Beginning Entity Framework Core 5

From Novice to Professional

Eric Vogel

Apress®

Beginning Entity Framework Core 5

From Novice to Professional

Eric Vogel

Apress®

Beginning Entity Framework Core 5: From Novice to Professional

Eric Vogel
Okemos, MI, USA

ISBN-13 (pbk): 978-1-4842-6881-0 ISBN-13 (electronic): 978-1-4842-6882-7
https://doi.org/10.1007/978-1-4842-6882-7

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Jonathan Gennick
Development Editor: Laura Berendson
Coordinating Editor: Jill Balzano

Cover image designed by Freepik (www.freepik.com)

Distributed to the book trade worldwide by Springer Science+Business Media LLC, 1 New York Plaza, Suite 4600, New York, NY 10004. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484268810. For more detailed information, please visit http://www.apress.com/source-code.

Printed on acid-free paper

Table of Contents

About the Author

Eric Vogel is a seasoned contributor to *Visual Studio Magazine* and Senior Software Developer at Red Cedar Solutions Group. He has been developing .NET Framework web and desktop solutions for 13 years. He holds a Bachelor of Science degree in computer science from Michigan State University. He is Acting President of the Greater Lansing User Group for .NET.

About the Technical Reviewer

Pieter Nijs is a Belgian .NET architect with a passion for mobile and cloud development. He has played a key role in several projects ranging from large consumer-facing healthcare, telecom, and media apps to smaller LOB applications. As a mobile development expert at Xpirit Belgium, he loves helping customers implement mobile-first and cloud-first applications. Pieter is primarily interested in the Microsoft stack, so his interest and expertise translate to technologies like .NET, C#, XAML, Xamarin, UWP, Azure, Azure DevOps, and so on. Both at work and in his spare time, Pieter is constantly working and playing with these and other new technologies. He likes to tell everybody about the things he does, sharing his knowledge. Hence, you can find him speaking at conferences, giving trainings, and blogging at blog.pieeatingninjas.be. Since 2017, Pieter has been receiving the Microsoft MVP Award in the Windows Development category for sharing his passion and expertise with the community.

Introduction

This book is aimed at readers who have a beginner's knowledge of the .NET Framework who are looking to use Entity Framework (EF) Core 5 for a side project or business application. No prior knowledge of Entity Framework Core 5 is required. The book guides the user through the basics of Entity Framework Core 5 up to some more advanced concepts and culminates in creating an ASP.NET Core Razor Pages web application that has full create, read, update, and delete (CRUD) capability.

We will be using the NUnit Framework to test Entity Framework Core 5 behavior before using it in a full web application. You will first go over how to query data and then how to insert, update, and delete data. Later in the book, we will go over more advanced techniques like how to aggregate data, use navigation properties to get related data, and call custom raw SQL and stored procedures.

The later chapters also cover basic authentication, authorization, and reporting in an ASP.NET Core Razor Pages web application. You will learn how to query and manipulate a SQL Server database by testing each facet through NUnit integration tests.

PART I

Getting Started

CHAPTER 1

Installation

In order to use Entity Framework Core 5, you will need some tools. These tools include Visual Studio (VS) 2019 and some NuGet packages for your solution.

Install Visual Studio

The first step is getting a version of Visual Studio 2019. The Community Edition of Visual Studio 2019 for Windows can be used for all code in this book. If you are running macOS, you can also opt to install Visual Studio for Mac. I'll be using Windows for the remainder of the book, but the steps are about the same for macOS.

You can download Visual Studio 2019 from https://visualstudio.microsoft.com/downloads/ as seen in Figure 1-1.

E. Vogel, *Beginning Entity Framework Core 5*, https://doi.org/10.1007/978-1-4842-6882-7_1

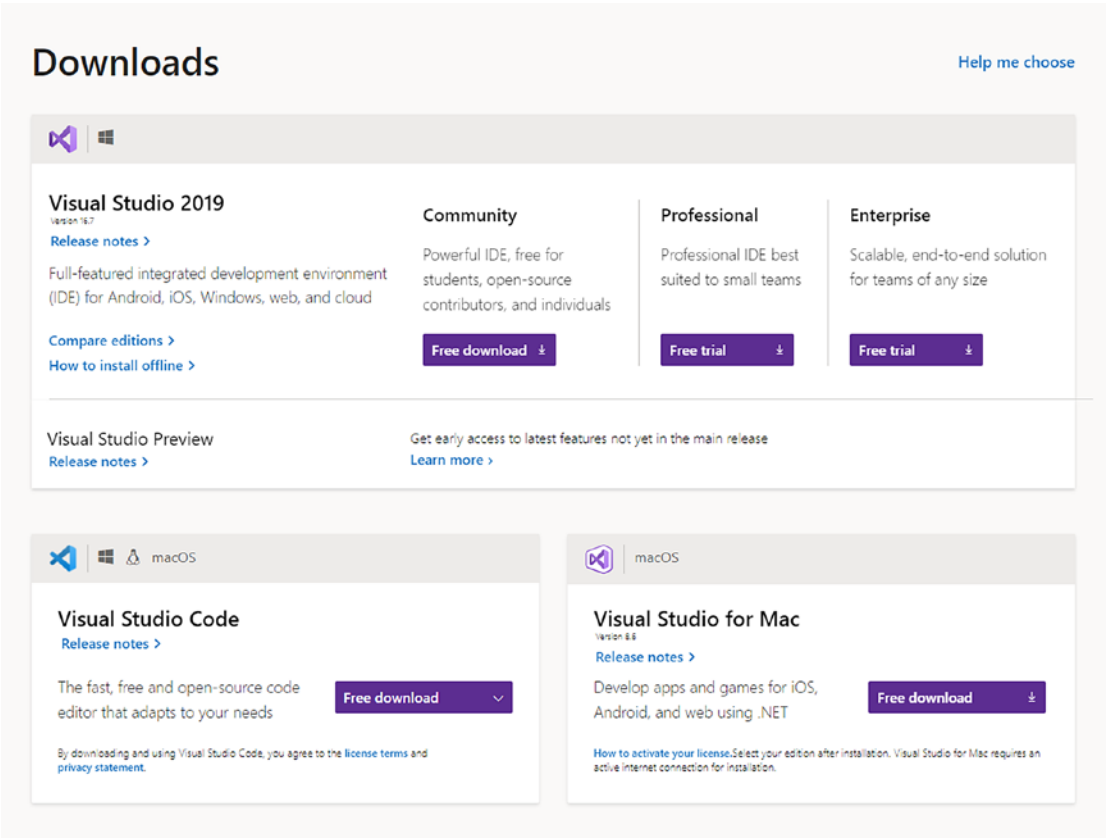

Figure 1-1. *Download VS 2019*

Next, install the latest Visual Studio 2019 and select the .NET Core option as seen in Figure 1-2.

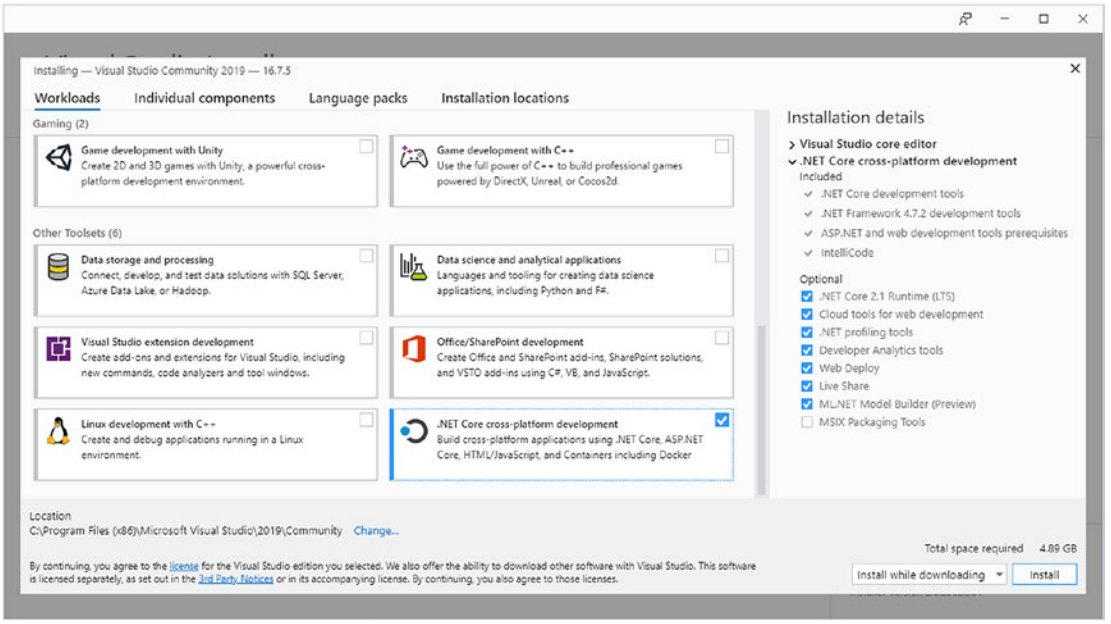

Figure 1-2. *Install .NET Core Support*

Entity Framework Core 5 will run on both .NET Core 3.1 and .NET 5. For this book, we will be using .NET Core 3.1.

Create a Project

Entity Framework Core 5 works on a variety of application types from console apps, desktop apps, and web apps. For this book, we will be creating an ASP.NET Core MVC (Model-View-Controller) app.

We will now create the project in Visual Studio and install all the required tools to use Entity Framework Core 5. First, create a new project in Visual Studio as seen in Figure 1-3.

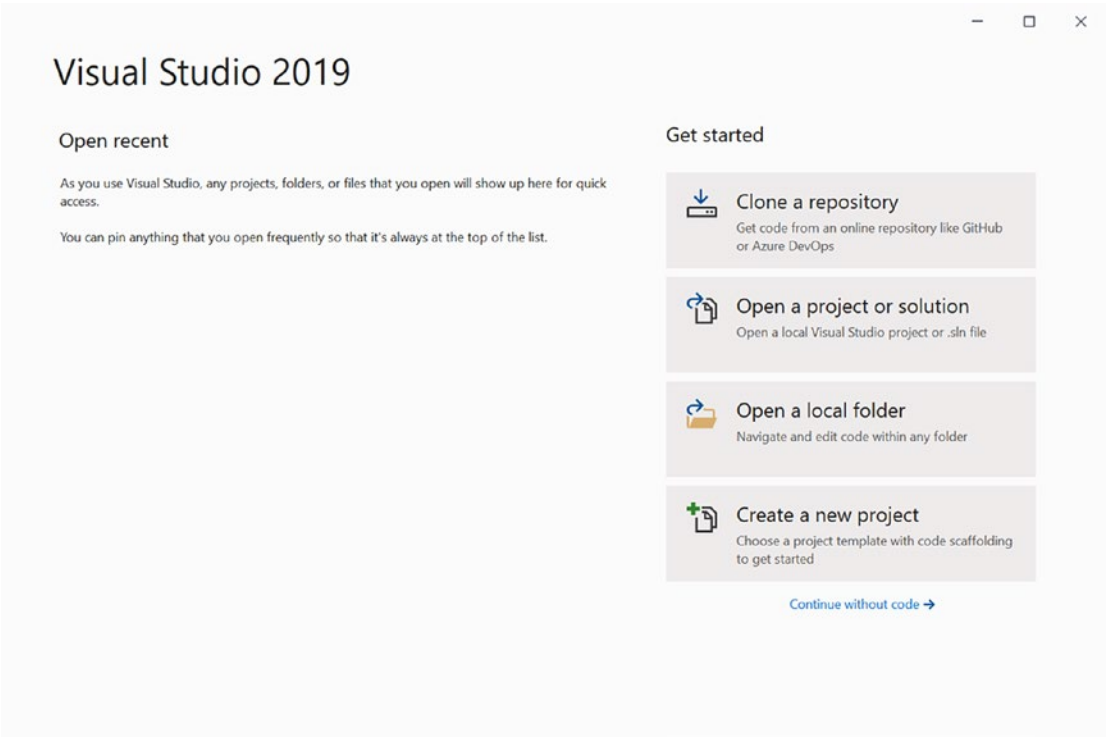

Figure 1-3. *Create a New Visual Studio 2019 Project*

Then create a new ASP.NET Core web application as seen in Figure 1-4.

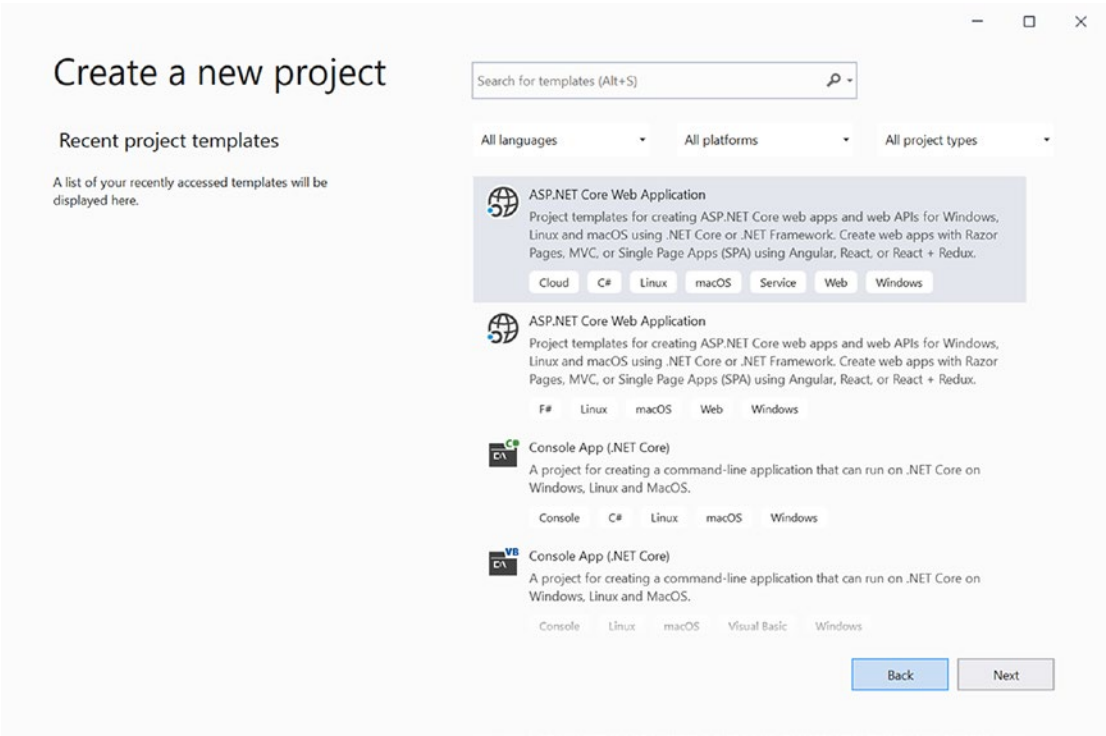

Figure 1-4. *New ASP.NET Core Web App*

Next, name your ASP.NET Core web application. I named my app EFCore5WebApp as seen in Figure 1-5.

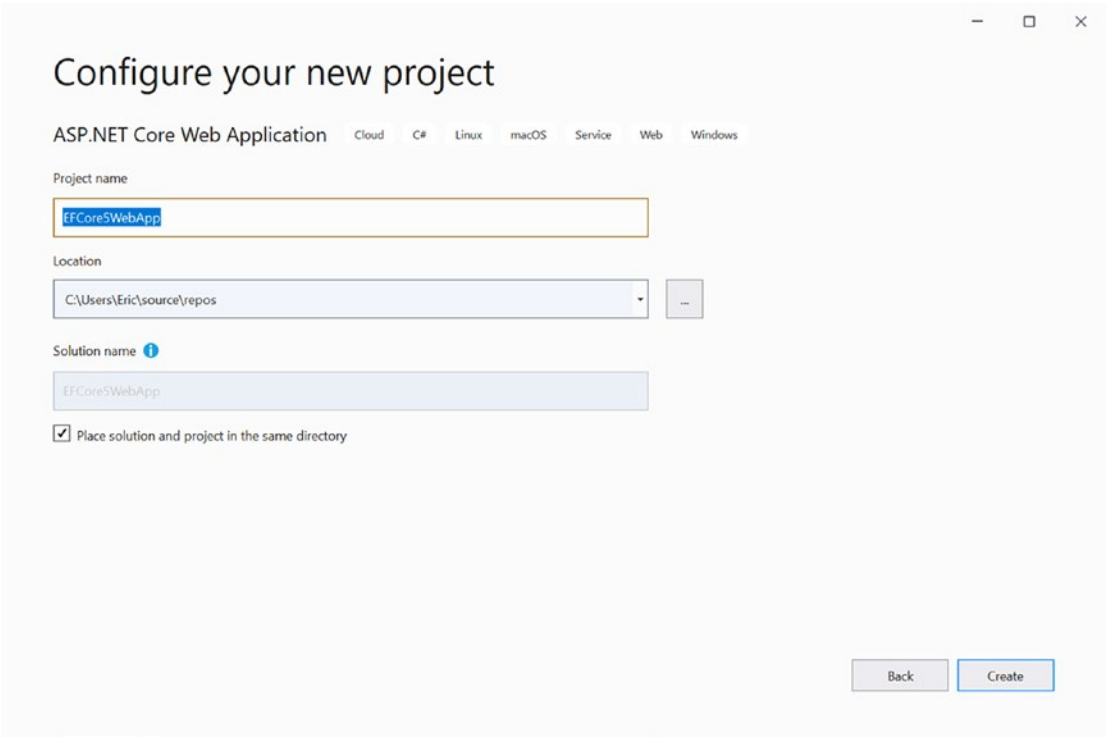

Figure 1-5. *Naming the Web App*

After that, select ASP.NET Core 3.1 as the target framework and select the Model-View-Controller template as seen in Figure 1-6.

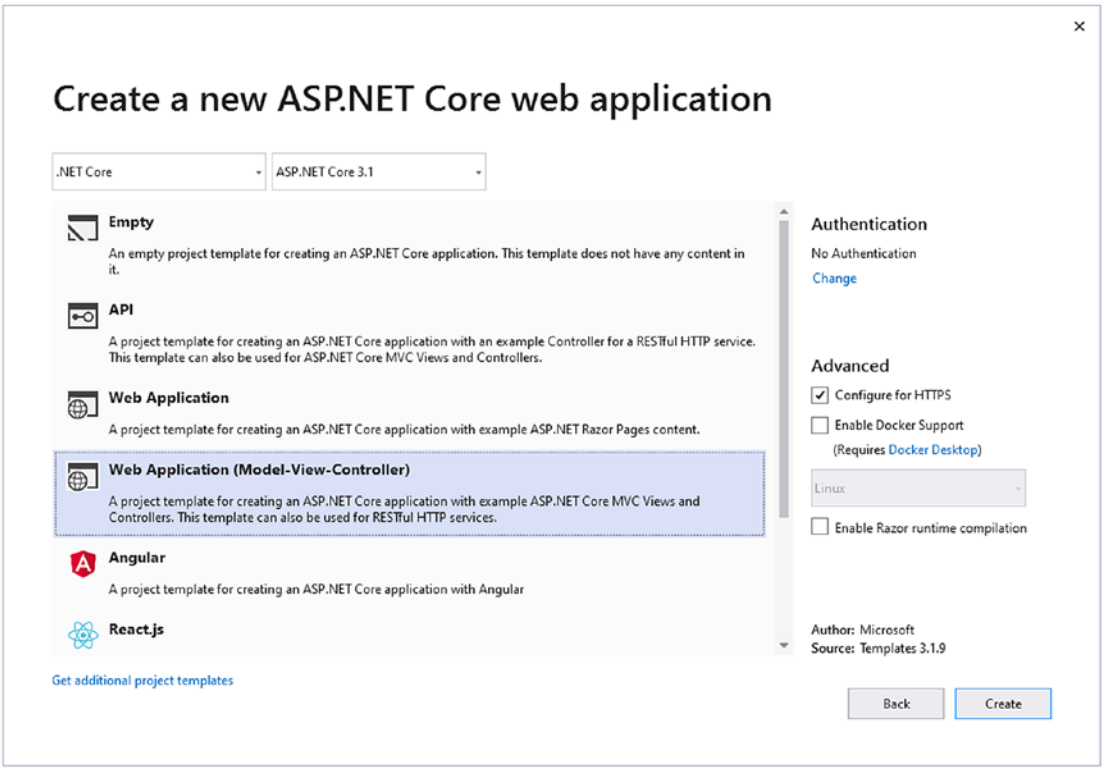

Figure 1-6. *Creating an ASP.NET Core 5.0 MVC App*

Then click the "Create" button to create the project and solution. After this, you will see the generated project opened in Visual Studio.

Install Entity Framework

Entity Framework Core 5 has providers for Microsoft SQL Server, SQLite, Cosmos, and in-memory databases. For this book, we will be using the SQL Server provider.

Entity Framework is installed through the NuGet Package Manager. We will be installing the Entity Framework Core 5 SQL Server NuGet package, which will allow us to interact with a Microsoft SQL Server instance.

Now it is time to install the Entity Framework Core 5 SQL Server NuGet package into the web app. Open the NuGet installer and search for "Entity Framework" and click the "Include prerelease" checkbox as shown in Figure 1-7.

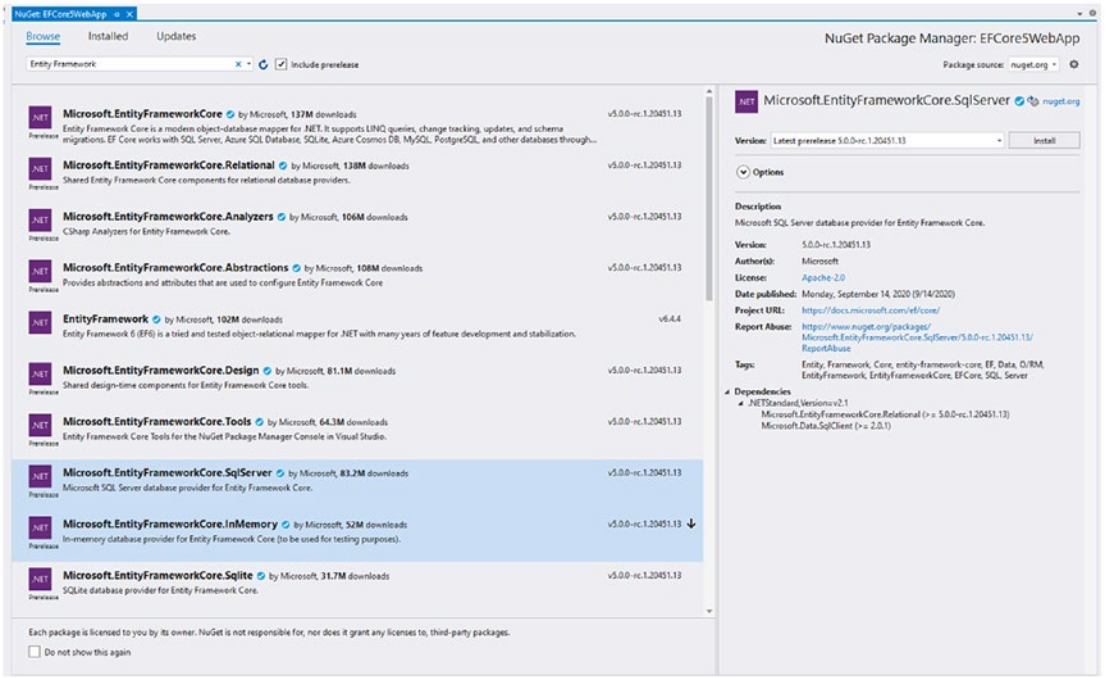

Figure 1-7. *Install the EF Core 5 SQL Server NuGet Package*

Click the Install button, and you should be prompted to confirm your installation as seen in Figure 1-8. Make sure the version is 5.* and that the package name is Microsoft.EntityFrameworkCore.SqlServer, which will install Entity Framework Core 5 with SQL Server support.

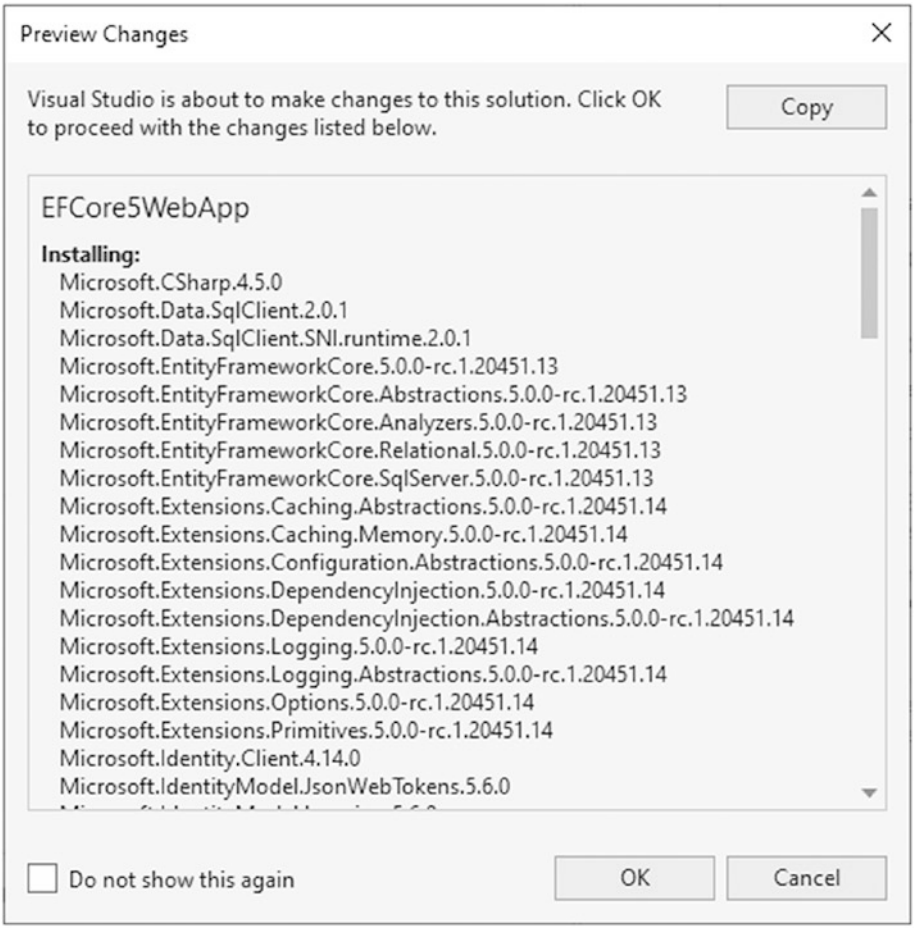

Figure 1-8. *Confirm EF Core SQL Server NuGet Installation*

Lastly, you'll be prompted to accept the license to install the EF Core 5 NuGet package as seen in Figure 1-9.

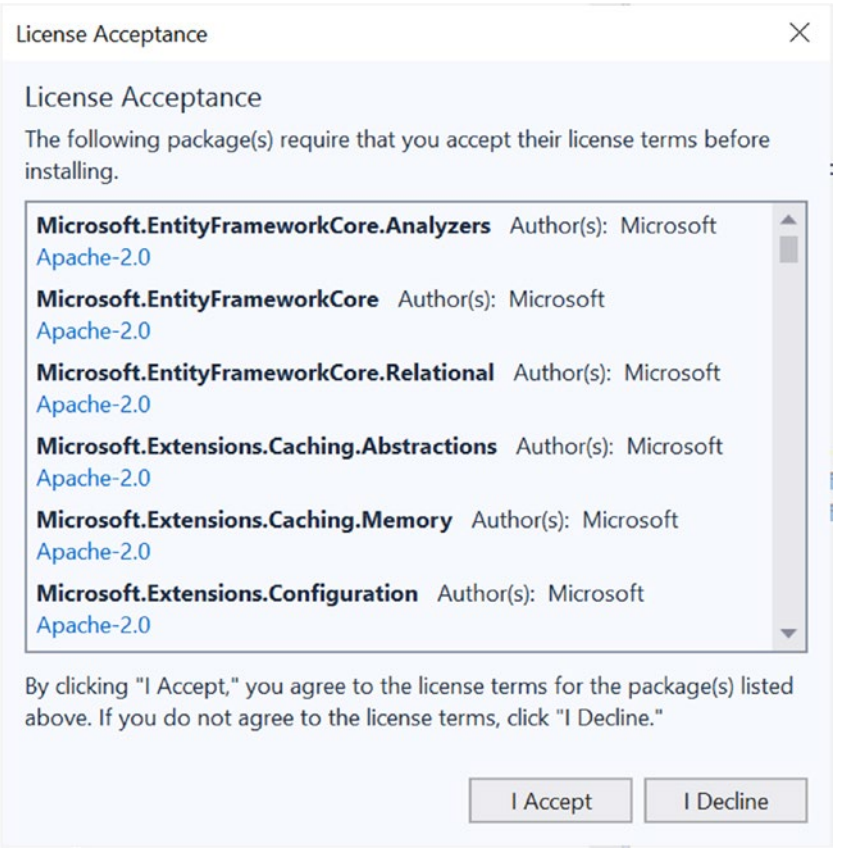

Figure 1-9. *Confirm EF Core 5 NuGet License*

Click the I Accept button, and the package will be installed.

Install the Core Tools Package

Finally, you'll need to install the Entity Framework 5 Core Tools package, which will allow you to create and update your database from the NuGet Package Manager Console. Open up the NuGet Package Manager and search for "Entity Framework Tools" and install the package as seen in Figure 1-10. Make sure to install the package named "Microsoft.EntityFrameworkCore.Tools" and that the version is 5.*.

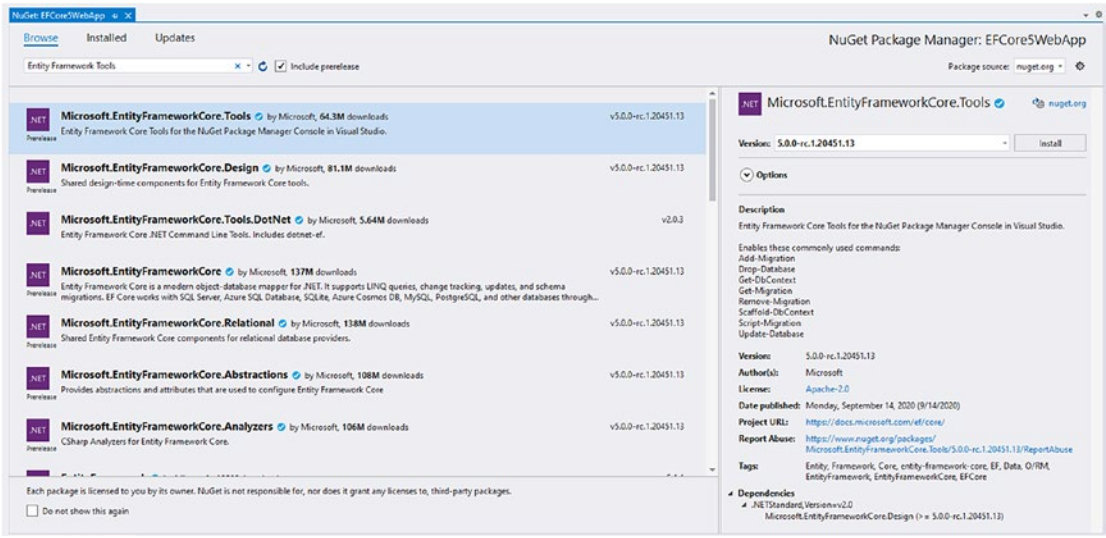

Figure 1-10. *Install the EF Core 5 Tools NuGet Package*

You have now successfully installed the needed tools to use Entity Framework Core 5. In the next chapter, we will cover how to move the Entity Framework 5 NuGet installation to a data access layer (DAL) project and how to structure your application using an N-tier architecture. You are well on your way to learning how to effectively use Entity Framework Core 5.

SQL Server Database

For this book, we will be using SQL Server Express LocalDb that is installed automatically by Visual Studio 2019 Preview as part of the .NET Core workload. Feel free to use your own full SQL Server instance. I will cover how to set the database connection string and create the database from code in Chapter 5.

Summary

In this chapter, you've installed the tools needed to use Entity Framework Core 5 with a Microsoft SQL Server instance. You've installed the latest Visual Studio 2019 and Entity Framework Core 5 into our solution.

With this groundwork lain, we can move on to structuring our application in the next chapter to effectively use Entity Framework Core 5. A good architecture goes a long way in allowing the application to be easily tested through unit and integration tests and eases maintenance of the application over time.

CHAPTER 2

Project Structure

Creating the structure of your application is an important step. In this chapter, I will detail how to use a multilayered architecture to separate concerns in the application. This will make your application easier to maintain and extend.

Core Project

The Core project is where the commonly used code shared across the entire application will reside. This is where the entity classes used by Entity Framework will live. This is also where common interfaces and utility code will exist.

To get started, open the solution you created in Chapter 1. Then add a new .NET Standard C# class library project named "EFCore5WebApp.Core" to the solution as seen in Figure 2-1.

E. Vogel, *Beginning Entity Framework Core 5*, https://doi.org/10.1007/978-1-4842-6882-7_2

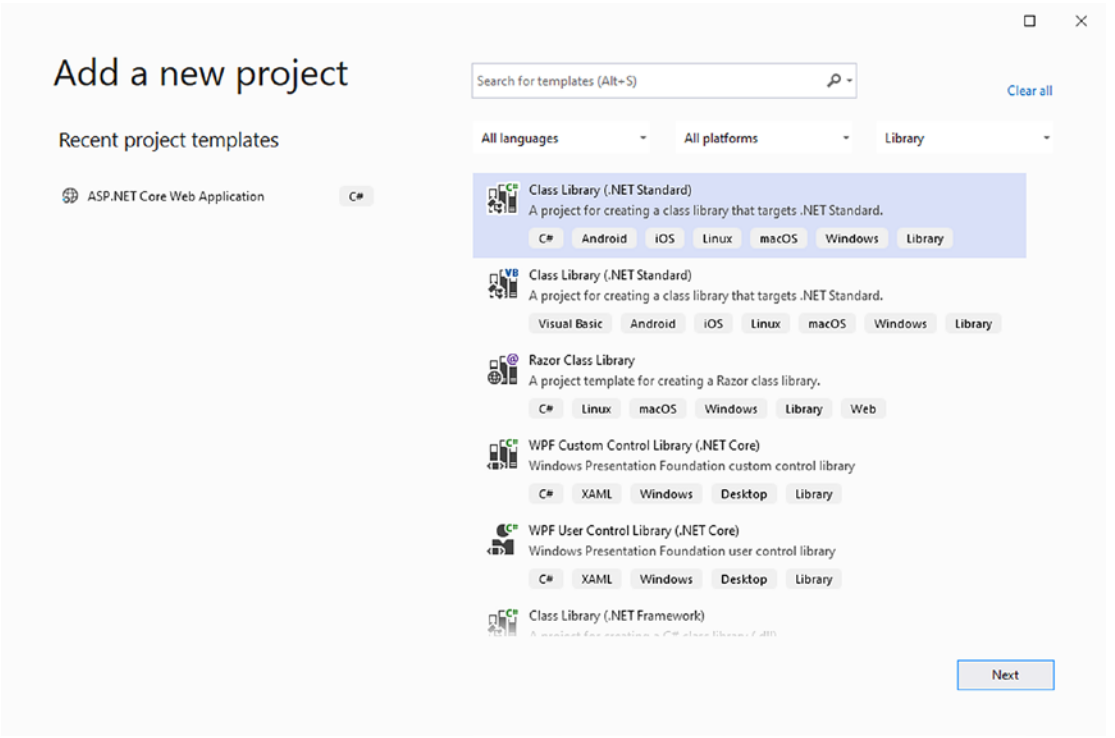

Figure 2-1. *New Class Library*

On the next screen, name your project "EFCore5WebApp.Core" as seen in Figure 2-2.

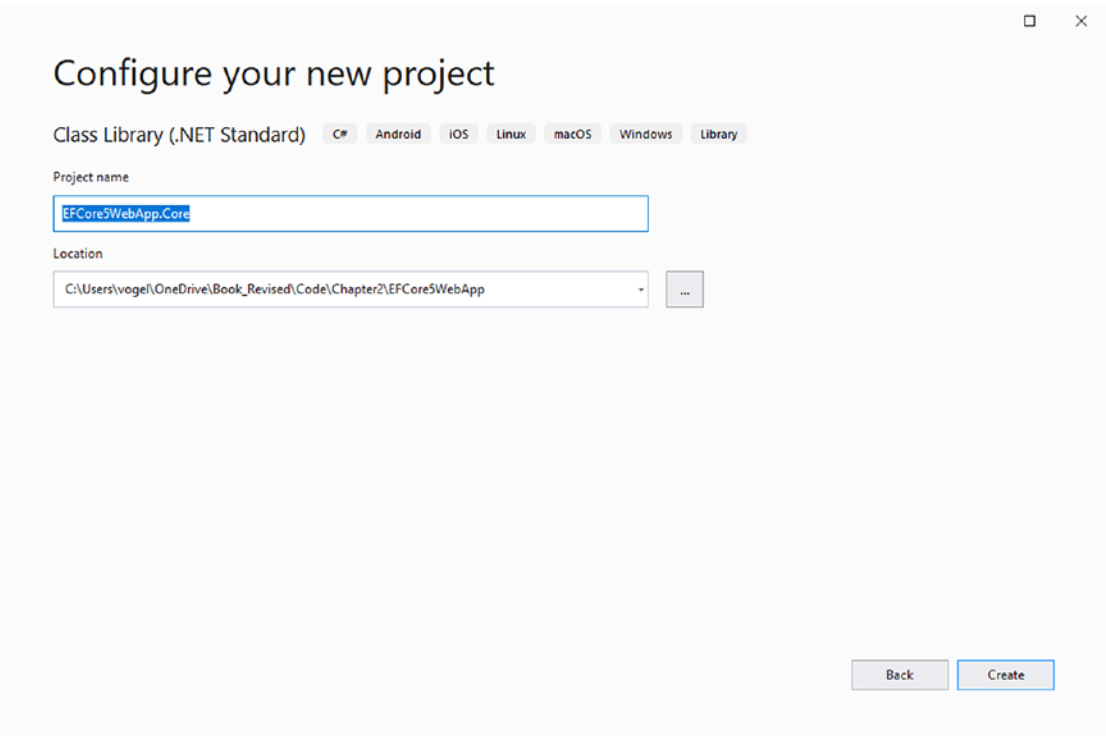

Figure 2-2. *Naming the Core Project*

Next, create a new NUnit Test Project (.NET Core) template project named "EFCore5WebApp.Core.Tests". This project will be where we'll create any needed unit tests for shared application code. You can easily find the correct template by searching for "NUnit" in the template dialog as seen in Figure 2-3.