

# CROWNIX REPORT & ERS 7

## 표준 프레임워크 연동 가이드

---

(주)엠투소프트

## CONTENTS

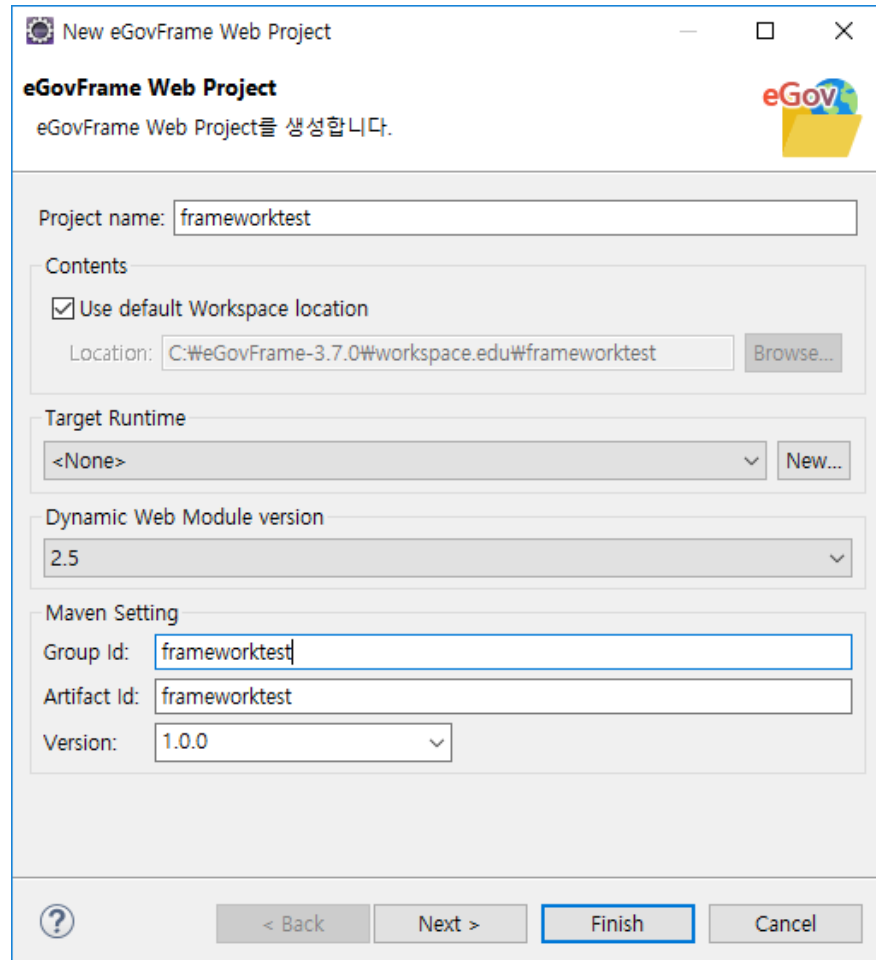
<b>1. 표준프레임워크 적용에 따른 로컬 개발환경 설정.....</b>	<b>3</b>
1.1. 프로젝트 준비.....	3
1.2. 파일 복사 .....	8
1.3. TOMCAT SERVER.XML 수정. ....	9
1.4. DATASERVER, REPORTINGSERVER MANAGER 최초접근 .....	9
1.5. 제품 사용 .....	10
<b>2. DATASERVER DATASOURCE설정 .....</b>	<b>12</b>
2.1. JDBC연결 .....	12
2.2. JNDI연결.....	12
<b>3. 데모.....</b>	<b>13</b>
3.1. 파일복사 .....	13
3.2. TOMCAT JNDI 설정 (CONTEXT.XML) .....	13
3.3. DATASOURCE 설정.....	14
3.4. EGOVBOARDMSTRLIST.JSP 수정 .....	15
3.5. EGOVTEMPLATELIST.JSP 수정 .....	16
3.6. EGOVRESTDEREGIST.JSP.....	17
<b>4. DEPLOY 가이드 .....</b>	<b>19</b>
4.1. WAR 배포 .....	19
4.2. EXPLODED 배포.....	19

## 1. 표준프레임워크 적용에 따른 로컬 개발환경 설정

### 1.1. 프로젝트 준비

1.1.1. eGovFrame-3.7.0설치 후 환경 구성된 eclipse 실행.

1.1.2. eGovFrame Web Project 생성.



**New eGovFrame Web Project**

**eGovFrame Web Project**  
eGovFrame Web Project를 생성합니다.

Project name:

Contents

☒ Use default Workspace location

Location:

Target Runtime

Dynamic Web Module version

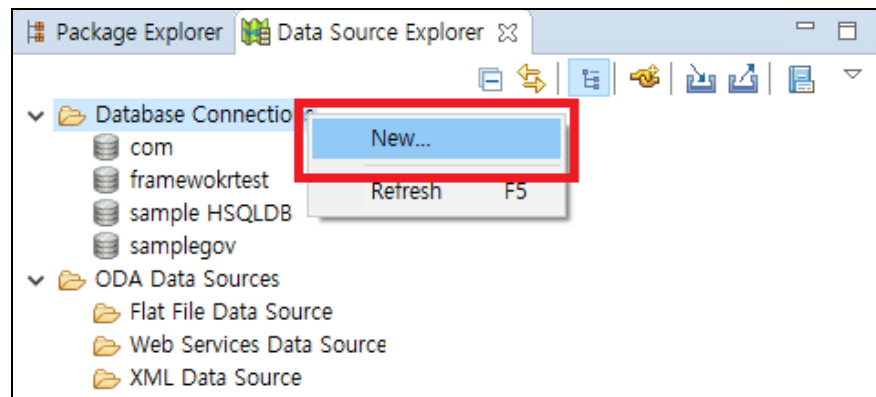
Maven Setting

Group Id:

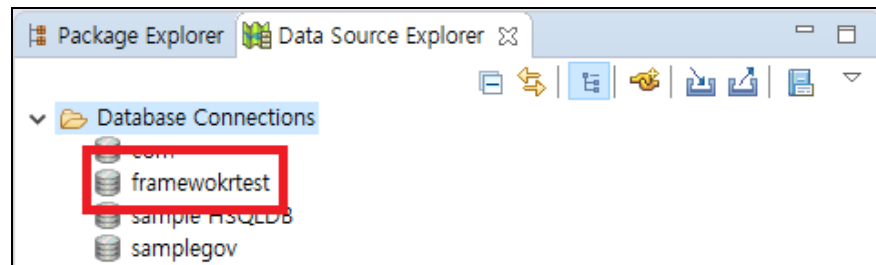
Artifact Id:

Version:

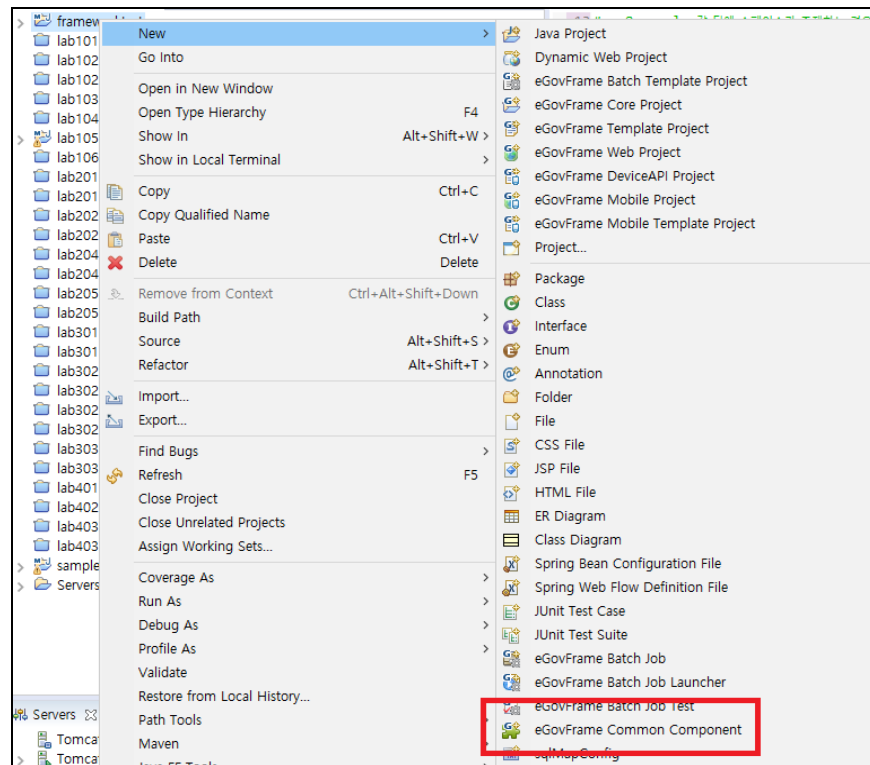
1.1.3. Data Source Explorer를 열어 새로운 DB Connection 생성.




1.1.4. 생성한 Connection 확인.



1.1.5. 프로젝트에서 eGovFrame Common Component 선택.



1.1.6. 설치할 컴포넌트를 선택 후 Next(테스트시에는 게시판 템플릿, 게시판을 선택하고 해당 컴포넌트의 Dependency로 생성된 컴포넌트를 사용한다.)



## Add eGovFrame Common Component

### Select Common Component

설치할 공통 컴포넌트를 선택하십시오.

Project name : **frameworktest**

- > ☐ 역할/권한관리
- > ☐ 암호화/복호화
- > ☐ 통계
- ▼ ☒ 게시판
  - ☒ 게시판 템플릿
  - ☒ 게시판
  - ☐ 댓글관리
  - ☐ 스크랩기능
  - ☐ 만족도조사
- > ☐ 커뮤니티

#### Description

사용자별, 부서별 권한을 부여하고 관리하는 기능을 제공함

#### Dependency

- N/A

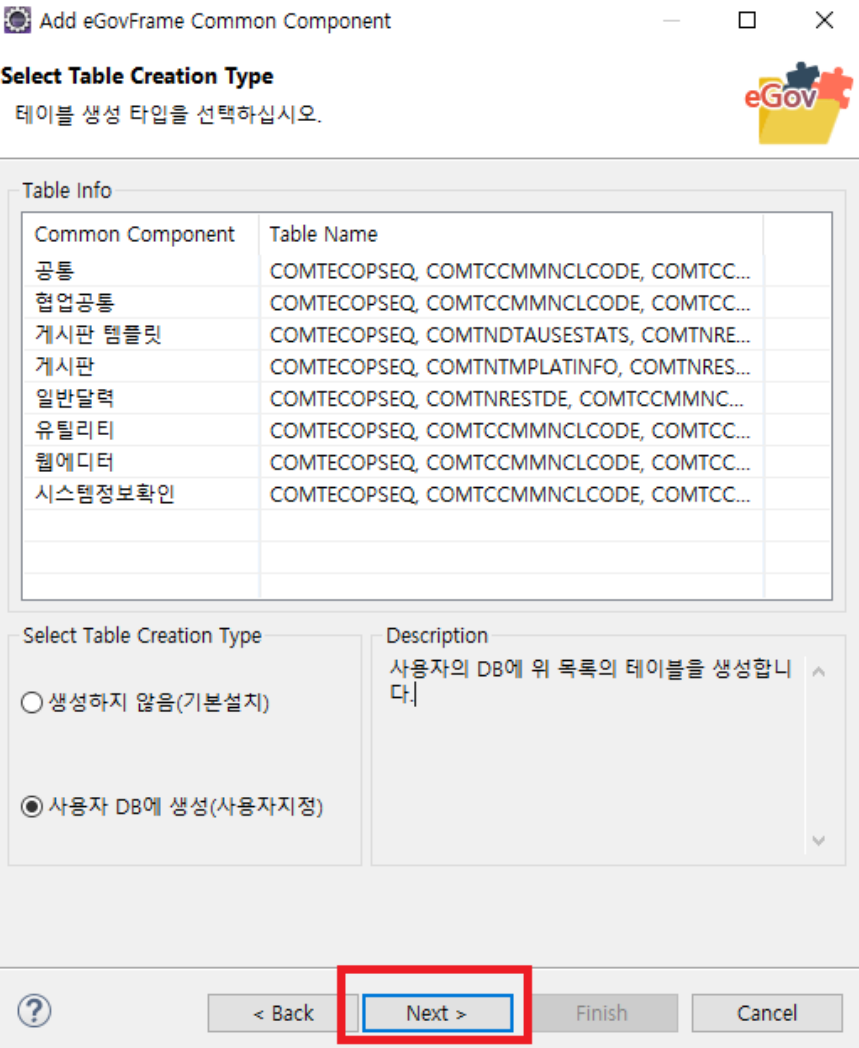
**Note:** 선택한 컴포넌트의 Dependency 컴포넌트를 수정했을 경우 해당 컴포넌트가 정상작동하지 않을 수 있습니다.

gpkisecureweb, libgpkiaapi\_jni, smeapi, ojdbc 라이브러리는 별도로 다운로드 해야 합니다. 닉서스에 추가하여 사용하는 것을 권장합니다.

범례) ☐ : 설치되지 않은 컴포넌트  
☒ : 설치할 컴포넌트  
☒ : 이미 설치된 컴포넌트

?
< Back
Next >
Finish
Cancel

1.1.7. 사용자 DB에 생성 선택 후 Next.



**Add eGovFrame Common Component**

**Select Table Creation Type**

테이블 생성 타입을 선택하십시오.

Table Info

Common Component	Table Name
공통	COMTECOPSEQ, COMTCCMMNCLCODE, COMTCC...
협업공통	COMTECOPSEQ, COMTCCMMNCLCODE, COMTCC...
게시판 템플릿	COMTECOPSEQ, COMTNDTAUSESTATS, COMTNRE...
게시판	COMTECOPSEQ, COMTNTMPLATINFO, COMTNRES...
일반달력	COMTECOPSEQ, COMTNRESTDE, COMTCCMMNC...
유틸리티	COMTECOPSEQ, COMTCCMMNCLCODE, COMTCC...
웹에디터	COMTECOPSEQ, COMTCCMMNCLCODE, COMTCC...
시스템정보확인	COMTECOPSEQ, COMTCCMMNCLCODE, COMTCC...

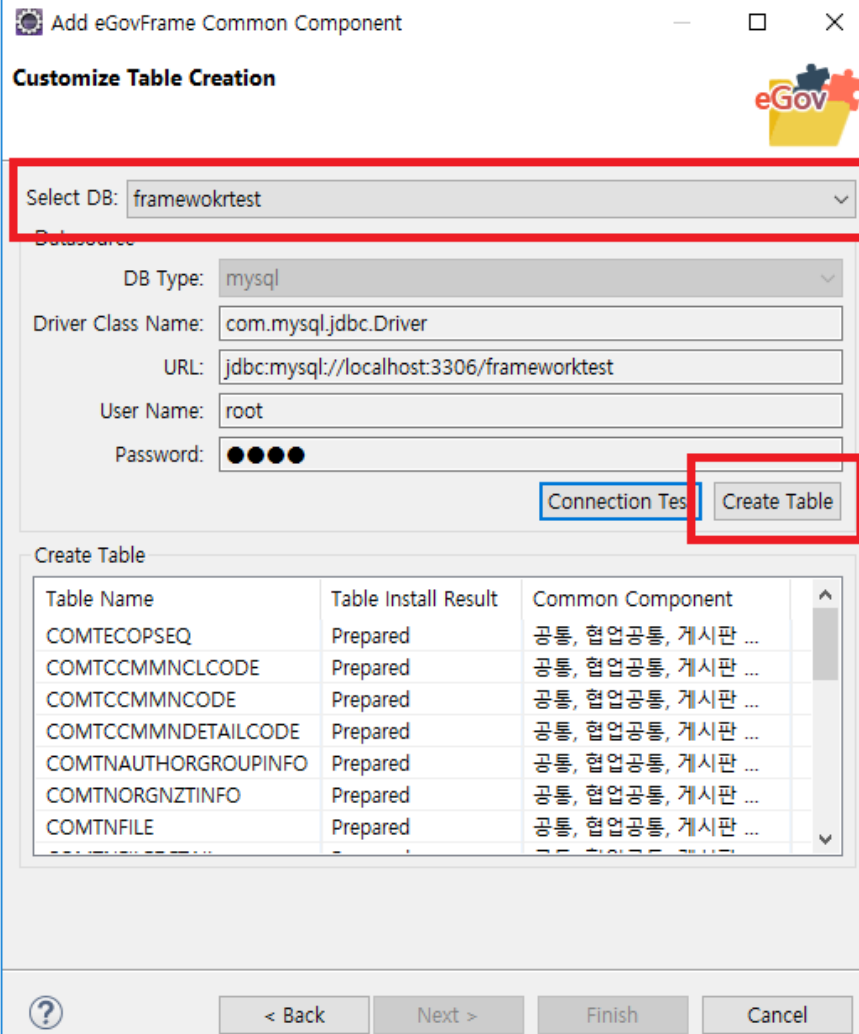
Select Table Creation Type

☐ 생성하지 않음(기본설치)
   
☒ 사용자 DB에 생성(사용자지정)

Description

사용자의 DB에 위 목록의 테이블을 생성합니다.

1.1.8. 사용할 DB선택 후 Create Table 클릭.



**Add eGovFrame Common Component**

**Customize Table Creation**

Select DB: frameworktest

DB Type: mysql

Driver Class Name: com.mysql.jdbc.Driver

URL: jdbc:mysql://localhost:3306/frameworktest

User Name: root

Password: ●●●●

Connection Test Create Table

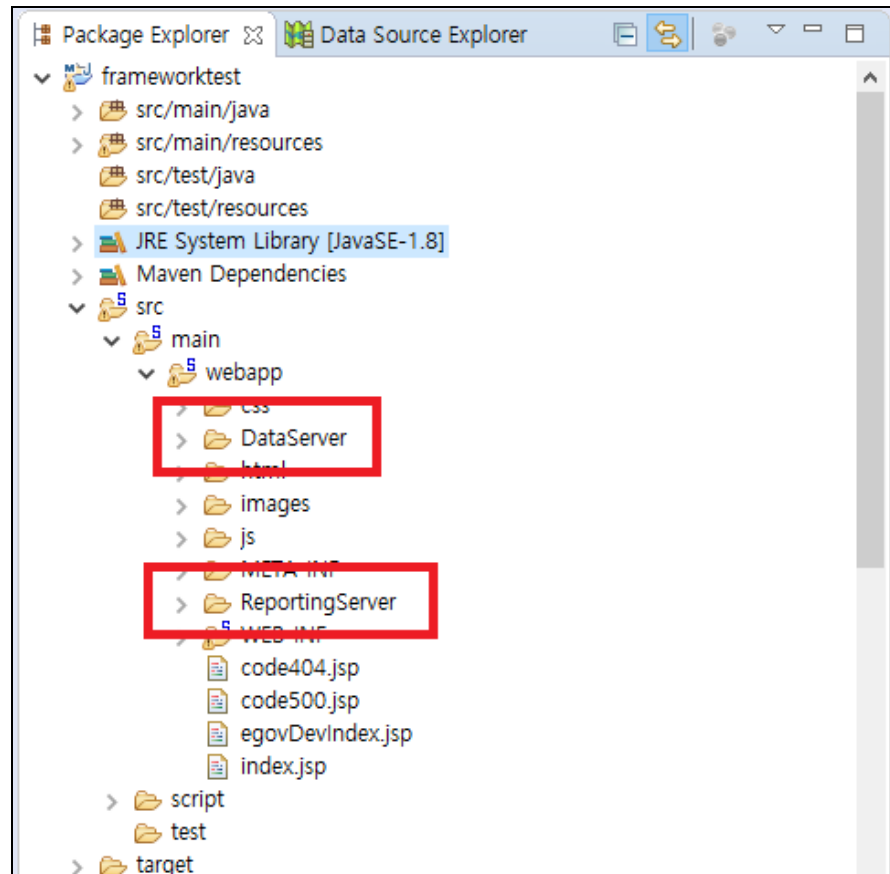
Table Name	Table Install Result	Common Component
COMTECOPSEQ	Prepared	공통, 협업공통, 게시판 ...
COMTCCMMNCLCODE	Prepared	공통, 협업공통, 게시판 ...
COMTCCMMNCCODE	Prepared	공통, 협업공통, 게시판 ...
COMTCCMMNDETAILCODE	Prepared	공통, 협업공통, 게시판 ...
COMTNAUTHORGROUPINFO	Prepared	공통, 협업공통, 게시판 ...
COMTNORGNZTINFO	Prepared	공통, 협업공통, 게시판 ...
COMTNFILE	Prepared	공통, 협업공통, 게시판 ...

< Back Next > Finish Cancel

1.1.9. 테이블 생성이 완료되면 Finish 클릭.(web.xml을 수정하시겠습니까? 문구 확인 시 Yes클릭.)

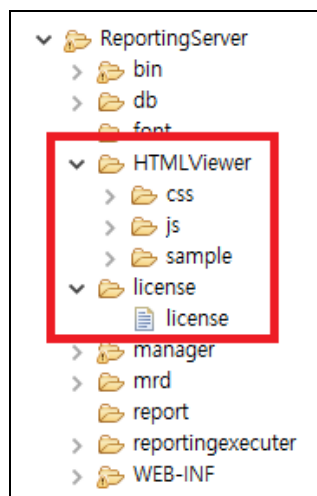
## 1.2. 파일 복사

1.2.1. 프로젝트의 src/main/webapp/ 경로에 ReportingServer와 DataServer를 복사한다.



1.2.2. 각 서버폴더 내부 /license 경로에 license 파일 확인.

1.2.3. 뷰잉에 필요한 HTML5 Viewer는 ReportingServer/HTMLViewer 폴더를 생성하여 복사한다.(경로참조가 가능하면 위치는 무관함.)





### 1.3. Tomcat server.xml 수정.

#### 1.3.1. Host태그 내부에 Context 태그 추가

(docBase예시 : "../wtpwebapps/프로젝트이름/ReportingServer")

```
<Host appBase="/webapps" autoDeploy="true" name="localhost" unpackWARs="true">

  <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs" pattern="%h %l %u %t &quot;%r&quot; %s" />
  <!-- <Context docBase="DBTest" path="/DBTest" reloadable="true" source="org.eclipse.jst.jee.server:DBTest"/> -->

  <Context docBase="../wtpwebapps/frameworktest/ReportingServer" path="/ReportingServer"/>
  <Context docBase="../wtpwebapps/frameworktest/DataServer" path="/DataServer"/>

  <Context docBase="samplegov" path="/samplegov" reloadable="true" source="org.eclipse.jst.jee.server:samplegov"/>
</Host>
```

### 1.4. DataServer, ReportingServer Manager 최초접근

자사의 DataServer와 ReportingServer는 배포할 경우 초기설정에서는 바로 사용이 불가능 하며 배포시 최초 한번은 사용자가 Manager에 접근하여 초기 설정된 admin 계정의 패스워드를 변경해야 한다.

#### 1.4.1. 접근하려고 하는 서버의 Manager경로로 페이지 오픈.

(ex : <http://localhost:8080/DataServer/manager/index.html>)

#### 1.4.2. 기본 설정된 ID/PW는 admin/admin이며 최초 접속일 경우 로그인 전에 패스워드 변경 UI를 확인 할 수 있다.



The image shows the login page for the Enterprise Reporting System Server Manager. It features a logo at the top consisting of three interlocking cubes in yellow and orange. Below the logo, the text "Enterprise Reporting System" is in orange and "Server Manager" is in black. There are two input fields: "아이디 :" (ID) and "비밀번호 :" (Password). Below these fields is a "로그인" (Login) button.

## 1.5. 제품 사용

자사 ERS는 ReportingServer제품에서 .mrd확장자를 가진 템플릿 형태파일을 보고서 형태로 Reporting하면 HTML5 Viewer제품을 이용하여 브라우저에서 사용자가 보고서를 확인 할 수 있도록 하는 방식으로 동작하고 있다. mrd 파일은 자사의 Designer tool을 사용하여 작성이 가능하며 본 문서에서는 파일 생성에 관한 내용은 다루지 않도록 한다.

1.5.1. HTML5 Viewer는 기본적으로 css와 js파일로 구성되어 있으며 HTML문서에서 아래와 같이 3개 파일을 참조하여 사용한다.

```
<script src="../../js/jquery-1.11.0.min.js"></script>
<script src="../../js/crownix-viewer.min.js"></script>
<link rel="stylesheet" type="text/css" href="../../css/crownix-viewer.min.css">
```

1.5.2. 실행을 위해서는 HTML상에서 자사 보고서를 보이게 할 영역이 필요하고 m2soft.crownix.Viewer의 생성자에 ReportingServer주소와 영역의 id를 파라미터로 넘겨주어 객체를 생성하고, openFile('파일이름', '파라미터')함수를 실행하여 보고서를 open한다. 1.5.1의 3파일이 참조되어있다면 자유롭게 영역을 지정하여 HTML5Viewer제품을 실행할 수 있다.

```
<body style="margin:0;height:100%">
  <div id="crownix-viewer" style="position:absolute;width:100%;height:100%"></div>
  <script>
    window.onload = function(){
      var viewer = new m2soft.crownix.Viewer('http://localhost:8080/ReportingServer/service', 'crownix-viewer');
      viewer.openFile('sampleTest.mrd', '');
    };
  </script>
</body>
```

### 1.5.3. HTML5 Viewer실행화면

Crownix HTML5 Viewer x

file:///C:/eGovFrame-3.7.0/workspace.edu/samplegov/src/main/webapp/ReportingServer/HTMLViewer/sample...

100%

**DONGAH FIRE & MARINE INSURANCE CO.,LTD.**  
DongaBldg. 1213, Youido-Dong, Youngdeungpo-Gu, Seoul 200-607, Korea TEL: 02/789-7357-7523(Contract)  
02/879-2211-6(CLAIM) FAX: 02/679-4216

the Republic of Korea  
**STAMP**  
1000 Won

**MARINE CARGO INSURANCE POLICY**

Assured(s), etc. ID\_009 ID\_010

Policy No.	ID_011	Ref. No.
Claim, if any, payable at	ID_031	ID_016
ID_032	ID_032	ID_017
ID_061	ID_061	Amount insured ID_043 ID_050
ID_068	ID_068	ID_052 ID_053 ID_054 ID_055 ID_056 ID_057
ID_065 ID_066	ID_065 ID_066	ID_059 ID_060 ID_061
Survey should be approved by	ID_069 ID_070 ID_072 ID_073 ID_075 ID_076	ID_079 ID_081 ID_083
ID_087	ID_087	ID_089 ID_091 ID_093 ID_095 ID_012 ID_013
ID_018	ID_018	Conditions
ID_019	ID_019	Subject to the following Clauses as per back hereof.
ID_021	ID_021	- Institute Classification Clause.
Local Vessel or Conveyance	Form(interior port or place of loading)	- On-deck Clause
ID_022	ID_022	- Institute Radioactive Contamination Exclusion Clause.
Skip or Vessel	Sailing on or about	- Special Replacement Clause (applying to machinery)
ID_023 ID_024	ID_023 ID_024	- Co-insurance Clause (applicable case of Co-insurance)
ID_025 ID_026	ID_025 ID_026	- Label Clause (applying to labelled goods)
at and from	transhipped at	ID_020
ID_029 ID_030	ID_029 ID_030	ID_027
ID_034 ID_035	ID_034 ID_035	ID_028
ID_037 ID_038	ID_037 ID_038	ID_033
arrived at	thence to	ID_036
ID_041 ID_044	ID_041 ID_044	ID_039
ID_046 ID_047	ID_046 ID_047	ID_040
ID_049 ID_014	ID_049 ID_014	ID_045
Goods and Merchandises		ID_048
ID_061	ID_061	ID_015
ID_067	ID_067	ID_058
ID_071	ID_071	ID_064
ID_077	ID_077	ID_068
ID_080	ID_080	ID_074
ID_084	ID_084	ID_078
ID_088	ID_088	ID_082
ID_090	ID_090	ID_086
ID_091	ID_091	ID_090

ORIGINAL COPY  
DUPLICATE COPY

## 2. DataServer DataSource설정

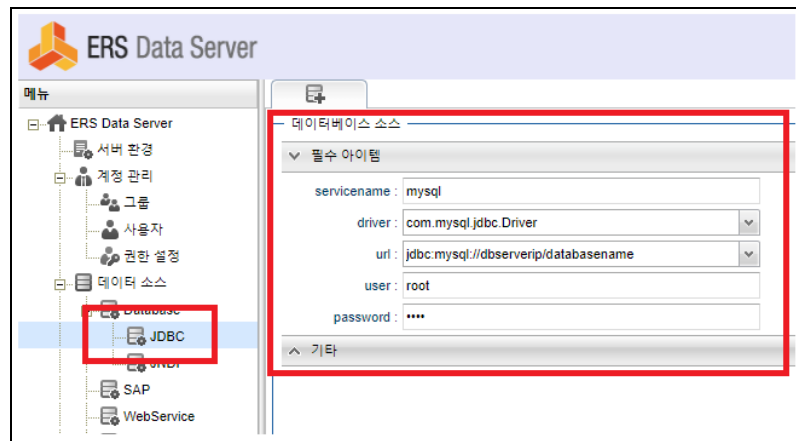
자사 제품군은 DataServer제품을 통해 DB와 통신하는 구조로 설계되어 있다. 때문에 DataSource를 DataServer에서 설정하여 사용한다.

현재 자사 DataServer 제품에서는 JDBC와 JNDI방식의 두 가지 방식으로 DataSource를 설정할 수 있다. 해당 파트에서는 DataSource의 설정 방법에 대해서 다루고 있으며 이는 DataServer제품이 배포 중 이어야 한다.

설정은 DataServer의 Manager에서 가능하다.

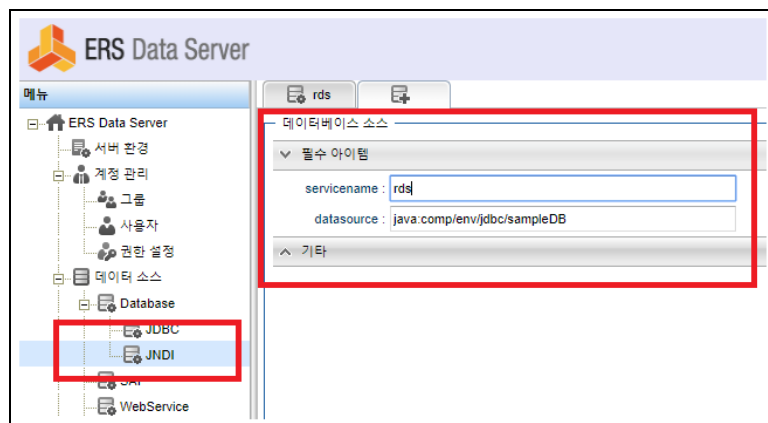
### 2.1. JDBC연결

2.1.1. 로그인 이후 보이는 메뉴에서 데이터소스 - 데이터베이스 JDBC탭에서 JDBC 방식의 데이터 소스 설정이 가능하다.



### 2.2. JNDI연결

2.2.1. JDBC와 같은 경로에서 JNDI를 선택하면 WAS에서 설정한 JNDI설정을 이용하여 데이터 소스 설정이 가능하다.

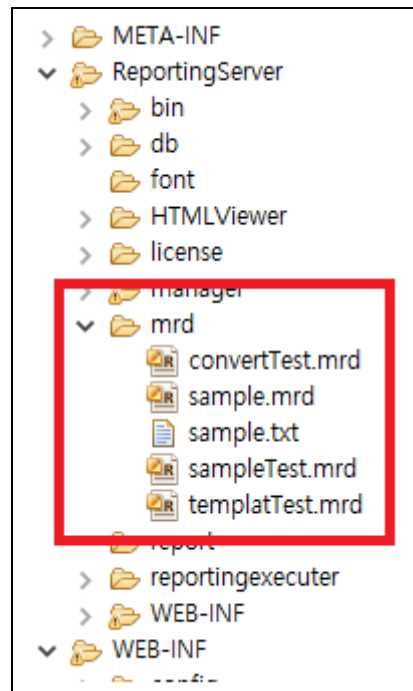


### 3. 데모

해당 파트에서는 호환성 인증 데모를 위한 설정을 다루고 있으며 1장 파트의 개발환경 설정이 완료된 이후 설정에 대해서만 설명한다.

#### 3.1. 파일복사

3.1.1.ReportingServer/mrd경로에 데모에 필요한 mrd를 복사한다.



#### 3.2. Tomcat JNDI 설정 (context.xml)

3.2.1. 톰캣의 context.xml 파일에 JNDI설정을 추가한다.

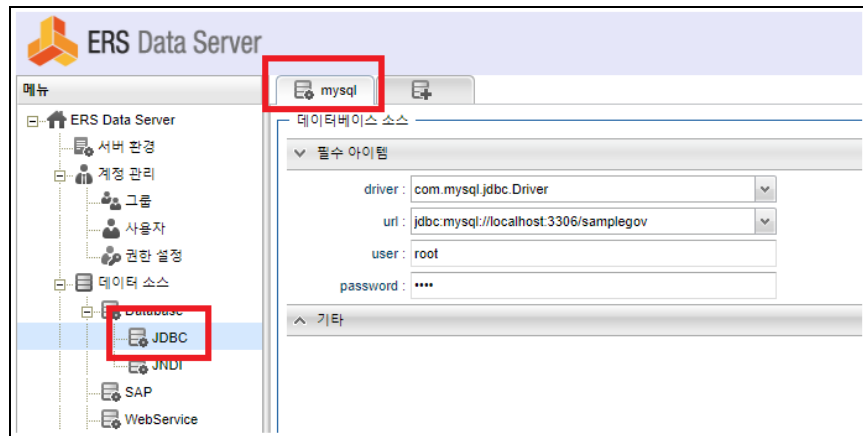
```
<!-- Default set of monitored resources. If one of these changes, the -->
<!-- web application will be reloaded. -->
<WatchedResource>WEB-INF/web.xml</WatchedResource>
<WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
<Resource name="jdbc/sampleDB" auth="Container"
  type="javax.sql.DataSource"
  username="root" password="root"
  driverClassName="com.mysql.jdbc.Driver"
  url="jdbc:mysql://localhost:3306/samplegov"
  maxActive="100"
  maxIdle="30"
  maxWait="10"/>
<!-- Uncomment this to disable session persistence across Tomcat restarts -->
<!--
<Manager pathname="" />
-->
```

### 3.3. DataSource 설정

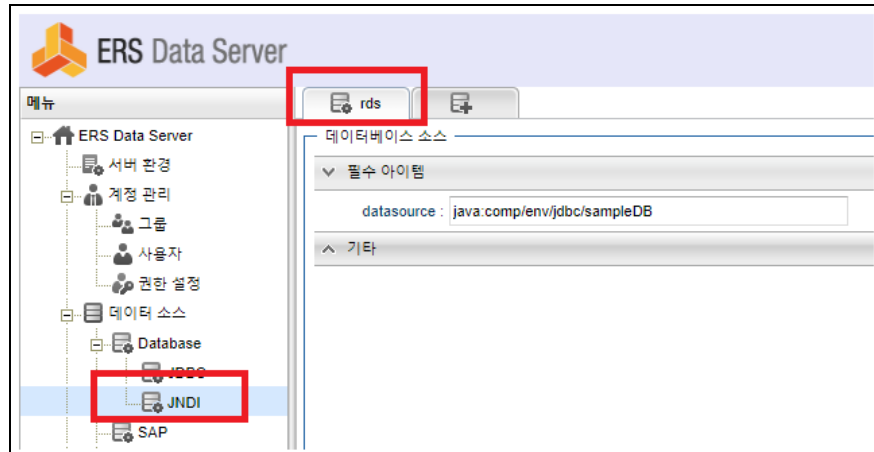
본 데모에서는 JDBC방식과 JNDI방식 모두를 사용하기 때문에 각각의 DataSource가 DataServer제품에 설정되어 있어야한다.

mrd파일에 대해서는 자세하게 다루고 있지 않기 때문에 데모에서는 샘플 mrd에 설정된 DataSource에 맞춰 DataServer를 설정하도록 한다.

#### 3.3.1. DataServer Manager 설정 - JDBC(DataSource Name : mysql)



#### 3.3.2. DataServer Manager 설정 - JNDI(DataSource Name : rds)



### 3.4. EgovBoardMstrList.jsp 수정

해당 파일에서는 HTML5Viewer를 직접 참조하지 않고 경로에 있는 sample.html파일을 popup창에서 보이도록 하는 기능을 구현하며 sample.html파일은 JDBC방식의 DataSource를 사용하는 mrd를 오픈한다.

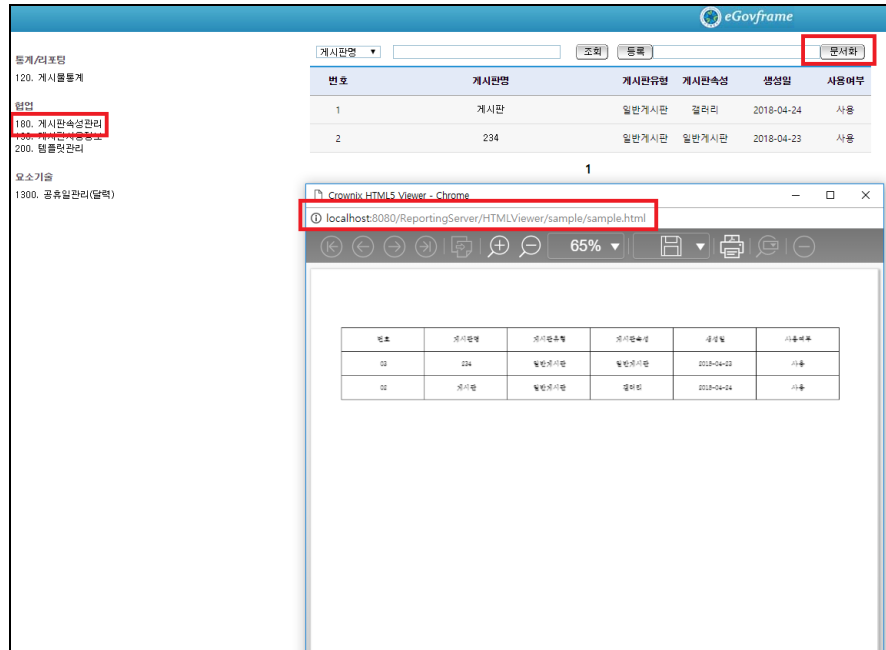
#### 3.4.1.스크립트 부분에 popup창을 오픈 하는 함수를 만든다.

```
function fn_egov_inquire_brdMstr(bbsId){
    document.frm.bbsId.value = bbsId;
    document.frm.action = "<:url value='/cop/bbs/SelectBBSMasterInf.do'/">;
    document.frm.submit();
}
function openMrd(){
    window.open("/ReportingServer/HTMLViewer/sample/sample.html", "crownix", "width=500, height=500");
}
</script>
<title>게시판 목록조회</title>
```

#### 3.4.2. 아래와 같이 원하는 위치에 만든 함수를 호출하는 UI를 구성한다.

```
<td>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td>
<td><span class="button"><a href="<:url value='/cop/bbs/addBBSMaster.do'/"> onclick="fn_egov_insert_addBrd
<td width="35%">
    <input id="searchMrd" type="text" value="" style="width: 95%; border: 1px solid #ccc;" />
</td>
<td><span class="button"><a href="#" onclick="openMrd(); return false;">문서화</a></span></td>
</tr>
</table>
</th>
</tr>
...</td>
```

### 3.4.3. 실행화면



### 3.5. EgovTemplateList.jsp 수정

해당파일은 HTML5Viewer를직접 사용하여 프레임워크에서 제공되는 템플릿 내부에 HTML5Viewer를 embedding된 형태로 보이게 하는 기능을 구현하며 이때는 JNDI방식의 DataSource를 사용한 mrd를 오픈한다.

#### 3.5.1. HTML5Viewer실행에 필요한 파일들을 다음과같이 참조한다.

```
<link href="<c:url value='/ReportingServer/HTMLViewer/css/crownix-viewer.min.css' />" rel="stylesheet" type="text/css">
<script type="text/javascript" src="<c:url value='/ReportingServer/HTMLViewer/js/jquery-1.11.0.min.js' />"></script>
<script type="text/javascript" src="<c:url value='/ReportingServer/HTMLViewer/js/crownix-viewer.min.js' />"></script>
```

#### 3.5.2. Viewer를 실행하는 동작을 하는 함수를 생성한다.

```
function convertDocument(){
    var viewer = new m2soft.crownix.Viewer('http://localhost:8080/ReportingServer/service', 'crownix-viewer');
    viewer.openFile('templatTest.mrd','');
}
```

#### 3.5.3. HTML5Viewer가 보일 영역을 설정 한다.

```
</div>
</form>
<div id="crownix-viewer" style="position:absolute;width:1000px;height:500px;"></div>
</body>
</html>
```





3.6.2. 사용자가 입력한 데이터를 받아와 openFile()함수에서 사용하는 파라미터의 형태로 가공하여 오픈 시 전달하는 동작을 하는 함수 생성.

```
function convertDocument(){
    var viewer = new m2soft.crownix.Viewer('http://localhost:8080/ReportingServer/service', 'crownix-viewer');

    var dataList = [
        $('input[name=vrestdeDe]').val(),
        $('input[name=restdeNm]').val(),
        $('textarea[name=restdeDc]').val(),
        $('select[name=restdeSeCode]').find(":selected").text()
    ];

    var rdata = '/rdata [' + dataList + ']';
    viewer.openFile('convertTest.mrd', rdata);
}
```

3.6.3. 3.5.3.과 같이 HTML5 Viewer가 보일 영역을 설정한다.

3.6.4. 3.5.4.과 같이 생성한 함수를 실행하도록 하는 UI를 구성한다.

3.6.5. 동작화면

날짜	2018-04-02
유일명	공유일
유일설명	테스트용 공유일입니다.
유일구분	법정유일

## 4. Deploy 가이드

해당 파트는 자사의 ReportingServer와 DataServer제품을 따로 배포하는 상황에 대하여 다루고 있으며, 설정방식은 두 제품이 동일하기 때문에 ReportingServer의 배포에 대해서만 다루도록 한다. 제품별로 배포하는 경우에도 1.4파트의 최초접근과 관련한 설정이 되어야 정상동작이 가능하다.

### 4.1. war 배포

4.1.1. 커맨드 창을 열어 directory형태의 제품 폴더로 이동.

4.1.2. Jar cvf [파일이름].war \* 명령어를 실행.

```
관리자: 명령 프롬프트
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>d:

D:\>cd D:\ReportingServer

D:\ReportingServer>
D:\ReportingServer>jar cvf ReportingServer.war *
Manifest를 추가함
추가하는 중: bin/(입력 = 0) (출력 = 0)(0%를 저장함)
```

4.1.3. 같은 경로에 생성된 war파일을 WAS에서 설정한 배포경로로 복사  
(테스트시 경로 : tomcat/webapps)

4.1.4. WAS 기동.

### 4.2. Exploded 배포

4.2.1. war배포와 동일한 경로에 war 압축 전 Directory를 그대로 카피한 뒤 WAS기동.