Contents:

Overview of the FromScratch Project

Setting up a development environment that is configured for the work you need to do in your organisation is important. This takes time however. So how do you improve yourself as a developer and learn something new?

Some challenges you might face.

1. A desire to learn the code to interact with a new product but this requires a running example of the new product.
2. Helpful experts share code snippets that do not include key resources such as which version of the tools was used, which external products are imported, what other files are needed but are not included in the code snippet that was shared with you

Our Solution:

Provide end to end, complete examples as a github configured to run in github codespaces.
Provide detailed instructions on how the project was setup so you can run locally or you can create a similar project on your own.

Why Codespaces:
Github Created Codespaces as an answer to the problem developers of all levels from beginners to experts have faced, the “It doesn’t run on my machine” problem.

CodeSpaces provide you with the IDE, the repo, and the environment.

Going from Codespaces to local:

Once you get started with a new tool or a new language you certainly want to set up a local environment. Having a known working example in codespaces, should help you towards that goal.

In addition for the FromScratch project we have included the steps we took to build the environment that we used for these examples.

Running the Code in Codespaces

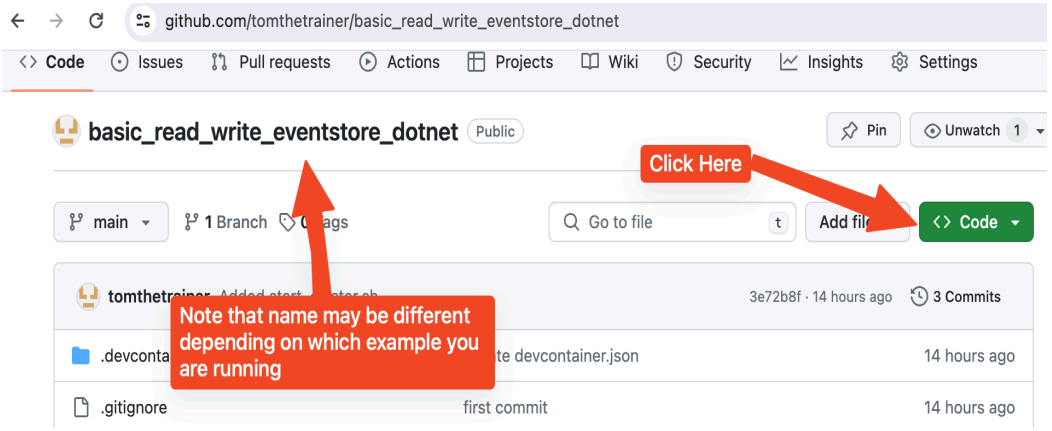
Here are the steps to run the code in the FromScratch repos in github codespaces.

Requirements:

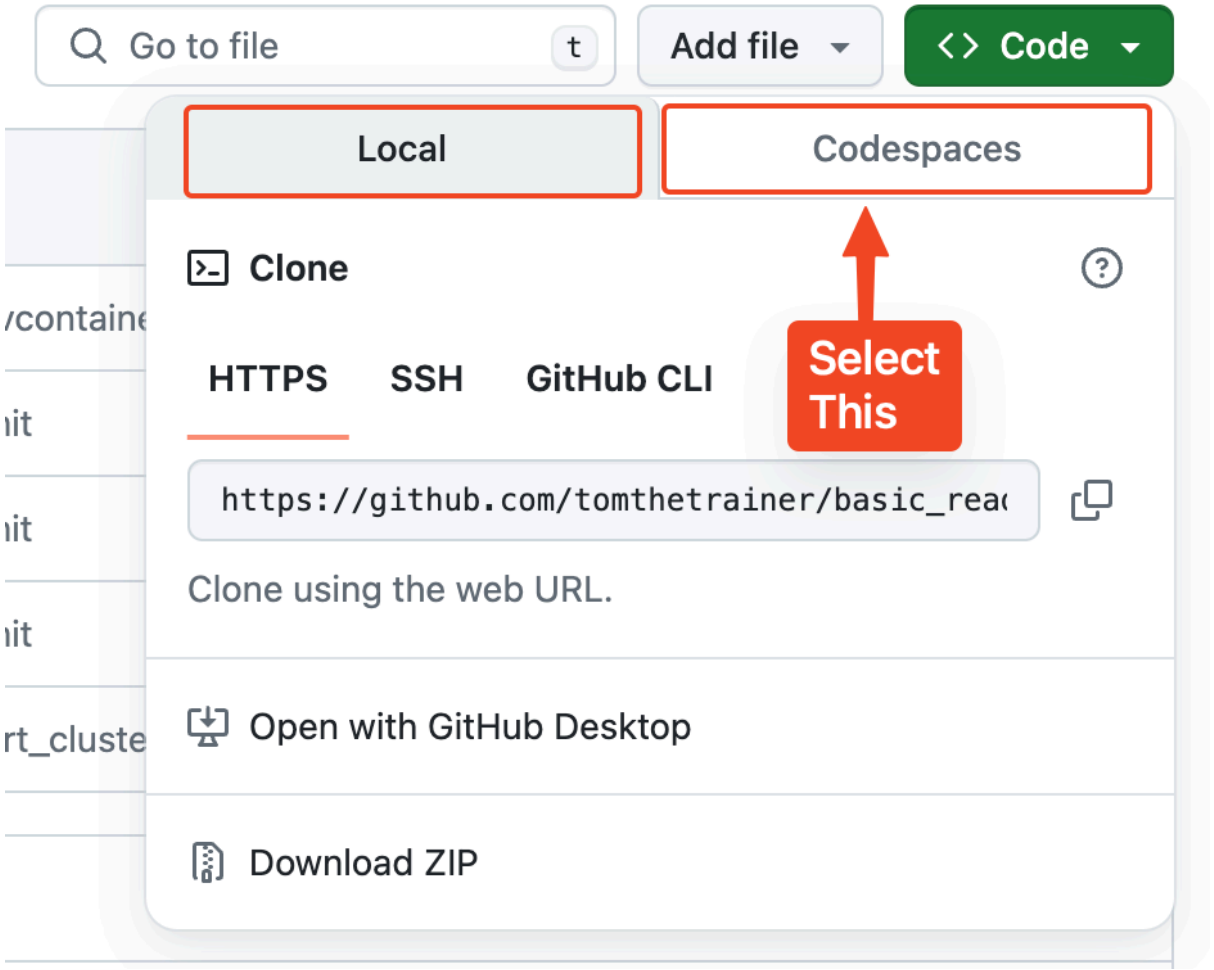
1. A github account
2. A browser

Steps:

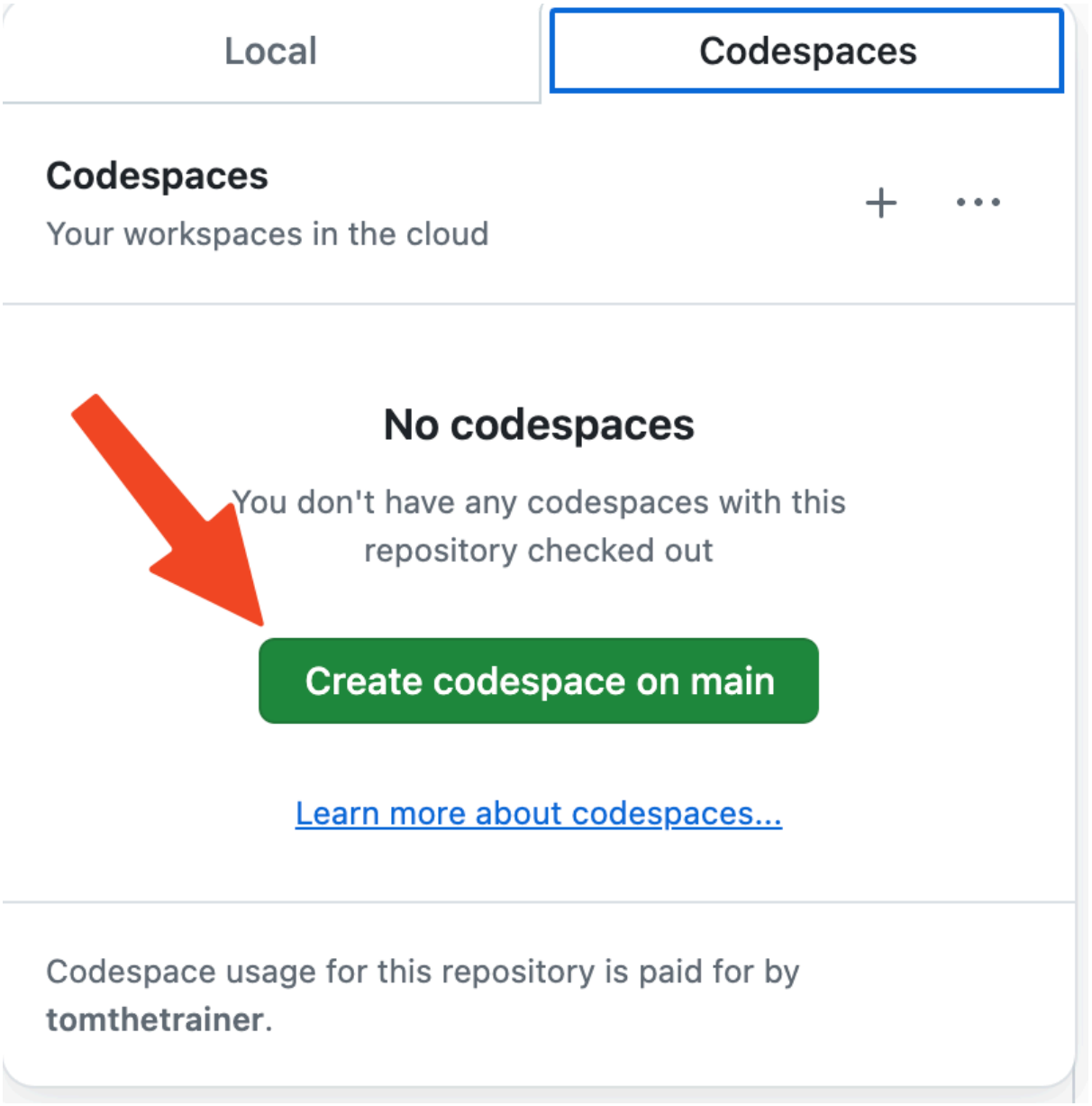
1. Point your browser to the FromScratch repo you would like to use, choices are node.js, dotnet, Java, and Python
2. Click on the Green Button labelled "<> code"



3.Given the choice of two tabs “Local” or “Codespaces”, choose “Codespaces”



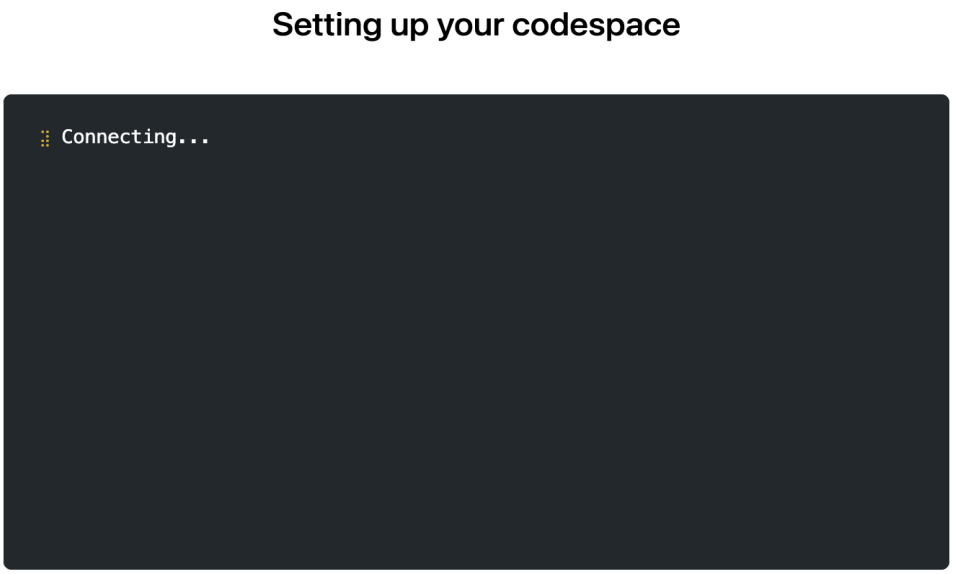
4. Click on the green button labelled “create codespace on main”



5. Wait for your codespace to launch

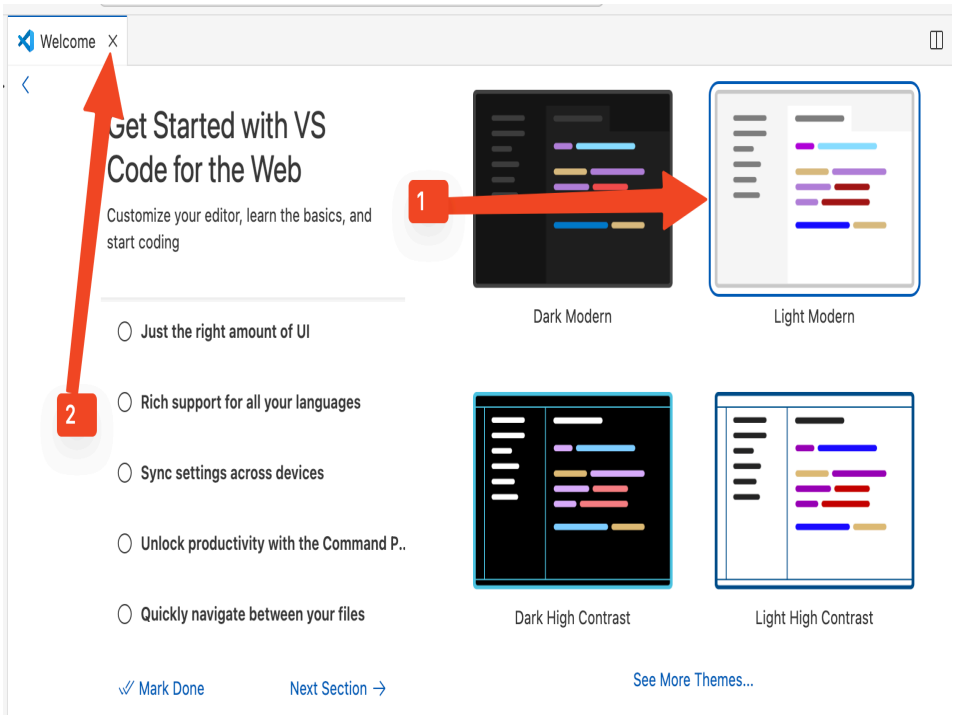
Depending on the configurations of the container, launching your codespace may take a few seconds, to a few minutes.

While it launches you will see this image.



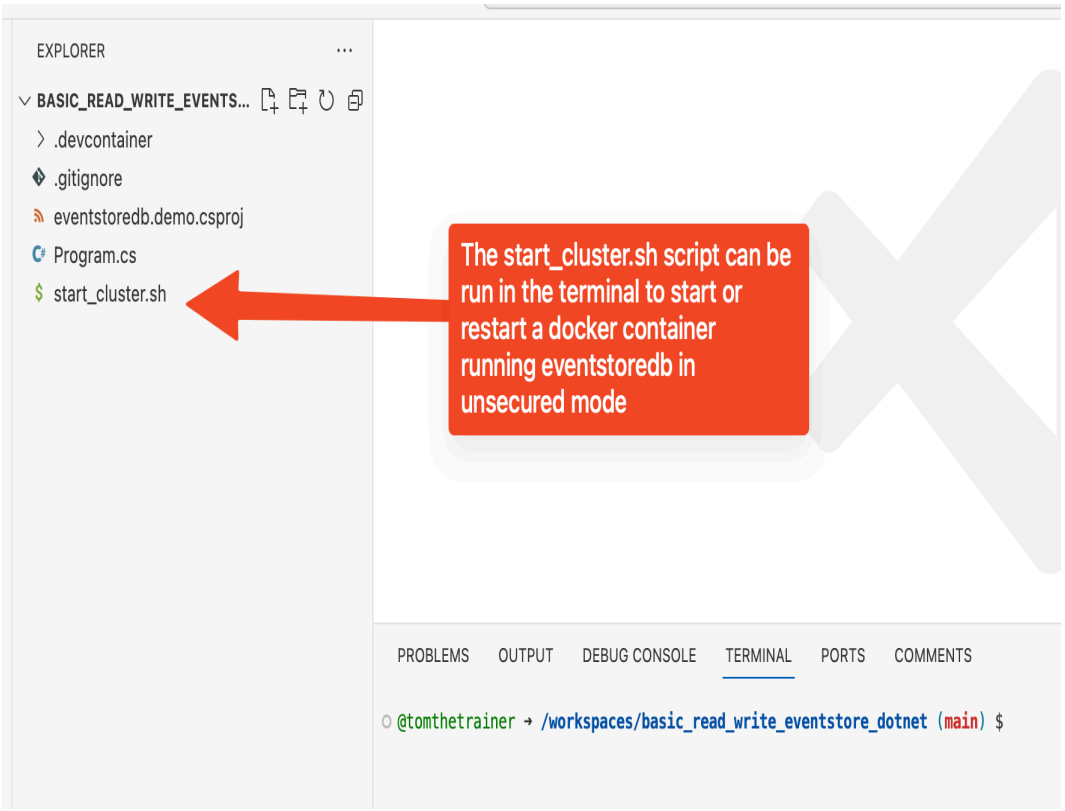
6. Make some formatting choices in the welcome page of vscode.

You can choose a theme click on “arrow 1” and close click on “arrow 2” or you can just close the ‘Welcome’ tab and accept the defaults

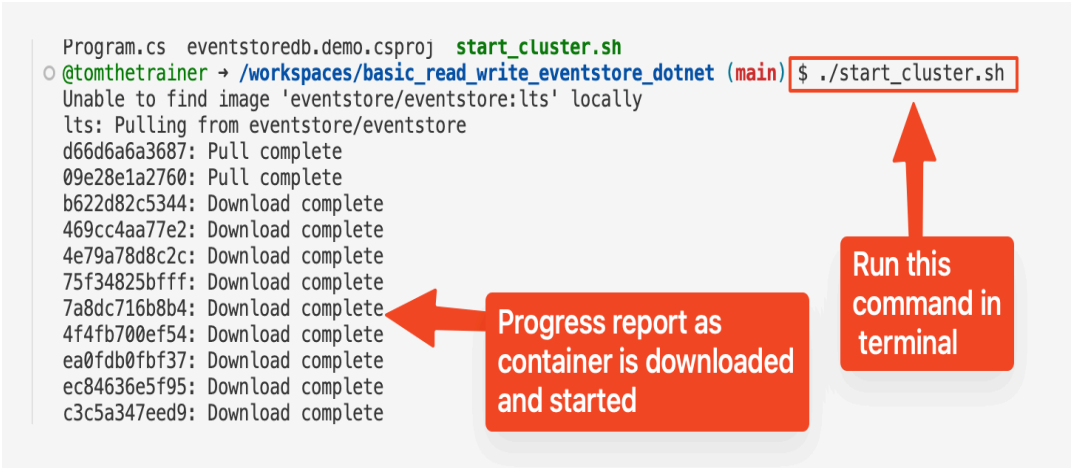


7. Use the 'start_cluster.sh' script to launch a docker container running eventstoredb.

This will be the eventstore instance that the "FromScratch" code will write and read events to.



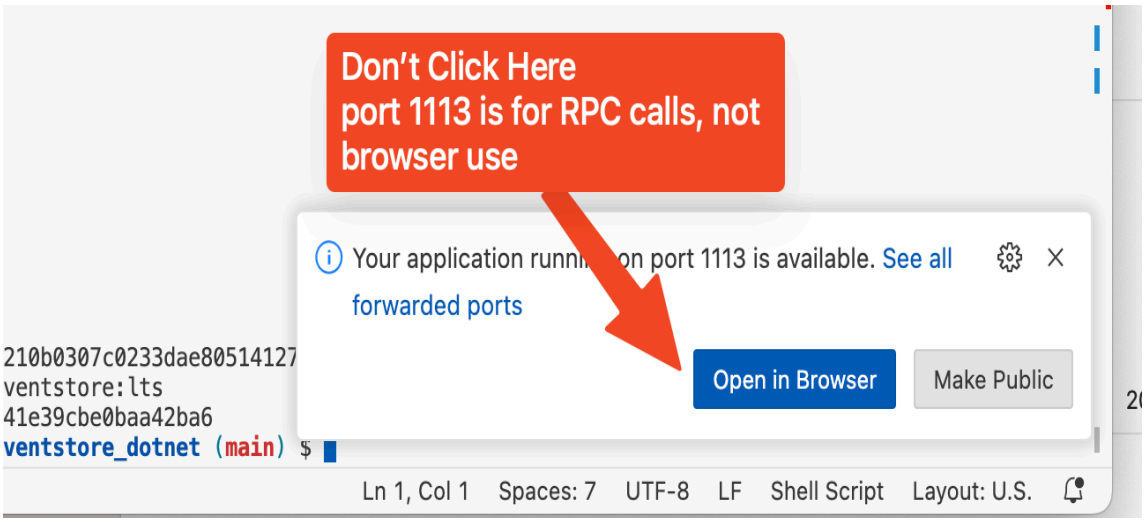
Open a terminal and run './start_cluster.sh'



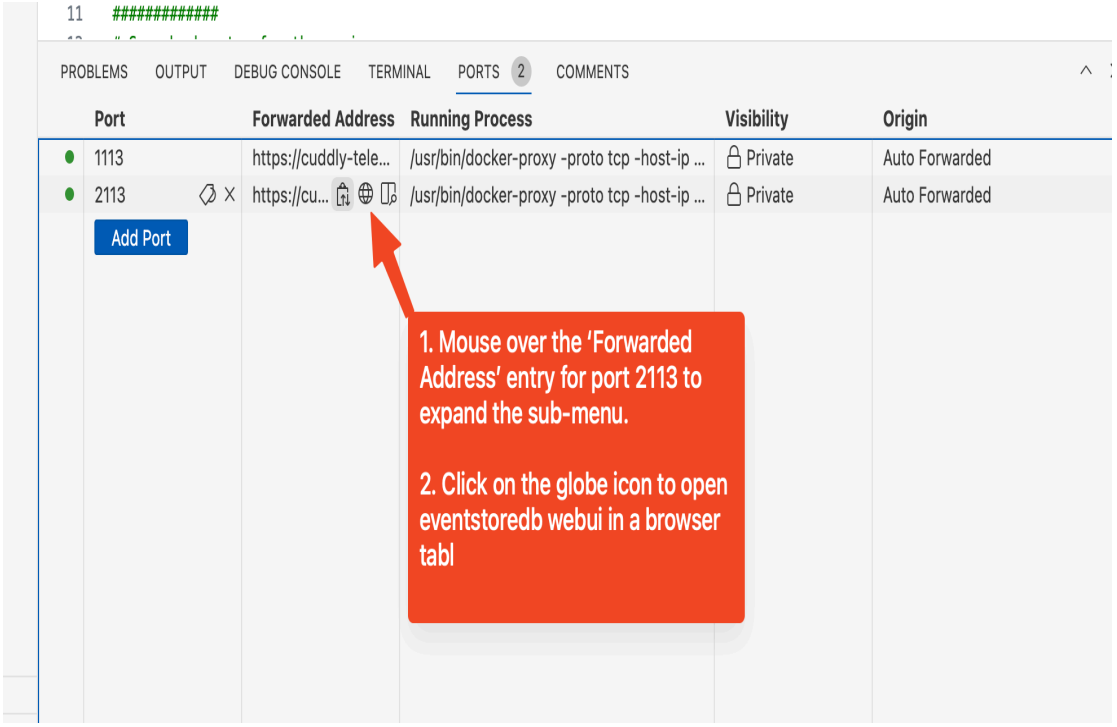
8. Open the webui of the eventstore running in the docker container.

Eventstoredb uses ports :1113 and :2113. Open up the webui 'port :2113' in a browser tab.

Don't click here as 1113 is used for RPC calls and is not the webui port.

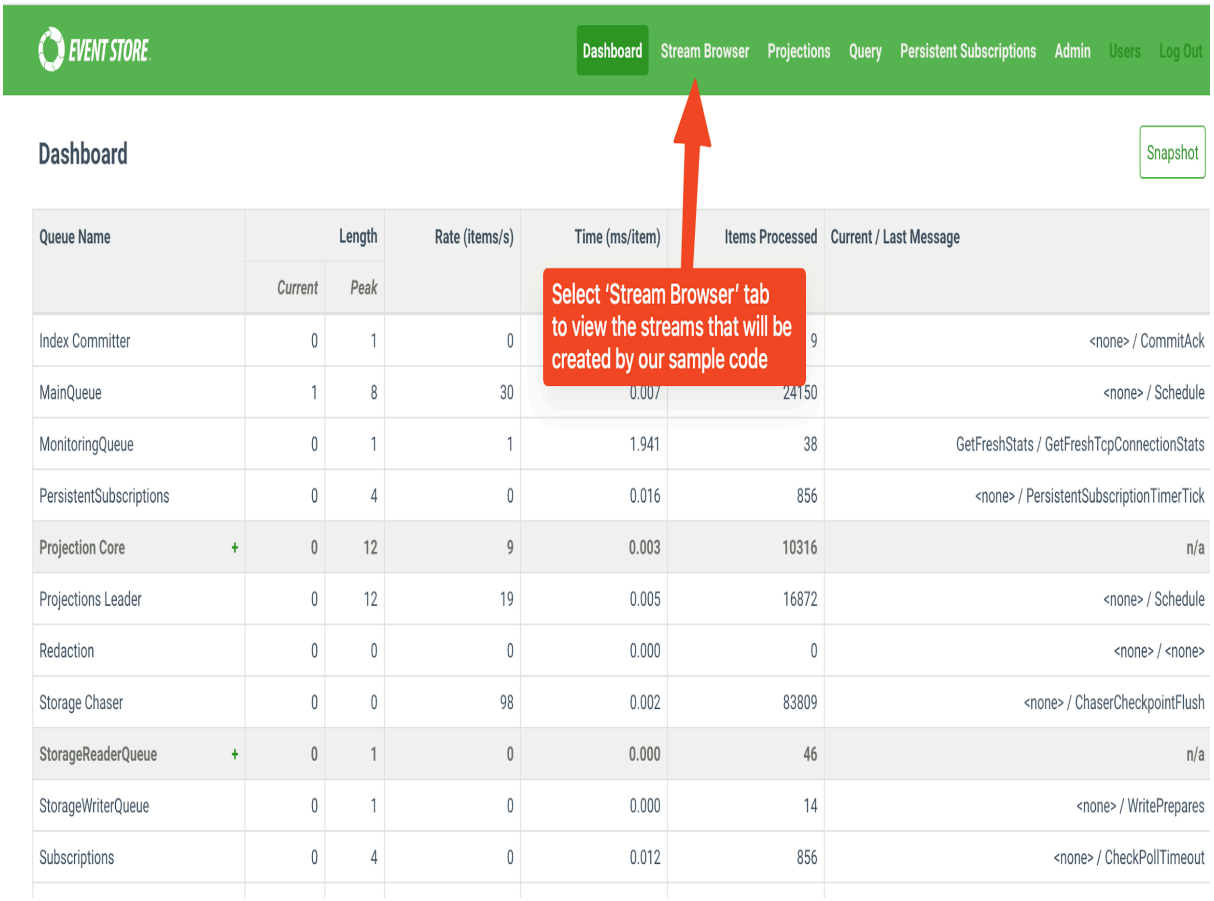


Click here instead.



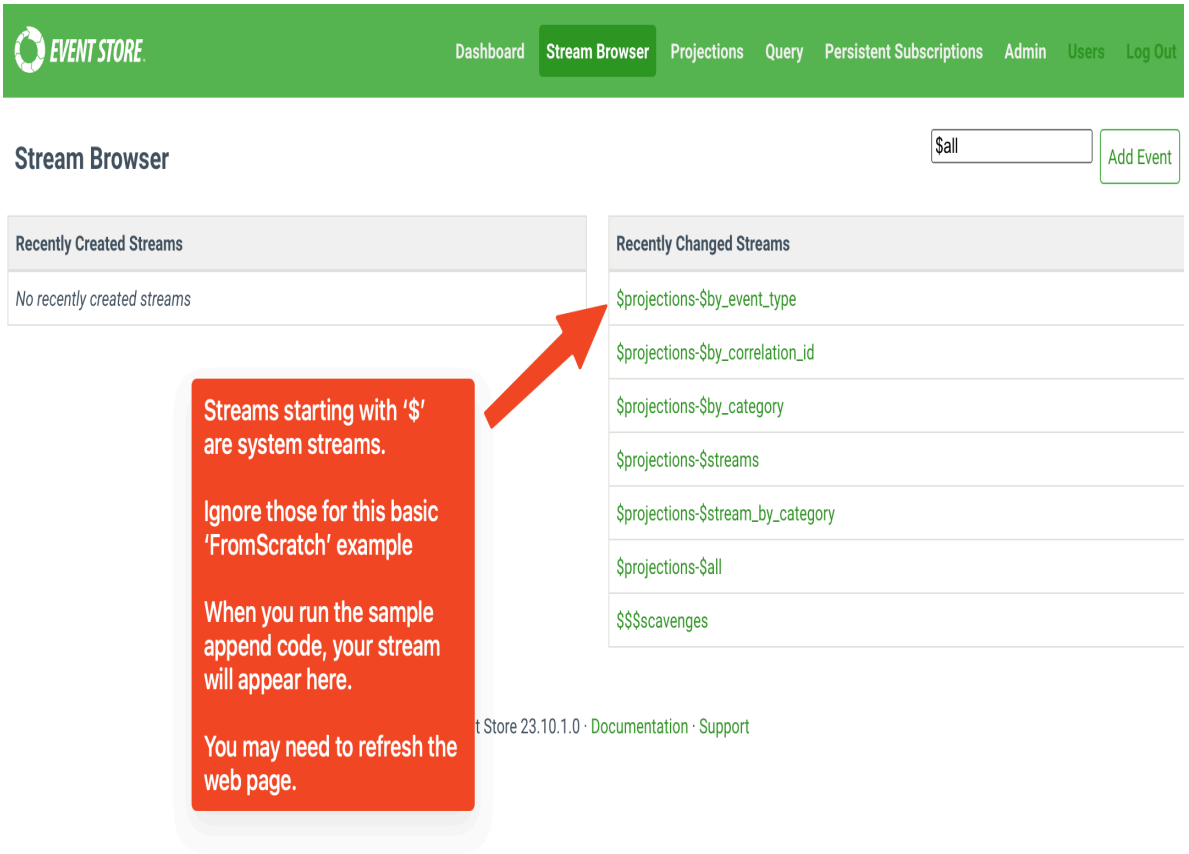
9. Select the stream browser tab from the eventstoredb webui.

The events written in the demo will be visible here after running the sample append code.



Queue Name	Length		Rate (items/s)	Time (ms/item)	Items Processed	Current / Last Message
	Current	Peak				
Index Committer	0	1	0		9	<none> / CommitAck
MainQueue	1	8	30	0.007	24150	<none> / Schedule
MonitoringQueue	0	1	1	1.941	38	GetFreshStats / GetFreshTcpConnectionStats
PersistentSubscriptions	0	4	0	0.016	856	<none> / PersistentSubscriptionTimerTick
Projection Core +	0	12	9	0.003	10316	n/a
Projections Leader	0	12	19	0.005	16872	<none> / Schedule
Redaction	0	0	0	0.000	0	<none> / <none>
Storage Chaser	0	0	98	0.002	83809	<none> / ChaserCheckpointFlush
StorageReaderQueue +	0	1	0	0.000	46	n/a
StorageWriterQueue	0	1	0	0.000	14	<none> / WritePrepares
Subscriptions	0	4	0	0.012	856	<none> / CheckPollTimeout

Stream Browser view explained



Congratulations you have successfully started the cluster, and viewed the stream browser from the eventstoredb webui.

Next Steps: Follow the code instructions for the language of your choice, node.js, python, dotnet, or java.