PySpark Installation & Configuration Guide (Windows)

This document is a step-by-step guide to install and configure PySpark correctly on Windows systems. It is intended to avoid common technical issues and to serve as onboarding guidance for new members of the Data Analytics team.

Overview

PySpark requires Python and Java. Installing PySpark via pip includes Apache Spark and is sufficient for local development and Databricks-style workflows. Hadoop installation is not required unless working with secured or on-prem HDFS clusters.

1. Python Installation

To use PySpark, Python and pip must be installed on the local machine. Download Python from https://www.python.org/downloads/ and follow the installation steps. Ensure that pip is selected during installation.

Validate the installation by running the following commands in PowerShell, Command Prompt, or a Linux terminal:

1 python --version (Expected output: Python 3.X.X)

2 pip --version

2. PySpark Installation

Install the PySpark interface using pip by running the following command:

1 pip install pyspark

3. Java & Hadoop

Java must be installed, preferably Java 8, as it is the most compatible version with Spark. Download the JDK from https://www.oracle.com/java/technologies/downloads/#java8 and follow the installation steps.

Verify the Java installation by running the following command:

1 Java -version

Optional: Hadoop bineries need to be configured as well. Since the Hadoop community never officially packaged Windows binaries, a widely used and trusted GitHub mirror is commonly relied upon: https://github.com/steveloughran/winutils. This repository is maintained by a Hadoop committer and is considered the de-facto standard.

4. Directories & Environment Variables

Create a simple directory in the C drive, called hadoop for example, then create a folder inside it called 'bin'. After downloading the repository ZIP file and extracting it, open the folder that matches the PySpark-compatible version and copy all files into the bin folder.

Spark needs to locate Java at runtime; therefore, environment variables must be created. Open PowerShell or Command Prompt and run the following commands:

1 setx JAVA_HOME path_to_your_JDK_between_double_quotations

2 setx PATH "%PATH%;%JAVA_HOME%\bin"

3 (Optional) setx HADOOP_HOME path_to_your_hadoop_directory_in_C_drive

```
(Optional) setx PATH "%PATH%;path_to_bin_folder_in_hadoop_folder"
```

By running the previous commands, environment variables for the dependencies were created for Spark to use.

5. Validation & Testing

To make sure everything was configured correctly, open a new PowerShell or Command Prompt and run the following command:

```
spark-submit --version
```

If the setup is correct, Spark version information will be displayed, indicating that PySpark is ready to execute Python scripts that use the PySpark interface and related libraries.

Note: This guide is for Windows operating systems only. Although the steps are similar on Linux, there are platform-specific differences.