My humble solution

Located in Github:

Will show:

* Use of docker containers
  + Have docker compose using a yml file to:
    - Spin up a local SQL server
    - Spin up a local .NET Core Blazor website
    - Spin up a local .NET Core service
    - Spin up a Kafka Confluent cluster
    - Spin up an Apache NIFI instance
* Apache NIFI will be used to monitor a directory for CSV file
  + Once encountered, the file will be loaded and every row will be send to SQL
  + As well as some overhead information in SQL
* The .NET core service is looking for rows in the database and will pump them onto the kafka topic
* The website will be monitoring the topic and showing the rows

More:

* Possibly create Parquet out of CSV rows
* Use local CosmosDB/Azure Function?

<https://docs.microsoft.com/nl-nl/azure/cosmos-db/local-emulator?tabs=ssl-netstd21>

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-develop-vs?tabs=in-process>

* Use ML.NET

Docker:

Latest version SQL 2019, and then use developer edition

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