

WebJars 사용

프로젝트명 : B12WebJars


준비사항

- 의존설정 : Spring Web, Lombok
- JSP 사용을 위한 설정을 한다.
- Refresh Gradle Project 를 눌러 적용한다.


WebJars란..??

CDN 방식을 주로 사용하는 Bootstrap 이나 jQuery 와 같은 웹 라이브러리를 Jar 파일로 패키징하여 메이븐(Maven)이나 그레이들(Gradle)을 통해 프로젝트에 포함할 수 있게 해준다.

URL : <https://www.webjars.org/>

 **WebJars** Documentation

WebJars are client-side web libraries (e.g. jQuery & Bootstrap) packaged into JAR (Java Archive) files.

- Explicitly and easily manage the client-side dependencies in JVM-based web applications
- Use JVM-based build tools (e.g. Maven, Gradle, sbt, ...) to download your client-side dependencies
- Know which client-side dependencies you are using
- Transitive dependencies are automatically resolved and optionally loaded via RequireJS
- Deployed on [Maven Central](#)
- Public CDN, generously provided by:  **JSDELIVR**

WebJars come in four flavors:

NPM WebJars

- Contents mirror NPM package
- GroupId: `org.webjars.npm`
- ArtifactId: NPM Package or URL-based Name

Bower GitHub WebJars

- Contents mirror Bower package
- GroupId: `org.webjars.bowergithub.[GITHUB_ORG]`
- ArtifactId: GitHub Repo Name

검색하기

Classic 만 체크한 후 라이브러리를 검색한다. 빌드툴은 Gradle을 선택한다.

Search Results

☐ NPM ☐ Bower GitHub ☐ Bower Original ☒ Classic

Add a WebJar

Name	Versions	Build Tool:	Files
Alpaca - Easy Forms for jQuery	1.5.24	<input type="button" value="SBT / Play 2"/> <input type="button" value="Maven"/> <input type="button" value="Ivy"/> <input checked="" type="button" value="Gradle"/> <input type="button" value="Buildr"/> <input type="button" value="Leiningen"/>	6428 Files
Backstretch	2.1.16	compile 'org.webjars:jquery-backstretch:2.1.16'	2 Files
blockui	2.70-1	compile 'org.webjars:jquery-blockui:2.70-1'	6 Files

우리는 jquery, bootstrap 을 검색한다.

검색된 결과에서 그레이들에 추가할 내용을 복사해 둔다.

jquery	3.7.1	+	compile 'org.webjars:jquery:3.7.1'	7 Files
Bootstrap	5.3.2	+	compile 'org.webjars:bootstrap:5.3.2'	296 Files

compile 은 implementation 으로 수정하면 된다.

```
39     implementation 'org.apache.tomcat.embed:tomcat-embed-jasper'
40
41     implementation 'org.webjars:jquery:3.7.1'
42     implementation 'org.webjars:bootstrap:5.3.2'
43
44 }
```

저장한 후 Refresh 한다.

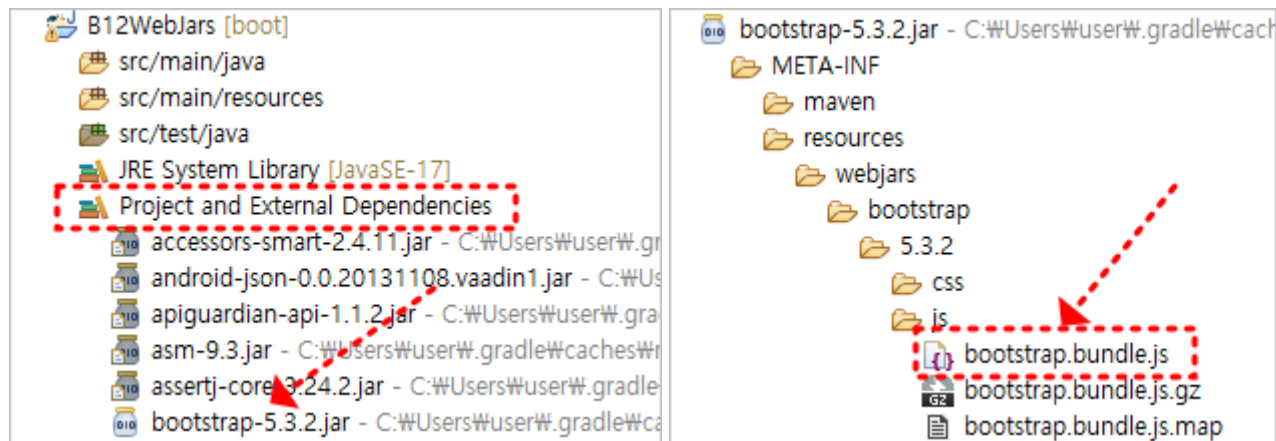
뷰 : webapp/WEB-INF/views/home.jsp

<head> 영역에 이와같이 추가하면 된다.

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Insert title here</title>
8 <link rel="stylesheet" href="/webjars/bootstrap/5.3.2/css/bootstrap.css" />
9 <script src="/webjars/bootstrap/5.3.2/js/bootstrap.bundle.js"></script>
10 <script src="/webjars/jquery/3.7.1/jquery.js"></script>
11 </head>
```

/webjars 로 시작하는 경로는 아래와 같이 찾을 수 있다.

프로젝트 ⇒ Project and External Dependencies 하위를 클릭한다.



필요한 라이브러리 파일을 찾아 우클릭 ⇒ Copy Qualified Name 으로 경로를 복사한다.
붙여넣기 하면 아래와 같은데 /webjars 부터 사용하면 된다.

```
/META-INF/resources/webjars/bootstrap/5.3.2/js/bootstrap.bundle.js
```

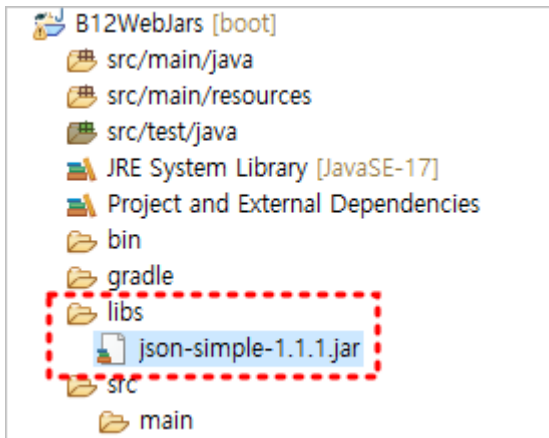
```
12<body>
13  <h2>Spring boot Project 세팅하기</h2>
14  <ul>
15    <li><a href="/">루트</a></li>
16  </ul>
17
18  <h2>Webjars - 부트스트랩</h2>
19  <button type="button" class="btn">Basic</button>
20  <button type="button" class="btn btn-primary">Primary</button>
21  <button type="button" class="btn btn-secondary">Secondary</button>
22  <button type="button" class="btn btn-success">Success</button>
23  <button type="button" class="btn btn-info">Info</button>
24  <button type="button" class="btn btn-warning">Warning</button>
25  <button type="button" class="btn btn-danger">Danger</button>
26  <button type="button" class="btn btn-dark">Dark</button>
27  <button type="button" class="btn btn-light">Light</button>
28  <button type="button" class="btn btn-link">Link</button>
29
30  <h2>Webjars - jQuery</h2>
31  <button type="button" id="myBtn" class="btn btn-warning">
32    클릭하세요
33  </button>
34  <script>
35    $(function(){
36      $('#myBtn').click(function(){
37        alert("jQuery 동작하나요?");
38      });
39    });
40  </script>
41 </body>
42 </html>
```

여기까지 작성하세요.

외부 라이브러리 사용하기

json-simple 라이브러리 다운로드 [[메이븐저장소](#)]

다운로드 한 파일은 프로젝트에 libs 폴더를 만든 후 복사한다.



그레이들 : build.gradle

```
41     implementation 'org.webjars:jquery:3.7.1'
42     implementation 'org.webjars:bootstrap:5.3.2'
43
44     implementation fileTree(dir: 'libs', include: ['*.jar'])
45 }
46
```

libs 폴더에 있는 모든 jar 파일을 라이브러리로 사용하겠다는 설정

뷰 : webapp/WEB-INF/views/home.jsp

```
12<body>
13    <h2>Spring boot Project 세팅하기</h2>
14    <ul>
15        <li><a href="/">루트</a></li>
16        <li><a href="/json.do">simple-json 라이브러리 사용하기</a></li>
17    </ul>
18
```

컨트롤러 : com.edu.springboot.MainController.java

```
23@RequestMapping("/json.do")
24@ResponseBody
25public String json() {
26    JSONArray arr = new JSONArray();
27    JSONObject obj = new JSONObject();
28
29    arr.add("손오공");
30    arr.add("저팔계");
31    arr.add("사오정");

```

```

33         obj.put("서유기", arr);
34         obj.put("result", 1);
35
36         return obj.toJSONString();
37     }
38 }

```

퀴즈]

다음과 같은 JSON 객체를 출력하는 프로그램을 작성하시오.

```

{
    "name":"홍길동",
    "age":99,
    "hobby":["자전거","수영","축구"],
    "friend":{
        "mid":["손오공","저팔계","사오정"],
        "high":{
            "circle":["유비","관우","장비"],
            "class":["이몽룡","성춘향"]
        }
    }
}

```

- 요청명 : jsonQuiz.do
- 컨트롤러 : com.edu.springboot.MainController.java

정답 [[바로가기](#)]

```
50 @RequestMapping("/jsonQuiz.do")
51 @ResponseBody
52 public String jsonQuiz() {
53     JSONObject objResult = new JSONObject();
54
55     //취미
56     JSONArray arr1 = new JSONArray();
57     arr1.add("자전거"); arr1.add("수영"); arr1.add("축구");
58
59     //중딩친구
60     JSONArray arrMidFriend = new JSONArray();
61     arrMidFriend.add("손오공"); arrMidFriend.add("저팔계");
62     arrMidFriend.add("사오정");
63
```

```
64     //고딩친구
65     JSONArray arrCircle = new JSONArray();
66     arrCircle.add("유비"); arrCircle.add("관우");
67     arrCircle.add("장비");
68     JSONArray arrClass = new JSONArray();
69     arrClass.add("이몽룡"); arrClass.add("성춘향");
70     JSONObject objHighFriend = new JSONObject();
71     objHighFriend.put("circle", arrCircle);
72     objHighFriend.put("class", arrClass);
73
74     //중딩,고딩통합
75     JSONObject objFriend = new JSONObject();
76     objFriend.put("mid", arrMidFriend);
77     objFriend.put("high", objHighFriend);
```

```
79     //결과
80     objResult.put("name", "홍길동");
81     objResult.put("age", 99);
82     objResult.put("hobby", arr1);
83     objResult.put("friend", objFriend);
84     return objResult.toJSONString();
85 }
86 }
```