Master spring batch course

Reader,writer, at the enterprise level

Batch processing – processing an amount of data without interaction

A picture containing text, screenshot, diagram, design

Description automatically generated

Batch processing

* Process large amounts of data
* Automation
* Robust – handle invalid data
* Reliability – keep track of what went wrong and where
* Performance –

A computer screen shot of a computer

Description automatically generated with low confidence

Why batch

* U don’t have always all the required information required immediately
* Delayed process –
* Better use of computer resources

Why spring batch

* Dependency injection
* Out of box components
* Allows for declarative skipping and retry

Spring batch – lightweight and easy to learn

Hello World batch job

Batch domain language

1. Task based step vs chunk based step
2. Chunk based step has 3 main parts 🡪 ItemReader , ItemProcessor, ItemWriter
3. Task based step used for things like doing initiatilization, copy a file, sending notification and so on
4. Chunk based step 🡪 read database, process it, write to a file

Spring batch architecture

ItemReader

1

Step

Job Launcher

Job

1 \* 1

ItemProcessor

1

ItemWriter

1

Job Repository

Job Repository – main component responsible for maintaining the state of the job

* Implemented by a relational database
* Shared by all main components
* Hellowolrd job – job repository was implemented by h2 database

Job Launcher – responsible for execution of job

* Validates if a job should be run
* Restarting etc

Within a job there are multiple steps

After each step is executed, job repo is updated – eg how many items being read, processed and written

Job has 1 to many steps

Each step will perform a task

Each step has only 1 item reader , 1 item processor and 1 item writer

A job needs to be launched with a job launcher, and metadata about the currently running process needs to be stored in the job repo

Job – contains job name, and definition and ordering of step instances

Job Instance - refers to the concept of a logic job run

Job Execution

Job has many job instances. 1 job instance has many job executions

How is one job instance distinguished from another – the answer is job parameters

Eg. Scheduled\_date=2020-03-01

Scheduled\_date=2020-03-02

Job instance 1 – lets say failed

Job instance 1 – 2nd time it ran , it passed. So this job instance has 2 job executions

Job Execution –

-single attempt to run a job

-an execution may end in failure or success

-job instance corresponding to a given execution is not considered to be complete unless the execution completes successfully

JobExecutionContext is a map that stores what happened during a run

Step is a domain object that encapsulates an independent phase of a batch job

Job has many steps

1 step has many step executions

1 job execution can have many step executions (say one step fails)

StepExecution – a single attempt to execute a step

Step execution is created only when step is actually started

ExecutionContext – key and value pair (map)

Allows u to pass values between steps

A jobexecution context is available to all steps – so use this to pass data from step to step

Chunk based step

Itemreader – reads ONE ITEM AT A TIME – returns null when nothing to read

Itemprocessor

Itemwriter – write ONE BATCH OR CHUNK of items at a time