

Examen Semana 4

Emmanuel Reyes Hernandez



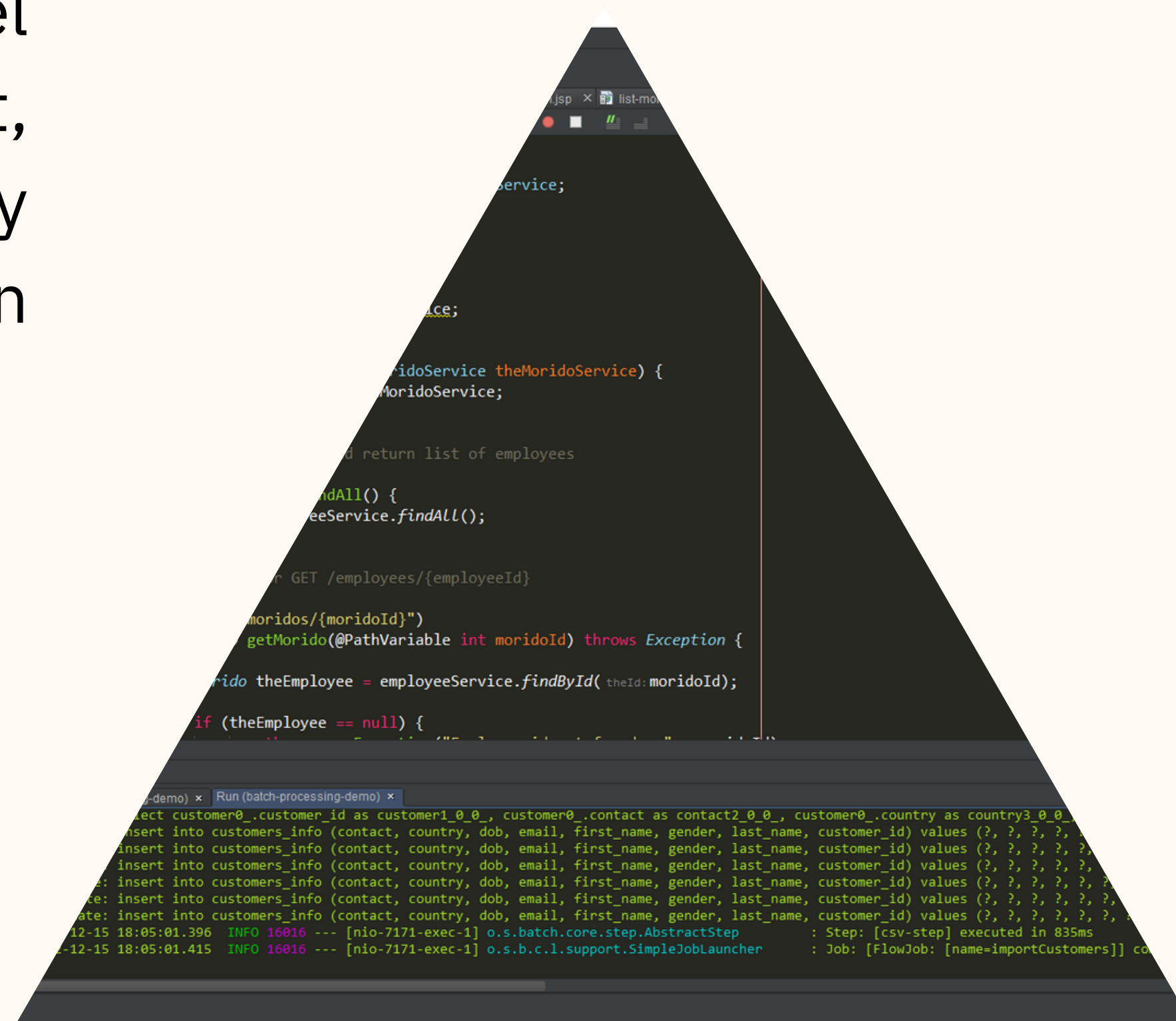
Xideral

Introducción



En esta semana practicamos el funcionamiento de springboot, exponiendo un servicio rest y consumiéndolo por medio de un servidor tomcat.

Vimos el funcionamiento de batch.

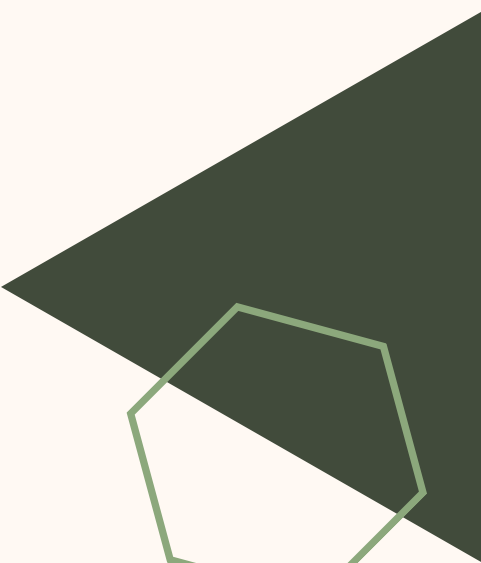




Git

Git funciona como un almacenamiento en la nube con la función de llevar a cabo un versionamiento y control por ramas, esto nos ayudara a construir un proyecto en colaboración y evitar conflictos que puedan surgir en el camino.

Nos da muchas ayudas para guardar y respaldar nuestro proyecto como el que algún integrante corrompa su proyecto pueda bajar el proyecto en una versión anterior y resolver el problema que haya surgido.



git checkout (Nombre de la rama): te cambia a la rama que especifiques

git checkout -b (Nombre de la rama): Crea una nueva rama y te cambia a esa rama

git branch: Muestra todas las ramas que existen y en la que estas posicionado

git stash: Sirve para ocultar cambios en la rama en que te encuentres

Branches (Ramas)

En git existen las ramas (branches) que son una representacion de la rama main en la que solo los commits se veran afectados en esa rama y no estaran en la rama main hasta que se haga un pull request validando que los cambios realizados no afecten al proyecto

Merge

Un merge es la manera en la que se une los cambios que hay en el repositorio con los que tienes localmente y si se encuentra algún conflicto y estos se deberan resolver de manera manual

git merge: Nos permite hacer un commit con los cambios de la rama main con los cambios locales

git diff: Muestra cuales son los cambios que hay y cuales son los que entran en conflicto

git log: Muestra un historico de los cambios locales

git status: Muestra los archivos que se han modificado y falta por agregar en el commit

Pull Request

Son las peticiones que se realizan para integrar los cambios a la rama principal y esta deberán ser verificadas por los que administren el repositorio

Fork

Es una copia directa de un repositorio y todos los cambios que se hagan sera necesario actualizar el fork para realizar un pull.

Rebase

Un git rebase integra la rama main con la rama en la que te encuentras y la integra en una sola, con la diferencia de que este no almacena el historial de los commits hechos anteriormente. Desafortunadamente este comando se esta considerado como una mala practica ya que se perderá el histórico.

Clean

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt.

Stash

Nos permite ocultar nuestros cambios o quitarlos temporalmente de la pila de cambios que se tenga en ese momento

Cherry pick

Es una comando mediante la cual, podemos traer commits específicos desde otra rama, pero sin la necesidad de unir sus cambios o fusionar ambas ramas en 1 sola



Api Rest

Rest se creó como una arquitectura a principios del 2000, y utiliza al protocolo HTTP para generar la transferencia de datos. Este es un protocolo que permite la transferencia de datos. Éstos pueden viajar bajo el formato HTML, JSON, XML, etc.

Verbos de HTTP

POST: Sirve para guardar nueva información o registros.

GET: Solicitar recursos que ya están guardados, la consulta a estos resultados se puede hacer de manera general o de manera específica

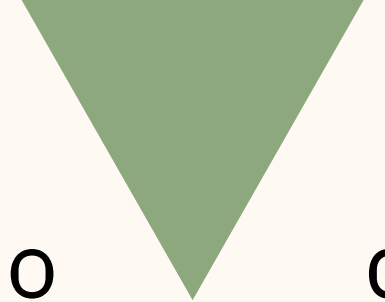
PUT: Se actualizar o modificar un recurso existente, para actualizar un registro se establecerá una ruta al registro específico respetando una nomenclatura parecida

DELETE: Sirve para eliminar un registro existente, para eliminar un registro

servicio Rest


Una API REST es una forma de permitir que diferentes programas de ordenador se comuniquen entre sí a través de Internet. Ya que la comunicación debe darse a través de protocolos y estándares para enviar y recibir datos, estas APIs están diseñadas bajo los principios de REST (que significa Representational State Transfer).





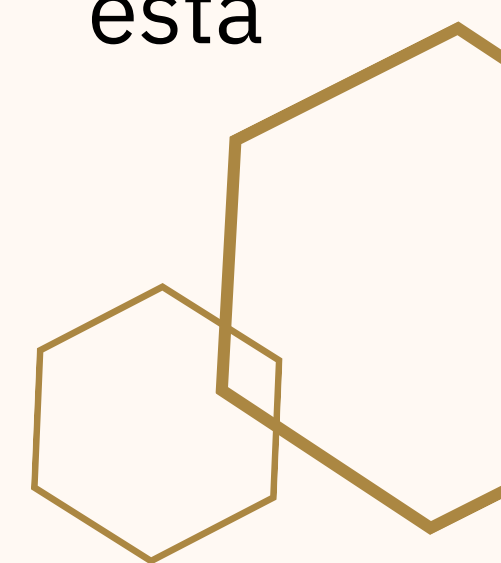
Vimos como levantar un servicio Rest el cual sera consumido por otro servicio el cual solo consume los datos en formato Json y este a su vez los muestra de manera grafica en una tabla la cual se encuentra en una base de datos.

Para este proyecto fue necesario crear una base de datos, por parte del back end se realiza la conexion y se establecen las reglas de negocio



donde se definira como es que se manejara un POST, GET, DELETE al hacer la peticion con ciertos parametros, el como manejaran los datos y si es que estos se modificaran.

La conexion se logra por medio de Hibernate o jdbc que son las dependencias que logran esta conexion, el proyecto esta estructurado con MVC



Estructura

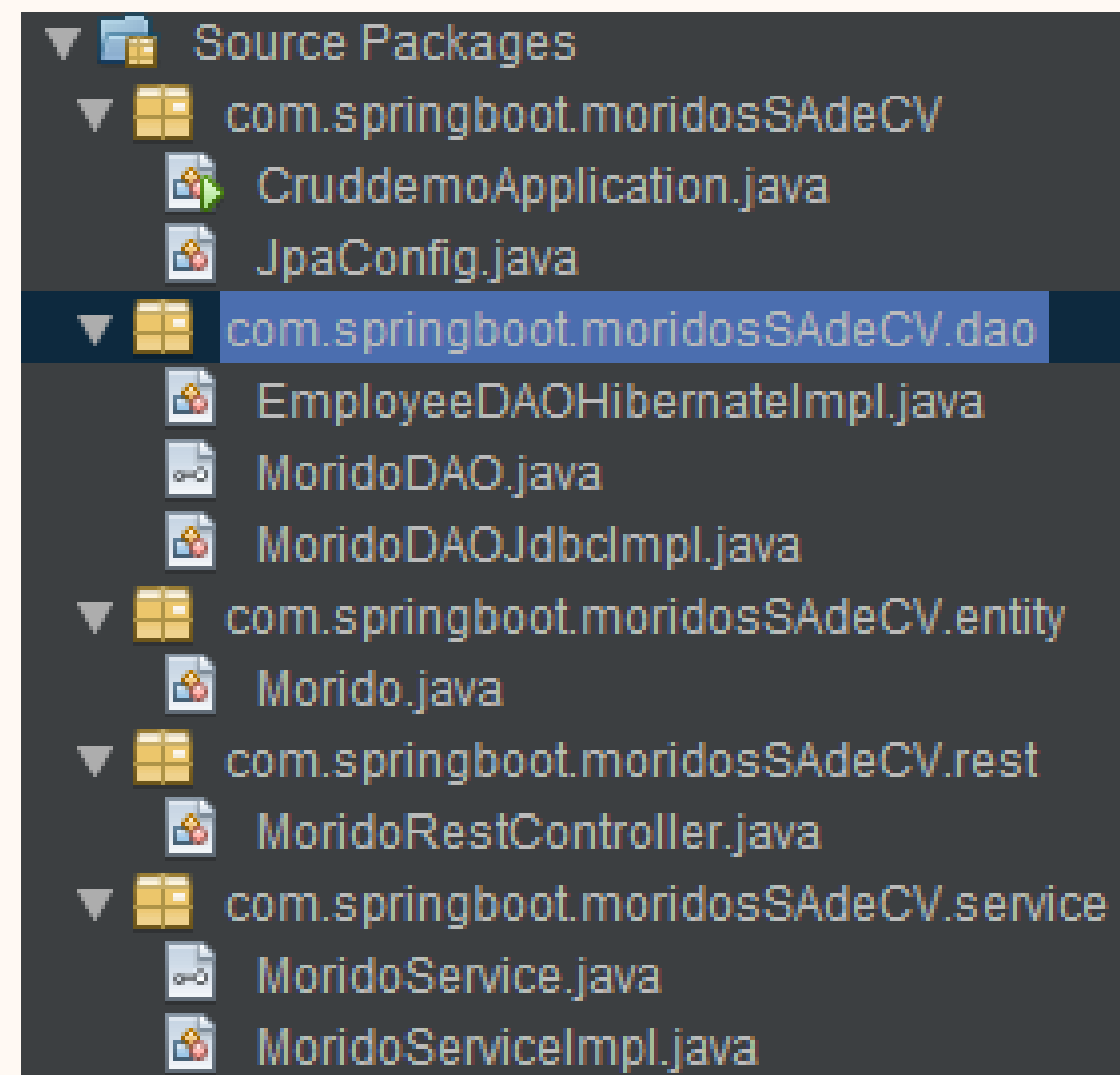
El proyecto esta estructurado de la siguiente manera:

La entity: Es entidad que se crea por cada registro y se almacena en ella.

Service: Aqui solo es el medio por el cual llevan los datos solamente el cual apunta y refiere a DAO.

Dao: Se encarga de definir la logica en la que se presentan y se almacenan los datos.

Controller: Aqui se reciben los datos de las vistas y este a su vez contiene la regla de negocia que nos dice que hacer con esos datos.





Estructura del crud

El crud se estructura de la siguiente manera:

Config: Aqui se configura el view y se mapea las solicitudes por componentes.

Service: Se encargar de recibir los datos que son mandados.

Model: Aqui se instancian los objetos por medio de los datos recibido

Controller: Se realiza la logica de negocio y manda las solicitudes .



Batch

Batch es un procesamiento por lotes el cual se realiza trabaja mediante lotes de trabajos, a menudo simultáneamente, en orden secuencial y sin parar.

También es un comando que garantiza que los trabajos grandes se calculen en partes pequeñas, para mejorar la eficiencia durante el proceso de depuración.

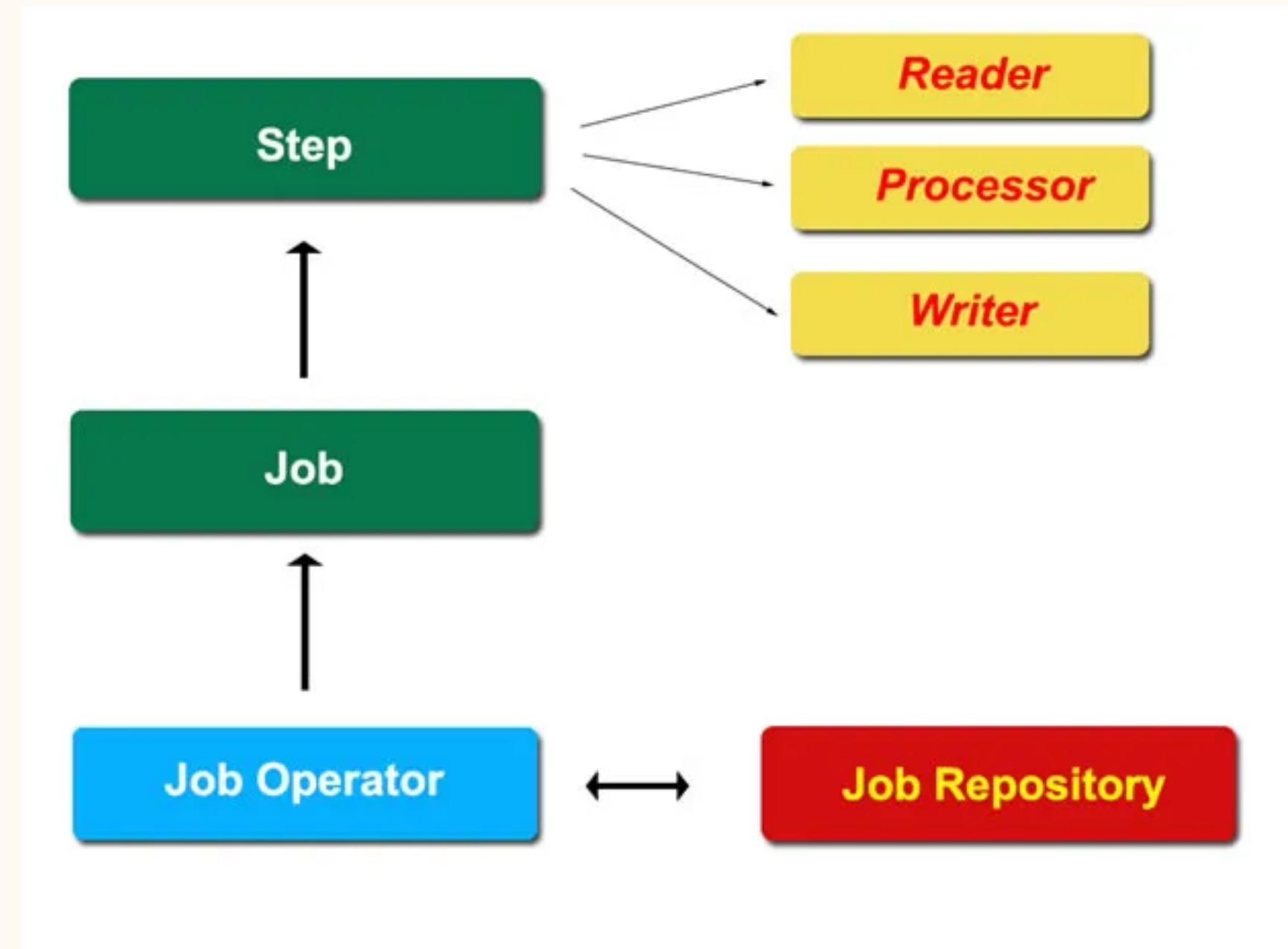


En un proceso batch puede contar con uno o varios step y este a su vez puede contar con uno o mas jobs, la finalidad de realizar una tarea masiva por pequeños lotes es que pueda realizar operaciones de manera rapida y eficaz, en un job puede contener una entrada, procesamiento y una salida.

En la entrada puede ser por un documento, json, Sql, etc.

El procesamiento es la logica que se le daran a estos datos.

La salida es como queremos almacenar estos datos.





Gracias por Todo lo aprendido Miguel.