

# Build an ASP.NET Core MVC App with EF Core

## One-Day Hands-On Lab

### Lab 0

Welcome to the Build an ASP.NET Core App with EF Core One-Day Hands-On Lab. Prior to starting the rest of the workshop, you must have the .NET (Core) 6.0.100+ SDK, .NET (Core) 6.0.0+ runtime, access to a local SQL Server Database, an appropriate .NET development IDE, and a SQL Server IDE.

SQL Server options:

- [Any OS] Docker Community with SQL Server 2019,
- [Windows] SQL Server LocalDb (installed with Visual Studio 2022), or
- [Windows] SQL Server 2019 Developer Edition (or above)

Supported .NET development IDEs include:

- [Windows] Visual Studio 2022
- [macOS] Visual Studio for the Mac 8.8+
- [Any OS] Visual Studio Code 1.62.0+

Supported SQL Server IDEs include:

- [Windows] SQL Server Management Studio
- [Any OS] Azure Data Studio

SQL Server Management Studio or Azure data Studio is recommended.

## Part 0: Permissions

You must have admin permissions on your machine to complete this hands-on lab.

## Part 1: Installing the Prerequisites

### Step 1: Install/Confirm .NET (Core) 6 Runtime and SDK

- Download and install the latest .NET 6 SDK, ASP.NET Core Runtime, and .NET Runtime from <http://dot.net> (the .NET Desktop Runtime is not used for the hands-on lab).
- Check the version of the .NET 6 Runtime by entering:

```
dotnet --list-runtimes
```

- The response will be (at the time of this writing):

```
Microsoft.AspNetCore.App 6.0.0  
Microsoft.NETCore.App 6.0.0
```

- Microsoft.AspNetCore.App leverages the ASP.NET Core shared framework. Any assets in the ASP.NET Core shared framework will not be deployed with your app and are pre-compiled for better application startup time. Microsoft.AspNetCore.App also uses version roll-forward to work with later versions of the .NET Core framework installed on the target machine.
- Check the version of the .NET 6 SDK by entering:

```
dotnet --list-sdks
```

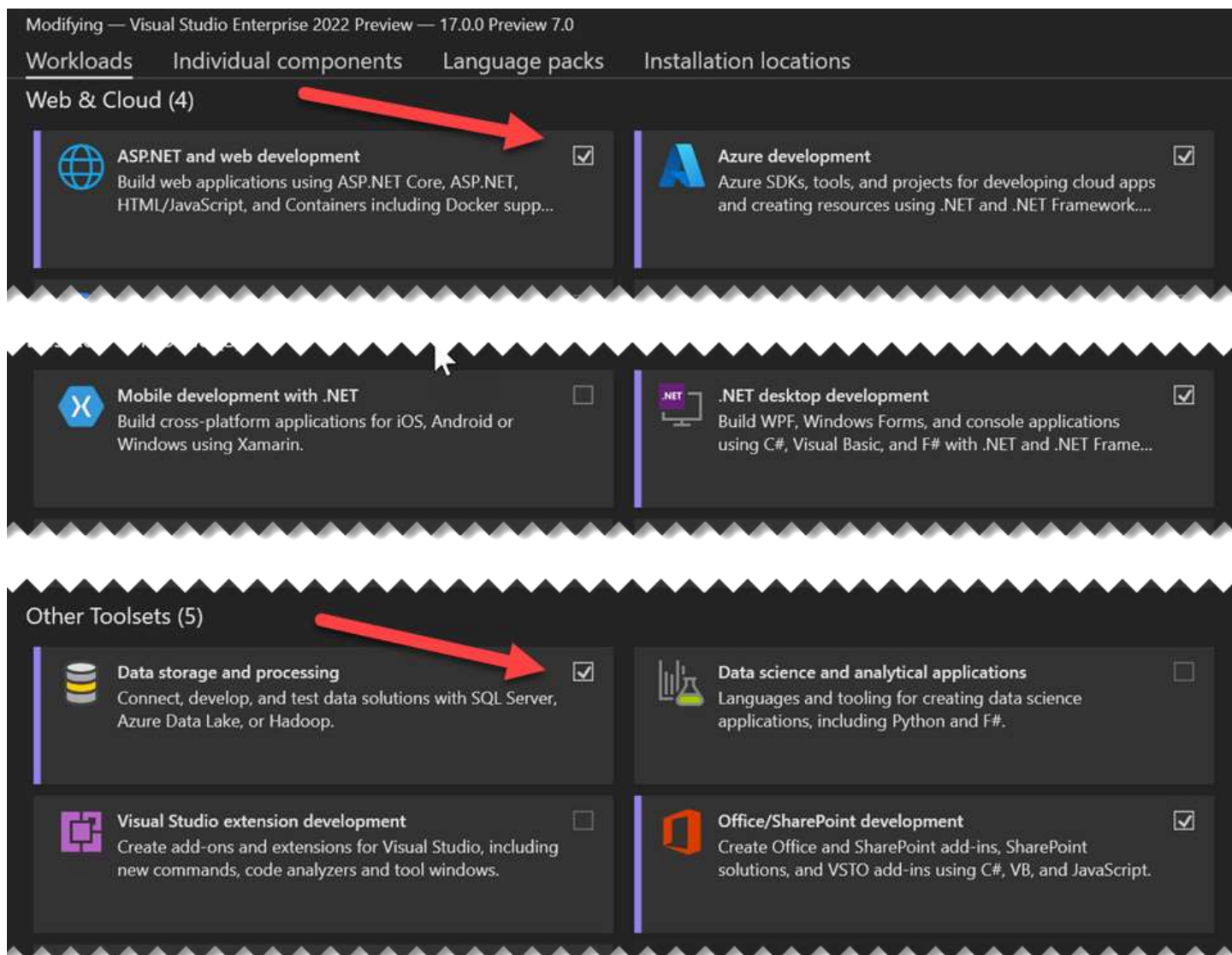
- The response will be (at the time of this writing):

```
6.0.100
```

## Step 2: Install an IDE

### Option 1: Download and install any edition of Visual Studio 2022

- If you already have Visual Studio 2022 installed or don't plan on using VS 2022, continue to the next step.
- Download Visual Studio 2022 (any edition) from the Visual Studio home page: <https://www.visualstudio.com>
  - a) The Community Edition is free, and has everything you need to complete this Hands-On Lab
- Start the installer
  - b) The new installation experience has separate workloads based on what type of work you intend to do. For this lab, select the “**ASP.NET and web development**” workload as well as the “**Data storage and processing**” workloads. The “**.NET desktop development**” workload is not used in this lab, but you can use it to experiment with the new WPF and Winforms frameworks.

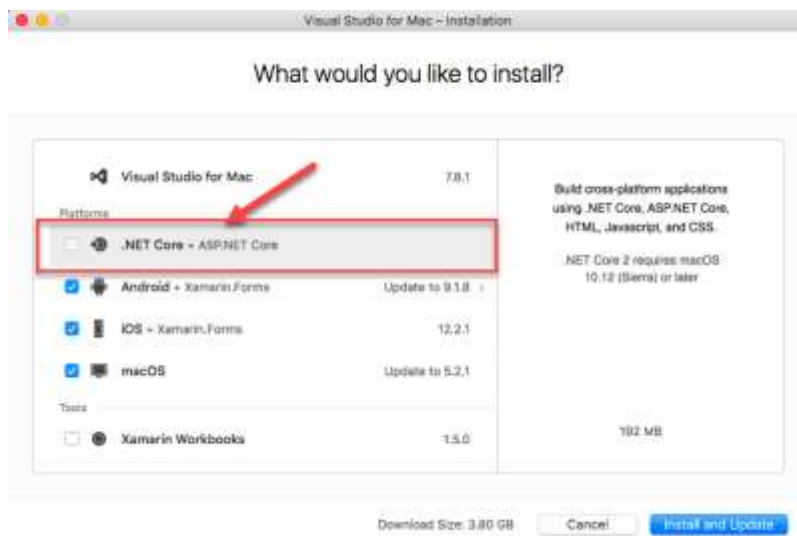


### Option 2: Download and install Visual Studio Code 1.62.0+

- If you already have VS Code 1.62.0 or above installed or don't plan on using VS Code, continue to the next step.
- Download Visual Studio Code from <https://visualstudio.microsoft.com/>.
- Install the "Microsoft C# extension (powered by OmniSharp)" extension.

### Option 3: Download and install Visual Studio for the Mac (8.8+)

- If you already have VS Mac version 8.8 or above installed or don't plan on using VS Mac, continue to the next step.
- Download Visual Studio for the Mac from <https://visualstudio.microsoft.com/>
- Select .NET Core from the install screen (image from 7.8.1)



### Step 3: Download and install SQL Server Tooling (SSMS or ADS)

Neither of these are required for the workshop but having one installed makes it easier to work with the database. You only need to install one.

#### Windows: Download/Install SQL Server Management Studio (SSMS)

- Download and install the free SQL Server Management Studio 18.9.2+ from <https://aka.ms/ssmsfullsetup>

#### Mac/Windows: Download/Install Azure Data Studio (ADS)

- Download and install the free Azure Data Studio 1.33.1+ from <https://docs.microsoft.com/en-us/sql/azure-data-studio/download-azure-data-studio?view=sql-server-ver15>

### Step 4: Install Docker Desktop (macOS/Linux)

Docker is a containerization platform that runs on Windows, MacOS, and Linux.

**NOTE:** If you are using a Windows based machine, Docker is optional for this workshop. If you are not on a Windows/Linux machine and don't have SQL Server 2019 installed on your laptop, Docker is required.

- Download and install Docker Desktop from <https://www.docker.com/products/docker-desktop>
  - a) Select the edition for your operating system (Windows/Mac)
  - b) During installation, when prompted what type of containers to use, select Linux containers (and not Windows containers), even if you are on a Windows machine. This is the type of container that will be loaded and is not related to your computer's operating system.
  - c) This is a free tool but requires you to have a Docker user id and password

## Step 5: Pull the SQL Server Image and Create the Local Container

A Docker image is like a class definition, while a Docker Container is like an instance of that class. To run SQL Server in Docker, you must first pull the image from Docker Hub, and then create a container using that image.

- Pull the SQL Server 2019 for Linux image. Enter the following command:

```
docker pull mcr.microsoft.com/mssql/server:2019-latest
```

- When creating an image, there are two required environment variables, “ACCEPT\_EULA” and “SA\_PASSWORD”. An optional environment variable “MSSQL\_PID” sets the product version. The host port mapping to the image port needs to be set, and a friendly name added. Create the container using the following command:

a) **NOTE:** On Windows, use double quotes ("). On Mac and Linux, use single quotes (').

```
docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=P@ssw0rd" -p 5433:1433 --name AutoLot -d mcr.microsoft.com/mssql/server:2019-latest
```

## Step 6: [Optional] Download and install SQL Server 2019 Developer (only if NOT using Docker AND on a Windows/Linux machine)

If you already have a version of SQL Server 2019 installed on your machine you can use that version.

- Download the SQL Server 2019 Developer Edition from <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

## Clone the Repo for the Hands-On Lab

The repo will become public the day before the workshop.

[https://github.com/skimedic/dotnetcore\\_hol\\_oneday](https://github.com/skimedic/dotnetcore_hol_oneday)

## Summary

These are all the tools you need to complete this Hands-on Lab.