16.3.1

Key Features of Spark

You and Jennifer are really starting to get the hang of this! In your next conversation with SellBy, they tell you they've been using Spark to handle their datasets, and ask if you'll be able to use it, too. Thankfully, you and Jennifer had already set up a working meeting to dig into Spark, so you let them know that you'll be up and running in no time.

Hadoop is an ecosystem for handling big data. Expect to spend significant time configuring multiple servers or computers, as well as researching which technology can best deliver your big data solution. With the growing interest in big data and the ease of access to cloud technology, which we'll cover later, Hadoop is no longer required. New technologies allow more flexibility in data processing. One of these technologies is Spark.

Apache Spark (Spark) is a unified analytics engine for large-scale data processing. Spark lets you write applications in code that can run on Hadoop. However, Spark doesn't have to run on Hadoop, as it can run in stand-alone mode or in the cloud. Spark can be 100 times faster than Hadoop. Just like Hadoop's MapReduce, Spark works with data spread across a cluster, or a group of computers that work together.

Spark uses in-memory computation instead of a disk-based solution, which means it doesn't need to talk to the HDFS each time and can retain as much as HDFS can in-memory. Spark uses lazy evaluation, which delays the evaluation of an expression or command until the value is needed.

For example, when you direct Spark to count all the product reviews and then group them by star rating, Spark is ready to start, but you'll need to initiate the task—at this point, no counting or grouping has been done, only the instructions have been given. Once you give the go-ahead, Spark will then count and group the reviews all at once.

↻ Retake