**Create a Java program to demonstrate JDBC and Spring Integration**

**Pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.javarticles.spring.integration.jms</groupId>

<artifactId>springintegrationjms</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework.integration</groupId>

<artifactId>spring-integration-core</artifactId>

<version>4.1.2.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework.integration</groupId>

<artifactId>spring-integration-jdbc</artifactId>

<version>4.1.2.RELEASE</version>

<scope>compile</scope>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>5.1.26</version>

</dependency>

</dependencies>

<properties>

<spring.version>4.1.4.RELEASE</spring.version>

</properties>

</project>

package com.info;

public class Article {

    private int id;

    private String name;

    private String author;

    public int getId() {

        return id;

    }

    public void setId(int id) {

        this.id = id;

    }

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

    public String getAuthor() {

        return author;

    }

    public void setAuthor(String author) {

        this.author = author;

    }

    public String toString() {

        StringBuilder sb = new StringBuilder("id: ");

        sb.append(id).append(", name: ").append(name).append(", tags: ")

                .append(tags).append(", category: ").append(category)

                .append(", author").append(author);

        return sb.toString();

    }

}

package com.info;

import java.sql.ResultSet;

import java.sql.SQLException;

import org.springframework.jdbc.core.RowMapper;

public class ArticleRowMapper implements RowMapper<Article> {

    public Article mapRow(ResultSet rs, int rowNum) throws SQLException {

        String name = rs.getString("name");

        String category = rs.getString("category");

        String author = rs.getString("author");

        String tags = rs.getString("tags");

        int id = rs.getInt("id");

        Article article = new Article();

        article.setId(id);

        article.setAuthor(author);

        article.setName(name);

        return article;

    }

}

CREATE TABLE `articles` (

  `ID` INT(10) UNSIGNED NOT NULL AUTO\_INCREMENT,

  `NAME` VARCHAR(100) NOT NULL,

  `AUTHOR` VARCHAR(50) NOT NULL,

  `SENT` INT,

  PRIMARY KEY (`ID`)

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8;

**JdbcMessageHandler**

package com.javarticles.spring.integration.jdbc;

import java.util.List;

public class JdbcMessageHandler {

public void handleMessage(List<Article> articleList) {

System.out.println("In JdbcMessageHandler:" + articleList);

}

}

**ApplicationContext.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>" xmlns:int="<http://www.springframework.org/schema/integration>"

    xmlns:jdbc="<http://www.springframework.org/schema/jdbc>" xmlns:int-jdbc="<http://www.springframework.org/schema/integration/jdbc>"

    xmlns:context="<http://www.springframework.org/schema/context>" xmlns:p="<http://www.springframework.org/schema/p>"

    xsi:schemaLocation="<http://www.springframework.org/schema/beans>

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

<http://www.springframework.org/schema/integration>

<http://www.springframework.org/schema/integration/spring-integration.xsd>

<http://www.springframework.org/schema/integration/jdbc>

<http://www.springframework.org/schema/integration/jdbc/spring-integration-jdbc.xsd>

<http://www.springframework.org/schema/jdbc>

<http://www.springframework.org/schema/jdbc/spring-jdbc-3.0.xsd>">

    <int-jdbc:inbound-channel-adapter id="dataChannel"

        query="select \* from articles where author='Joe' and sent = 0"

        update="update articles set sent = 1 where id in (:id)"

        data-source="dataSource" row-mapper="articleRowMapper">

        <int:poller fixed-rate="100">

            <int:transactional />

        </int:poller>

    </int-jdbc:inbound-channel-adapter>

    <int:service-activator input-channel="dataChannel" ref="jdbcMessageHandler" />

    <bean id="jdbcMessageHandler"

        class="com.javarticles.spring.integration.jdbc.JdbcMessageHandler" />

    <bean id="transactionManager"

        class="org.springframework.jdbc.datasource.DataSourceTransactionManager">

        <property name="dataSource" ref="dataSource" />

    </bean>

    <int:poller default="true" fixed-rate="100" />

    <int:channel id="dataChannel">

        <int:queue />

    </int:channel>

    <jdbc:initialize-database data-source="dataSource"

        enabled="true">

        <jdbc:script location="classpath:db-schema.sql" />

        <jdbc:script location="classpath:db-test-data.sql" />

    </jdbc:initialize-database>

    <bean id="dataSource"

        class="org.springframework.jdbc.datasource.DriverManagerDataSource">

        <property name="driverClassName" value="com.mysql.jdbc.Driver" />

        <property name="url" value="jdbc:mysql://localhost/test" />

        <property name="username" value="root" />

        <property name="password" value="mnrpass" />

    </bean>

    <bean id="articleRowMapper"

        class="com.javarticles.spring.integration.jdbc.ArticleRowMapper" />

</beans>

package com.javarticles.spring.integration.jdbc;

import java.io.IOException;

import java.sql.SQLException;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class SpringIntegrationJdbcInboundRowMapperExample {

public static void main(String[] args) throws InterruptedException, IOException, SQLException {

ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext(

"jdbcInboundApplicationContext.xml");

Thread.sleep(1000);

context.close();

}

}