http://www.egovframe.go.kr/wiki/doku.php?id=egovframework:rte2:brte:batch_example:event_notice_template_mgmt

Template Example for Event Notification

Outline

Refer to the following example of EgovEventNoticeTriggerFunctionalTests for how the event is notified via SMS, e-mail, etc. using EventNoticeTrigger:

Description

Settings

Configuring Jobs

Check eventNoticeTriggerJob.xml for job configuration of event notification template.

A job has the configuration <tasklet> and <listener> to call EgovEventNoticeCallListener and register the associated classes of EgovEventNoticeCallProcessor and EgovEmailEventNoticeTrigger as beans.

```
<job id="eventNoticeTriggerJob" xmlns="http://www.springframework.org/schema/batch">
	<step id="eventNoticeTriggerStep1">
		<tasklet>
			<chunk reader="itemReader" writer="itemWriter" commit-interval="2" />
		</tasklet>
		<listeners>
			<listener ref="EventNoticeCallListener" />
		</listeners>
	</step>
</job>
<bean id="EventNoticeCallListener"
	class="egovframework.brte.sample.example.listener.EgovEventNoticeCallProcessor" />
<bean id="EmailEventNoticeTrigger"
	class="egovframework.brte.sample.example.event.EgovEmailEventNoticeTrigger" />
```

Configuring Classes

Check out the class EgovEmailEventNoticeTrigger to send an Email.

Refer to EgovEventNoticeTrigger for notification of th erelevant information in implementation of job. A job comprises a total of three compositions as follows:

1. Input e-mail communication information for the Class EgovEmailEventNoticeTrigger.

Receiver: You can input either single or multiple receivers in the form of String Arrays.
Sender: You can input the sender's e-mail address in the form of String Arrays.

```
public class EgovEmailEventNoticeTrigger extends EgovEventNoticeTrigger {

	// E-Mail Receiver Configuration (String Arrays)
	String[] emailList = { "****@*****" };

	// E-Mail Sender Configuration
	String emailFromAddress = "****@*****";

}
```

2. You can use the information loaded from the Parameter invoke(). Refer to the following example for how the contents is comprised of using JobExecution:

```
public void invoke(JobExecution jobExecution) {
	try {

		// Title of E-mail
		String emailSubjectTxt = jobExecution.getJobInstance().getJobName()  + "Execution Report";

		// Contents of E-mail
		String emailMsgTxt = "============ Notice ============"
				+ "\nJobName : " + jobExecution.getJobInstance().getJobName()
				+ "\nExitStatus : " + jobExecution.getExitStatus().getExitCode();

		postMail(emailList, emailSubjectTxt, emailMsgTxt, emailFromAddress);
	}
}
```

3. Input SMTP Server Information to send E-mails out. Refer to the following example for SMTP Server Information configured in postMail() and input the username and password of the concerned SMTP server in SMTPAuthenticator class.

✔ You may change SMTP server otherwise as per your convenience.

username : Input the username of the mail server.
password : Input the password of the mail server.

```
public class EgovEmailEventNoticeTrigger extends EgovEventNoticeTrigger {

	private class SMTPAuthenticator extends javax.mail.Authenticator {

		public PasswordAuthentication getPasswordAuthentication() {

			String username = "****@****";
			String password = "********";
			return new PasswordAuthentication(username, password);
		}
	}
}
```

Check out the Class EgovEventNoticeCallProcessor.

EgovEventNoticeCallProcessor inherited EgovStepPostProcessor among other eGovFramework processors and hands off the parameters of StepExecution related to post-Step jobs and calls the Method EgovEmailEventNoticeTrigger.invoke.

The Class EgovEmailEventNoticeTrigger is called in EgovEventNoticeCallProcessor as a bean for Job configuration (see Job Configuration for more information).

```
public class EgovEventNoticeCallProcessor<T,S> extends EgovStepPostProcessor<T,S> {
	@Autowired
	EgovEmailEventNoticeTrigger egovEmailEventNoticeTrigger;

	public ExitStatus afterStep(StepExecution stepExecution) {

		egovEmailEventNoticeTrigger.invoke(stepExecution);
		return stepExecution.getExitStatus();
	}
}
```

✔ The example assumes that the Job Configuration File configures Listener in Step and inherited the concerned processor to implement the job related to Step. Note, however, that the Annotation related to Listener and Event Notification Trigger Method are available for for expansive use (make sure you register EventNoticeListener in Job configuration as a bean).

```
public class EventNoticeListener {
	@Autowired
	EgovEmailEventNoticeTrigger egovEmailEventNoticeTrigger;


	// Implement the invoke to send e-mails upon completion of the Job
	@AfterJob
	public ExitStatus sendJobNotice(JobExecution jobExecution) {

		// Use e-mail services
		egovEmailEventNoticeTrigger.invoke(jobExecution);
		return jobExecution.getExitStatus();
	}

	 // Implement the invoke to send e-mails upon completion of Step
	@AfterStep
	public ExitStatus sendStepNotice(StepExecution stepExecution) {

		// Use e-mail services
		egovEmailEventNoticeTrigger.invoke(stepExecution);
		return stepExecution.getExitStatus();
	}

	// Implement the invoke to send e-mails when error occurs while reading
	@OnReadError
	public void sendErrorNotice(Exception e) {

		// Use e-mail services
		egovEmailEventNoticeTrigger.invoke(e);
	}
}
```

Composition and Implementation of JunitTest

Composition of JunitTest

Implement the event notification template and verify the result of batch by comprising @Test as follows:

✔ See Junit Test Description for Batch Execution Environment .
✔ assertEquals("COMPLETED", jobExecution.getExitStatus().getExitCode()) : Make sure you check the batch execution result is COMPLETED.

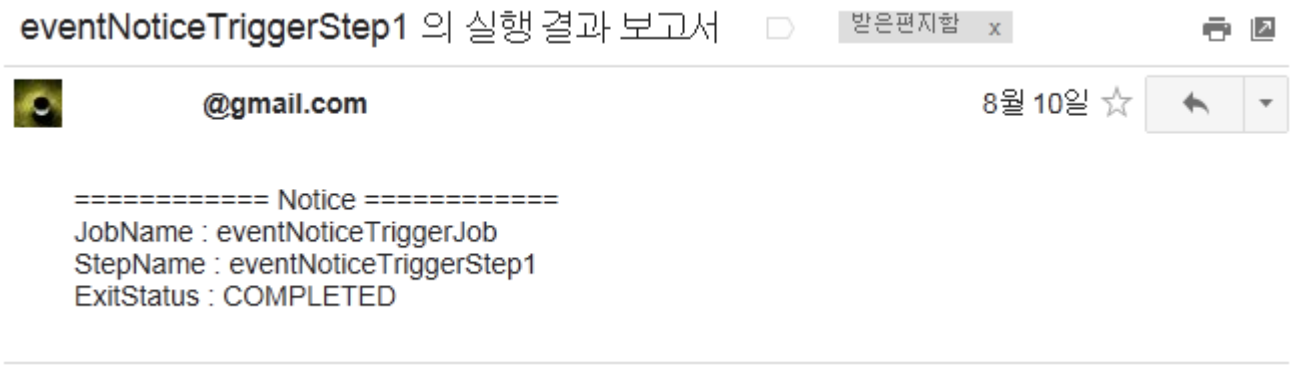
```
@ContextConfiguration(locations = { "/egovframework/batch/jobs/eventNoticeTriggerJob.xml" })
public class EgovEventNoticeTriggerFunctionalTests extends EgovAbstractIoSampleTests {

	@Test
	public void testUpdateCredit() throws Exception {
		JobExecution jobExecution = jobLauncherTestUtils.launchJob(getUniqueJobParameters());

		// The web pages will determine success and failure by reception of mails on the webpage.
		assertEquals("COMPLETED", jobExecution.getExitStatus().getExitCode());
	}
}
```

Implementation of JunitTest

See Implementation of JunitTest for more information.

Verify Result

The test result reflects the result of batch implementation only. The event notification is to be determined by e-mail reception.



References

- Template Management for Event Notification
- SMTP Server
  - naver SMTP Server : smtp.naver.com
  - daum SMTP Server : smtp.hanmail.net
  - Google SMTP Server : smtp.google.com