Spring DATA Neo4j Module

Spring DATA module is suitable to implement DAO layer for all kinds of Databases. It improves the application development process by avoiding some unwanted artifacts DAOImplementations.

- Relational Databases Oracle, MySQL etc.
- No SQL Databases MongoDB, Redis, Apache Cassandra etc.
- No SQL Graph Databases Neo4j etc.

In this Chapter, we are going to discuss about Spring DATA Neo4j Module Architecture. This Architecture is almost similar to for all types of databases mentioned above. It differs only in few component names

Spring DATA Neo4j Architecture

Before starting Spring DATA Neo4j Development, first we will understand this module architecture.

The following diagram shows the Architecture of Spring DATA Neo4j Module

Components of Sprint DATA Neo4J are briefly mentioned below -

Spring DATA Neo4j Repositories

It has provided different API to supports different scenarios

- GraphRepository
- GraphTemplate
- CrudRepository
- PaginationAndSortingRepository

These are Java Classes. Each one has a specific purposes to perform Neo4j DB Operations
<table>
<thead>
<tr>
<th>S.No.</th>
<th>Spring DATA Neo4j Class</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>GraphRepository</td>
<td>It is used to perform Basic Neo4j DB operations.</td>
</tr>
<tr>
<td>2.</td>
<td>GraphTemplate</td>
<td>Like other modules, it is Spring Template to perform Neo4j DB operations.</td>
</tr>
<tr>
<td>3.</td>
<td>CrudRepository</td>
<td>It is used to perform Neo4j CRUD operations with Cypher Query Language CQL.</td>
</tr>
<tr>
<td>4.</td>
<td>PaginationAndSortingRepository</td>
<td>It is used to perform Paging and Sorting of Neo4j CQL Query results.</td>
</tr>
</tbody>
</table>

**Neo4j Drivers**

When we use Spring DATA Neo4j Module API to perform some Database Operations, Spring DATA Neo4j Jars internally uses these drivers Jars to interact and perform Neo4j DB Operations.

**Neo4j Graph Database**

It is actual Neo4j DB server where our Graph data is stored.

We will discuss a simple example by using GraphRepository API in the coming chapter.