Summary & Highlights

In this lesson, you learned that:

* Spark Architecture has driver and executor processes, coordinated by the Spark Context in the Driver​.
* The Driver creates jobs and the Spark Context splits jobs into tasks which can be run in parallel in the executors on the cluster​. Stages are a set of tasks that are separated by a data shuffle. Shuffles are costly, as they require data serialization, disk and network I/O.​ The driver program can be run in either client Mode (connecting the driver outside the cluster) or cluster mode (running the driver in the cluster).
* Cluster managers acquire resources and run as an abstracted service outside the application. Spark can run on Spark Standalone, Apache Hadoop YARN, Apache Mesos or Kubernetes cluster managers, with specific set-up requirements.​ Choosing a cluster manager depends on your data ecosystem and factors such as ease of configuration, portability, deployment, or data partitioning needs. Spark can also run using local mode, which is useful for testing or debugging an application.
* 'spark-submit’ is a unified interface to submit the Spark application, no matter the cluster manager or application language​. Mandatory options include telling Spark which cluster manager to connect to; other options set driver deploy mode or executor resourcing. To manage dependencies, application projects or libraries must be accessible for driver and executor processes, for example by creating a Java or Scala uber-JAR​. Spark Shell simplifies working with data by automatically initializing the SparkContext and SparkSession variables and providing Spark API access.
* Running Spark on IBM Cloud provides enterprise security and easily ties in IBM big data solutions for AIOps, IBM Watson and IBM Analytics Engine​. Spark’s big data processing capabilities work well with AIOps tools, using machine learning to identify events or patterns and help report or fix issues​. IBM Spectrum Conductor manages and deploys Spark resources dynamically on a single cluster and provides enterprise security.​ IBM Watson helps you focus on Spark’s machine learning capabilities by creating automated production-ready environments for AI​. IBM Analytics Engine separates storage and compute to create a scalable analytics solution alongside Spark’s data processing capabilities.
* You can set Spark configuration using properties (to control application behavior), environment variables (to adjust settings on a per-machine basis) or logging properties (to control logging outputs)​. Spark property configuration follows a precedence order, with the highest being configuration set programmatically, then spark-submit configuration and lastly configuration set in the spark-defaults.conf file​. Use Static configuration options for values that don’t change from run to run or properties related to the application, such as the application name​. Use Dynamic configuration options for values that change or need tuning when deployed, such as master location, executor memory or core settings​.
* Use Kubernetes to run containerized applications on a cluster, to manage distributed systems such as Spark with more flexibility and resilience. You can run Kubernetes as a deployment environment, which is useful for trying out changes before deploying to clusters in the cloud​. Kubernetes can be hosted on private or hybrid clouds, and set up using existing tools to bootstrap clusters, or using turnkey options from certified providers​. While you can use Kubernetes with Spark launched either in client or cluster mode, when using Client mode, executors must be able to connect with the driver and pod cleanup settings are required.