

21 Top Spring Batch Interview Questions (2023)

Description

Explain Spring Batch framework?

Spring batch is one of the core module of Spring framework used for developing robust batch applications. Batch processing is a technique which processes data in large group or volume with minimal interaction, instead of a single data element.

Spring Batch is a lightweight, comprehensive batch framework. Spring Batch builds upon the productivity, POJO-based development approach, and general ease of use capabilities people have come to know from the Spring Framework, while making it easy for developers to access and leverage more advanced enterprise services when necessary.

What is the use of Spring Batch? or When to use Spring Batch?

Batch processing is a technique which processes data in large group or volume with minimal human interaction, instead of a single data element.

Spring Batch can be used for use cases like reading and writing to files, transforming data from one form to other form, reading from or writing to databases, creating reports using source data, import data from one format and export to other format from one database to other etc. When it comes to processing millions or even more records concurrently, we can achieve it using TaskExecutor in Spring Batch.

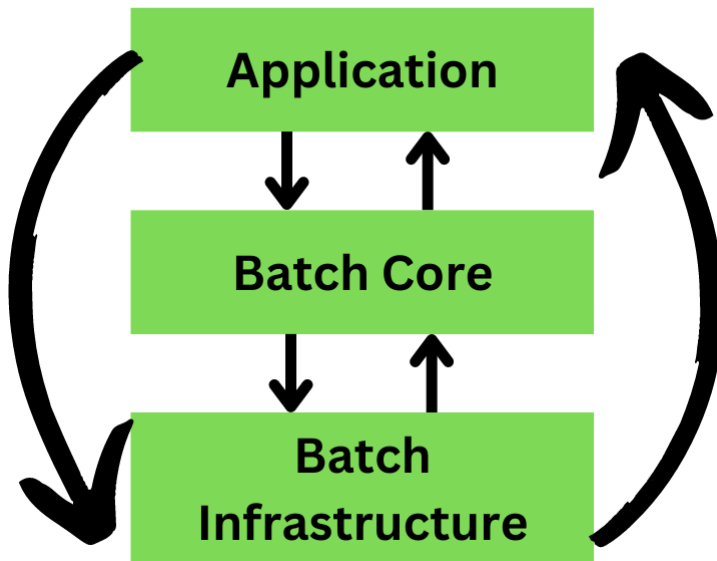
If batch processing didn't exist, then for database write operation we would need to insert each row by writing insert statement. For this kind of scenario we would use batch processing which would save the time by processing chunks of data.

Spring Batch provides reusable functions that are essential in processing large volumes of records, including logging/tracing, transaction management, job processing statistics, job restart, skip, and resource management. It also provides more advanced technical services and features that will enable extremely high-volume and high performance batch jobs through optimization and partitioning techniques. Simple as well as complex, high-volume batch jobs can leverage the framework in a highly scalable manner to process significant volumes of information.

Explain the Spring Batch Architecture framework

Lets look at Spring Batch framework architecture and understand each component of the architecture thoroughly.

Architecture of Spring Batch



Application

This component contains all the jobs and code we write using the Spring Batch framework

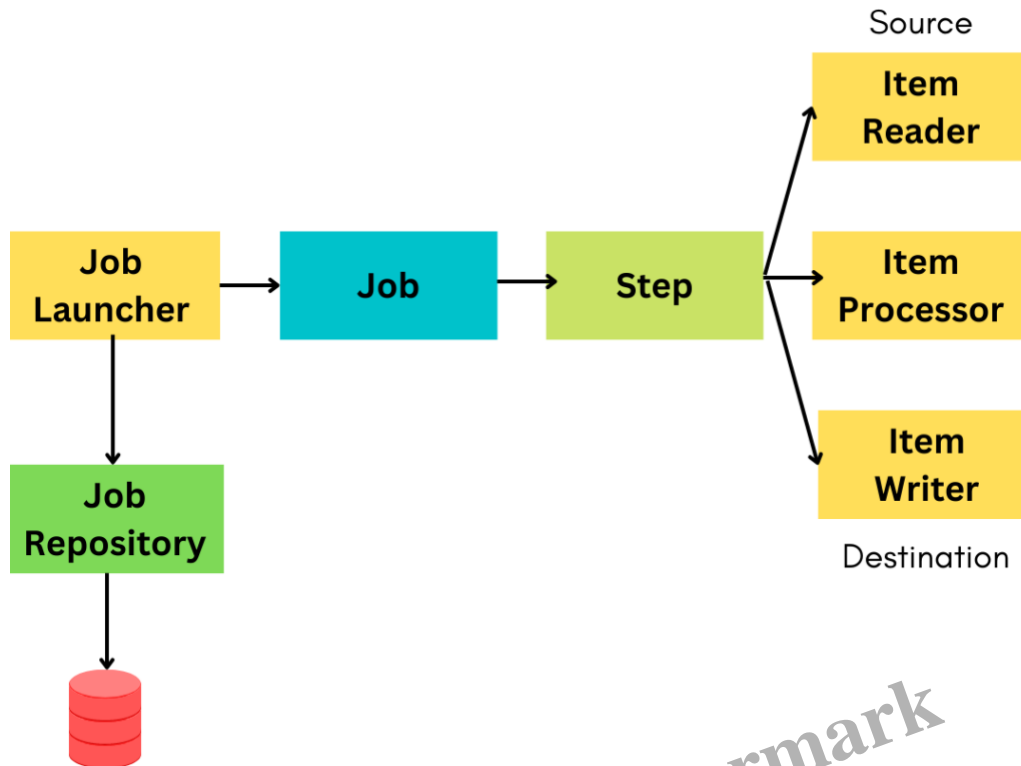
Batch Core

This component contains all the API classes that are needed to control and launch a Batch Job.

Batch Infrastructure

This component contains the readers, writers and services used by both application and Batch core components.

Components of Spring Batch



Job Launcher

Job Launcher is the first component of the spring batch architecture which is responsible for launching the Spring Job. You can also say that this is the entry point to initiate any job in spring batch.

Job

A Job can be defined as a work to be executed by Spring batch.

Step

Step is an independent part within the Job. A Job can contain one or more Steps. Each Step has three components i.e. ItemReader, ItemProcessor and ItemWriter.

ItemReader

ItemReader is used to read the data from source for making it available for ItemProcessor.

ItemProcessor

ItemProcessor is used for processing the data that is read from source by ItemReader. ItemProcessor contains the actual business logic for processing the data.

ItemWriter

ItemWriter is used to write the data to destination after processing is done by ItemProcessor.

Job Repository

Job Repository in Spring Batch provides Create, Retrieve, Update and Delete operations for JobLauncher, Job and Step implementation.

How Spring Batch works

First key component is the Job Launcher which is used to launch spring batch jobs. You can also say this is an entry point to initiate any job in spring batch. It has a run method which will trigger the Job component. Once Job Launcher calls the run method, immediately it will create another component i.e. job. A job can be defined as a work to be executed by the spring batch, this work might involve a simple or a complex task. Once Job Launcher launches a job, immediately it will call another component that is Job repository. This job repository helps to maintain state of job whether it is success or failed. Suppose a spring batch job is running and error occurs, how does spring batch know that error has occurred, and the jobs needs to be rerun, so we need to save the state of the job. State management is an important aspect when running huge volume of data, this is achieved using job repository. Job component will talk to another ocmponent i.e. Step. Step is nothing but combination of 3 components ItemReader, ItemProcessor, ItemWriter, where Itemreader will read the data from source, Itemprocessor will process the data, Itemwriter will help you to write the data to the destination. A Job can have multiple steps, and each step will have Itemreader, Itemprocessor and Itemwriter

What is JobLauncher

JobLauncher is one of the key components in Spring Batch. JobLauncher is an interface which is used to launch spring batch jobs. It can also be called as an entry point to initiate any Job in spring batch. Job can be defined as a work to be executed by spring batch, work can either be a simple or complex task.

What is a Job in Spring Batch

A job in Spring Batch is a process, executed without interruption from start to finish. A Job in Spring Batch is made up of many steps. Each Step is a READ-PROCESS-WRITE task or a single operation task (tasklet).

What is a Step in Spring Batch Job

Step is an independent part of a Job. Each step consists of an ItemReader, ItemProcessor and an ItemWriter. A Job can consist of one or more steps.

What is an ItemReader in Spring Batch

An ItemReader is used to read data into a Spring Batch application from a particular source.

What is an ItemProcessor in Spring Batch

ItemProcessor is used to apply business logic to the data read from the source using ItemReader.

What is an ItemWriter in Spring Batch

ItemWriter writes the data to a particular destination after processing the data using ItemProcessor.

What is a JobRepository in Spring Batch

JobRepository in spring batch is used to maintain the state of the Job. Job Repository is responsible for persistence of batch meta-data entities. "JobRepository" is the mechanism in Spring Batch that makes all this persistence possible. It provides CRUD operations for JobLauncher, Job, and Step.

What is Tasklet in Spring Batch

Spring Batch provides a Tasklet interface, which will be called to perform a single task only, like clean or delete or set up resources before or after any step execution.

How to schedule a Spring Batch Job

Spring Batch Jobs can be executed periodically on a fixed schedule using CRON expressions that are passed to Spring TaskScheduler. CRON expressions in scheduling are used to represent the details of the schedule.

Spring Batch Jobs can be configured in two steps:

- Using the `@EnableScheduling` annotation above the main class which is already annotated with `@SpringBootApplication`
 - Creating a method annotated with `@Scheduled` and providing recurrence details with the job. Then add the job execution logic inside this method.
-

What is the difference between Tasklet and chunk in Spring Batch

In Spring batch, Tasklet is an interface, which is called to perform a single task only, like clean or set up resources before or after any step execution. Example can be to clean up the resource (folders) after a batch job is completed.

Spring Batch uses a ‘chunk-oriented’ processing style in its most common implementation. Chunk oriented processing refers to reading the data one at a time and creating ‘chunks’ that are written out within a transaction boundary. Once the number of items read equals the commit interval, the entire chunk is written out by the `ItemWriter`, and then the transaction is committed.

Does Spring Batch run in parallel

At a high level, there are two modes of parallel processing:

- Single-process, multi-threaded
- Multi-process

These break down into categories as well, as follows:

- Multi-threaded Step (single-process)
 - Parallel Steps (single-process)
 - Remote Chunking of Step (multi-process)
 - Partitioning a Step (single or multi-process)
-

What is default chunk size in Spring Batch

The default batch chunk size is 1.

What is ExecutionContext in Spring Batch

An `ExecutionContext` is a set of key-value pairs containing information that is scoped to either `StepExecution` or `JobExecution`. Spring Batch persists the `ExecutionContext`, which helps in cases where you want to restart a batch run (e.g., when a fatal error has occurred, etc.). All that is needed is to put any object to be shared between steps into the context and the framework will take care of the rest. After restart, the values from the prior `ExecutionContext` are restored from the database and applied.

What is Step scope in Spring Batch

LifeSpan of step scoped beans is tied to the lifecycle of a step i.e beans are created and destroyed at the beginning and at the end of a step respectively. The annotation used to declare a step scoped

bean is @StepScope.

A spring batch StepScope object is one which is unique to a specific step and not a singleton. As the default bean scope in Spring is a singleton. By specifying a spring batch component being StepScope means that Spring Batch will use the spring container to instantiate a new instance of that component for each step execution.

This is often useful for doing parameter late binding where a parameter may be specified either at the StepScope or the JobExecutionContext level and needs to be substituted for a placeholder, much like your example with the filename requirement.

Another useful reason to use StepScope is when you decide to reuse the same component in parallel steps. If the component manages any internal state, its important that it be StepScope based so that one thread does not impair the state managed by another thread (e.g, each thread of a given step has its own instance of the StepScope component).

How do I know if Spring Batch is running

Spring-batch-Admin UI is used to view the status of the jobs (failed/running/completed, etc). With proper set-up of Spring Batch Admin UI, you can even view the status of the several tasks inside different jobs.

Is Spring Batch single threaded

Multiple jobs can be run simultaneously. There are two main types of Spring Batch Parallel Processing, Single Process, Multi-threaded, or Multi-process. These are also divided into subcategories, as follows:

1. Multi-threaded Step (Step with many threads, single process)
 2. Parallel Steps (Steps in the Same Direction, single process)
 3. Remote Chunking of Step (Step Chunking from afar, multi-process)
 4. Partitioning a Step (Creating a Step Partition, single or multi-process)
-

Is Spring Batch a scheduler

Spring Batch is not a scheduling framework. Quartz, Tivoli, Control-M are some of the good enterprise schedulers available in both the commercial and open source spaces. Spring Batch is intended to work in conjunction with a scheduler rather than replace a scheduler.

Category

1. Interview

Date Created

December 1, 2022

Author

kk-ravi144gmail-com
