CRT020: Databricks Certified Associate Developer for Apache Spark 2.4 with Python 3 – Assessment

Summary

Databricks Certified Associate Developer for Apache Spark 2.4 & Python 3 validates your knowledge of the core components of the DataFrames API and confirms that you have a rudimentary understanding of the Spark Architecture.

To register for this certification please click the button below and follow the instructions to create a certification account and process payment.

REGISTER (HTTPS://GO.PROCTORU.COM/INVITATIONS/OPEN-INVITE?TOKEN=MZAZMTK2MG9WZW4TAW52AXRL)

Categories

All Categories (/categories)

Public Trainings (/category/public-trainings)

Self Paced (/category/self-paced)

Certifications (/category/certifications)

Description

The test is approximately 180 minutes with a series of randomly selected, multiple-choice questions, and another set of randomly selected coding challenges. Implementation of the coding challenges is completed within the Databricks product. Candidates are advised to become familiar with our online programming environment by signing up for the free version of Databricks, the Community Edition (https://databricks.com/signup#signup/community). Memorization of the APIs is not required, and access to the Programming and API docs will be made available during the exam.

NOTE: This certification is not affiliated with the Apache Software Foundation.

Prerequisites

None

Expectation of Knowledge

Spark Architecture Components

Candidates are expected to be familiar with the following architectural components and their relationship to each other:

- Driver
- Executor
- Cores/Slots/Threads
- Partitions

Spark Execution

Candidates are expected to be familiar with Spark's execution model and the breakdown between the different elements:

- Jobs
- Stages
- Tasks

Spark Concepts

Candidates are expected to be familiar with the following concepts:

- Caching
- Shuffling
- Partitioning
- Wide vs. Narrow Transformations
- DataFrame Transformations vs. Actions vs. Operations
- High-level Cluster Configuration

DataFrames API

Candidates are expected to have a command of the following APIs.

SparkContext

Candidates are expected to know how to use the **SparkContext** to control basic configuration settings such as **spark.sql.shuffle.partitions**.

SparkSession

Candidates are expected to know how to:

- Create a **DataFrame**/**Dataset** from a collection (e.g. list or set)
- Create a **DataFrame** for a range of numbers
- Access the **DataFrameReaders**
- Register User Defined Functions (UDFs).

DataFrameReader

Candidates are expected to know how to:

- Read data for the "core" data formats (CSV, JSON, JDBC, ORC, Parquet, text and tables)
- How to configure options for specific formats
- How to read data from non-core formats using **format()** and **load()**
- How to specify a DDL-formatted schema
- How to construct and specify a schema using the **StructType** classes

DataFrameWriter

Candidates are expected to know how to:

- Write data to the "core" data formats (csv, json, jdbc, orc, parquet, text and tables)
- Overwriting existing files
- How to configure options for specific formats
- How to write a data source to 1 single file or N separate files
- How to write partitioned data
- How to bucket data by a given set of columns

DataFrame

- Have a working understanding of every action such as **take()**, **collect()**, and **foreach()**
- Have a working understanding of the various transformations and how they work such as producing a distinct set, filtering data, repartitioning and coalescing, performing joins and unions as well as producing aggregates
- Know how to cache data, specifically to disk, memory or both
- Know how to uncache previously cached data
- Converting a **DataFrame** to a global or temp view.
- Applying hints

Row & Column

Candidates are expected to know how to work with row and columns to successfully extract data from a **DataFrame**

Spark SQL Functions

When instructed what to do, candidates are expected to be able to employ the multitude of Spark SQL functions. Examples include, but are not limited to:

- Aggregate functions: getting the first or last item from an array or computing the min and max values of a column.
- Collection functions: testing if an array contains a value, exploding or flattening data.
- Date time functions: parsing strings into timestamps or formatting timestamps into strings
- Math functions: computing the cosign, floor or log of a number
- Misc functions: converting a value to crc32, md5, sha1 or sha2
- Non-aggregate functions: creating an array, testing if a column is null, not-null, nan, etc
- Sorting functions: sorting data in descending order, ascending order, and sorting with proper null handling
- String functions: applying a provided regular expression, trimming string and extracting substrings.
- UDF functions: employing a UDF function.
- Window functions: computing the rank or dense rank.

Memorization of every command, their parameters, and return types are not necessary, in that access to the Spark API docs and Databricks docs are provided during the exam. However, a failure to efficiently identify and use these commands can result in an excessive amount of time being spent researching those commands which in turn may result in an inability to complete the exam in the allowed amount of time.

Register

To register for this certification please click the button below and follow the instructions to create a certification account and process payment.

REGISTER (HTTPS://GO.PROCTORU.COM/INVITATIONS/OPEN-INVITE?TOKEN=MZAZMTK2MG9WZW4TAW52AXRL)

PRODUCT
(HTTPS://DATABRICKS.COM/PRODUCT/DATABRICKS)

APACHE SPARK
(HTTPS://DATABRICKS.COM/SPARK/ABOUT)

SOLUTIONS
(HTTPS://DATABRICKS.COM/SOLUTIONS)

CUSTOMERS
(HTTPS://DATABRICKS.COM/CUSTOMERS)

(https://twitter.com/d

(https://www.linkedir

(https://www.faceboc

(https://databricks.cc

(https://www.youtub