

Date : 15/04/2021

Spring Boot 9AM

Mr. RAGHU

Spring Boot Batch API

Batch API : Processing Large set of data and transfer
from source to destination.

Source/Destination: Database, FileSystem.

Ex#1

Job#1 MySQL Database to CSV(Excel) File

Job#2 CSV to MongoDB Import

a. JOB : Complete Batch Processing Task is called as Job
b. STEP : Every JOB contains at least one STEP and at most
n-STEPS. STEP is a PART of JOB.

c. Reader/Item Reader : It will read data from source.

d. Processor/Item Processor : It will process (calculate,
logic checking..etc) data given by Reader.

e. Writer/Item Writer : Write data to destination,
data given by processor.

f. JobLauncher : It will start executing of Job.

g. Job Repository : It is a memory that holds executed
Job Details (status, start and end time,...etc)

*) 1 Step = 1 Reader + 1 Processor + 1 Writer

Q) What is the difference b/w JDBC Batch API and Spring Boot batch
API?

A)

JDBC Batch API: execute multiple Insert/Update SQLs
at a time using one network call.

Ex: 5 Students insert using one n/w call.

Spring Boot batch API: Transfer data/process data

Read from Source-> Process->Write to Destination

Ex: Data Export , Database Migration.

*) Reader and Processor executed once per record/item.

*) Writer executed based on chunk (size) given by programmer
ex chunk=10, then for every 10 items one time
writer is executed.

*) Reader GenericType (ex:String) and Processor Input Type
(ex:String) must match.

*) Processor Output Type (ex: Employee) and Writer Generic
Type (ex: Employee) must match.

*) To create one step we need

-> Step Name

-> Reader object

-> Processor Object

-> Writer Object

```
pom.xml
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-batch</artifactId>
</dependency>
```

--API Details-----

```
org.sf.batch
+ ItemReader<T>
+ read():T
```

```
org.sf.batch
+ ItemProcessor<I, O>
+ process(I item):O
```

```
org.sf.batch
+ ItemWriter<T>
+ write(List<T>):void
```

*) All are interface given by Spring Boot batch API
inside package 'org.springframework.batch.item'.

*) Spring Batch API has provided Impl classes for
Reader and Writer interfaces.

Ex:

```
FlatFileItemReader<T>
JdbcCursorItemReader<T>
JpaItemWriter<T>
MongoItemWriter<T>
...etc
```

*) Process logic depends on concept/data that
you are working. So, we need to define this.