Functional Programming with Scala

Project Title: NistJsonToMongo

Group Name: 9

ROBERT Suzanne

IG5_Polytech

November 19, 2024

About the Project

- **My Specific Contribution:**
  - I worked on the connection between NIST JSON data and MongoDB.
  - I implemented the data pipeline using Scala and Spark.
  - I ensured compatibility between Spark, Scala, and MongoDB by resolving dependency issues in ‘build.sbt‘.
  - I developed the logic to filter and process data from NIST JSON files and store them in MongoDB.

- **Methodology or Approach:**
  - Designed a data ingestion pipeline:
    - Loaded JSON files from the ‘data‘ directory using Apache Spark.
    - Filtered CVE data for 2023 and 2024 using Spark transformations.
    - Stored the filtered data in MongoDB using the MongoDB Spark Connector.
  - Used Spark SQL to query and analyze CVE data based on impact scores and severity.
  - Debugged and tested the pipeline iteratively to ensure compatibility and correctness.

Challenges and Learning

- **Challenges Faced and Their Resolutions:**
  - **Dependency Issues in ‘build.sbt‘:**
    - Encountered unresolved dependencies with the MongoDB Spark Connector.
    - Resolved by ensuring compatibility between Scala 2.13, Spark 3.5.0, and MongoDB libraries.
  - **Managing ‘.env‘ Files:**
    - Replaced ‘dotenv-scala‘ with a manual environment variable loader to handle configuration.
  - **Processing Large JSON Files:**
    - Optimized Spark transformations to handle large files efficiently.
- **Learning Gained:**
  - Gained hands-on experience with Scala, Spark, and MongoDB integration.
  - Learned how to debug dependency and compatibility issues in ‘build.sbt‘.
  - Improved understanding of data pipelines and how to process large datasets using Spark.
- **Future Improvements:**
  - Add real-time monitoring for the data pipeline using Spark Streaming.
  - Optimize MongoDB queries for faster analysis.
  - Extend support for additional data formats beyond JSON.