 27 Ch: 10 SPC CRLF

Solution Explorer Team Explorer

Package Manager Console Error List ... Output Web Publish Activity

Ready Add to Source Control X 5

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

- Top Bar:** File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q), WebApi.
- Toolbar:** Standard icons for file operations like Open, Save, Print, etc.
- Status Bar:** Debug, Any CPU, IIS Express, Live Share.
- Solution Explorer:** Shows the solution 'WebApi' with one project 'WebApi' containing files: Connected Services, Dependencies, Properties, appsettings.json, Program.cs, and Startup.cs.
- Properties:** Standard properties pane.
- Code Editor:** The file 'Startup.cs' is open. The code defines two methods:

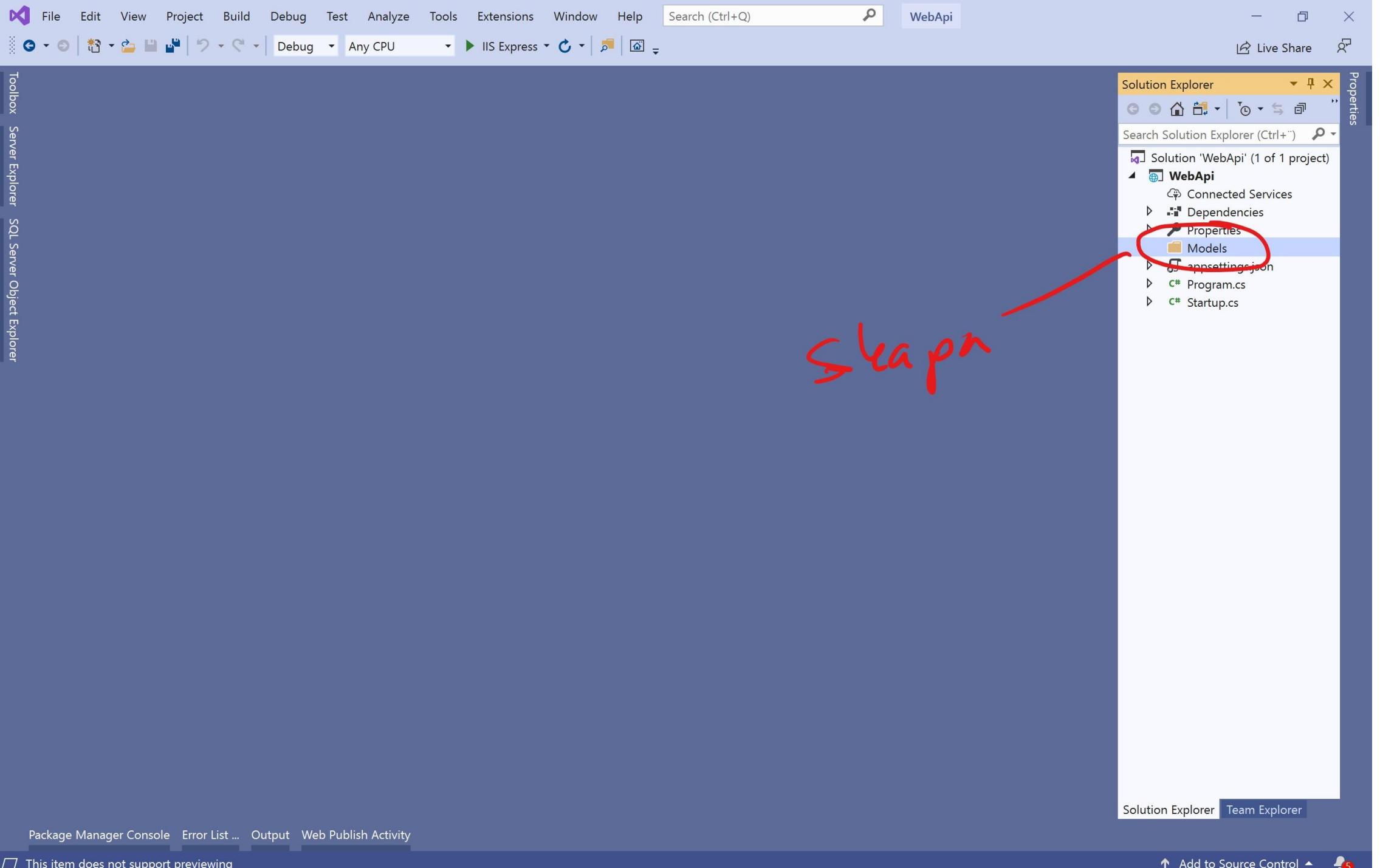
```
public void ConfigureServices(IServiceCollection services)
{
    services.AddControllers();
}

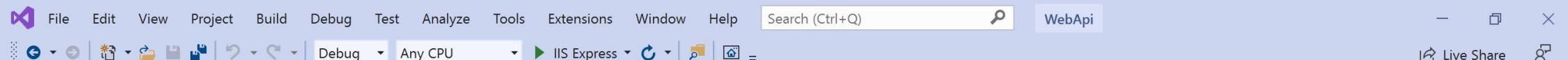
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    app.UseRouting();

    app.UseAuthorization();

    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllers();
    });
}
```

A large red rectangular box highlights the entire body of the `Configure` method, from the opening brace at line 30 to the closing brace at line 42.
- Toolbox:** Standard .NET development tools.
- Bottom Bar:** Package Manager Console, Error List ..., Output, Web Publish Activity, Solution Explorer, Team Explorer.





Solution Explorer

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties

Models
appsettings.json
Program.cs
Startup.cs

View in Browser (Microsoft Edge)
Browse With...

Add

- Scope to This
- New Solution Explorer View
- Exclude From Project

Cut (Ctrl+X)
Copy (Ctrl+C)
Delete (Del)

Rename

Open Folder in File Explorer

Properties (Alt+Enter)

Class... (Shift+Alt+C)

Toolbox

Server Explorer

SQL Server Object Explorer

Package Manager Console

Error List ...

Output

Web Publish Activity

Live Share

Properties

Solution Explorer Team Explorer

A red arrow points from the bottom left towards the "Class..." option in the context menu.

The screenshot shows the Visual Studio 2022 interface with the "WebApi" project selected in the Solution Explorer. A context menu is open over the "Models" item in the project tree. The menu includes options for viewing the item in a browser, adding new items, cutting, copying, deleting, renaming, opening in the file explorer, and viewing properties. At the bottom of the menu, the "Class..." option is highlighted with a red arrow pointing towards it from the bottom left. The status bar at the bottom shows "Ready" and "Add to Source Control".

A screenshot of the Microsoft Visual Studio IDE interface. The main window shows a code editor with the file `Product.cs` open, containing the following C# code:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5
6 namespace WebApi.Models
7 {
8     public class Product
9     {
10    }
11 }
12
```

The Solution Explorer panel on the right shows the project structure for 'WebApi' (1 of 1 project), which includes the `Models` folder containing `Product.cs`, and files like `appsettings.json`, `Program.cs`, and `Startup.cs`. The `Properties` node is also visible.

At the bottom of the screen, there are several status bars and toolbars. The status bar at the very bottom includes the following information: `120 %`, `No issues found`, `Ln: 1 Ch: 1 SPC CRLF`, `Solution Explorer`, `Team Explorer`, and a notification icon with the number `5`.



WebApi

Product.cs* ▾ X

WebApi

WebApi.Models.Product

Product(int id, string name, string description, double price)

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5
6  namespace WebApi.Models
7  {
8      public class Product
9      {
10         public int Id { get; set; }
11         public string Name { get; set; }
12         public string Description { get; set; }
13         public double Price { get; set; }
14
15         //constructors
16
17         public Product()
18         {
19         }
20
21
22         public Product(int id, string name, string description, double price)
23         {
24             Id = id;
25             Name = name;
26             Description = description;
27             Price = price;
28         }
29     }
30 }
31
```

109 % ▾ No issues found

Solution Explorer ▾ Team Explorer

Live Share

Properties

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties
- Models
- Product.cs
- appsettings.json
- Program.cs
- Startup.cs

Toolbox

Server Explorer

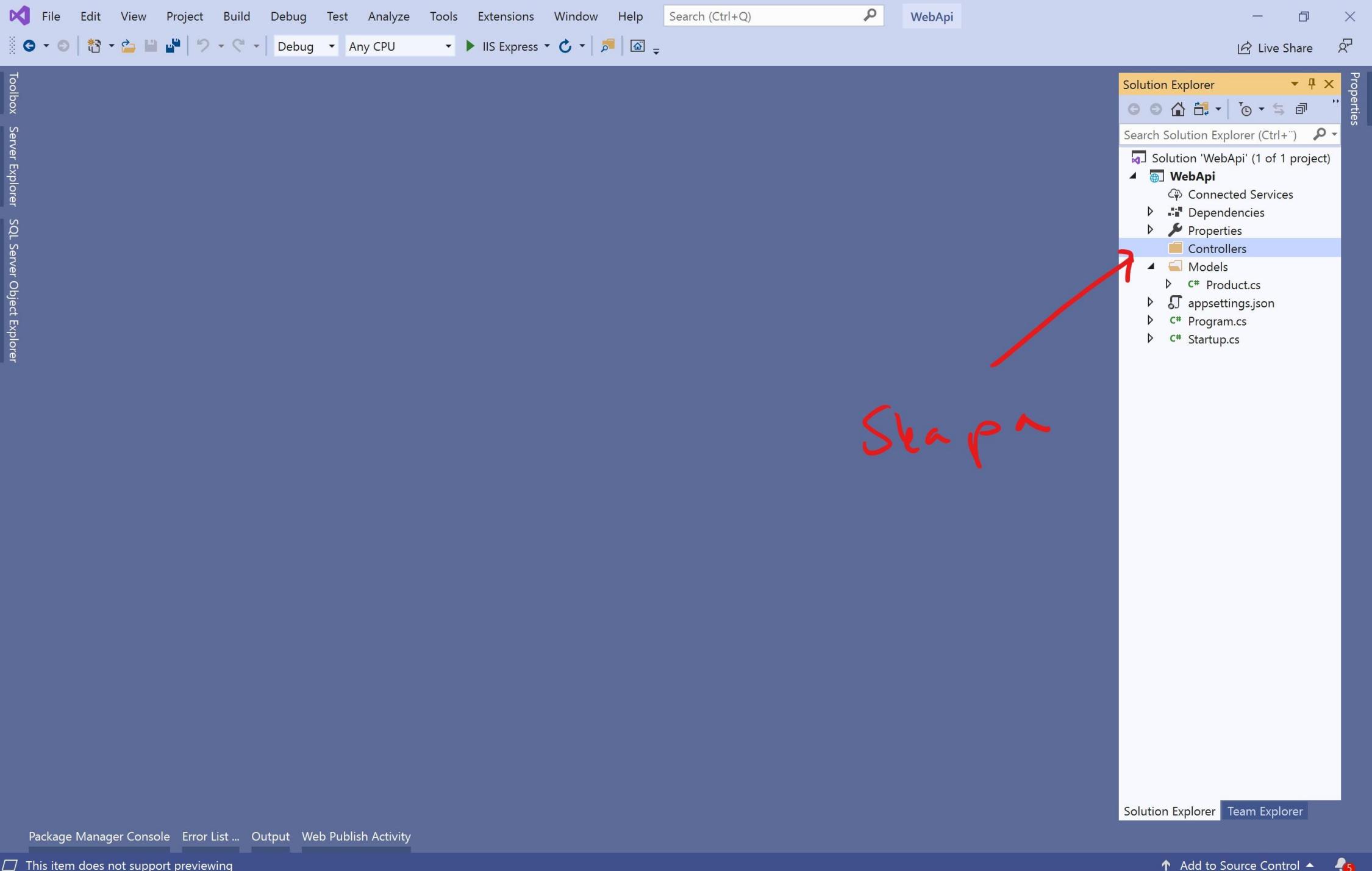
SQL Server Object Explorer

Package Manager Console Error List ... Output Web Publish Activity

Ready

Add to Source Control

5



View in Browser (Microsoft Edge)

Browse With...

Add

Scope to This

New Solution Explorer View

Exclude From Project

Cut

Ctrl+X

Copy

Ctrl+C

Delete

Del

Rename

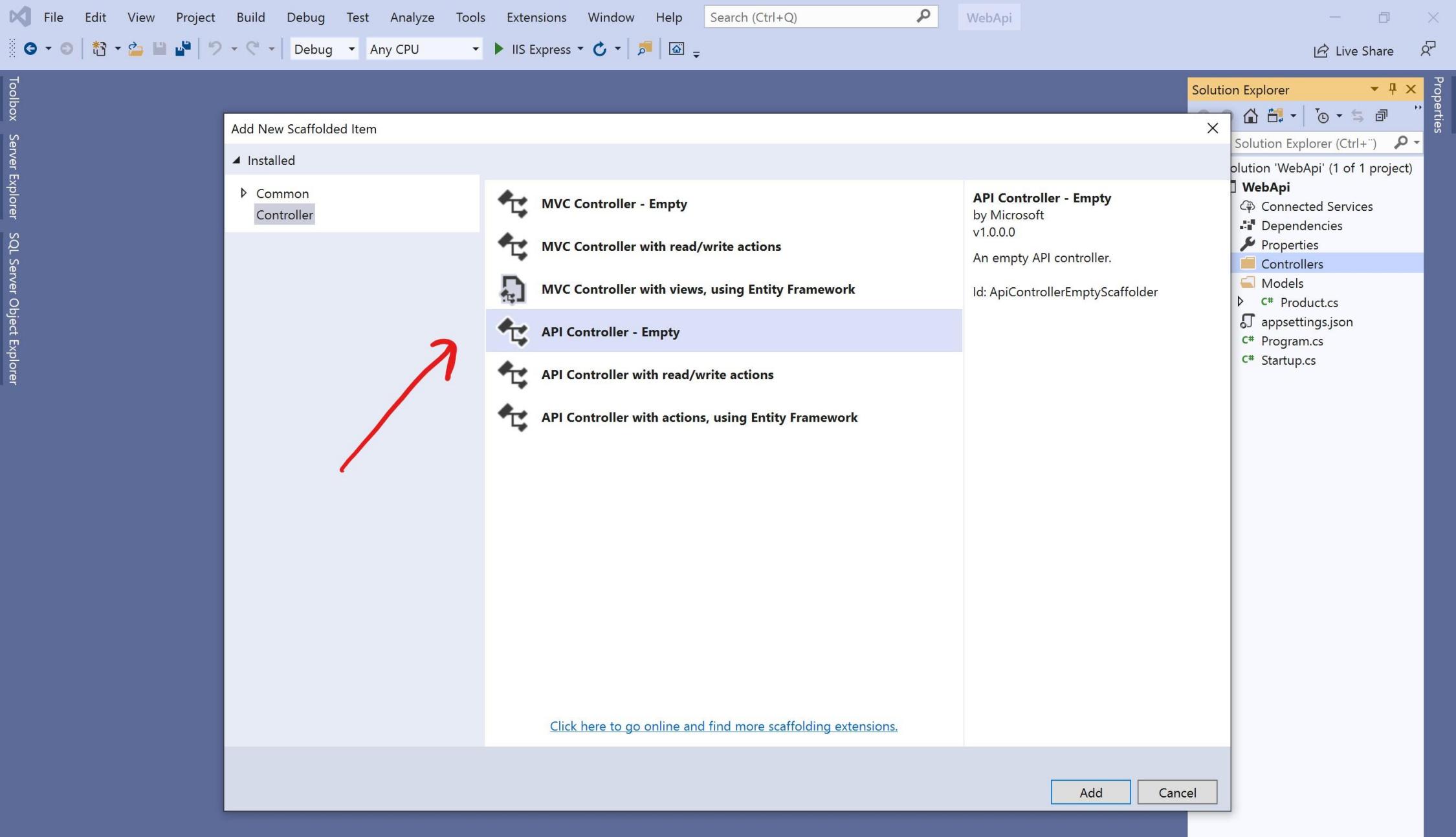
Open Folder in File Explorer

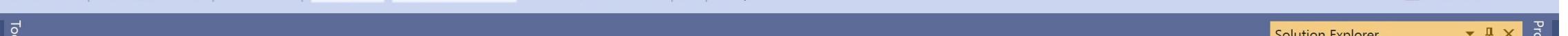
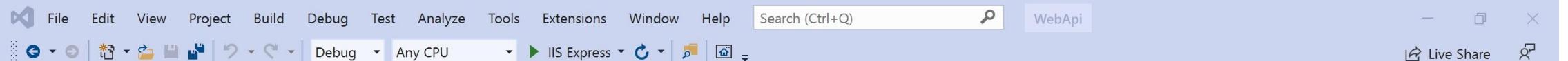
Properties

Alt+Enter



- Controller...
- New Item... Ctrl+Shift+A
- Existing Item... Shift+Alt+A
- New Scaffolded Item...
- New Folder
- From Cookiecutter...
- Container Orchestrator Support...
- Docker Support...
- Application Insights Telemetry...
- Client-Side Library...
- New Azure WebJob Project
- Existing Project as Azure WebJob
- Class... Shift+Alt+C





Solution Explorer

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties
- Controllers
- Models
- Product.cs
- appsettings.json
- Program.cs
- Startup.cs

Add Empty API Controller

Controller name: ProductsController

Add Cancel

A red arrow points from the 'ProductsController' text input field towards the 'Add' button.

Properties

Solution Explorer Team Explorer

Package Manager Console Error List ... Output Web Publish Activity

This item does not support previewing

Add to Source Control

5

A screenshot of the Microsoft Visual Studio IDE interface, showing a Web API project named "WebApi".

The top menu bar includes: File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q), and a magnifying glass icon.

The toolbar contains icons for Back, Forward, Stop, Refresh, and various project management tools.

The title bar shows "WebApi" and the current file "ProductsController.cs".

The left sidebar includes: Toolbox, Server Explorer, SQL Server Object Explorer, and Solution Explorer.

The main code editor displays the "ProductsController.cs" file:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5 using Microsoft.AspNetCore.Http;
6 using Microsoft.AspNetCore.Mvc;
7
8 namespace WebApi.Controllers
9 {
10     [Route("api/[controller]")]
11     [ApiController]
12     public class ProductsController : ControllerBase
13     {
14     }
15 }
```

The Solution Explorer on the right shows the project structure:

- Solution 'WebApi' (1 of 1 project)
 - WebApi
 - Connected Services
 - Dependencies
 - Properties
 - Controllers
 - ProductsController.cs
 - Models
 - Product.cs
 - appsettings.json
 - Program.cs
 - Startup.cs

The status bar at the bottom shows: 109 %, No issues found, Ln: 1 Ch: 1 SPC CRLF, and Solution Explorer Team Explorer.

The bottom navigation bar includes: Package Manager Console, Error List ..., Output, Web Publish Activity, Ready, Add to Source Control, and a notifications icon.

Screenshot of Microsoft Visual Studio showing a Web API project named "WebApi".

The main window displays the code for `ProductsController.cs` in the `WebApi.Controllers` namespace. A red arrow points to the `using WebApi.Models;` statement.

```
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using Microsoft.AspNetCore.Http;
6  using Microsoft.AspNetCore.Mvc;
7  using WebApi.Models; -----^
8
9  namespace WebApi.Controllers
10 {
11     [Route("api/[controller]")]
12     [ApiController]
13     public class ProductsController : ControllerBase
14     {
15         List<Product> products = new List<Product>()
16         {
17             new Product {
18                 Id = 1006368,
19                 Name = "Austin and Barbeque AABQ Wifi Food Thermometer",
20                 Description = "Termometer med WiFi för en optimal innertemperatur",
21                 Price = 399
22             },
23             new Product {
24                 Id = 1009334,
25                 Name = "Andersson Elektrisk Tändare ECL 1.1",
26                 Description = "Elektrisk stormssäker tändare helt utan gas och bränsle",
27                 Price = 89
28             },
29             new Product {
30                 Id = 1002266,
31                 Name = "Weber Non-Stick Spray",
32                 Description = "BBQ-oljespray som motverkar att råvaror fastnar på gallret",
33                 Price = 99
34             }
35         };
36     }
37 }
```

The Solution Explorer on the right shows the project structure:

- WebApi
- Connected Services
- Properties
- Controllers
- Models
- Product.cs
- appsettings.json
- Program.cs
- Startup.cs

At the bottom, the status bar shows "109 %", "No issues found", "Ln: 37 Ch: 2 SPC CRLF", and "Solution Explorer Team Explorer".



ProductsController.cs* X

WebApi Diagnostic Tools

WebApi.Controllers.ProductsController Properties

products

```
23     new Product {
24         Id = 1009334,
25         Name = "Andersson Elektrisk Tändare ECL 1.1",
26         Description = "Elektrisk stormsäker tändare helt utan gas och bränsle",
27         Price = 89
28     },
29     new Product {
30         Id = 1002266,
31         Name = "Weber Non-Stick Spray",
32         Description = "BBQ-oljespray som motverkar att råvaror fastnar på gallret",
33         Price = 99
34     }
35 };
36
37
38 [HttpGet] // Get ALL Products
39
40 public IEnumerable<Product> Get()
41 {
42     return products;
43 }
44
45
46
47
48
49 }
50 }
```

Search Solution Explorer (Ctrl+Shift+F) X

Solution Explorer X

WebApi (1 of 1 project)

- Connected Services
- Dependencies
- Properties
- Controllers
 - ProductsController.cs
- Models
 - Product.cs
 - appsettings.json
 - Program.cs
 - Startup.cs

Toolbox Diagnostic Tools

Server Explorer Properties

SQL Server Object Explorer

120 % No issues found

Ln: 50 Ch: 2 SPC CRLF

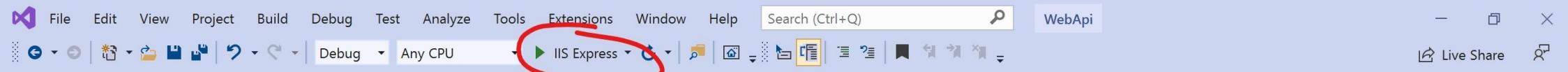
Solution Explorer Team Explorer

Package Manager Console Error List Output Web Publish Activity

Ready Add to Source Control

Live Share Diagnostic Tools

Diagnostic Tools Properties



ProductsController.cs*

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using Microsoft.AspNetCore.Http;
6  using Microsoft.AspNetCore.Mvc;
7  using WebApi.Models;
8
9  namespace WebApi.Controllers
10 {
11     [Route("api/[controller]")]
12     [ApiController]
13     public class ProductsController : ControllerBase
14     {
15         public List<Product> products = new List<Product>()
16         {
17             new Product {
18                 Id = 1006368,
19                 Name = "Austin and Barbeque AABQ Wifi Food Thermometer",
20                 Description = "Termometer med WiFi för en optimal innertemperatur",
21                 Price = 399
22             },
23             new Product {
24                 Id = 1009334,
25                 Name = "Andersson Elektrisk Tändare ECL 1.1",
26                 Description = "Elektrisk stormsäker tändare helt utan gas och bränsle",
27                 Price = 89
28             },
29             new Product {
30                 Id = 1002266,
31                 Name = "Weber Non-Stick Spray",
32                 Description = "BBQ-oljespray som motverkar att råvaror fastnar på gallret",
33                 Price = 99
34             }
35         };
36
37         [HttpGet] // Get ALL Products
38         public IEnumerable<Product> Get()
39         {
40             return products;
41         }
42     }
43 }
```

No issues found

90 %

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) WebApi Live Share Diagnostic Tools Properties

ProductsController.cs* WebApi WebApi.Controllers.ProductsController products

Solution Explorer

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

- WebApi
 - Connected Services
 - Dependencies
 - Properties
 - Controllers
 - ProductsController.cs
 - Models
 - Product.cs
 - appsettings.json
 - Program.cs
 - Startup.cs

Toolbox Server Explorer SQL Server Object Explorer

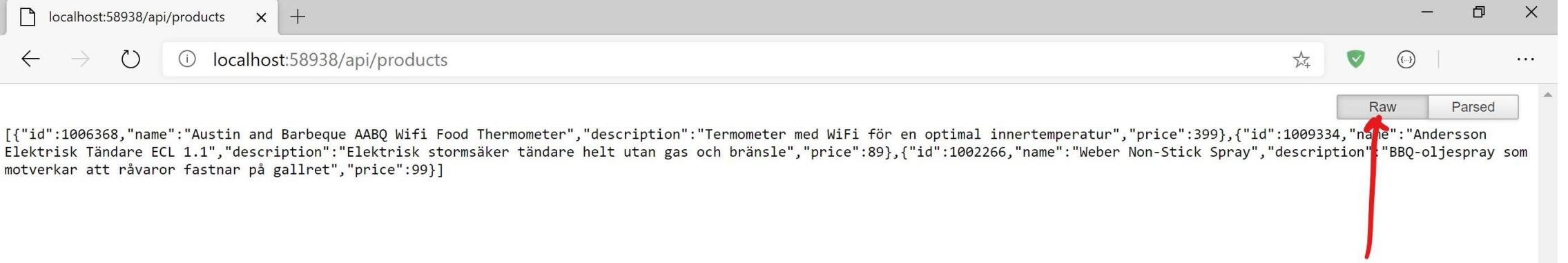
Package Manager Console Error List Output Web Publish Activity

Ready Add to Source Control

localhost:58938/api/products

```
[  
  {  
    "id": 1006368,  
    "name": "Austin and Barbeque AABQ Wifi Food Thermometer",  
    "description": "Termometer med WiFi för en optimal innertemperatur",  
    "price": 399  
  },  
  {  
    "id": 1009334,  
    "name": "Andersson Elektrisk Tändare ECL 1.1",  
    "description": "Elektrisk stormssäker tändare helt utan gas och bränsle",  
    "price": 89  
  },  
  {  
    "id": 1002266,  
    "name": "Weber Non-Stick Spray",  
    "description": "BBQ-oljespray som motverkar att råvaror fastnar på gallret",  
    "price": 99  
  }]
```

↑
*Automatiskt
JSON*

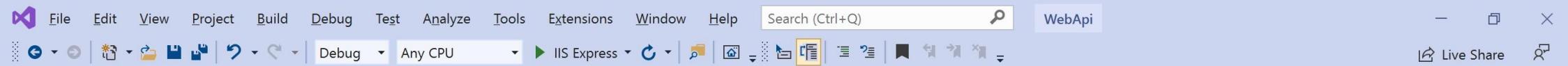


localhost:58938/api/products

localhost:58938/api/products

Raw Parsed

```
[{"id":1006368,"name":"Austin and Barbeque AABQ Wifi Food Thermometer","description":"Termometer med WiFi för en optimal innertemperatur","price":399}, {"id":1009334,"name":"Andersson Elektrisk Tändare ECL 1.1","description":"Elektrisk stormssäker tändare helt utan gas och bränsle","price":89}, {"id":1002266,"name":"Weber Non-Stick Spray","description":"BBQ-oljespray som motverkar att råvaror fastnar på gallret","price":99}]
```



ProductsController.cs* WebApi.Controllers.ProductsController

```
31     Name = "Weber Non-Stick Spray",
32     Description = "BBQ-oljespray som motverkar att råvaror fastnar på gallret",
33     Price = 99
34   }
35 }
36
37 [HttpGet] // Get ALL Products
38 public IEnumerable<Product> Get()
39 {
40   return products;
41 }
42
43
44 [HttpGet("{id}")] // Get a specific Product
45 public Product Get(int id)
46 {
47
48 }
49
50 }
51 }
```

A red arrow points from the text "Get a specific Product" in the code to the line "[HttpGet("{id}")]".

Solution Explorer Solution 'WebApi' (1 of 1 project)
WebApi
Connected Services
Dependencies
Properties
Controllers
ProductsController.cs
Models
Product.cs
appsettings.json
Program.cs
Startup.cs

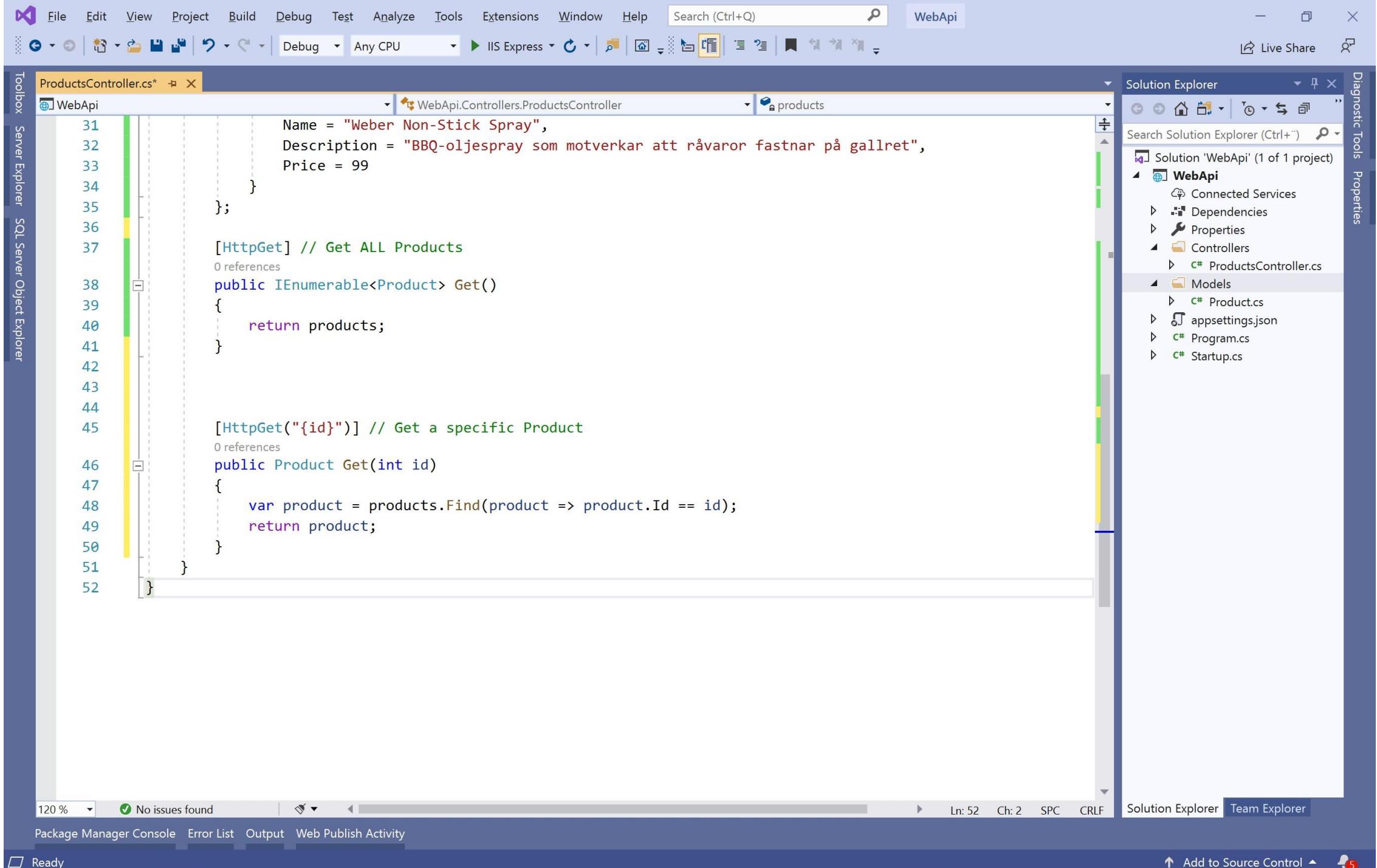
Toolbox Server Explorer SQL Server Object Explorer

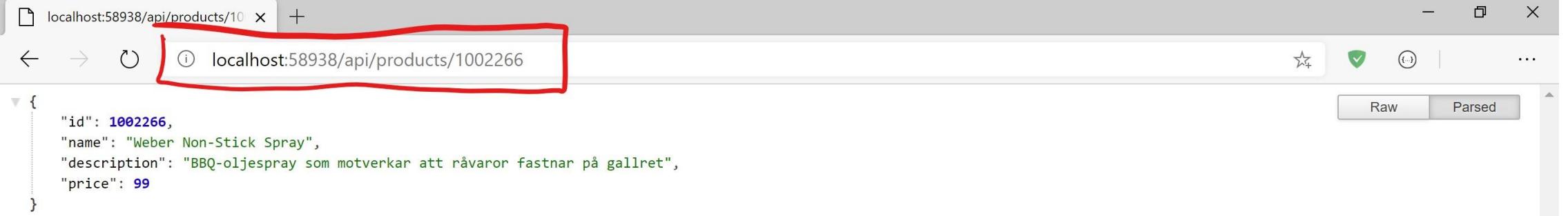
120% 1 0

Ln: 48 Ch: 1 SPC CRLF

Package Manager Console Error List Output Web Publish Activity

Ready





The screenshot shows a browser developer tools Network tab with a red box highlighting the URL bar and the response body. The URL bar contains 'localhost:58938/api/products/1002266'. The response body is a JSON object:

```
{  
  "id": 1002266,  
  "name": "Weber Non-Stick Spray",  
  "description": "BBQ-oljespray som motverkar att r\u00e5varor fastnar p\u00e5 gallret",  
  "price": 99  
}
```

At the top right of the response area, there are 'Raw' and 'Parsed' buttons.