Inbuilt Event Handling

Create Maven project add spring core, context and beans dependencies in your pom.xml

<!-- Spring Core -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>5.3.22</version>

</dependency>

<!-- Spring context-->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.22</version>

</dependency>

<!-- Spring Beans -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>5.3.22</version>

</dependency>

Create Listeners for start and stop context.

**package** com.allstate.eventhandling.event;

**import** org.springframework.context.ApplicationListener;

**import** org.springframework.context.event.ContextStartedEvent;

**public** **class** StartEventListener **implements**

ApplicationListener<ContextStartedEvent>{

**public** **void** onApplicationEvent(ContextStartedEvent event) {

System.***out***.println("Start Event executed");

}

}

package com.allstate.eventhandling.event;

import org.springframework.context.ApplicationListener;

import org.springframework.context.event.ContextStoppedEvent;

public class StopEventListener implements

ApplicationListener<ContextStoppedEvent>{

public void onApplicationEvent(ContextStoppedEvent event) {

System.out.println("Stop Event Executed");

}

}

Register both the listeners in beans.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"*

*http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd"*>

<bean id=*"startel"* class=*"com.allstate.eventhandling.event.StartEventListener"*></bean>

<bean id=*"stopel"* class=*"com.allstate.eventhandling.event.StopEventListener"*></bean>

</beans>

Now try to trigger this events from App.java

package com.allstate.eventhandling;

import org.springframework.context.ApplicationContext;

import org.springframework.context.ConfigurableApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.allstate.eventhandling.custom.CustomEventpublisher;

import com.allstate.eventhandling.custom.CustomeEvent;

public class App

{

public static void main( String[] args )

{

ConfigurableApplicationContext context= new ClassPathXmlApplicationContext("beans.xml");

context.start();

System.out.println("Completed the task");

context.stop();

}

}

Let’s Implement CustomEvent

Create CustomEvent class

**package** com.allstate.eventhandling.custom;

**import** org.springframework.context.ApplicationEvent;

**public** **class** CustomeEvent **extends** ApplicationEvent{

**public** CustomeEvent(Object source) {

**super**(source);

}

@Override

**public** String toString() {

**return** "This is the message from CustomeEvent";

}

}

Create CustomEventPublisher to publish the event

**package** com.allstate.eventhandling.custom;

**import** org.springframework.context.ApplicationEventPublisher;

**import** org.springframework.context.ApplicationEventPublisherAware;

**public** **class** CustomEventpublisher **implements** ApplicationEventPublisherAware{

**private** ApplicationEventPublisher publisher;

**public** **void** setApplicationEventPublisher(ApplicationEventPublisher publisher) {

**this**.publisher=publisher;

}

**public** **void** publish() {

CustomeEvent ce= **new** CustomeEvent(**this**);

publisher.publishEvent(ce);

}

}

Now to listen this event create Listener.

**package** com.allstate.eventhandling.custom;

**import** org.springframework.context.ApplicationListener;

**public** **class** CustomEventlistener **implements** ApplicationListener<CustomeEvent>{

**public** **void** onApplicationEvent(CustomeEvent event) {

System.***out***.println(event);

}

}

Register both publisher and listener to your beans.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"*

*http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd"*>

<bean id=*"startel"* class=*"com.allstate.eventhandling.event.StartEventListener"*></bean>

<bean id=*"stopel"* class=*"com.allstate.eventhandling.event.StopEventListener"*></bean>

<bean id=*"cel"* class=*"com.allstate.eventhandling.custom.CustomEventlistener"*></bean>

<bean id=*"cep"* class=*"com.allstate.eventhandling.custom.CustomEventpublisher"*></bean>

</beans>

Trigger event from App.java

package com.allstate.eventhandling;

import org.springframework.context.ApplicationContext;

import org.springframework.context.ConfigurableApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.allstate.eventhandling.custom.CustomEventpublisher;

import com.allstate.eventhandling.custom.CustomeEvent;

public class App

{

public static void main( String[] args )

{

ConfigurableApplicationContext context= new ClassPathXmlApplicationContext("beans.xml");

// context.start();

// System.out.println("Completed the task");

// context.stop();

CustomEventpublisher cep=(CustomEventpublisher)context.getBean("cep");

cep.publish();

}

}

Save and check the output