



# ASP.NET CORE WEB API

**WebApi med Entity Framework (databaseFirst)**

# 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

 Search for templates (Alt+S)[Clear all](#)

## Recent project templates

ASP.NET Core Web Application

C#

Azure Functions

C#

Blazor App

C#

Console App (.NET Core)

C#

Windows Forms App (.NET Framework)

Visual Basic

Console App (.NET Framework)

C#

Azure Resource Group

C#

ASP.NET Web Application (.NET Framework)

C#

ASP.NET Web Application (.NET Framework)

Visual Basic

C# 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](#)

C# 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:\



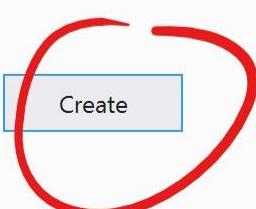
Solution name 

WebApi

Place solution and project in the same directory

Back

Create



# Create a new ASP.NET Core web application

.NET Core    ASP.NET Core 3.1

**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

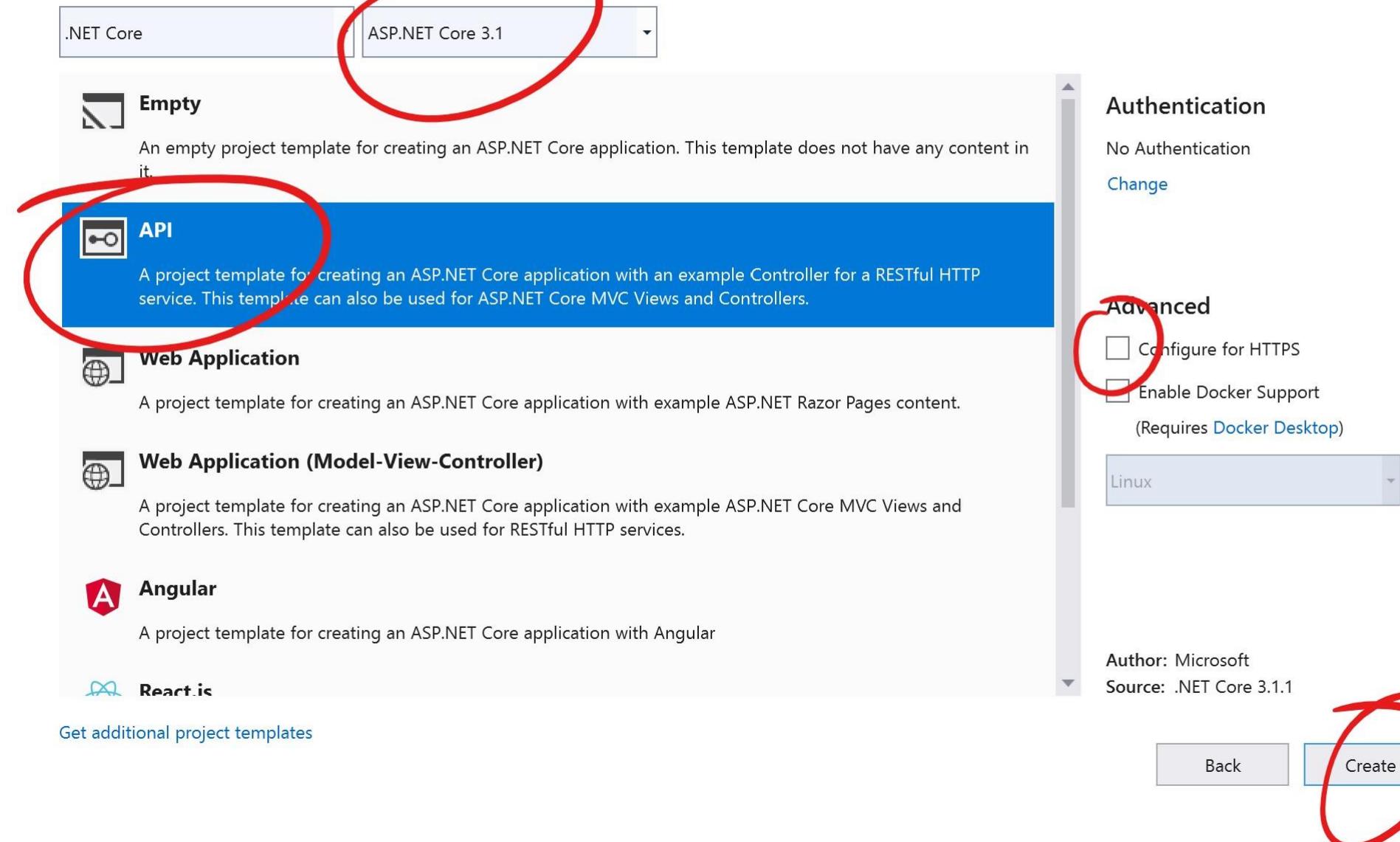
**Authentication**  
No Authentication  
[Change](#)

**Advanced**  
 Configure for HTTPS  
 Enable Docker Support  
(Requires Docker Desktop)

Linux

Author: Microsoft  
Source: .NET Core 3.1.1

Back **Create**



ASP.NET Core

Learn about the .NET platform, create your first application and extend it to the cloud.

Build Your App

Connect To The Cloud

Learn Your IDE

ASP.NET Core documentation

.NET application architecture

Published your app to Azure

Get started with ASP.NET on Azure

See our productivity guide

Write code faster

TA BOOM !

Toolbox Server Explorer SQL Server Object Explorer

Solution Explorer Properties

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties
- Controllers
- WeatherForecastController.cs
- appsettings.json
- Program.cs
- Startup.cs
- WeatherForecast.cs

Error List ... Output Web Publish Activity

Restore completed in 38,44 ms for C:\Users\HansMatti-Lasse\source\repos\WebApi\WebApi\WebApi.csproj.

Add to Source Control

Debug Any CPU IIS Express Add to Source Control ▾

Live Share

WebApi

Overview

Connected Services

Service References

Publish

# ASP.NET Core

Learn about the .NET platform, create your first application and extend it to the cloud.

{ } Build Your App

Cloud icon Connect To The Cloud

ASP.NET Core documentation

.NET application architecture

See our project Write code

Get started with ASP.NET on Azure

Manage NuGet Packages...

Manage Client-Side Libraries...

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

Toolbox

Server Explorer

SQL Server Object Explorer

Solution Explorer

Properties

Error List ... Output Web Publish Activity

Ready

ASP.NET Core

Build

Rebuild

Clean

View

Analyze and Code Cleanup

Pack

**Manage NuGet Packages...**

Connected Services

Dependencies

Properties

Controllers

appsettings.json

Program.cs

Startup.cs

New Solution Explorer View

File Nesting

Edit Project File

Add

Manage Client-Side Libraries...

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

Solution Explorer Team Explorer

Screenshot of the Microsoft Visual Studio IDE showing the NuGet Package Manager interface for a project named "WebApi".

The left pane displays a list of packages under the "Browse" tab, filtered for "Microsoft.AspNetCore.Mvc.NewtonsoftJson". A red arrow points from the search bar to the package listing.

The right pane shows the details for the selected package, "Microsoft.AspNetCore.Mvc.NewtonsoftJson", version 3.1.2. The "Install" button is highlighted with a red circle.

**NuGet Package Manager: WebApi**

Package source: nuget.org

**Microsoft.AspNetCore.Mvc.NewtonsoftJson** by Microsoft, 4,62M downloads v3.1.2

ASP.NET Core MVC features that use Newtonsoft.Json. Includes input and output formatters for JSON and JSON PATCH.

**Microsoft.AspNetCore.Mvc.Formatters.Json** by Microsoft, 98,2M downloads v2.2.0

ASP.NET Core MVC formatters for JSON input and output and for JSON PATCH input using Json.NET.

**Microsoft.AspNetCore.Mvc.Core** by Microsoft, 116M downloads v2.2.5

ASP.NET Core MVC core components. Contains common action result types, attribute routing, application model conventions, API explorer, application parts, filters, formatters, model binding, an...

**Microsoft.AspNetCore.Mvc.Abstractions** by Microsoft, 109M downloads v2.2.0

ASP.NET Core abstractions and interfaces for action invocation and dispatching, authorization, action filters, formatters, model binding, routing, validation, and more.

**Microsoft.AspNetCore.Mvc.DataAnnotations** by Microsoft, 91M downloads v2.2.0

ASP.NET Core metadata and validation system using System.ComponentModel.DataAnnotations.

**Microsoft.AspNetCore.JsonPatch** by Microsoft, 79,9M downloads v3.1.2

ASP.NET Core support for JSON PATCH.

**Microsoft.AspNetCore.Mvc.ApiExplorer** by Microsoft, 89,1M downloads v2.2.0

ASP.NET Core MVC API explorer functionality for discovering metadata such as the list of controllers and actions, and their URLs and allowed HTTP methods.

**Microsoft.AspNetCore.Http.Abstractions** by Microsoft, 192M downloads v2.2.0

ASP.NET Core HTTP object model for HTTP requests and responses and also common extension methods for registering middleware in an IApplicationBuilder.

**Microsoft.AspNetCore.Http** by Microsoft, 170M downloads v2.2.2

**.NET Microsoft.AspNetCore.Mvc** nuget.org

Version: Latest stable 3.1.2 **Install**

**Options**

**Description**

ASP.NET Core MVC features that use Newtonsoft.Json. Includes input and output formatters for JSON and JSON PATCH.

This package was built from the source code at <https://github.com/aspnet/AspNetCore/tree/c3acdac86dad91c3d3fbc3b93ecc6b7ba494bdc>

**Version:** 3.1.2  
**Author(s):** Microsoft  
**License:** Apache-2.0  
**Date published:** Tuesday, February 18, 2020 (2/18/2020)  
**Project URL:** <https://asp.net/>  
**Report Abuse:** <https://www.nuget.org/packages/Microsoft.AspNetCore.Mvc.NewtonsoftJson/3.1.2/ReportAbuse>  
**Tags:** aspnetcore, aspnetcoremvc, json

**Dependencies**

.NETCoreApp, Version=v3.1  
Microsoft.AspNetCore.JsonPatch (>= 3.1.2)  
Newtonsoft.Json (>= 12.0.2)  
Newtonsoft.Json.Bson (>= 10.0.2)

**Solution Explorer**

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

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

**Properties**

Error List ... Output Web Publish Activity

Add to Source Control ▾

Screenshot of the Visual Studio IDE showing the NuGet Package Manager and Solution Explorer.

The NuGet Package Manager window displays the results for the search term "Microsoft.EntityFrameworkCore.SqlServer". A red arrow points to the "Microsoft.EntityFrameworkCore.SqlServer" package entry, which has 51,3M downloads. To its right, a red circle highlights the "Install" button in the details panel.

The Solution Explorer window shows the project structure for "WebApi".

Bottom navigation bar: Output, Web Publish Activity, Solution Explorer, Team Explorer.

Page Header: File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q), WebApi, Live Share.

Toolbar icons: Back, Forward, Refresh, Stop, Save, Home, etc.

Search Bar: Search (Ctrl+Q).

Solution Explorer: Solution 'WebApi' (1 of 1 project) -> WebApi -> Connected Services, Dependencies, Properties, Controllers, appsettings.json, Program.cs, Startup.cs.

NuGet Package Manager: WebApi -> Microsoft.EntityFrameworkCore.SqlServer v3.1.2 -> Description, Version: 3.1.2, Author(s): Microsoft, License: Apache-2.0, Date published: Wednesday, February 19, 2020 (2/19/2020), Project URL: https://docs.microsoft.com/ef/core/, Report Abuse: https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.SqlServer/3.1.2/ReportAbuse, Tags: Framework, Core, entity-framework-core, Data, EF, O/RM, EntityFramework, EFCore, EntityFrameworkCore, Entity, SQL, Server, Dependencies: .NETStandard, Version=v2.0, Microsoft.EntityFrameworkCore.Relational (>= 3.1.2), Microsoft.Data.SqlClient (>= 1.0.19269.1).

Screenshot of the Visual Studio IDE showing the NuGet Package Manager and Solution Explorer.

The NuGet Package Manager window shows the search results for "Microsoft.EntityFrameworkCore.Tools". A red arrow points to the download count of 40,9M for the top result, "Microsoft.EntityFrameworkCore.Tools".

The Solution Explorer window shows the project structure for "WebApi". A red circle highlights the "Install" button for the selected package "Microsoft.EntityFrameworkCore.Tools".

**NuGet Package Manager: WebApi**

Package source: nuget.org

**Microsoft.EntityFrameworkCore.Tools** by Microsoft, 40,9M downloads

Entity Framework Core Tools for the NuGet Package Manager Console in Visual Studio.

**Microsoft.EntityFrameworkCore.Design** by Microsoft, 49,5M downloads

Shared design-time components for Entity Framework Core tools.

**Microsoft.EntityFrameworkCore.Tools.DotNet** by Microsoft, 4,57M downloads

Entity Framework Core .NET Command Line Tools. Includes dotnet-ef.

**Microsoft.EntityFrameworkCore** by Microsoft, 78,9M downloads

Entity Framework Core is a lightweight and extensible version of the popular Entity Framework data access technology.

**Microsoft.EntityFrameworkCore.Relational** by Microsoft, 79,3M downloads

Shared Entity Framework components for relational database providers.

**Microsoft.EntityFrameworkCore.Analyzers** by Microsoft, 53,6M downloads

CSharp Analyzers for Entity Framework Core.

**Microsoft.EntityFrameworkCore.Abstractions** by Microsoft, 54,4M downloads

Provides abstractions and attributes that are used to configure Entity Framework Core

**Microsoft.EntityFrameworkCore.SqlServer** by Microsoft, 51,3M downloads

Microsoft SQL Server database provider for Entity Framework Core.

**Microsoft.EntityFrameworkCore.InMemory** by Microsoft, 35,1M downloads

Entity Framework Core In-Memory provider

**.NET Microsoft.EntityFrameworkCore** nuget.org

Version: 3.1.2

Install

Options

Description

Entity Framework Core Tools for the NuGet Package Manager Console in Visual Studio.

Enables these commonly used commands:

- Add-Migration
- Drop-Database
- Get-DbContext
- Scaffold-DbContext
- Script-Migrations
- Update-Database

Version: 3.1.2

Author(s): Microsoft

License: Apache-2.0

Date published: Wednesday, February 19, 2020 (2/19/2020)

Project URL: <https://docs.microsoft.com/ef/core/>

Report Abuse: <https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.Tools/3.1.2/ReportAbuse>

Tags: EntityFramework, O/RM, Framework, EntityFrameworkCore, Core, Entity, entity-framework-core, EF, EFCore, Data

**Solution Explorer**

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

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

**Properties**

**Output** Web Publish Activity

Ready Add to Source Control

The screenshot shows the Microsoft Visual Studio IDE interface. A red arrow points from the top-left towards the **Server Explorer** tab in the toolbar. Another red arrow points from the bottom-left towards the **Properties** tab in the Server Explorer pane.

**Server Explorer** (highlighted by a red arrow):

- Shows a tree view of database connections. One connection, `sqldb.mdf`, is selected.
- Context menu options include Refresh, Delete, Change View, Modify Connection..., Close Connection, New Query, Detach Database, Browse In SQL Server Object Explorer, and Rename.
- Properties pane (Alt+Enter): Shows the `Product` entity with fields: ProductId, Quantity, Id, Name, Description, Price, Views, Stored Procedures, Functions, Synonyms, Types, and Assemblies.

**Solution Explorer**:

- Shows the solution 'WebApi' (1 of 1 project) containing the `WebApi` project.
- Project files listed: Connected Services, Dependencies, Properties, Controllers, `appsettings.json`, `Program.cs`, and `Startup.cs`.

**Properties** (highlighted by a red arrow):

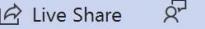
- Shows the `Product` entity with fields: ProductId, Quantity, Id, Name, Description, Price, Views, Stored Procedures, Functions, Synonyms, Types, and Assemblies.

**Central Area:**

- .NET Core** landing page: "Build Your App", "Connect To The Cloud", and "Learn Your IDE".
- Links: ASP.NET Core documentation, .NET application architecture, Publish your app to Azure, Get started with ASP.NET on Azure, See our productivity guide, and Write code faster.

**Bottom Navigation:**

- Solution Explorer and Team Explorer tabs.
- Status bar: Output, Web Publish Activity, Ready, Add to Source Control, and a notification icon with the number 5.

Debug Any CPU IIS Express Add to Source Control 

WebApi 

Overview

Connected Services

Service References

Publish

# ASP.NET Core

Learn about the .NET platform, create your first application and extend it to the cloud.

{ } Build Your App

ASP.NET Core documentation

.NET application architecture

Cloud icon Connect To The Cloud

Cloud icon Publish your app to Azure

Cloud icon Get started with ASP.NET on Azure

Tools icon Learn Your IDE

Tools icon See our productivity guide

Tools icon Write code faster

*kopiert*

Properties

sqlDb.mdf Connection

Name	C:\USERS\HANSMATTIN-LASSEI\DESKTOP\SQLDB.MDF
Case Sensitive	False
<b>Connection String</b>	Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\HansMattin-Lassei\Desktop\SQLDB.MDF;Integrated Security=True
Owner	AzureAD\HansMattin-Lassei
Primary File Path	C:\Users\HansMattin-Lassei\Desktop\SQLDB.MDF
Provider	.NET Framework Data Provider
State	Open
Type	Microsoft SQL Server
Version	13.0.0.4001

Connection String

File Edit View Project Build Debug Test Analyze Tools

Extensions Window Help Search (Ctrl+Q) WebApi

IIS Express Live Share

WebApi

Overview Connected Services Service References Publish

Get Tools and Features... Connect to Database... Connect to Server... Add SQL Server... SQL Server Data Lake Code Snippets Manager... Ctrl+K, Ctrl+B Choose Toolbox Items... NuGet Package Manager Python Create GUID External Tools... Command Line Import and Export Settings... Customize... Options... .NET application architecture

create your first application and extend it to the cloud.

Package Manager Console Manage NuGet Packages for Solution... Package Manager Settings

Connect To The Cloud

Publish your app to Azure Get started with ASP.NET on Azure

See our productivity guide Write code faster

Learn Your IDE

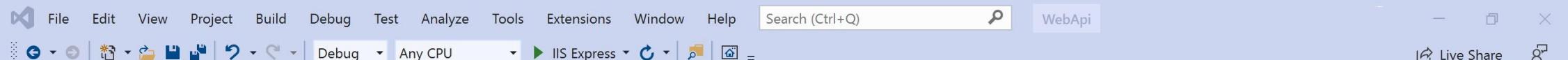
Solution Explorer Team Explorer

Output Web Publish Activity

Add to Source Control

Ready

5



WebApi

WebApi

\*scaffold.txt - Notepad

File Edit Format View Help

Scaffold-DbContext "in med connectionstring" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -Context ApiContext

A red arrow points upwards from the bottom of the image towards the word 'Scaffold' in the code editor window. The code editor contains the command 'Scaffold-DbContext "in med connectionstring" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -Context ApiContext'. The 'WebApi' tab is selected in the tabs above the editor. The status bar at the bottom shows 'Ln 7, Col 4'.

WebApi

Overview

Connected Services

Service References

Publish

# ASP.NET Core

Learn about the .NET platform, create your first application and extend it to the cloud.

{ } Build Your App

Cloud icon Connect To The Cloud

Keys icon Learn Your IDE

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'WebApi' (1 of 1 project)

WebApi

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

Properties

Package Manager Console

Package source: All Default project: WebApi

packages may include dependencies which are governed by additional licenses. Follow the package source (feed) URL to determine any dependencies.

Package Manager Console Host Version 5.4.0.6292

Type 'get-help NuGet' to see all available NuGet commands.

PM> Scaffold-DbContext "Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Default\Documents\sqlitedb.mdf;Integrated Security=True;Connect Timeout=30" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -Context ApiContext



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

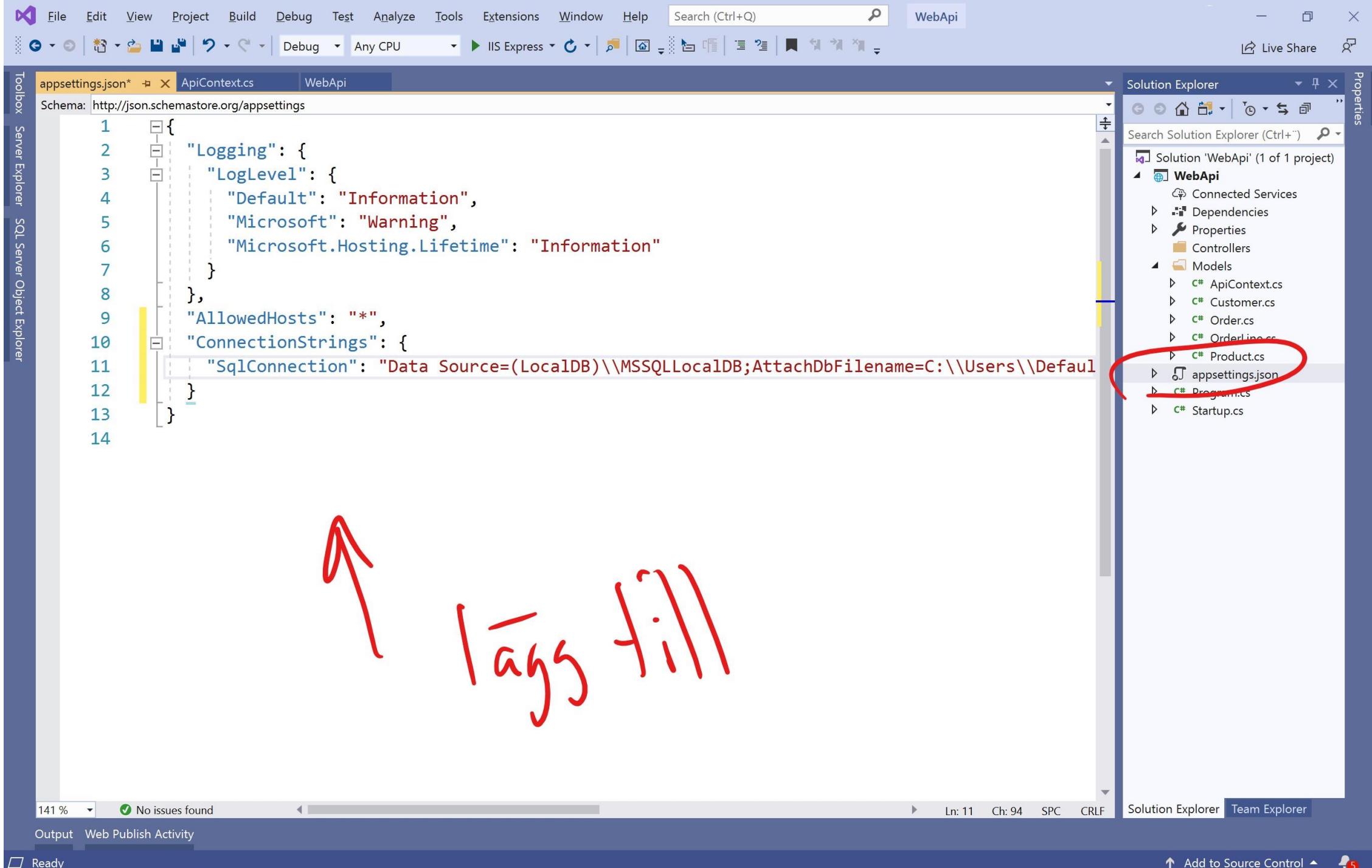
The Solution Explorer pane shows the project structure:

- Connected Services
- Dependencies
- Properties
- Controllers
- Models (containing ApiContext.cs, Customer.cs, Order.cs, OrderLine.cs, Product.cs)
- appsettings.json
- Program.cs
- Startup.cs

The ApiContext.cs file is open in the code editor. It defines a partial class ApiContext that inherits from DbContext. It has three constructors: a default constructor, a constructor taking DbContextOptions<ApiContext>, and a constructor taking a DbSet<Customer> for the Customer property. The code editor shows 3 references for the ApiContext class.

```
1 using System;
2 using Microsoft.EntityFrameworkCore;
3 using Microsoft.EntityFrameworkCore.Metadata;
4
5 namespace WebApi.Models
6 {
7     public partial class ApiContext : DbContext
8     {
9         public ApiContext()
10    {
11    }
12
13        public ApiContext(DbContextOptions<ApiContext> options)
14            : base(options)
15        {
16        }
17
18        public virtual DbSet<Customer> Customer { get; set; }
19        public virtual DbSet<Order> Order { get; set; }
20        public virtual DbSet<OrderLine> OrderLine { get; set; }
21        public virtual DbSet<Product> Product { get; set; }
22    }
}
```

The status bar at the bottom indicates the code is 141% zoomed, has 0 errors, 1 warning, and the current line is Ln: 1 Ch: 1 SPC CRLF.





WebApi

Live Share

Toolbox

Server Explorer

SQL Server Object Explorer

appsettings.json ApiContext.cs WebApi

WebApi.Models.ApiContext

\* OnConfiguring(DbContextOptionsBuilder optionsBuilder)

```
17
18     public virtual DbSet<Customer> Customer { get; set; }
19     public virtual DbSet<Order> Order { get; set; }
20     public virtual DbSet<OrderLine> OrderLine { get; set; }
21     public virtual DbSet<Product> Product { get; set; }
22
23     protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
24     {
25         if (!optionsBuilder.IsConfigured)
26         {
27             #warning To protect potentially sensitive information in your connection string, you should
28             optionsBuilder.UseSqlServer("Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFile
29         }
30     }
31
32     protected override void OnModelCreating(ModelBuilder modelBuilder)
33     {
34         modelBuilder.Entity<Customer>(entity =>
35         {
36             entity.Property(e => e.FirstName)
37                 .IsRequired()
38                 .HasMaxLength(50);
39
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
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
```

Solution Explorer

Properties

Search Solution Explorer (Ctrl+Shift+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties
- Controllers
- Models
- ApiContext.cs (highlighted with a red arrow)
- Customer.cs
- Order.cs
- OrderLine.cs
- Product.cs
- appsettings.json
- Program.cs
- Startup.cs

TA BOET

141% 0 1 ← → ⏪ ⏩ Ln: 25 Ch: 1 SPC CRLF

Output Web Publish Activity

Item(s) Saved Add to Source Control

The screenshot shows the Microsoft Visual Studio IDE interface. 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 title bar displays "WebApi". The toolbar contains icons for Back, Forward, Refresh, Stop, Start, and other development tools. The status bar at the bottom shows "141 %", "No issues found", "Ln: 32 Ch: 50 SPC CRLF", "Solution Explorer", and "Team Explorer".

The main area shows the code editor with the file "WebApi.Models.ApiContext.cs" open. The code defines a DbContext class with DbSet properties for Customer, Order, OrderLine, and Product. It also overrides OnConfiguring and OnModelCreating methods to configure the EntityProperties for FirstName and LastName.

```
17      0 references
18      public virtual DbSet<Customer> Customer { get; set; }
19      0 references
20      public virtual DbSet<Order> Order { get; set; }
21      0 references
22      public virtual DbSet<OrderLine> OrderLine { get; set; }
23      0 references
24      protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
25      {
26      }
27
28      0 references
29      protected override void OnModelCreating(ModelBuilder modelBuilder)
30      {
31          modelBuilder.Entity<Customer>(entity =>
32          {
33              entity.Property(e => e.FirstName)
34                  .IsRequired()
35                  .HasMaxLength(50);
36
37              entity.Property(e => e.LastName)
38                  .IsRequired()
39                  .HasMaxLength(50);
30);
```

The Solution Explorer on the right lists the project "WebApi" with its files: Connected Services, Dependencies, Properties, Controllers, Models (containing ApiContext.cs, Customer.cs, Order.cs, OrderLine.cs, Product.cs), appsettings.json, Program.cs, and Startup.cs.

Startup.cs\* appsettings.json ApiContext.cs\* WebApi

WebApi.WebApi.Startup Startup(IConfiguration configuration)

lags till

```
12  using WebApi.Models;
13  using Microsoft.EntityFrameworkCore;
14
15  namespace WebApi
16  {
17      public class Startup
18      {
19          public IConfiguration Configuration { get; }
20
21          public void ConfigureServices(IServiceCollection services)
22          {
23              services.AddControllers();
24
25              services.AddDbContext<ApiContext>(options =>
26                  options.UseSqlServer(Configuration.GetConnectionString("SqlConnection")));
27
28              services
29                  .AddMvc(options => options.EnableEndpointRouting = false)
30                  .SetCompatibilityVersion(CompatibilityVersion.Latest)
31                  .AddNewtonsoftJson(options =>
32                      options.SerializerSettings.ReferenceLoopHandling =
33                          Newtonsoft.Json.ReferenceLoopHandling.Ignore);
34
35          }
36
37      }
38
39
40
41  }
```

117 % No issues found Ln: 16 Ch: 1 SPC CRLF

Output Web Publish Activity

Add to Source Control

Ready

Toolbox Server Explorer SQL Server Object Explorer

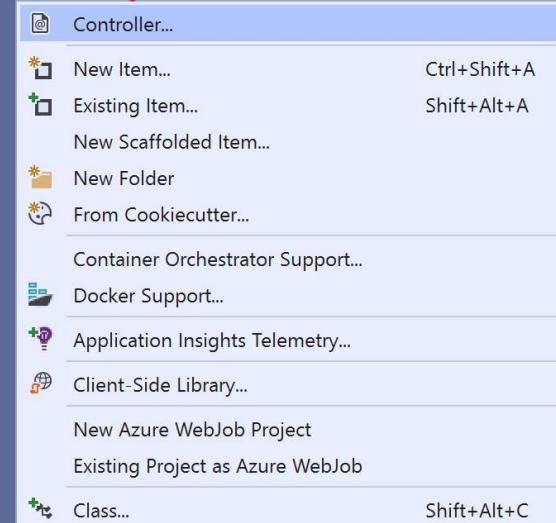
Solution Explorer Properties

Search Solution Explorer (Ctrl+F) Solution 'WebApi' (1 of 1 project)

WebApi Connected Services Dependencies Properties Controllers Models

- C# ApiContext.cs
- C# Customer.cs
- C# Order.cs
- C# OrderLine.cs
- C# Product.cs
- appsettings.json
- C# Program.cs
- C# Startup.cs

Solution Explorer Team Explorer



View in Browser (Microsoft Edge)

Browse With...

Add

Scope to This

New Solution Explorer View

Exclude From Project

Cut

Copy

Delete

Rename

Open Folder in File Explorer

Properties

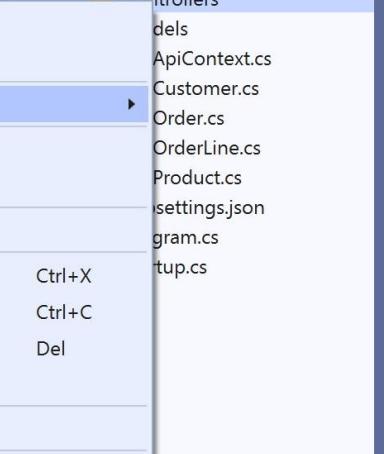
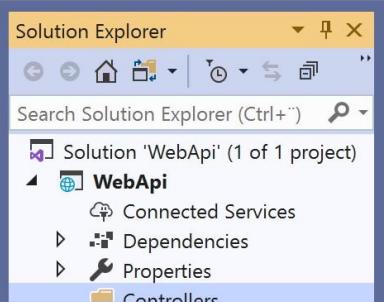
Ctrl+Shift+A  
Shift+Alt+A

Ctrl+X

Ctrl+C

Del

Alt+Enter



Toolbox Server Explorer SQL Server Object Explorer

Add New Scaffolded Item

Installed

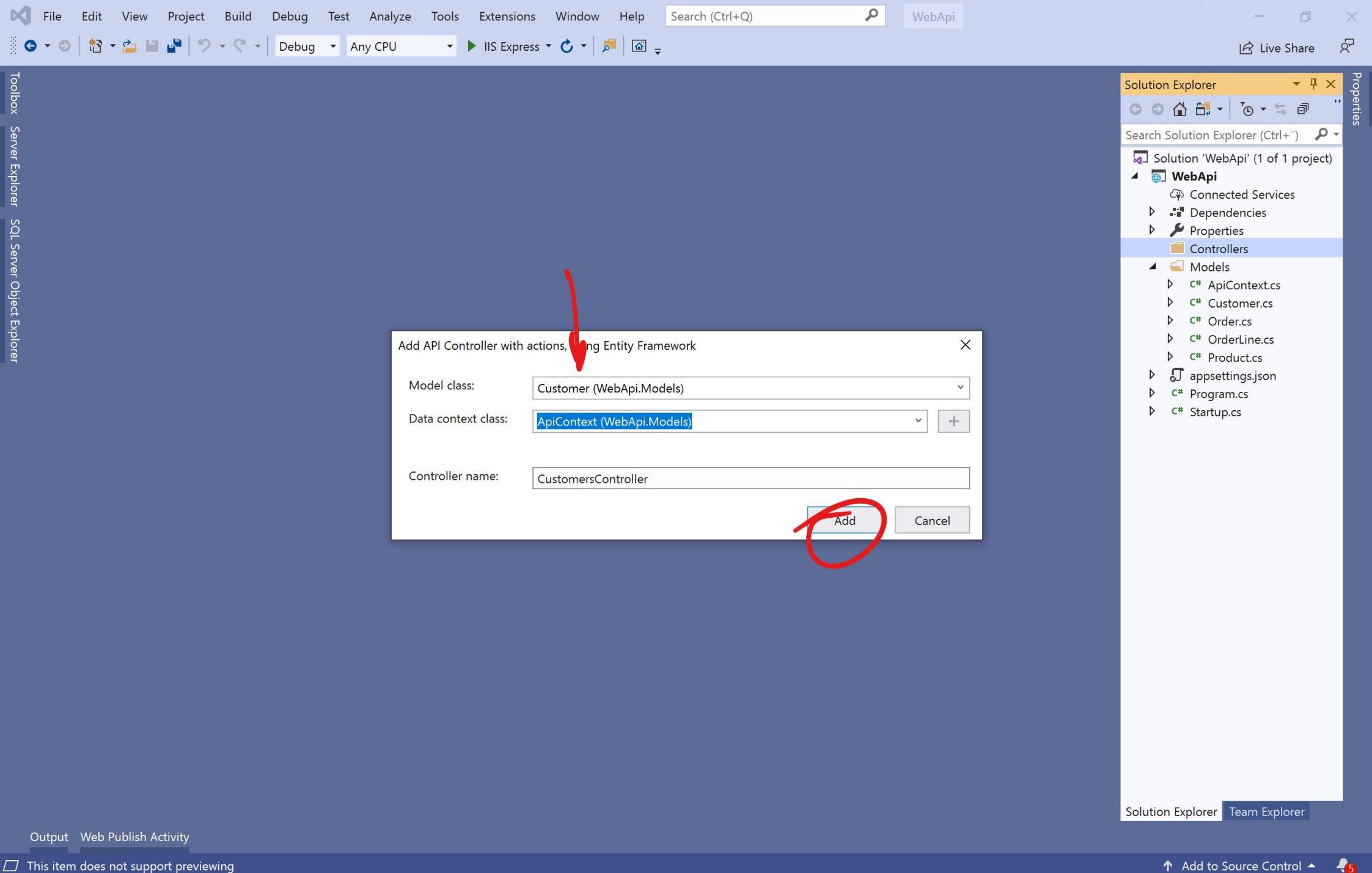
- Common Controller
  - MVC Controller - Empty
  - MVC Controller with read/write actions
  - MVC Controller with views, using Entity Framework
  - API Controller - Empty
  - API Controller with read/write actions
  - API Controller with actions, using Entity Framework**

An API controller with REST actions to create, read, update, delete, and list entities from an Entity Framework data context.

Id: ApiControllerWithContextScaffolder

[Click here to go online and find more scaffolding extensions.](#)

Solution Explorer Solution Explorer (Ctrl+Shift+E) Solution 'WebApi' (1 of 1 project)  
WebApi Connected Services Dependencies Properties Controllers Models C# ApiContext.cs C# Customer.cs C# Order.cs C# OrderLine.cs C# Product.cs appsettings.json C# Program.cs C# Startup.cs



Debug Any CPU IIS Express \_context

Live Share

Toolbox Server Explorer SQL Server Object Explorer

CustomersController.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 Microsoft.EntityFrameworkCore;
8 using WebApi.Models;
9
10 namespace WebApi.Controllers
11 {
12     [Route("api/[controller]")]
13     [ApiController]
14     public class CustomersController : Controller
15     {
16         private readonly ApiContext _context;
17
18         public CustomersController(ApiContext context)
19         {
20             _context = context;
21         }
22
23         // GET: api/Customers
24         [HttpGet]
25         public async Task<ActionResult<IEnumerable<Customer>>> GetCustomer()
26         {
27             return await _context.Customer.ToListAsync();
28         }
29
30         // GET: api/Customers/5
31         [HttpGet("{id}")]
32         public async Task<ActionResult<Customer>> GetCustomer(int id)
33         {
34             var customer = await _context.Customer.FindAsync(id);
35             if (customer == null)
36             {
37                 return NotFound();
38             }
39             return Ok(customer);
40         }
41
42         // PUT: api/Customers/5
43         [HttpPut("{id}")]
44         public async Task<ActionResult<Customer>> PutCustomer(int id, Customer customer)
45         {
46             if (id != customer.Id)
47             {
48                 return BadRequest();
49             }
50             _context.Update(customer);
51             await _context.SaveChangesAsync();
52             return Ok(customer);
53         }
54
55         // POST: api/Customers
56         [HttpPost]
57         public async Task<ActionResult<Customer>> PostCustomer(Customer customer)
58         {
59             _context.Add(customer);
60             await _context.SaveChangesAsync();
61             return CreatedAtAction("GetCustomer", new { id = customer.Id }, customer);
62         }
63
64         // DELETE: api/Customers/5
65         [HttpDelete("{id}")]
66         public async Task<ActionResult<Customer>> DeleteCustomer(int id)
67         {
68             var customer = await _context.Customer.FindAsync(id);
69             if (customer == null)
70             {
71                 return NotFound();
72             }
73             _context.Remove(customer);
74             await _context.SaveChangesAsync();
75             return Ok(customer);
76         }
77     }
78 }
```

Properties

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'WebApi' (1 of 1 project)

WebApi

- Connected Services
- Dependencies
- Properties

Controllers

- \* CustomersController.cs
- Models
- \* ApiContext.cs
- \* Customer.cs
- \* Order.cs
- \* OrderLine.cs
- \* Product.cs

ppsettings.json

program.cs

Startup.cs

View in Browser (Microsoft Edge)

Browse With...

Add

Scope to This

New Solution Explorer View

Exclude From Project

Cut

Copy

Delete

Rename

Open Folder in File Explorer

Properties

Alt+Enter

117 % No issues found

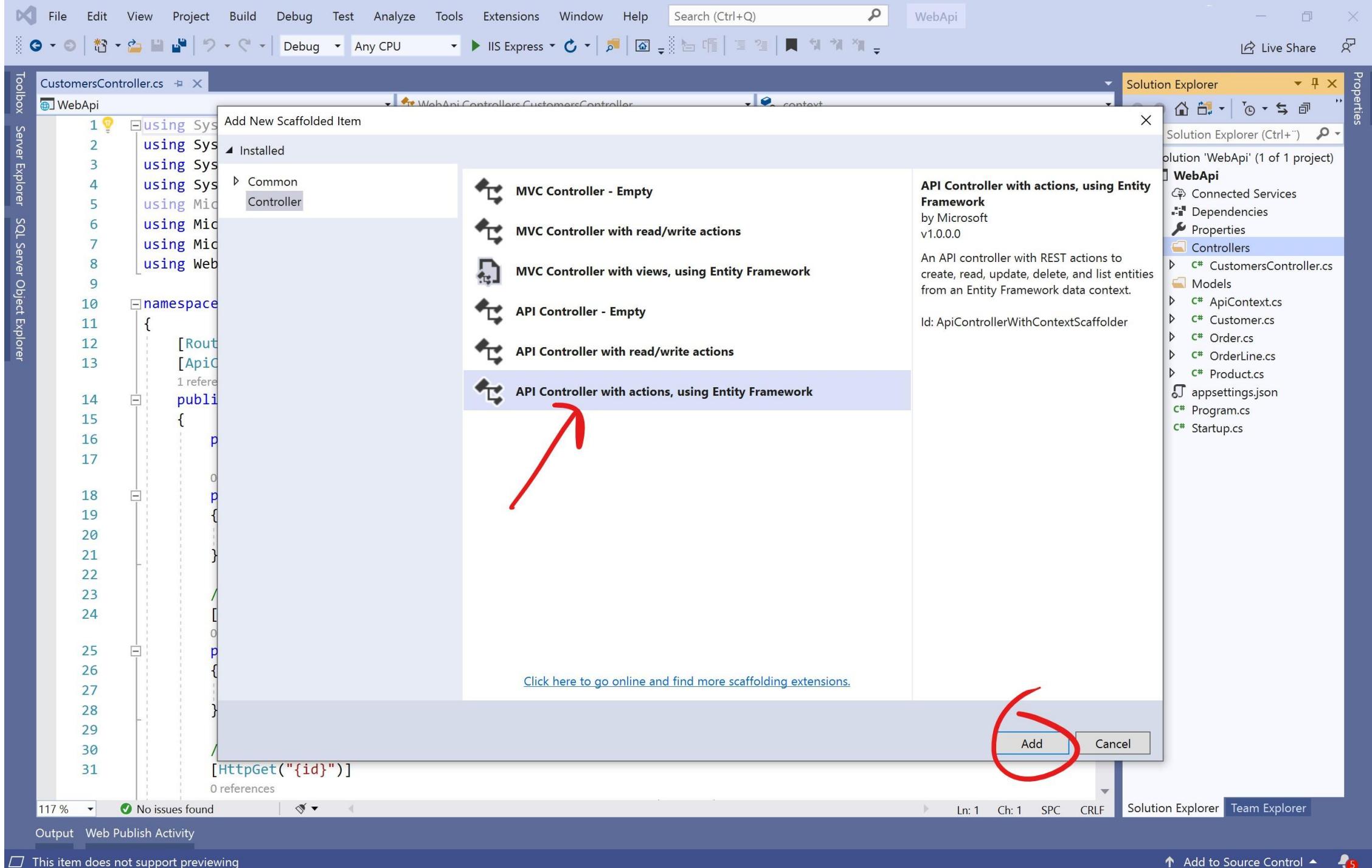
Ln: 1 Ch: 1 SPC CRLF

Output Web Publish Activity

This item does not support previewing

Add to Source Control

Team Explorer



Screenshot of Microsoft Visual Studio showing the creation of an API controller using Entity Framework.

The current file is `CustomersController.cs` in the `WebApi.Controllers` namespace. A modal dialog titled "Add API Controller with actions, using Entity Framework" is open, prompting for:

- Model class: `Product (WebApi.Models)`
- Data context class: `ApiContext (WebApi.Models)`
- Controller name: `ProductsController`

A red arrow points from the top right towards the "Add" button in the dialog. A red circle highlights the "Add" button itself.

The Solution Explorer shows the project structure:

- `WebApi` project
- `Controllers` folder contains `CustomersController.cs`
- `Models` folder contains `ApiContext.cs`, `Customer.cs`, `Order.cs`, `OrderLine.cs`, and `Product.cs`
- `appsettings.json`
- `Program.cs`
- `Startup.cs`

The status bar at the bottom indicates "117 %", "No issues found", and "Ln: 1 Ch: 1 SPC CRLF".



ProductsController.cs > CustomersController.cs

WebApi WebApi.Controllers.ProductsController \_context

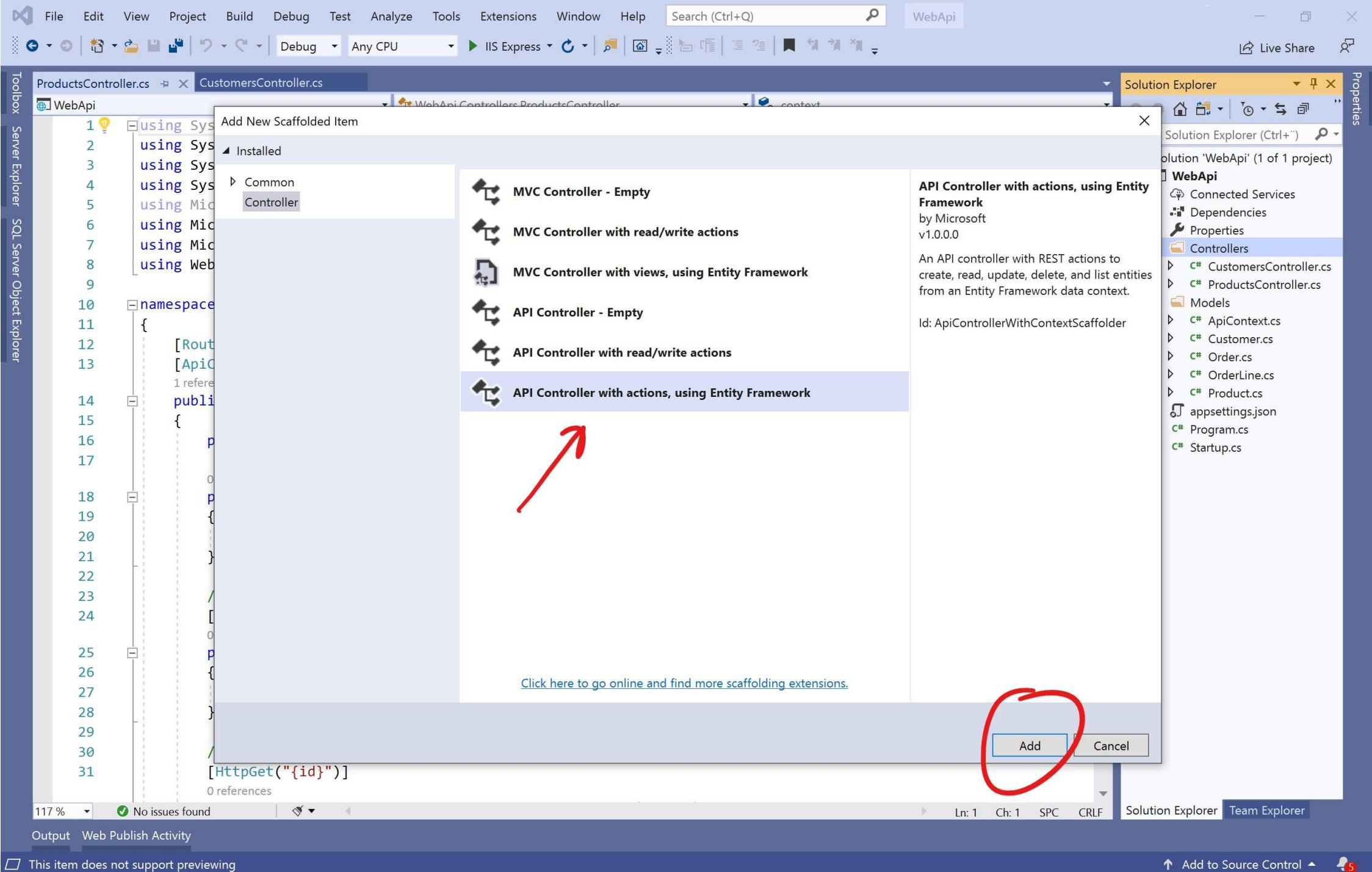
```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5 using Microsoft.AspNetCore.Http;
6 using Microsoft.AspNetCore.Mvc;
7 using Microsoft.EntityFrameworkCore;
8 using WebApi.Models;
9
10 namespace WebApi.Controllers
11 {
12     [Route("api/[controller]")]
13     [ApiController]
14     public class ProductsController : ControllerBase
15     {
16         private readonly ApiContext _context;
17
18         public ProductsController(ApiContext context)
19         {
20             _context = context;
21         }
22
23         // GET: api/Products
24         [HttpGet]
25         public async Task<ActionResult<IEnumerable<Product>>> GetProduct()
26         {
27             return await _context.Product.ToListAsync();
28         }
29
30         // GET: api/Products/5
31         [HttpGet("{id}")]
32         public async Task<ActionResult<Product>> GetProduct(int id)
33         {
34             var product = await _context.Product.FindAsync(id);
35             if (product == null)
36             {
37                 return NotFound();
38             }
39             return Ok(product);
40         }
41
42         // PUT: api/Products/5
43         [HttpPut("{id}")]
44         public async Task<ActionResult<Product>> UpdateProduct(int id, [FromBody] Product product)
45         {
46             if (id != product.Id)
47             {
48                 return BadRequest();
49             }
50
51             _context.Update(product);
52             await _context.SaveChangesAsync();
53             return Ok(product);
54         }
55
56         // POST: api/Products
57         [HttpPost]
58         public async Task<ActionResult<Product>> CreateProduct([FromBody] Product product)
59         {
60             _context.Add(product);
61             await _context.SaveChangesAsync();
62             return CreatedAtAction("GetProduct", new { id = product.Id }, product);
63         }
64
65         // DELETE: api/Products/5
66         [HttpDelete("{id}")]
67         public async Task<ActionResult<Product>> DeleteProduct(int id)
68         {
69             var product = await _context.Product.FindAsync(id);
70             if (product == null)
71             {
72                 return NotFound();
73             }
74             _context.Remove(product);
75             await _context.SaveChangesAsync();
76             return Ok(product);
77         }
78     }
79 }
```

117 % No issues found

Output Web Publish Activity

This item does not support previewing

Add to Source Control



ProductsController.cs    CustomersController.cs

WebApi

WebApi.Controllers.ProductsController

\_context

using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Threading.Tasks;  
using Microsoft.AspNetCore.Http;  
using Microsoft.AspNetCore.Mvc;  
using Microsoft.EntityFrameworkCore;  
using WebApi.Models;

namespace WebApi.Controllers

[Route("api/[controller]")]  
[ApiController]

public class ProductsController : ControllerBase

{  
 private readonly ApiCo

public ProductsController(ApiCo

{  
 \_context = context;

}

// GET: api/Products  
[HttpGet]

public async Task<ActionResult<IEnumerable<Product>>> GetProduct()

{  
 return await \_context.Product.ToListAsync();

// GET: api/Products/5  
[HttpGet("{id}")]

Model class: Order (WebApi.Models)  
Data context class: ApiContext (WebApi.Models)  
Controller name: OrdersController

Add    Cancel

Solution Explorer    Properties

Search Solution Explorer (Ctrl+F)

Solution 'WebApi' (1 of 1 project)

WebApi

Connected Services  
Dependencies  
Properties  
Controllers

CustomersController.cs  
ProductsController.cs

Models

ApiContext.cs  
Customer.cs  
Order.cs  
OrderLine.cs  
Product.cs

appsettings.json  
Program.cs  
Startup.cs

Output    Web Publish Activity

No issues found

117 %

Ln: 1 Ch: 1 SPC CRLF

Add to Source Control

OrdersController.cs\* ProductsController.cs CustomersController.cs

WebApi

```
20 } _context = context;
21
22 // GET: api/Orders
23 [HttpGet]
24 0 references
25 public async Task<ActionResult<IEnumerable<Order>>> GetOrder()
26 {
27     return await _context.Order
28         .Include(order => order.Customer)
29         .Include(order => order.OrderLine)
30             .ThenInclude(order => order.Product)
31         .ToListAsync();
32 }
33
34 // GET: api/Orders/5
35 [HttpGet("{id}")]
36 0 references
37 public async Task<ActionResult<Order>> GetOrder(int id)
38 {
39     var order = await _context.Order
40         .Include(order => order.Customer)
41         .Include(order => order.OrderLine)
42             .ThenInclude(order => order.Product)
43         .Where(order => order.Id == id)
44         .FirstOrDefaultAsync();
45
46     if (order == null)
47     {
48         return NotFound();
49     }
50
51     return order;
52 }
```

Tags till

Solution Explorer

Search Solution Explorer (Ctrl+F)

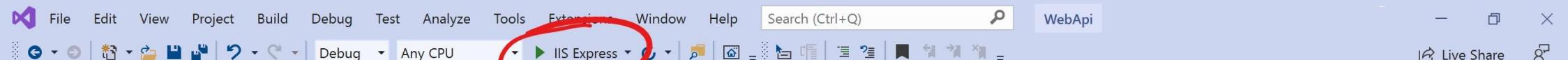
Solution 'WebApi' (1 of 1 project)

- WebApi
  - Connected Services
  - Dependencies
  - Properties
  - Controllers
    - CustomersController.cs
    - OrdersController.cs
    - ProductsController.cs
  - Models
    - ApiContext.cs
    - Customer.cs
    - Order.cs
    - OrderLine.cs
    - Product.cs
  - appsettings.json
  - Program.cs
  - Startup.cs

Properties

Output Web Publish Activity

Ready Add to Source Control

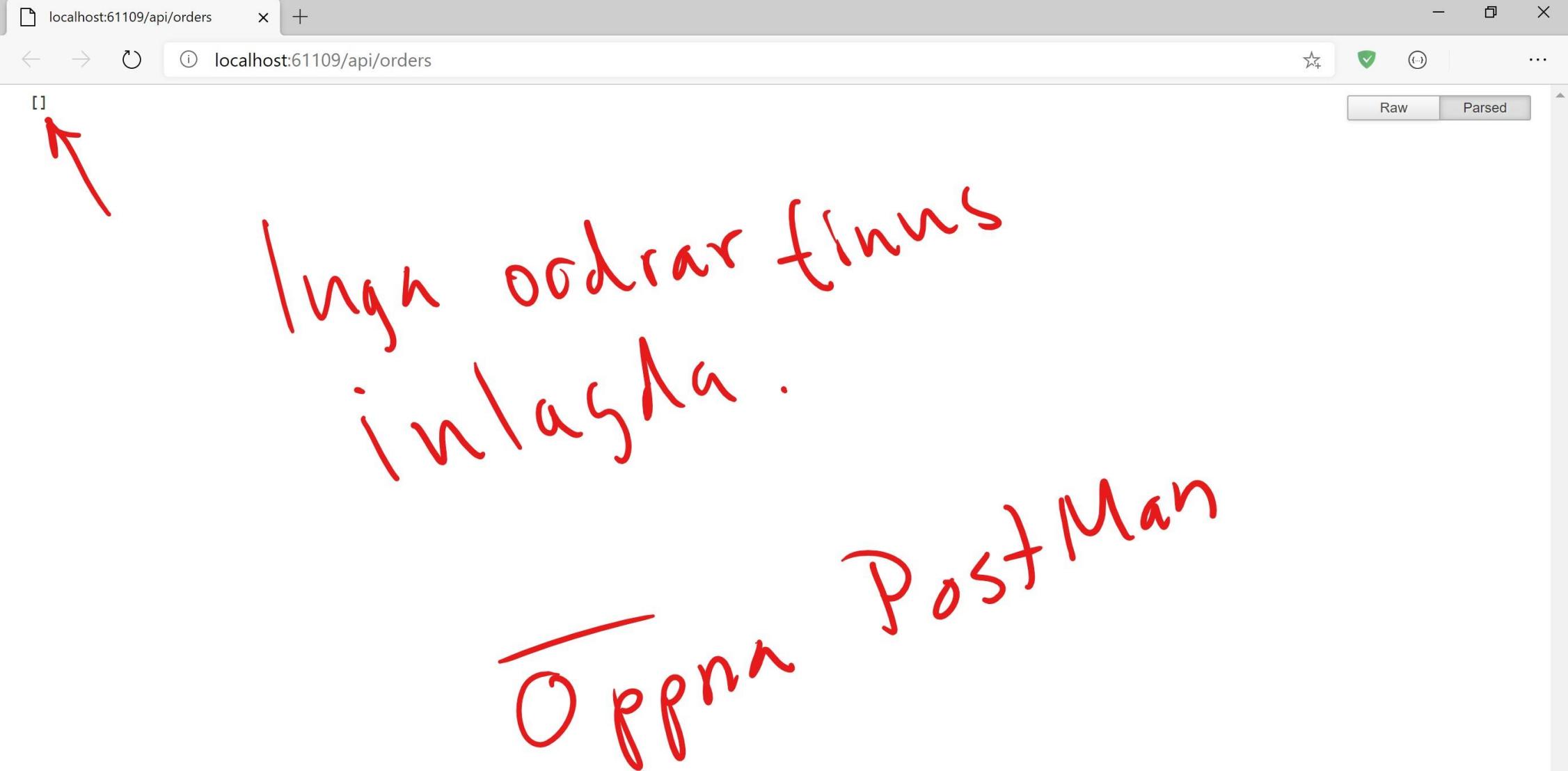


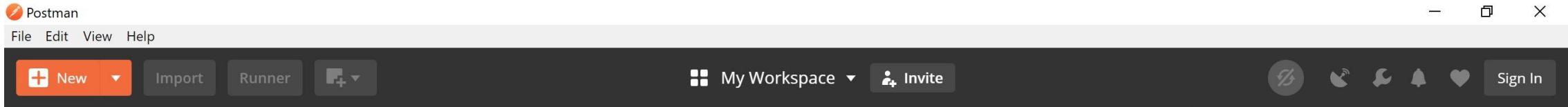
Schema: <http://json.schemastore.org/launchsettings.json>

```
1  {
2      "$schema": "http://json.schemastore.org/launchsettings.json",
3      "iisSettings": {
4          "windowsAuthentication": false,
5          "anonymousAuthentication": true,
6          "iisExpress": {
7              "applicationUrl": "http://localhost:61109",
8              "sslPort": 0
9          }
10     },
11     "profiles": {
12         "IIS Express": {
13             "commandName": "IISExpress",
14             "launchBrowser": true,
15             "launchUrl": "api/orders",
16             "environmentVariables": {
17                 "ASPNETCORE_ENVIRONMENT": "Development"
18             }
19         },
20         "WebApi": {
21             "commandName": "Project",
22             "launchBrowser": true,
23             "launchUrl": "api/orders",
24             "applicationUrl": "http://localhost:5000",
25             "environmentVariables": {
26                 "ASPNETCORE_ENVIRONMENT": "Development"
27             }
28         }
29     }
30 }
```

Andra x!!!  
api/orders

The screenshot shows the Visual Studio IDE interface with the 'WebApi' project open. The 'Solution Explorer' pane on the right lists files like 'Connected Services', 'Properties', 'launchSettings.json', 'Controllers', 'Models', and 'appsettings.json'. The 'Properties' node is highlighted with a red circle. The 'launchSettings.json' file is open in the main editor, showing configuration for 'IIS Express' and 'WebApi' profiles. Red arrows point from the handwritten note 'Andra x!!!' and 'api/orders' to the 'launchUrl' entries in both profiles. The 'IIS Express' profile points to 'api/orders', and the 'WebApi' profile also points to 'api/orders'. The 'applicationUrl' for the 'WebApi' profile is set to '<http://localhost:5000>'. The 'Properties' node in the Solution Explorer is also circled in red.





1.

2.

3.

4.

5.

Untitled Request

POST http://localhost:61109/api/cus... + ...

POST http://localhost:61109/api/customers

No Environment

Comments (0)

Send

Save

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {  
2   "firstName": "Anna",  
3   "lastName": "Nyman"  
4 }
```

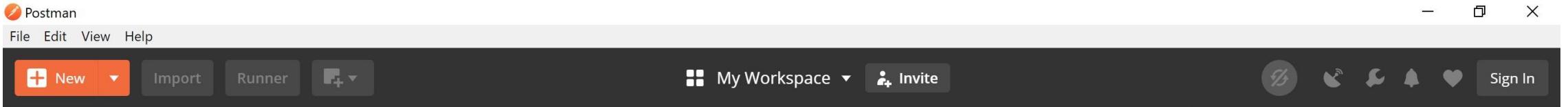
Status: 201 Created Time: 528ms Size: 286 B Save Response

Body Cookies Headers (6) Test Results

Ny Customer skapad  
och fick Id 1

Pretty Raw Preview Visualize JSON

```
1 {  
2   "id": 1,  
3   "firstName": "Anna",  
4   "lastName": "Nyman",  
5   "order": []  
6 }
```



1. Untitled Request

2. POST http://localhost:61109/api/products

3. Body (9)

4. Send

5. { "name": "Coca-Cola", "description": "Vanlig Coca-Cola", "price": 15 }

The screenshot shows a step-by-step guide for creating a POST request in Postman. Step 1 points to the 'Untitled Request' label. Step 2 points to the 'POST' method and the URL 'http://localhost:61109/api/products'. Step 3 points to the 'Body' tab which contains 9 items. Step 4 points to the 'Send' button, which is highlighted with a red circle. Step 5 points to the JSON payload in the body editor.

6. Status: 201 Created Time: 126ms Size: 316 B Save Response

7. { "id": 1, "name": "Coca-Cola", "description": "Vanlig Coca-Cola", "price": 15.0, "orderLine": [] }

8. Ny produkt skapt! Ett id 1

The screenshot shows the response from the POST request. Step 6 shows the status 201 Created, time 126ms, size 316 B, and a 'Save Response' button. Step 7 shows the JSON response object with an id of 1. Step 8 is a handwritten note in red: 'Ny produkt skapt! Ett id 1'.

Postman

File Edit View Help

New Import Runner + ... My Workspace Invite No Environment Comments (0) Sign In

POST http://localhost:61109/api/pro... + ... Untitled Request POST http://localhost:61109/api/products Send Save

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 {  
2   "name": "Coca-Cola Zero",  
3   "description": "Sockerfri Coca-Cola",  
4   "price": 15  
5 }
```

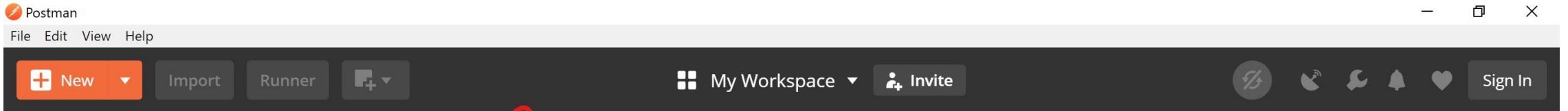
Body Cookies Headers (6) Test Results Status: 201 Created Time: 78ms Size: 324 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {  
2   "id": 2,  
3   "name": "Coca-Cola Zero",  
4   "description": "Sockerfri Coca-Cola",  
5   "price": 15.0,  
6   "orderLine": []  
7 }
```

En till produkt shoppen och titta id 2

The screenshot shows the Postman interface with a handwritten guide overlaid. Step 1 points to the 'New' button. Step 2 points to the URL input field. Step 3 points to the JSON body editor where a product object is defined. Step 4 points to the 'Send' button, which is circled in red. Step 5 points to the response body, which displays a newly created product with an ID of 2.



1. Untitled Request

2. POST http://localhost:61109/api/orders

3. Body (JSON)

4. Send

5. Status: 201 Created

Detta motsvarar data från ett gratiskortformular

```
1 {  
2   "customerId": 1,  
3   "orderDate": "2020-01-01",  
4   "dueDate": "2020-02-01",  
5   "billingTerms": "30 dagar netto",  
6   "shippingTerms": "fritt vårt lager",  
7   "orderLine": [  
8     {  
9       "OrderLineNr": 1,  
10      "productId": 1,  
11      "quantity": 10  
12    },  
13    {  
14      "OrderLineNr": 2,  
15      "productId": 2,  
16      "quantity": 22  
17    }  
18  ]  
19 }
```

Body Headers (6) Test Results

Pretty Raw Preview Visualize JSON

1 {  
2 "id": 1.

Status: 201 Created Time: 172ms Size: 560 B Save Response

Bootcamp

Detta motsvarar data från ett gratiskortformular



File Edit View Help

+ New Import Runner + ...

My Workspace Invite

No Environment

POST http://localhost:6109/api/ord... GET http://localhost:61109/api/orde...

Untitled Request

GET http://localhost:61109/api/orders

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (5) Test Results

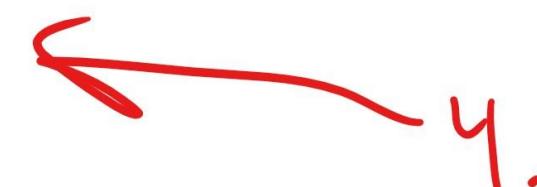
Status: 200 OK Time: 88ms Size: 745 B Save Response

Pretty Raw Preview Visualize JSON

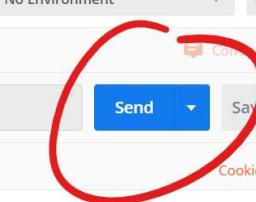
```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
```

```
{
  "id": 1,
  "customerId": 1,
  "orderDate": "2020-01-01T00:00:00",
  "dueDate": "2020-02-01T00:00:00",
  "billingTerms": "30 dagar netto",
  "shippingTerms": "fritt vårt lager",
  "customer": {
    "id": 1,
    "firstName": "Anna",
    "lastName": "Nyman",
    "order": []
  },
  "orderLine": [
    {
      "orderId": 1,
      "orderLineNr": 1,
      "productId": 1,
      "quantity": 10,
      "product": {
        "id": 1,
        "name": "Coca-Cola",
        "description": "Vanlig Coca-Cola",
        "price": 15.0000,
        "orderLine": []
      }
    },
    {
      "orderId": 1,
      "orderLineNr": 2,
      "productId": 2,
      "quantity": 22,
      "product": {
        "id": 2,
        "name": "Coca-Cola Zero",
        "description": "Sockerfri Coca-Cola",
        "price": 15.0000,
        "orderLine": []
      }
    }
  ]
}
```

3.



2.



4.



Bootcamp

