**Unit 1 Individual Project: Loading Streaming Data**

Cassi Mason

10/13/2024

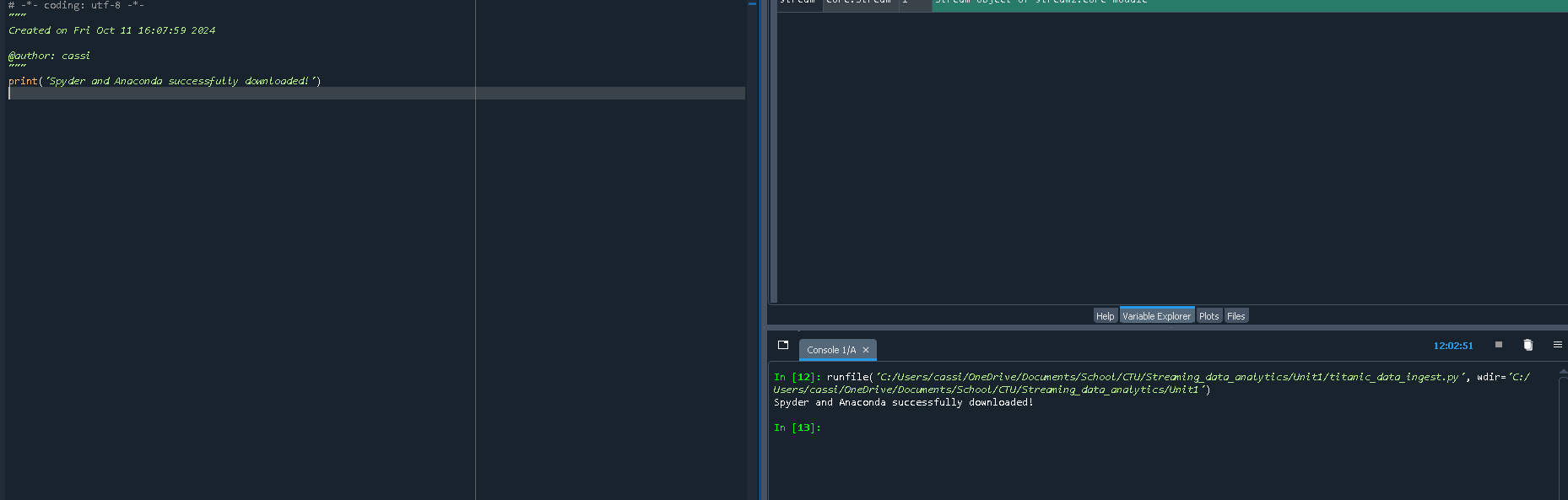
**Differences between streaming data analytics and at rest data analytics**

* Streaming data is constantly arriving, while data at rest has a defined start and end period
* There is a much higher volume of streaming data than data at rest
* Low latency with streaming data (quickly analyzing it), while there can be a much slower analysis process with data at rest
* Streaming data has a uniform load on the system because it is constantly being processed and analyzed, while data at rest changes the load as it idles between batches
* Streaming data can allow for real time actions to be taken based on business intelligence gained from analyzing it (movie recommendations, fixing machinery, etc.) while data at rest shows business intelligence after an event has happened.

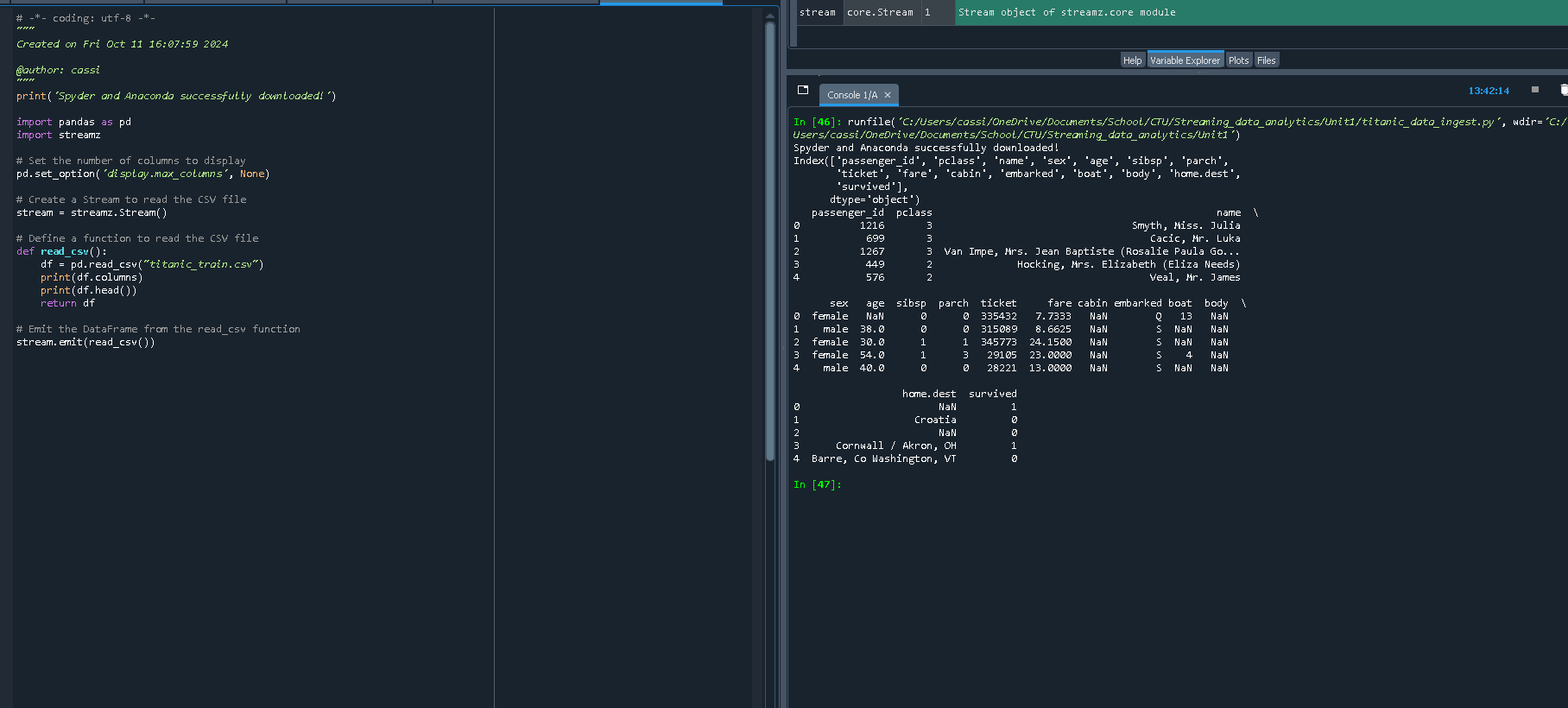
**How streaming data analytics works**

* Producers generate events
* Consumers process events: aggregate incoming data, send auto alerts in real time, etc.
* Events are transactions, activity on a website, clickstream, IoT, logs, records, etc.
* There is a buffer between the producers and consumers that allow consumers to process the data when they are ready.
* Data structure is append-only, so events cannot be altered, multiple consumers can read them without removing them. The retention period for keeping events in the buffer is predetermined
* Requires a lot of data and processing, so done in parallel scaling horizontally
* Sharding splits a data stream between several nodes, using a hash key for each record to keep track of where everything is
* Every shard is replicated on several nodes for failover and faster data transfer

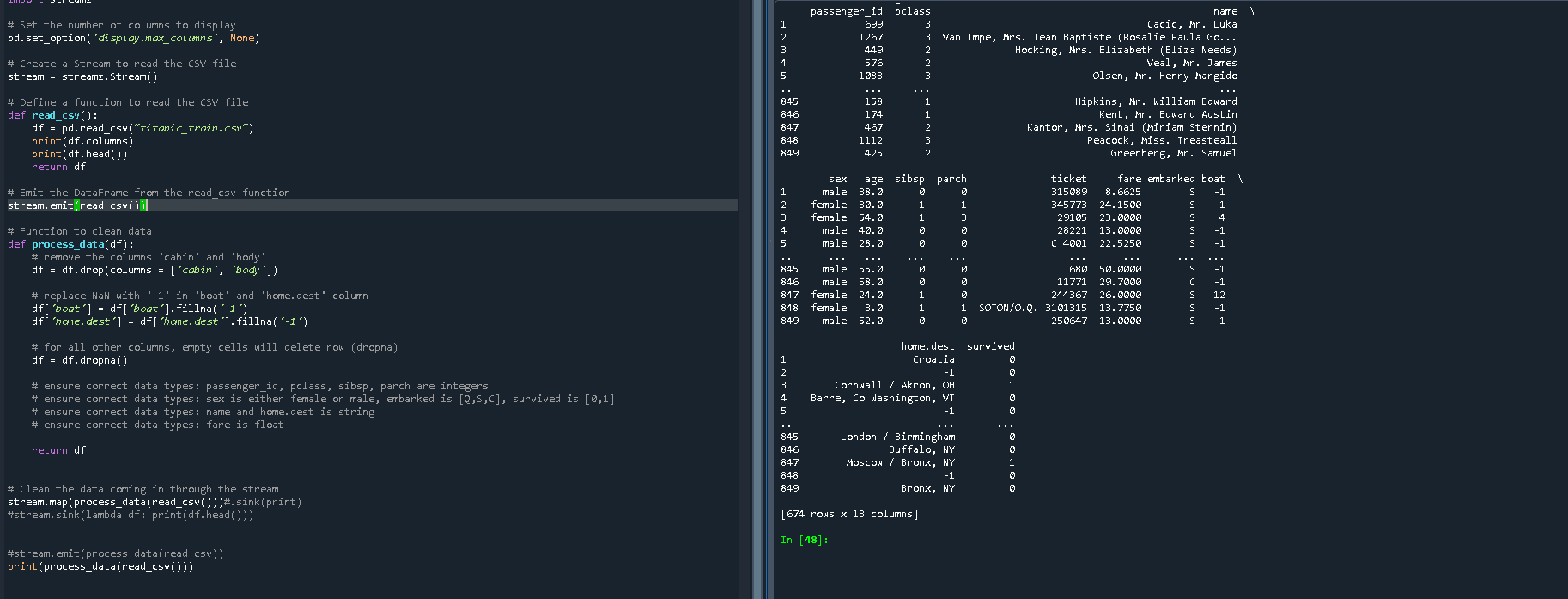
**Downloaded Spyder**

****

**Use streamz to create a stream of data that comes from the titanic data csv file**

****

**Clean the data, including dropping columns with mostly empty cells, replacing NaN with -1 to indicate no lifeboat and no known home destination, and deleting rows in other columns with empty values.**

****