

Repository Example

Outline

Refer to the following example for how to configure data processing (DB, File, etc.) when implementing the batch. See the example EgovRepositoryFunctionalTests for how to implement jdbcCursorIoJob and delimitedIoJob, where the resources comprise DBs and Files, respectively.

Description

Settings

Configuring Jobs

Check repositoryJob.xml, the Job configuration file for the Repository Example.

A Job is comprised of a pair of sub-jobs that uses either reader or writer depending on the type.

- jdbcCursorIoJob : A Job where resources comprise DBs
- delimitedIoJob : A Job where resources comprise Files

```
<!-- Register Jobs where Resources comprise DBs, in XML -->
<job id="jdbcCursorIoJob" xmlns="http://www.springframework.org/schema/batch">
```

```
  <step id="jdbcCursorIoStep1">
    <tasklet>
      <chunk reader="itemReaderDB" processor="itemProcessor" writer="itemWriterDB"
commit-interval="2" />
    </tasklet>
  </step>
</job>
```

```
<!-- Register Files where Resources comprise Files, in XML or JobParameter -->
```

```
<job id="delimitedIoJob" xmlns="http://www.springframework.org/schema/batch">
  <step id="delimitedIostep1">
    <tasklet>
      <chunk reader="itemReaderFile" processor="itemProcessor" writer="itemWriterFile"
commit-interval="2"/>
    </tasklet>
  </step>
</job>
```

Composition and Implementation of JunitTest

Composition of JunitTest

Junit Test is performed based upon the configuration **RepositoryJob** and the related classes. Batch is performed as per the configuration.

- ✓ See [JUnit Test](#) for runtime environment of batches for more information.
- ✓ assertEquals(BatchStatus.COMPLETED, jobExecution.getStatus()) : Make sure you check the batch execution result is COMPLETED.
- ✓ jobLauncher.run(jobRegistry.getJob("jdbcCursorIoJob"),
getUniqueJobParameters("jdbcCursorIoJob")) : Executes the job entitled jdbcCursorIoJob
- ✓ getUniqueJobParameters hands off the input and output resources required for JobParameter batches

```
@ContextConfiguration(locations = {  
    "/egovframework/batch/simple-job-launcher-context.xml",  
    "/egovframework/batch/jobs/repositoryJob.xml" })  
public class EgovRepositoryFunctionalTests {  
  
    @Test  
    public void testUpdateCredit() throws Exception {  
        // 1. 1. Implementation of Jobs where resources comprise DBs  
        JobExecution jobExecution =  
jobLauncher.run(jobRegistry.getJob("jdbcCursorIoJob"),getUniqueJobParameters("jdbcCurs  
orIoJob"));  
        assertEquals(BatchStatus.COMPLETED, jobExecution.getStatus());  
  
        // 2. 2. Implementation of Jobs where resources comprise Files  
        jobExecution =  
jobLauncher.run(jobRegistry.getJob("delimitedIoJob"),getUniqueJobParameters("delimitedIo  
Job"));  
        assertEquals(BatchStatus.COMPLETED, jobExecution.getStatus());  
    }  
  
    /**  
     * Method to configure JobParameter  
     */  
    protected JobParameters getUniqueJobParameters(String jobName) {  
  
        JobParametersBuilder builder = new JobParametersBuilder();  
        // Job Parameters Registrable  
        builder.addString("inputFile",  
"/egovframework/data/input/delimited.csv");  
        builder.addParameter("timestamp", new JobParameter(new  
Date().getTime()));  
  
        JobParameters jobParameters = builder.toJobParameters();  
        return jobParameters;  
    }  
}
```

Implementation of JunitTest

See [Implementation of JunitTest](#) for more information.

Verify Result

- You can check out the data processing result where the Repository Type is DB, in the Table CUSTOMER.

Status Result1				
	ID	VERSION	NAME	CREDIT
1	1	0	customer1	100005
2	2	0	customer2	100005
3	3	0	customer3	100005
4	4	0	customer4	100005

- Find delimitedOutput.csv for data processing result where the Repository Type is File.

	A	B	C	D	E
1	customer1	15			
2	customer2	25			
3	customer3	35			
4	customer4	45			
5	customer5	55			
6	customer6	65			
7					

References

- [ItemReader](#)
- [ItemWriter](#)