



# EventStoreDB .NET (C#) Instructions for Running Locally

## Overview

Welcome to the .NET (C#) example of Event Store's **From Scratch** series. This series allows you to quickly overcome the common challenges of setting up and configuring a new development environment, and focus on advancing your EventStoreDB skills.

The **From Scratch** series provides working code examples for basic reads and writes to EventStoreDB, a tested environment to run the code, and instructions that clearly describe the steps required to run the code successfully.

Each **From Scratch** repository provides the following:

- A working Github Codespaces environment
- Instructions on running EventStoreDB locally
- Instructions to set up a similar project on your own

We recommend you progress through the **From Scratch** projects in the following order:

1. Run the code in Codespaces
2. Clone the For Scratch GitHub repo, and follow the instructions to run it locally
3. Build your own project

This document assumes you have successfully run the code in Codespaces. Your next step is downloading or cloning the GitHub repository and running the code on your computer. ***This is the recommended second stage in Event Store's From Scratch .NET series.***

Other clients in the **From Scratch** series include:

- Node
- Java
- Python

## Topics covered

1. Download or clone the Github repository
2. Install an IDE
3. Install .NET locally
4. Start a Docker container running EventStoreDB
5. Execute the code by running **dotnet run**

## Before you start

To run the code locally, you will need the following:

- A working .NET installation
- An IDE (\*optional)
- Docker

### 1. Download or clone the GitHub repository

On the GitHub repository's home page, select the green "Code" button. Ensure you are in the "Local" tab. Choose one of the following options to download the repo code to a local directory.

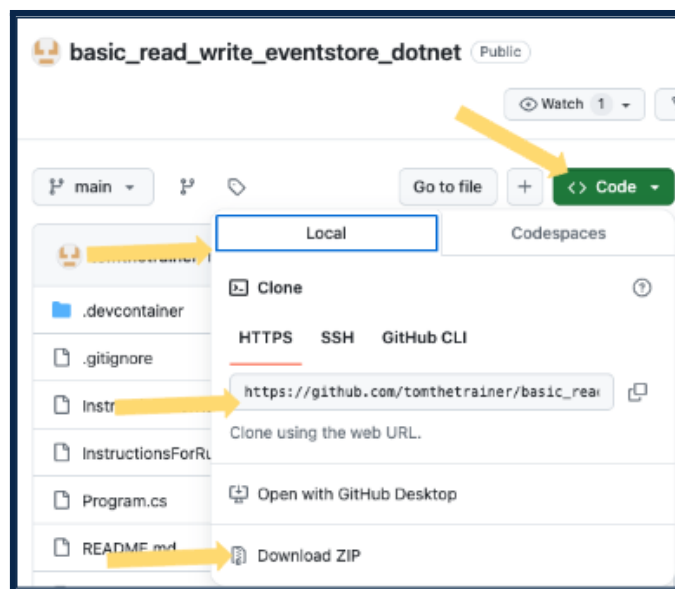
#### Using Git Clone

1. Copy the URL provided
2. Open a terminal window, and run the following command. (Replace "<repository\_url.git>" with the copied URL)

```
git clone <repository_url.git>
```

#### Download repo to a zip file

1. Select "Download ZIP" located at the bottom of the Local tab



## 2. Install an IDE

This is an optional step. If you prefer using an IDE, please follow these instructions to download VS Code.

<https://code.visualstudio.com/download>.

If you prefer an alternate IDE, please feel free to use the IDE of your choice.

If you opt not to use an IDE, the code can be run from the command line.

## 3. Install .NET locally

Similar to Java, .NET has a runtime environment to run compiled code. There is also an SDK, or software development kit, which includes a compiler to write and compile code.

If you are on a Mac, run the following command to access the SDK. If prompted, provide your system password.

```
brew install --cask .NET-sdk.
```

There are alternate methods for installing a .NET SDK. If you need additional support, please refer to the [.NET documentation](#).

After installing the .NET SDK, run the following command.

```
dotnet --version
```

It should return the version of .NET you are running. :

```
dotnet --version  
8.0.203
```

## 4. Start a Docker container running EventStoreDB

Install Docker by following these instructions: <https://docs.docker.com/engine/install/>

Download and run the EventStoreDB Docker container using the command line

The following command will start an unsecured single instance EventStoreDB cluster locally.

```
docker run -d --name esdb-node -it -p 2113:2113 -p 1113:1113 \
  eventstore/eventstore:lts --insecure --run-projections=All \
  --enable-external-tcp --enable-atom-pub-over-http
```

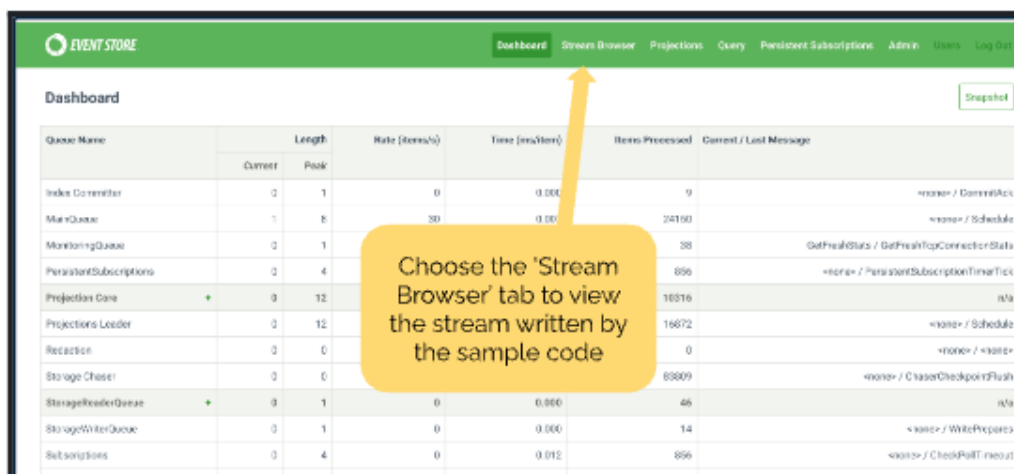
Like the Codespaces environment, you can view the EventStoreDB WebUI by pointing a browser to <http://localhost:2113/>.

## 5. Execute the code by running dotnet run

Running the following command in a terminal window to execute program.cs. This will write and read an event to the local EventStoreDB instance running in the Docker container.

```
dotnet run
```

After executing the code, visit the WebUI to verify the events were written to a stream, by viewing the "stream browser" tab.



Queue Name	Length		Rate (Items/s)	Time (ms/item)	Items Processed	Current / Last Message
	Current	Peak				
Index Generator	0	1	0	0.000	9	<none> / CommitAck
MainQueue	1	8	30	0.000	24190	<none> / Schedule
MonitoringQueue	0	1			38	GetFreshStats / GetFreshTopologyConnectionState
PersistentSubscriptions	0	4			896	<none> / PersistentSubscriptionTimerTick
Projection Core	9	12			10316	n/a
Projections Loader	0	12			16672	<none> / Schedule
Redaction	0	0			0	<none> / <none>
Storage Chaser	0	0			89009	<none> / ChaseCheckpointFlush
StorageReaderQueue	9	1	0	0.000	46	n/a
StorageWriterQueue	0	1	0	0.000	14	<none> / WritePrepared
Subscriptions	0	4	0	0.012	896	<none> / CheckPullTimeout

**Congratulations!** After running the sample program.cs you have succeeded in writing and reading events to and from EventstoreDB.

## Next Steps

Now that you have successfully written and read events in EventstoreDB locally, we recommend you follow the steps to build a local .NET project environment. Please continue to the **From Scratch** .NET instructions for setting up a local environment to continue your learning.

As you progress with your EventStoreDB skills, you can also find additional examples in the following repo:

<https://github.com/EventStore/samples>

In particular, we recommend the Quickstart examples here:

<https://github.com/EventStore/samples/tree/main/Quickstart>