

IBM WAS Liberty 서버 기반
전자정부 프레임워크(Spring)
개발 가이드 03

이정운과장 (juwlee@kr.ibm.com)
IBM Korea, WebSphere Technical Sales

목 차

1	Spring 관련 설정 변경	3
1.1	JDBC 설정 변경	3
1.2	트랜잭션 매니저 설정 변경	9

1 Spring 관련 설정 변경

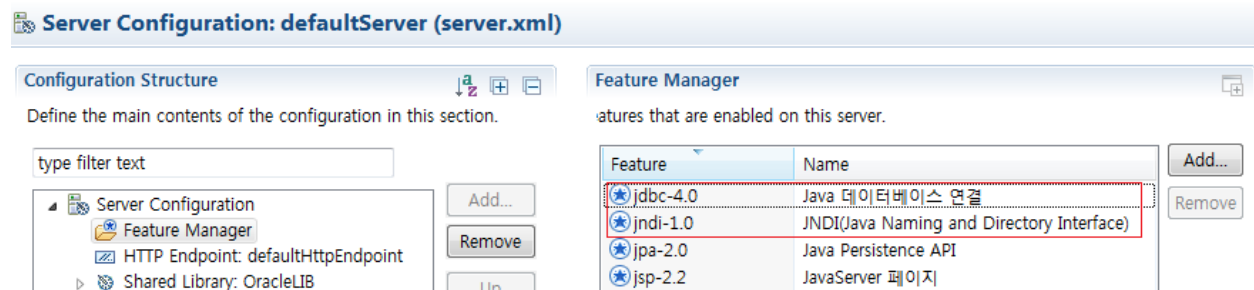
1.1 JDBC 설정 변경

전자정부 프레임워크의 경우 기본 샘플을 만드시게 되면 하단과 같이 DB 연결을 위한 JDBC 설정이 context-datasource.xml 파일에 들어가 있습니다.

```
<jdbc:embedded-database id="dataSource" type="HSQL">
  <jdbc:script location= "classpath:/db/sampledb.sql"/>
</jdbc:embedded-database>
```

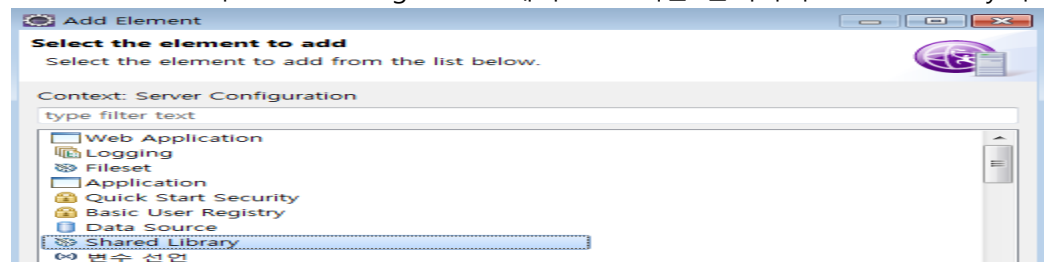
여기서 기본 Datasource 이외에 IBM WAS Liberty 서버의 JDBC 설정을 사용하시려면 다음과 같은 작업을 수행하셔야 합니다.(대상 DB 는 샘플로 Oracle 로 가정하며 Oracle 에 sample 을 돌리기 위한 table 이 만들어져 있다는 것을 가정하고 진행하도록 하겠습니다.)

1. server.xml 의 Feature Manager 에 **Add** 버튼 클릭하여 jdbc-4.0 과 jndi-1.0 기능 추가

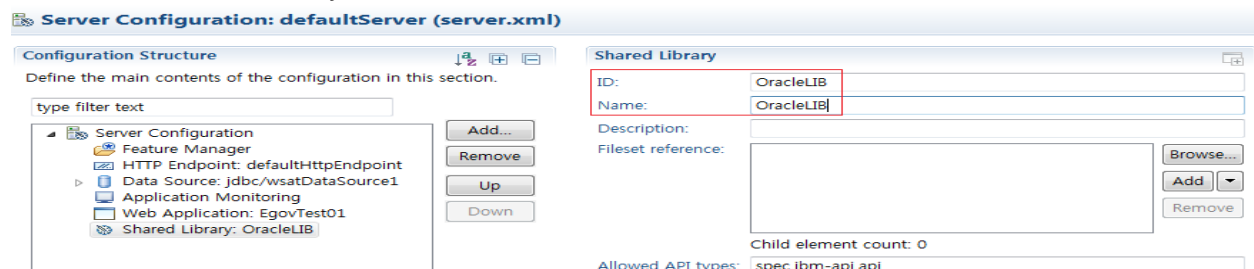


2. server.xml 에 JDBC 연결을 위한 JDBC Driver 용 파일 등록 (ojdbc6.jar)

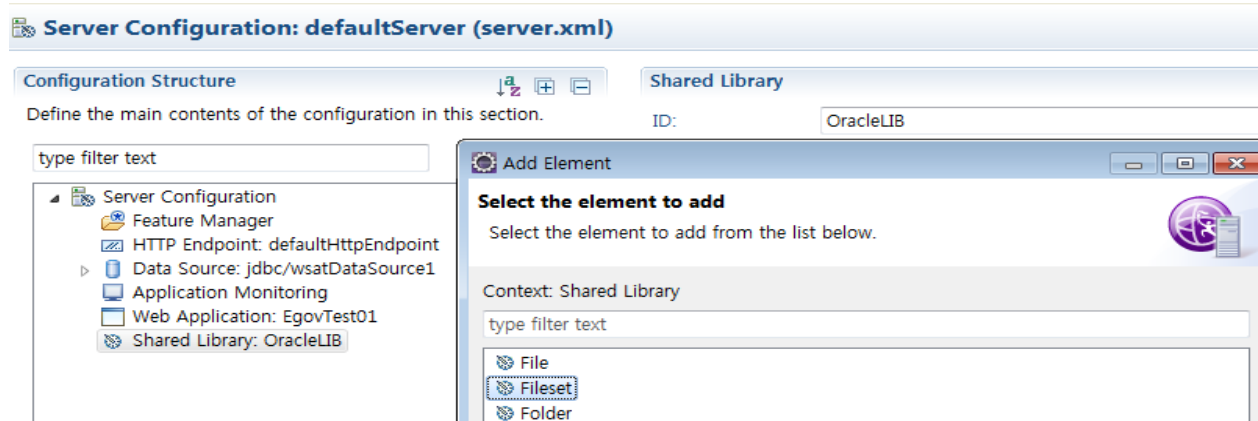
- server.xml 의 Server Configuration 에서 Add 버튼 클릭하여 Shared Library 추가



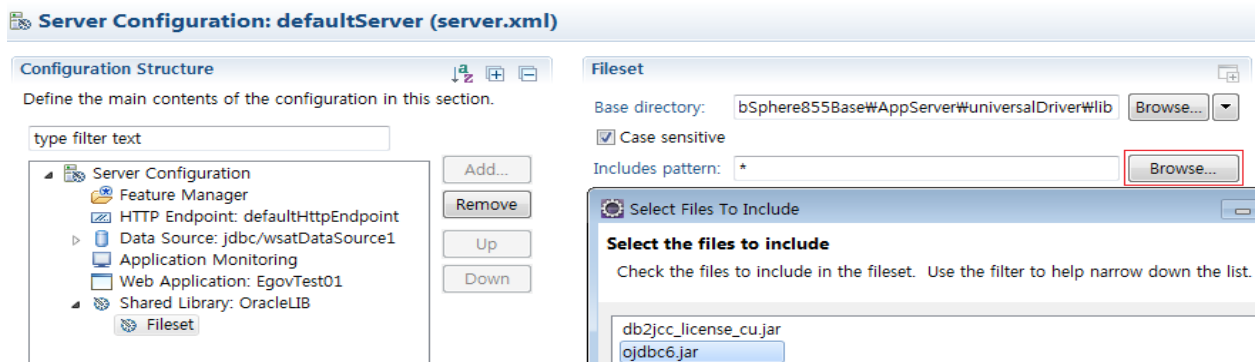
- 추가된 Shared Library 에 ID 와 Name 입력



- Shared Library 에서 **Add** 버튼 클릭하여 Fileset 추가

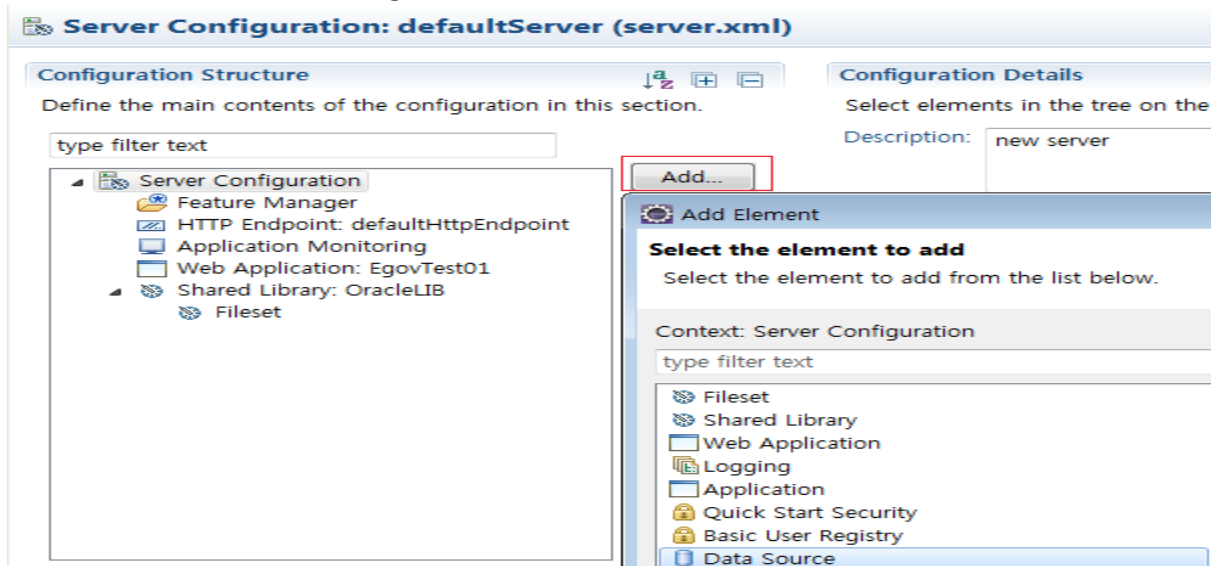


- 생성된 Fileset 설정에서 ojdbc6.jar 파일의 절대 경로를 입력한 후 Includes pattern 에서 Browse 버튼을 클릭하여 ojdbc6.jar 선택(해당 드라이버는 사전에 별도로 다운받아야 합니다.)



3. JDBC 연결을 위한 driver 등록을 완료 후에 JDBC Datasource 설정

- server.xml 의 Server Configuration 에서 Add 버튼 클릭하여 Data Source 추가



- 추가된 Data source 에 ID, JNDI Name, Type 을 입력

(이때, JNDI name 은 애플리케이션에서 사용할 JNDI 이름을 입력하면 되며 Type 은 connection pool 사용 여부와 트랜잭션의 종류에 따라서 가장 기본 형태이면서 Connection pool 을 사용하지 않는 javax.sql.DataSource, Connection pool 을 사용할 수 있는 javax.sql.ConnectionPoolDataSource, Connection pool 을 사용하면서 XA 까지 지원할 수 있는 javax.sql.XADataSource 중에 하나를 선택해서 사용할 수 있습니다.)

Server Configuration: defaultServer (server.xml)

The screenshot shows the 'Data Source' configuration page in the IDE. The 'Configuration Structure' pane on the left shows the hierarchy: Server Configuration > Shared Library: OracleLIB > Data Source: jdbc/wsatDataSource1. The main area contains the following fields:

- ID: oracleds
- JNDI name*: jdbc/wsatDataSource1
- JDBC driver reference: (empty dropdown)
- Connection manager reference: (empty dropdown)
- Type: javax.sql.ConnectionPoolDataSource
- Connection matching: MatchOriginalRequest
- Default container managed authentication data: (empty dropdown)
- Transaction isolation level: (empty dropdown)

- Data source 에서 **Add** 버튼을 클릭하여 Connection Manager 를 추가

Server Configuration: defaultServer (server.xml)

This screenshot shows the 'Add Element' dialog box open over the 'Data Source' configuration page. The dialog has a 'Context' field set to 'Data Source' and a list of elements to add. The 'Connection Manager' element is selected in the list.

Add Element
Select the element to add
Select the element to add from the list below.
Context: Data Source
type filter text

- Connection Manager
- DataDirect Connect for JDBC (sqlserver) Properties

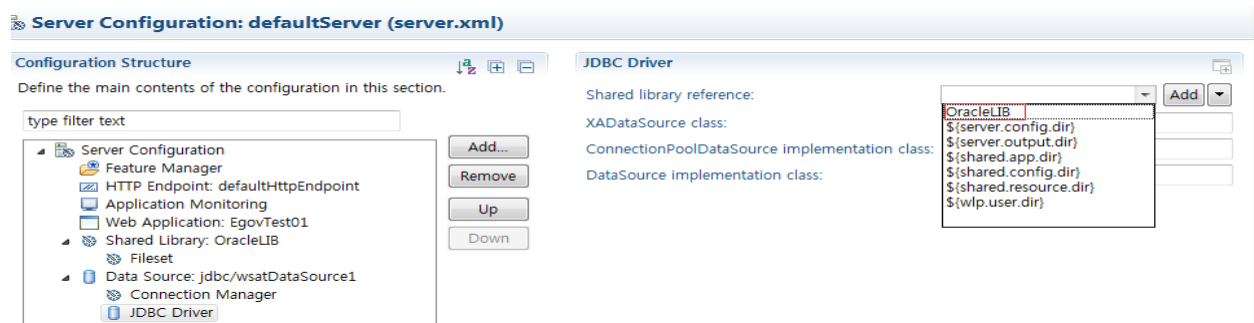
- Data source 에서 **Add** 버튼을 클릭하여 JDBC Driver 를 추가

This screenshot shows the 'Add Element' dialog box open over the 'Data Source' configuration page. The 'Context' is 'Data Source'. The list of elements includes various JDBC drivers, and 'JDBC Driver' is selected at the bottom of the list.

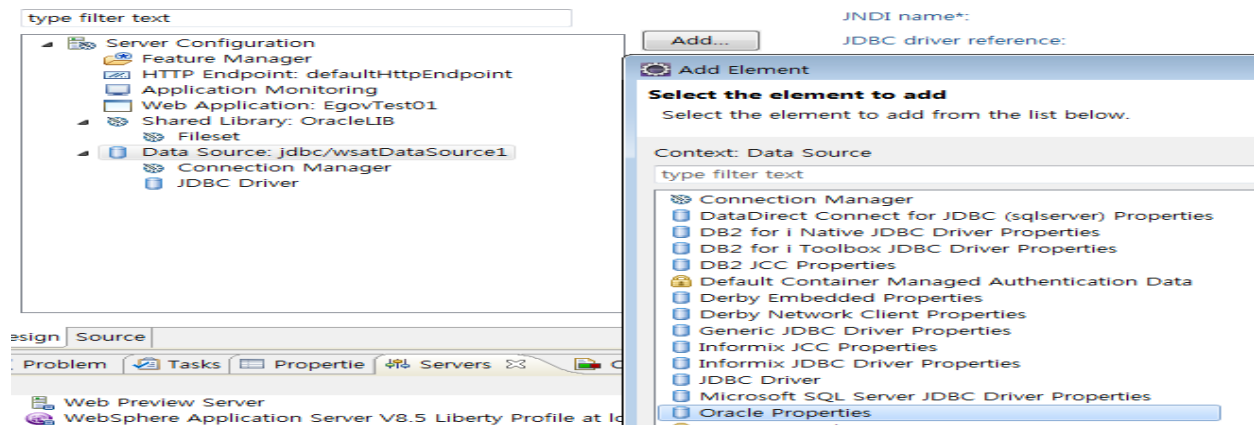
Add Element
Select the element to add
Select the element to add from the list below.
Context: Data Source
type filter text

- Connection Manager
- DataDirect Connect for JDBC (sqlserver) Properties
- DB2 for i Native JDBC Driver Properties
- DB2 for i Toolbox JDBC Driver Properties
- DB2 JCC Properties
- Default Container Managed Authentication Data
- Derby Embedded Properties
- Derby Network Client Properties
- Generic JDBC Driver Properties
- Informix JCC Properties
- Informix JDBC Driver Properties
- JDBC Driver

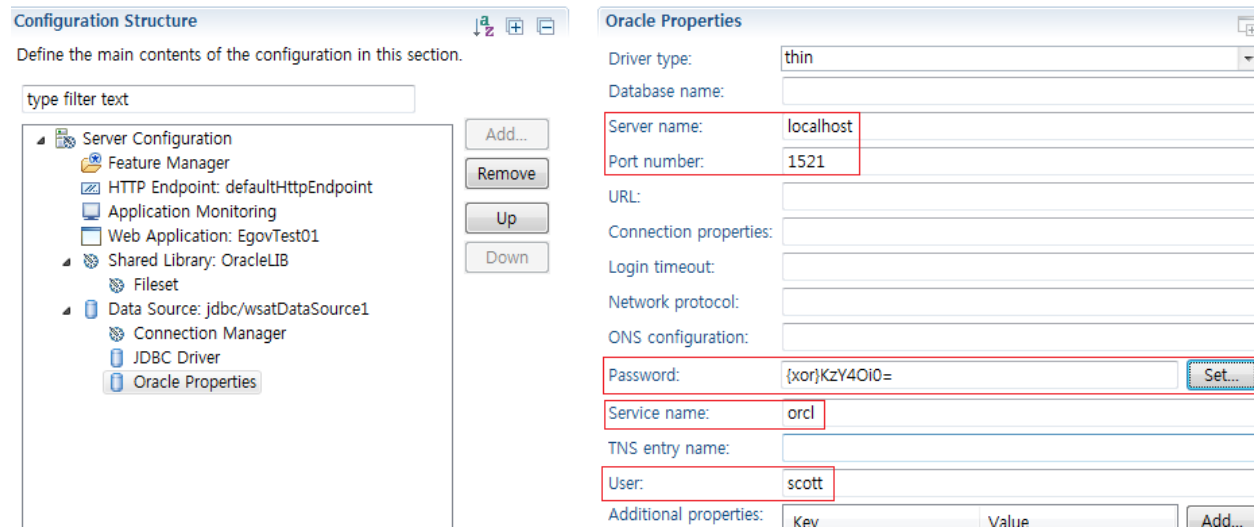
- 추가된 JDBC Driver 의 Shared library reference 의 콤보박스를 클릭하여 미리 만들어진 JDBC Driver file 정보인 OracleLIB 를 선택



- Data Source 에서 **Add** 버튼을 클릭하여 Oracle Properties 를 추가



- Oracle Properties 에 Oracle 에 연결하기 위한 연결정보를 환경에 맞게 하단과 같이 입력



4. 위와 같이 작업을 문제없이 수행하셨다면 IBM WAS Liberty 서버에서 Oracle 연결을 위한 JDBC 설정을 완료하신 것입니다.

(위에서와 같이 하나씩 작업하셔도 되고 하단의 text 를 server.xml 에 바로 붙여넣기 한 후 필요한 정보만 변경에서 한번에 만드실 수도 있습니다.)

```
<library id="OracleLIB" name="OracleLIB">
    <fileset dir="C:\IBM\WebSphere855Base\AppServer\universalDriver\lib" includes="ojdbc6.jar"></fileset>
</library>
<dataSource jndiName="jdbc/wsatDataSource1" id="oracleds"
    syncQueryTimeoutWithTransactionTimeout="false"
    type="javax.sql.ConnectionPoolDataSource">
    <connectionManager></connectionManager>
    <jdbcDriver libraryRef="OracleLIB"></jdbcDriver>
    <properties.oracle serviceName="orcl" user="scott" password="{xor}KzY4Oi0="></properties.oracle>
</dataSource>
```

```
<!--
<library id="OracleLIB" name="OracleLIB">
    <fileset dir="C:\IBM\WebSphere855Base\AppServer\universalDriver\lib" includes="ojdbc6.jar"></fileset>
</library>
<dataSource jndiName="jdbc/wsatDataSource1" id="oracleds"
    syncQueryTimeoutWithTransactionTimeout="false"
    type="javax.sql.ConnectionPoolDataSource">
    <connectionManager></connectionManager>
    <jdbcDriver libraryRef="OracleLIB"></jdbcDriver>
    <properties.oracle serviceName="orcl" user="scott" password="{xor}KzY4Oi0="></properties.oracle>
</dataSource>
```


5. JDBC 설정이 완료되었다면 전자정부 프레임워크에서 DB 연결을 위한 JDBC 설정이 들어있는 context-datasource.xml 파일을 하단과 같이 IBM WAS Liberty 서버에서 생성한 JDBC 를 사용하도록 변경합니다. (해당 가이드에서는 jee prefix 를 사용했는데 Spring 에서 해당 prefix 를 사용하기 위해서는 꼭 name space 에 대한 schema 정의도 같이 추가되어야 합니다. 있는 경우에는 생략)



```
<jee:jndi-lookup id="dataSource" jndi-name="jdbc/wsatDataSource1" resource-ref="true" />
<?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:jdbc="http://www.springframework.org/schema/jdbc"
    xmlns:jee="http://www.springframework.org/schema/jee"
    xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.0.
        http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee.xsd
        http://www.springframework.org/schema/jdbc http://www.springframework.org/schema/jdbc/spring-jdbc-3.0.xs">
    <jee:jndi-lookup id="dataSource" jndi-name="jdbc/wsatDataSource1" resource-ref="true" />
```

6. 마지막으로 전자정부 프레임워크의 샘플 프로젝트의 ibatis 가 Oracle 용 sqlMap 을 참조하도록 sql-map-config.xml 설정 파일을 하단과 같이 변경합니다.


```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sqlMapConfig PUBLIC "-//IBATIS.com//DTD SQL Map Config 2.0//EN" "http://www.ibatis.com/dtd/sql-map-config-2.dtd">
<sqlMapConfig>
    <!-- <sqlMap resource="egovframework/sqlmap/example/sample/EgovSample_Sample_SQL.xml"/> -->
    <!-- <sqlMap resource="egovframework/sqlmap/example/sample/EgovSample_Sample_SQL_mysql.xml"/> -->
    <sqlMap resource="egovframework/sqlmap/example/sample/EgovSample_Sample_SQL_oracle.xml"/>
</sqlMapConfig>
```

위와 같이 설정을 완료하신 후 샘플을 테스트 해보면 IBM WAS Liberty 서버의 JDBC Datasource 를 사용해서 동일한 결과를 얻을 수 있는 것을 확인할 수 있습니다.

Basic Board List 

← →   http://localhost:9080/web/

List Sample

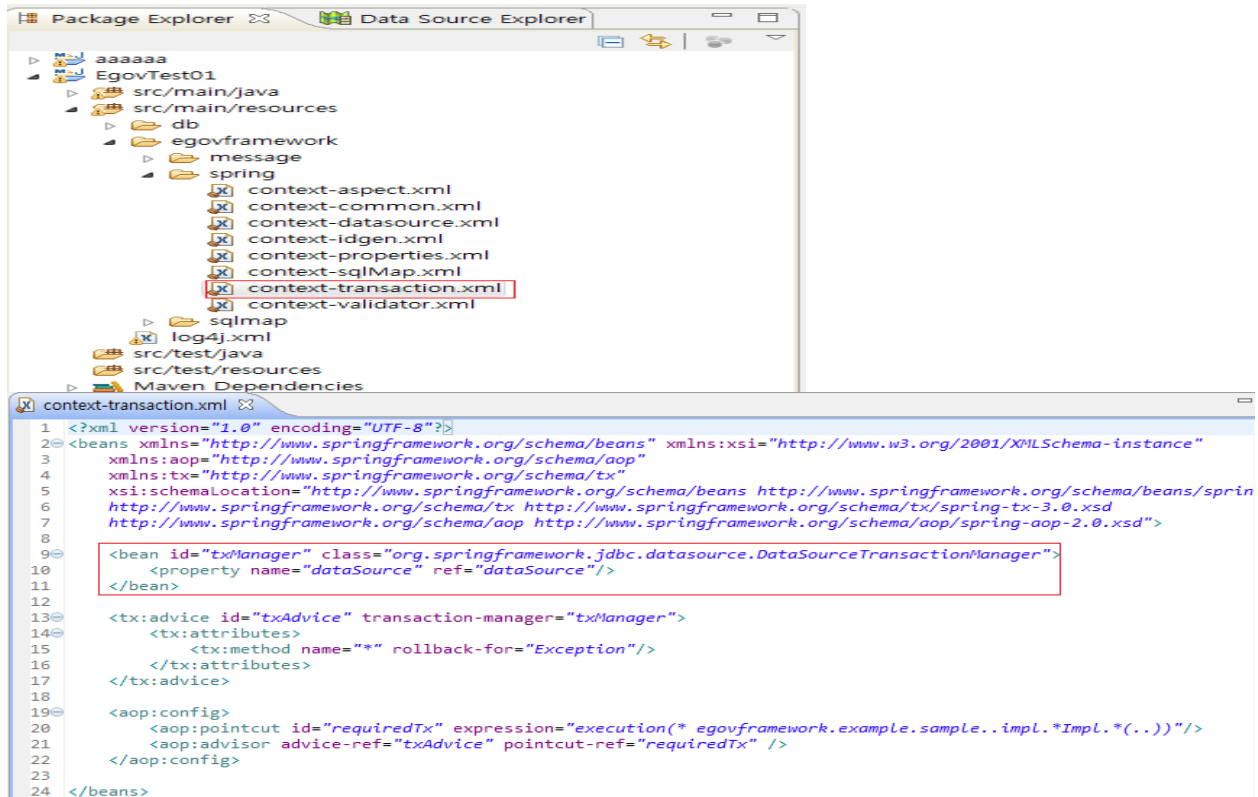
Name 

No	카테고리ID	카테고리명	사용여부	Description	등록자
1	Test0001	1ss	Y	1ss	1
2	SAMPLE-00350	aa	Y	aa	aa
3	SAMPLE-00340	ee	Y	ee	ee
4	SAMPLE-00330	3	Y	3	3
5	SAMPLE-00320	q	Y	q	q
6	SAMPLE-00310	a	Y	a	a
7	SAMPLE-00183	aaa2	Y	bbb	bbb
8	SAMPLE-00182	11111	Y	1111	1111

1

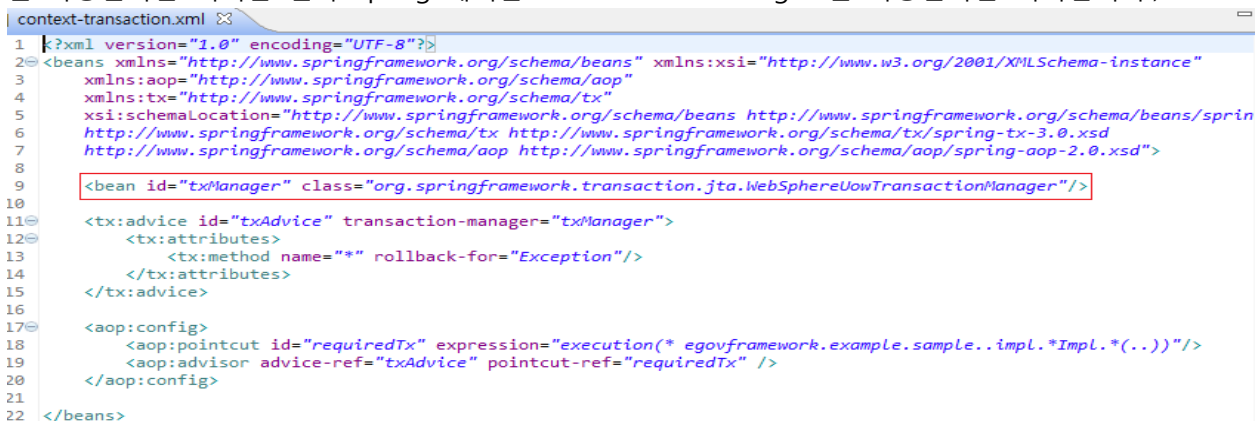
1.2 트랜잭션 매니저 설정 변경

전자정부 프레임워크의 경우 기본 샘플을 만드시게 되면 하단과 같이 트랜잭션 설정이 context-transaction.xml 파일에 미 Spring 내부의 트랜잭션 매니저를 사용하는 형태로 설정되어 있습니다.

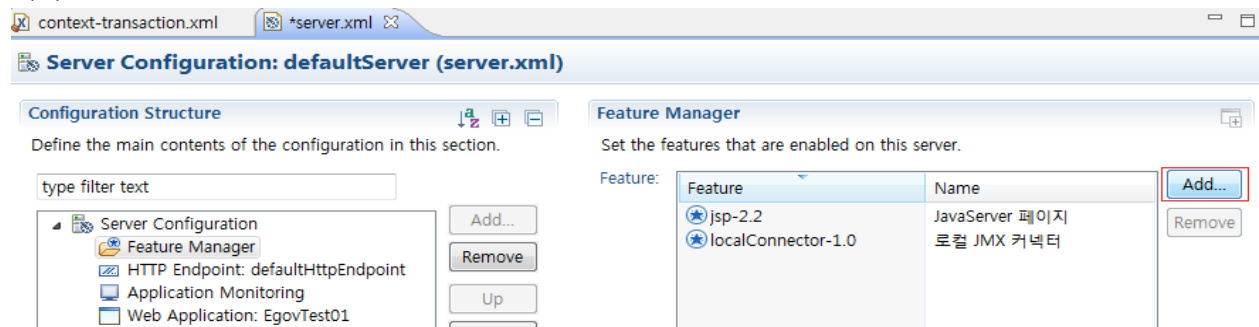


이를 IBM WAS Liberty 서버에서 제공하는 트랜잭션 매니저로 설정을 변경하시려면 다음과 같이 트랜잭션 매니저 설정 부분을 변경하면 됩니다.

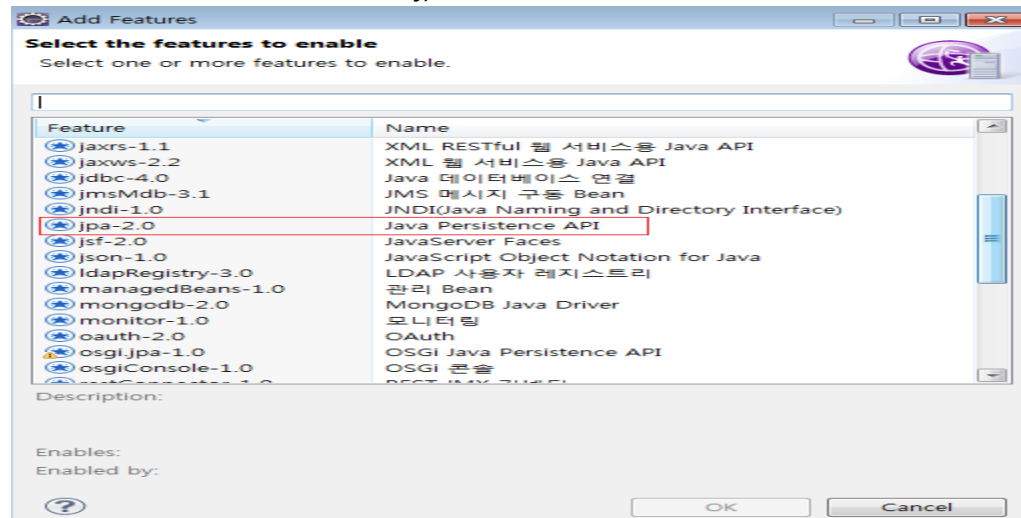
(WebSphereUowTransactionManager 사용하면 됩니다. 또는, <tx:jta-transaction-manager/> 를 사용해서 간단하게 적용할 수도 있습니다. - 여기서 IBM WAS Liberty 서버에서 제공하는 트랜잭션 매니저를 사용한다는 의미는 결국 Spring 에서는 JtaTransactionManager 를 사용한다는 의미입니다.)



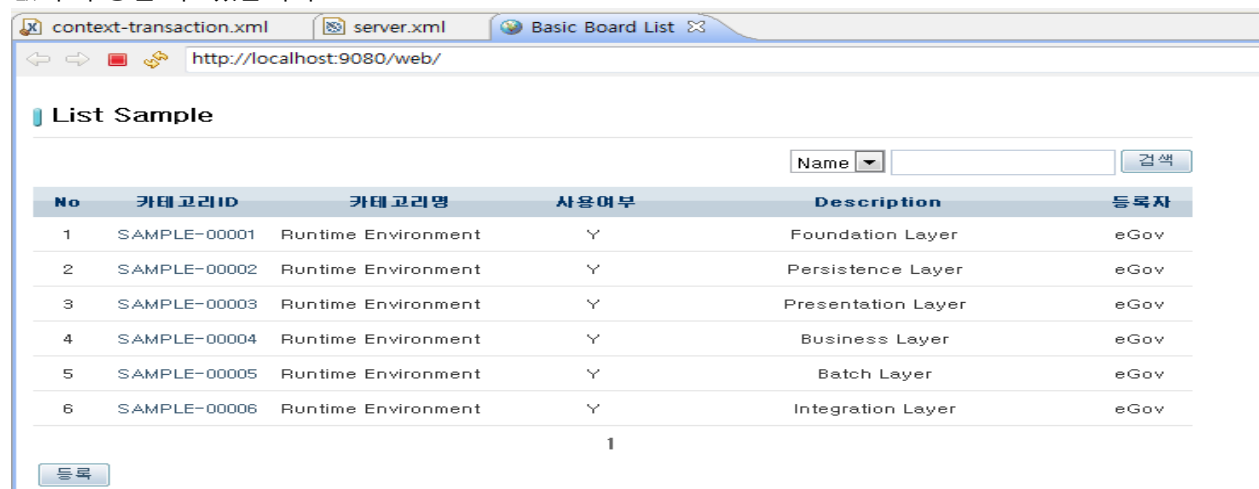
추가적으로 IBM WAS Liberty 서버의 server.xml 설정에서 **Feature Manager** 에서 **Add** 버튼을 클릭합니다.



그리고 나오는 Feature List 에서 jpa-2.0 을 선택하시면 됩니다.



이와 같이 작업하시면 IBM WAS Liberty 서버의 트랜잭션 매니저를 사용하면서 샘플 프로젝트를 문제 없이 구동할 수 있습니다.



단, JtaTransactionManager 를 사용했을 때 전자정부 프레임워크의 샘플 프로젝트가 수행은 되는데 insert 할 경우에 하단과 같은 Exception 이 나오는 것을 확인할 수 있습니다.



Could not commit JDBC transaction; nested exception is java.sql.SQLException: DSRA9350E: 글로벌 트랜잭션 중에는 Connection.commit 조작이 허용되지 않습니다.

이는 EgovSampleServiceImpl class 의 insertSample() 안에서 ID 를 자동으로 생성하기 위해 사용하는 EgovTableIdGnrService class(egovIdGnrService.getNextStringID() 부분이며 2009년도에 만들어진 소스)가 직접 DataSourceTransactionManager 를 생성하는 형태의 구조적인 버그에 기인하므로 이를 Transaction 에서 제외하거나 다른 방식으로 대체하시면 됩니다.

(실제 개발에서 JTATransaction 이 엮일 때 해당 class 를 사용하지 않으면 되며 샘플 프로젝트의 경우도 임시로 다른 형태로 ID 를 생성하거나 EgovTableIdGnrService 가 아닌 다른 class 를 통해서 ID 를 생성하면 됩니다.)

```
public String insertSample(SampleVO vo) throws Exception {
    log.debug(vo.toString());

    /** ID Generation Service */
    //String id = egovIdGnrService.getNextStringID();
    String id = "Test" + (System.currentTimeMillis() % 10000001);
    vo.setId(id);
    log.debug(vo.toString());

    sampleDAO.insertSample(vo);
    return id;
}
```