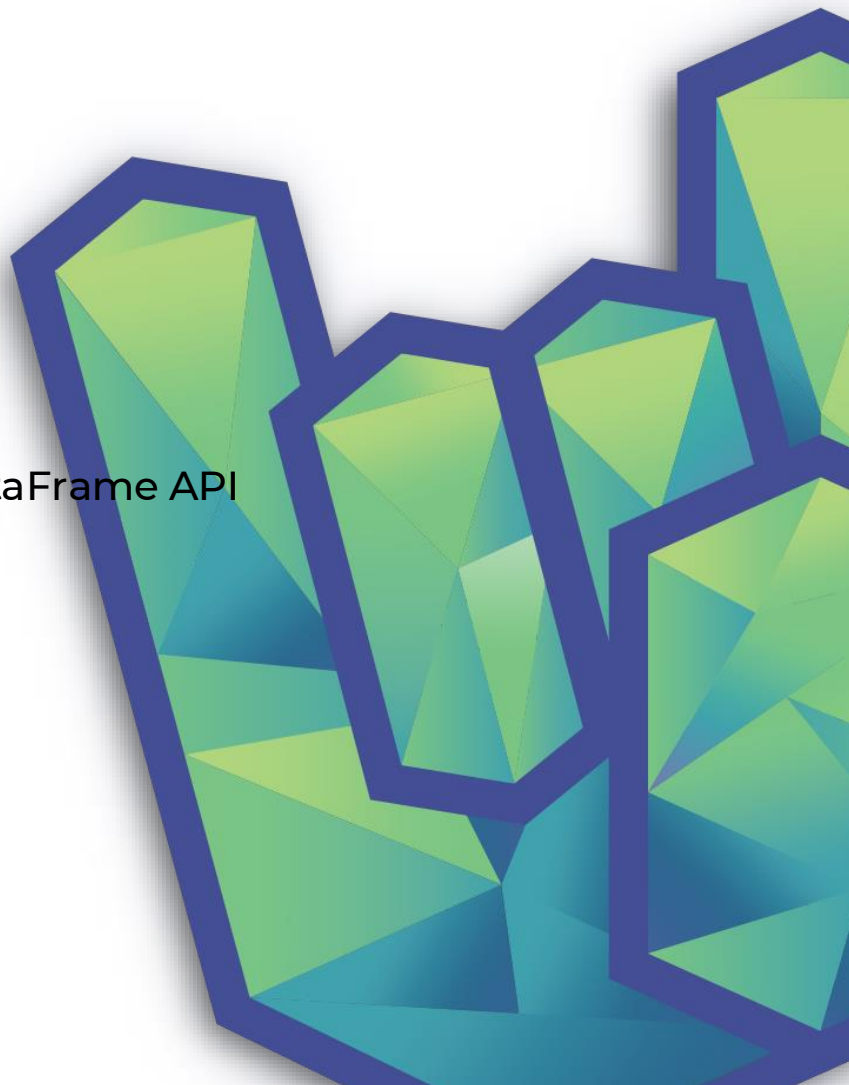


User Defined Aggregate Functions



Why

Custom aggregations unsupported by the DataFrame API



UDAFs

User defined aggregate functions: for whenever DataFrames API is not enough

```
// 1 - create an aggregator
object Concatenator extends Aggregator[String, String, String] {
  override def zero: String = ...
  override def reduce(buffer: String, newValue: String): String = ...
  override def merge(b1: String, b2: String): String = ...
  override def finish(finalBuffer: String): String = ...

  override def bufferEncoder: Encoder[String] = Encoders.STRING
  override def outputEncoder: Encoder[String] = Encoders.STRING
}

// 2 - register the UDF
val concatenatorUDAF = udaf(countWords)

// 3 - use the UDAF as a normal Spark function
val allCarNamesDF = carsDF
  .select(concatenatorUDAF(col("Name")).as("All_Cars"))
```

Spark rocks

