

Simple Processing (Tasklet) Example

Outline

This Section describes how simple processing works without use of interfaces such as `ItemReader` and `ItemWriter` in implementation of a job.

Description

Settings

Configuring Jobs

Check out `taskletJob.xml`, the job configuration file that you have for the simple processing (Tasklet) example.

What you can find is that the job has no chunk configuration and the bean `Adapter` referred by `<tasklet>` sends the `jobParameters` values to the property `targetObject`, as follows:

```
<job id="loopJob" xmlns="http://www.springframework.org/schema/batch">
    <step id="step1">
        <tasklet ref="adapter">
            <transaction-attributes propagation="REQUIRED"/>
        </tasklet>
    </step>
</job>

<bean id="adapter" class="org.springframework.batch.core.step.tasklet.MethodInvokingTaskletAdapter">
    <property name="targetObject" ref="value" />
    <property name="targetMethod" value="execute" />
</bean>

<bean id="value" class="egovframework.brte.sample.example.test.EgovTaskletJobFunctionalTests$TestBean"
scope="step">
    <property name="value" value="#{jobParameters[value]}" />
</bean>
```

Composition and Implementation of JunitTest

Composition of JunitTest

You can implement Junit Test using the `taskletJob` configuration and associated classes, by which batch is implemented and contents are verifiable.

- ✓ See [Junit Test Description for Batch Execution Environment](#) for structure of the Class Junit Test.
- ✓ `assertEquals(BatchStatus.COMPLETED, jobExecution.getStatus())`: Make sure the batch implementation is COMPLETED.
- ✓ `assertEquals("foo", value)` for Class `TestBean` : You need to determine whether "foo" in `jobParameter` can be found in the string as well (comparison between `jobParameter` and customized value)

```
@ContextConfiguration(locations = { "/egovframework/batch/simple-job-launcher-context.xml",
                                   "/egovframework/batch/jobs/taskletJob.xml",
                                   "/egovframework/batch/job-runner-context.xml" })
```

```
public class EgovTaskletJobFunctionalTests {
```

```
    ...
    @Test
```

```

    public void testLaunchJob() throws Exception {
        JobExecution jobExecution = jobLauncherTestUtils.launchJob(new
JobParametersBuilder().addString("value", "foo")
                                .addParameter( "timestamp", new JobParameter(new
Date().getTime()))).toJobParameters());
        assertEquals(BatchStatus.COMPLETED, jobExecution.getStatus());
    }
    public static class TestBean {
        //String value
        private String value;

        public void setValue(String value) {
            this.value = value;
        }

        public void execute() {
            assertEquals("foo", value);
        }
    }
}

```

Implementation of JunitTest

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

References

- [TaskletStep](#)