**Glossary of Key Terms You Will Learn in this Lesson**

* Join (Streams) - The process of combining one or more streams into an output stream, typically on some related key attribute.
* Filtering (Streams) - The process of removing certain events in a data stream based on a condition
* Aggregating (Streams) - The process of summing, reducing, or otherwise grouping data based on a key attribute
* Remapping (Streams) - The process of modifying the input stream data structure into a different output structure. This may include the addition or removal of fields on a given event.
* Windowing (Streams) - Defining a period of time from which data is analyzed. Once data falls outside of that period of time, it is no longer valid for streaming analysis.
* Tumbling Window (Streams) - The tumbling window defines a block of time which rolls over once the duration has elapsed. A tumbling window of one hour, started now, would collect all data for the next 60 minutes. Then, at the 60 minute mark, it would reset all of the data in the topic, and begin collecting a fresh set of data for the next 60 minutes.
* Hopping Window (Streams) - Hopping windows advance in defined increments of time. A hopping window consists of a window length, e.g. 30 minutes, and an increment time, e.g. 5 minutes. Every time the increment time expires, the window is advanced forward by the increment.
* Sliding Window (Streams) - Sliding Windows work identically to Hopping Windows, except the increment period is much smaller -- typically measured in seconds. Sliding windows are constantly updated and always represent the most up-to-date state of a given stream aggregation.
* Stream - Streams contain all events in a topic, immutable, and in order. As new events occur, they are simply appended to the end of the stream.
* Table - Tables are the result of aggregation operations in stream processing applications. They are a roll-up, point-in-time view of data.
* Stateful - Stateful operations must store the intermediate results of combining multiple events to represent the latest point-in-time value for a given key

**Combining Streams**

* Combining, or joining, streams is the action of taking one or more streams and creating a single new output stream.
* Joined streams always share some common attribute across the data in all of the streams. For example, we might use a user\_id to merge user streams.
* State must be kept as events flow through the join calculation, until all of the related data has arrived. Once this happens, the new event can be emitted, and the state can be flushed
  + If the related data never fully arrives, at some point the data in memory should be cleared
  + This process is typically accomplished through *windowing*, which is covered in a later section of this lesson.

**Filtering Streams**

* Filtering a stream is the process of removing unwanted or unneeded data from an input stream, and outputting the desired data into a new stream
* Filtering may be a step in joining or combining two or more streams
* Filtering is often desirable when data clients don’t need access to all data for throughput or security reasons
* Applying filters earlier, rather than later, in the processing pipeline, can allow stream processing calculations to scale better and analyze less data

**Remapping Streams**

* Remapping streams is the process of transforming an input event and outputting it in a different form to a new stream
* Remapping may be done in conjunction with other processing steps, such as filters or joins
* Remapping is commonly used for data health, application compatibility, and security reasons
* Example Scenario 1: Transforming one data serialization format to another. E.g., Avro -> JSON, or JSON-> Avro
* Example Scenario 2: Removing sensitive or unnecessary fields from an input payload
* Example Scenario 3: Transforming an input event into a format suitable for downstream use by moving data fields or renaming them

**Aggregating Streams**

* An aggregation involves taking two or more distinct events and creating one or more new events based on a transformation function
* Aggregate Functions: Max, Min, Sum, TopN, HIstograms, Sets, Lists, and more
* Aggregates in streaming applications almost always involve a timeframe, unless the source topic is compacted

**Tumbling Window**

* Tumbling windows represent a fixed period of time that rolls over after that period of time has elapsed
  + ex: A 15 minute tumbling window started now would include all data from now until the 15th minute. On the 15th minute, the data is cleared and a new 15 minute window is started.
* Tumbling windows do not overlap
* Tumbling windows do not have gaps between windowed periods

**Sliding Window**

* Similar to Hopping Window, except the increment is not directly configurable and updates in real-time
  + A sliding window of the last 12 hours always includes all of the last 12 hours of data. Data is expired as soon as it reaches the 12-hour threshold, and new data is added as soon as it is received.
* Sliding Windows have no gaps between windows
* Sliding Windows do overlap

**Comparing Streams and Tables**

Streams and tables are not opposing concepts. In practice, the differentiation of a stream from a table in a stream processing application serves as a description of the type of data that is produced. Applications that are performing aggregations across incoming data are creating tables. Applications that are transforming incoming data into an unending sequence of events are streams.

**Further Optional Reading - Data Storage**

* [**RocksDB**](https://rocksdb.org/)
* [**Kafka Streams State**](https://docs.confluent.io/current/streams/architecture.html?&_ga=2.265603023.1364268795.1565759077-2091975159.1565759077#state)

**Glossary of Key Terms for this Lesson (same as glossary at beginning of lesson)**

* Join (Streams) - The process of combining one or more streams into an output stream, typically on some related key attribute.
* Filtering (Streams) - The process of removing certain events in a data stream based on a condition
* Aggregating (Streams) - The process of summing, reducing, or otherwise grouping data based on a key attribute
* Remapping (Streams) - The process of modifying the input stream data structure into a different output structure. This may include the addition or removal of fields on a given event.
* Windowing (Streams) - Defining a period of time from which data is analyzed. Once data falls outside of that period of time, it is no longer valid for streaming analysis.
* Tumbling Window (Streams) - The tumbling window defines a block of time which rolls over once the duration has elapsed. A tumbling window of one hour, started now, would collect all data for the next 60 minutes. Then, at the 60 minute mark, it would reset all of the data in the topic, and begin collecting a fresh set of data for the next 60 minutes.
* Hopping Window (Streams) - Hopping windows advance in defined increments of time. A hopping window consists of a window length, e.g. 30 minutes, and an increment time, e.g. 5 minutes. Every time the increment time expires, the window is advanced forward by the increment.
* Sliding Window (Streams) - Sliding Windows work identically to Hopping Windows, except the increment period is much smaller -- typically measured in seconds. Sliding windows are constantly updated and always represent the most up-to-date state of a given stream aggregation.
* Stream - Streams contain all events in a topic, immutable, and in order. As new events occur, they are simply appended to the end of the stream.
* Table - Tables are the result of aggregation operations in stream processing applications. They are a roll-up, point-in-time view of data.
* Stateful - Stateful operations must store the intermediate results of combining multiple events to represent the latest point-in-time value for a given key