WAY2LEARN

LAB- Spring Batch – Skip

In this lab you will be understand how to skippable-exception-classes and listener

In this lab, you will be working on **01-Batch-Skip-start project**
Make sure , you have created batchdb in mysql database

1) In src/main/resource/Files/products.txt – open the file , check there is ExtraFieldForException in few rows

2) Go to src/main/resources and Open **batch-infrastructure-config.xml**

3) Job repository and JobLauncher configuration is already done

4) Open batch-config1_WithoutSkipConfig.xml file

5) Notice **Job** is defined with a step

6) Step has a tasklet/chunk configuration with reference to **reader** and **writer , commit-interval of 3**

7) ItemReader : Notice Configuration of **FlatFileItemReader** already available in XML file
It reads from **FlatFile**. We don't need to do anything here.

8) ItemWriter : Notice Configuration of **JdbcBatchItemWriter** already available in XML file
It writes to DB. We don't need to do anything here

9) Go to src/test/java and Run **BatchTest.java**

10) It fails with Exception : org.springframework.batch.item.file.FlatFileParseException

Congratulations!! You have completed the Lab – part 1 successfully!!

1) Open batch-config2.xml file

2) JOB,STEP,ItemReader and ItemWriter configuration are same

3) It reads from same file

4) TODO 1 : Go To Job Configuration and within chunk configuration of "readWrite" step , specify FlatFileParseException in <batch:skippable-exception-classes>

5) TODO 2 : Also specify skip-limit attribute for <batch:chunk> , value should be greater than 3 for this usecase to run successfully

6) Go to src/test/java and Run **BatchTest2.java**

7) It runs successfully without any exception

Congratulations!! You have completed the Lab – part 2 successfully!!

1) Open batch-config3.xml file

2) JOB,STEP,ItemReader and ItemWriter configuration is same

3) It reads from same file

4) Notice Job Configuration

5) The chunk configuration of "readWrite" step has <batch:skippable-exception-classes> already configured

6) TODO 1 : Decrease skip-limit attribute for <batch:chunk> , value should be less than 3 although usecase requires more than 3

7) Notice XML has AlwaysSkipItemSkipPolicy configured
<bean id="alwaysSkipItemSkipPolicy"
class="org.springframework.batch.core.step.skip.AlwaysSkipItemSkipPolicy"/>

8) TODO 2: In batch:chunk , specify an attribute skip-policy that refers to alwaysSkipItemSkipPolicy

9) Go to src/test/java and Run **BatchTest3.java**

10) It runs successfully without any exception **due to AlwaysSkipItemSkipPolicy although skip-limit is < 3**

Congratulations!! You have completed the Lab – part 3 successfully!!

1) Open batch-config4.xml file

2) JOB,STEP,ItemReader and ItemWriter configuration is same

3) It reads from same file

4) Notice Job Configuration

5) The chunk configuration of "readWrite" step has <batch:skippable-exception-classes> already configured

6) Skip-limit is 4

7) Notice XML has skipListener configured
<bean id="skipListener" class="com.way2learnonline.batch.support.DatabaseSkipListener"/>

8) Open this listener :
TODO 1: annotate log method with @OnSkipInRead
Due to this annotation , method will be called when exception occurs while reading

9) Notice , this method inserts line and line_number to skipped_product table using exception

10) TODO 2 In XML file , within <batch:tasklet> of "readWrite" step , specify <batch:listeners> tag and have reference to skipListener

11) Go to src/test/java and Run **BatchTest4.java**

12) It runs successfully without any exception

13) **Go to database and check** skipped_product table , it shows line and line_number that are skipped

14) **Congratulations!! You have completed the Lab successfully!!**