

# 애플리케이션 배포

## 1. WAS 설치

1) <http://tomcat.apache.org/> 에서 톰캣 8.5버전 다운로드

The screenshot shows the Apache Tomcat 8.5.46 download page. The left sidebar contains navigation links for Download, Documentation, Problems?, and Get Involved. The 'Download' section lists various versions, with 'Tomcat 8' highlighted. The 'Binary Distributions' section lists download links for different operating systems and architectures, with the '64-bit Windows zip (pgp, sha512)' link highlighted.

Quick Navigation  
[KEYS](#) | [8.5.46](#) | [Browse](#) | [Archives](#)

Release Integrity  
You **must** [verify](#) the integrity of the downloaded files. We provide OpenPGP keys of Tomcat's Release Managers. We also provide a checksum for your download, and make sure it is the same as ours.

Mirrors  
You are currently using <http://mirror.navercorp.com/apache/>. If you are using backup mirrors (at the end of the mirrors list) that should be available.

Other mirrors:  [Change](#)

8.5.46  
Please see the [README](#) file for packaging information. It explains why we use this version.

Binary Distributions

- Core:
  - [zip \(pgp, sha512\)](#)
  - [tar.gz \(pgp, sha512\)](#)
  - [32-bit Windows zip \(pgp, sha512\)](#)
  - [64-bit Windows zip \(pgp, sha512\)](#)
  - [32-bit/64-bit Windows Service Installer \(pgp, sha512\)](#)
- Full documentation:
  - [tar.gz \(pgp, sha512\)](#)
- Deployer:

2) C 드라이브에 server폴더 생성 후, 다운 받은 톰캣 파일 압축해제

The screenshot shows a Windows File Explorer window with the path 'C:\server\apache-tomcat-8.5.46'. The folder contains several subfolders and files. The 'LICENSE' file is selected.

이름	수정된 날짜	유형	크기
bin	2019-10-04 오전...	파일 폴더	
conf	2019-10-04 오전...	파일 폴더	
lib	2019-10-04 오전...	파일 폴더	
logs	2019-10-04 오전...	파일 폴더	
temp	2019-10-04 오전...	파일 폴더	
webapps	2019-10-04 오전...	파일 폴더	
work	2019-10-04 오전...	파일 폴더	
BUILDING	2019-09-16 오후...	텍스트 문서	20KB
CONTRIBUTING.md	2019-09-16 오후...	MD 파일	6KB
LICENSE	2019-09-16 오후...	파일	57KB
NOTICE	2019-09-16 오후...	파일	2KB
README.md	2019-09-16 오후...	MD 파일	4KB
RELEASE-NOTES	2019-09-16 오후...	파일	8KB
RUNNING	2019-09-16 오후...	텍스트 문서	17KB

## 2. 서버 배포 환경 구축

- 1) C:\server\apache-tomcat-8.5.46\conf\server.xml 파일에 포트번호 수정  
(vscode를 활용하여 server.xml 오픈 후 아래 포트를 찾아 포트번호 변경)


```
<Server port="8008" shutdown="SHUTDOWN">
```

```
<Connector port="8899" protocol="HTTP/1.1"  
            connectionTimeout="20000"  
            redirectPort="8443" />
```

```
<Connector port="8012" protocol="AJP/1.3" redirectPort="8443" />
```

## 3. WAS 실행

- 1) 명령 프롬프트(cmd)를 관리자 권한으로 실행
- 2) 설치한 톰캣 폴더 하위에 bin 폴더로 이동

 관리자: 명령 프롬프트

```
Microsoft Windows [Version 10.0.17763.737]  
(c) 2018 Microsoft Corporation. All rights reserved.  
  
C:\windows\system32>cd c:\server\apache-tomcat-8.5.46\bin  
c:\server\apache-tomcat-8.5.46\bin>_
```

- 3) startup.bat 실행하여 톰캣 실행(실행 상태 계속 유지할 것!)

```
c:\server\apache-tomcat-8.5.46\bin>startup.bat  
Using CATALINA_BASE:   "c:\server\apache-tomcat-8.5.46"  
Using CATALINA_HOME:   "c:\server\apache-tomcat-8.5.46"  
Using CATALINA_TMPDIR: "c:\server\apache-tomcat-8.5.46\temp"  
Using JRE_HOME:        "C:\Program Files\Java\jre1.8.0_212"  
Using CLASSPATH:       "c:\server\apache-tomcat-8.5.46\bin\bootstrap.jar;c:  
:\server\apache-tomcat-8.5.46\bin\tomcat-juli.jar"
```

```
선택 Tomcat  
04-Oct-2019 17:05:35.837 ???? [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory ??????  
???  
04-Oct-2019 17:05:35.852 ???? [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory ??????  
???  
04-Oct-2019 17:05:35.852 ???? [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory ??????  
???  
04-Oct-2019 17:05:35.858 ???? [main] org.apache.coyote.AbstractProtocol.start ????  
???  
04-Oct-2019 17:05:35.874 ???? [main] org.apache.coyote.AbstractProtocol.start ????  
???  
04-Oct-2019 17:05:35.874 ???? [main] org.apache.catalina.startup.Catalina.start Server startup in 1603 ms
```

#### 4. Build 용 도구 Maven 설치

- 1) <https://maven.apache.org/> 에서 Maven 최신버전 다운로드

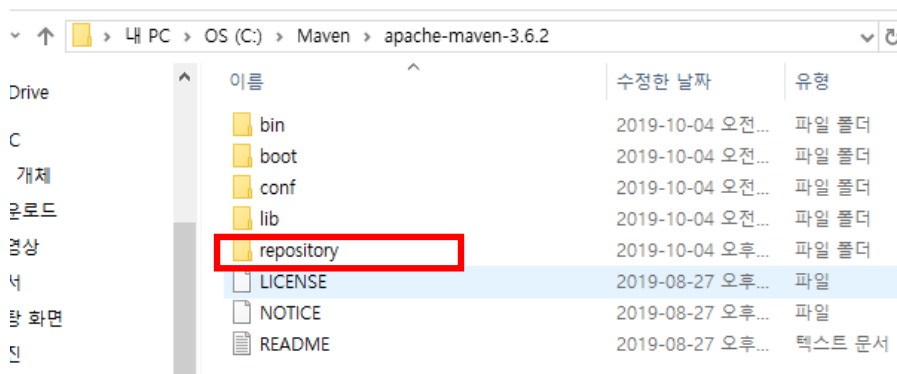


The screenshot shows the Apache Maven Project homepage. The 'Download, Install, Run Maven' link is highlighted with a red box. Below the link, there is a table of links and checksums for various Maven distributions.

	Link	Checksums
Binary tar.gz archive	<a href="#">apache-maven-3.6.2-bin.tar.gz</a>	<a href="#">apache-maven-3.6.2-bin.tar.gz.sha512</a>
Binary zip archive	<a href="#">apache-maven-3.6.2-bin.zip</a>	<a href="#">apache-maven-3.6.2-bin.zip.sha512</a>
Source tar.gz archive	<a href="#">apache-maven-3.6.2-src.tar.gz</a>	<a href="#">apache-maven-3.6.2-src.tar.gz.sha512</a>
Source zip archive	<a href="#">apache-maven-3.6.2-src.zip</a>	<a href="#">apache-maven-3.6.2-src.zip.sha512</a>

Below the table, there is a link to [Release Notes](#).

- 2) C 드라이브에 Maven 폴더 생성 후, 해당 폴더에 압축 해제하고  
내부에 Repository 폴더 생성



The screenshot shows a Windows File Explorer window with the path 'C:\Maven\apache-maven-3.6.2'. The 'repository' folder is highlighted with a red box. The contents of the folder are listed in the table below.

이름	수정된 날짜	유형
bin	2019-10-04 오전...	파일 폴더
boot	2019-10-04 오전...	파일 폴더
conf	2019-10-04 오전...	파일 폴더
lib	2019-10-04 오전...	파일 폴더
repository	2019-10-04 오후...	파일 폴더
LICENSE	2019-08-27 오후...	파일
NOTICE	2019-08-27 오후...	파일
README	2019-08-27 오후...	텍스트 문서

3) C:\Maven\apache-maven-3.6.2\conf\settings.xml 폴더를 열어

53 번째 줄

<localRepository>/path/to/local/repo</localRepository> 를 복사하여

바로 밑 주석 밖에 붙여넣기 후 내부 경로 값을

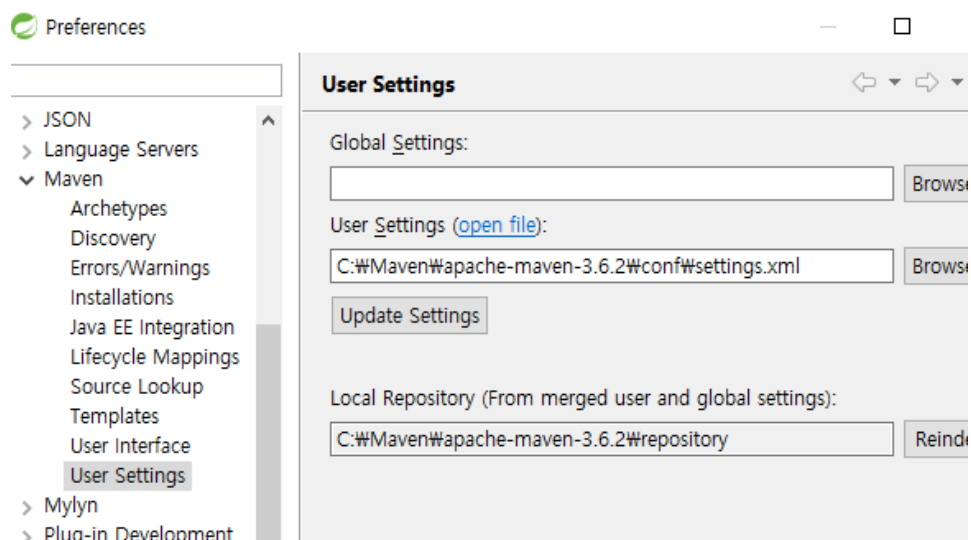
생성한 repository 폴더 경로(C:\Maven\apache-maven-3.6.2\repository)로 변경

```
46 <settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
47         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
48         xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.a
49 <!-- localRepository
50 | The path to the local repository maven will use to store artifacts.
51 |
52 | Default: ${user.home}/.m2/repository
53 <localRepository>/path/to/local/repo</localRepository>
54 -->
55 <localRepository>C:\Maven\apache-maven-3.6.2\repository</localRepository>
56
```

4) STS 실행 후 Maven Project 의 repository 경로 수정

- Window → Preferences → Maven → User Settings

→ User Settings 의 값을 3)에서 수정 settings.xml 로 변경



## 5. Spring Legacy Project 생성

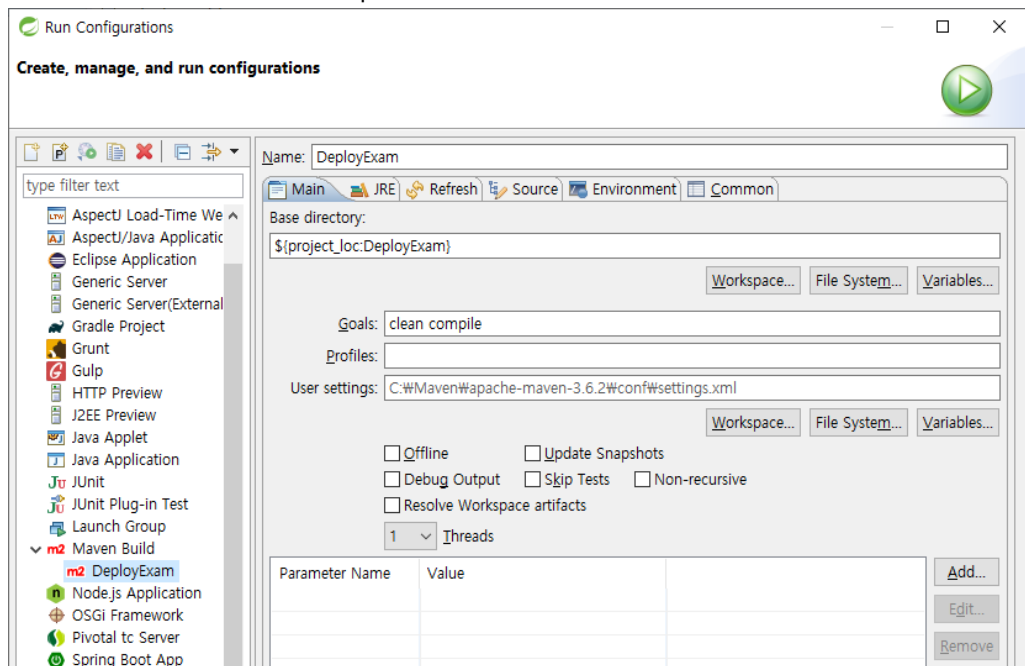
- 1) 프로젝트명 : DeployExam
- 2) 패키지명 : com.kh.deployExam

## 6. src/main/webapp 폴더에 DeployExam.jsp 파일 생성 후 다음 코드 작성

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<script type="text/javascript">
    function show(){
        var value = document.getElementById("btn").innerText;
        alert(value);
    }
</script>
</head>
<body>
    <button id="btn" onclick="show();">Already Clicked!</button>
</body>
</html>
```

## 7. 생성한 프로젝트를 빌드하여 문제 없는지 확인

- 1) 프로젝트 우클릭 → Run as → Run Configuration → Maven Build 선택
- 2) Goals 입력상자에 clean compile 작성



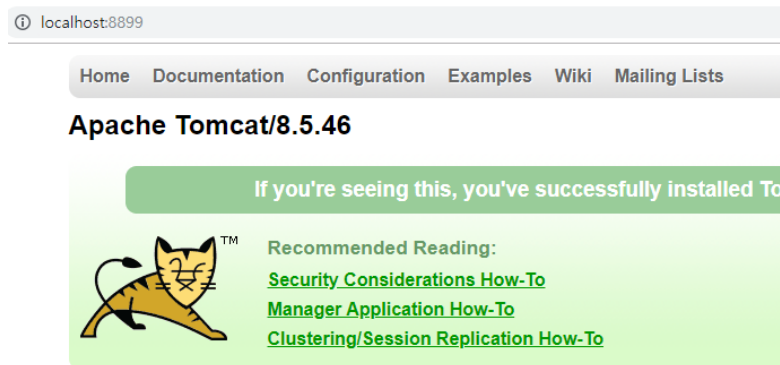
### 3) Apply 후 Run 실행

```
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ deployExam ---
[INFO] Deleting C:\workspace\ncsTest\DeployExam\target
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ deployExam ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent
[INFO] Copying 1 resource
[INFO] --- maven-compiler-plugin:2.5.1:compile (default-compile) @ deployExam ---
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 1 source file to C:\workspace\ncsTest\DeployExam\target\classes
[INFO] BUILD SUCCESS
[INFO] Total time: 0.764 s
[INFO] Finished at: 2019-10-04T17:30:14+09:00
[INFO]
```

## 8. 배포도구(톰캣)를 활용하여 서버 작동 여부 확인

### 1) 톰캣 정상 작동 여부 확인을 위하여 브라우저 주소창에

<http://localhost:8899/> < 접속 확인



## 9. 빌드가 완료된 프로젝트를 Maven 을 이용해 배포

### 1) C:\server\apache-tomcat-8.5.46\conf\tomcat-users.xml 파일을 열어

다음 코드 추가

```
<role rolename="manager-script"/>
<role rolename="manager-gui"/>
<role rolename="manager-jmx"/>
<role rolename="manager-status"/>
<user username="admin" password="admin" roles="manager-
gui,manager-script,manager-status,manager-jmx"/>
```

```
44 | <role rolename="manager-script"/>
45 | <role rolename="manager-gui"/>
46 | <role rolename="manager-jmx"/>
47 | <role rolename="manager-status"/>
48 | <user username="admin" password="admin" roles="manager-gui,manager-script,manager-status,manager-jmx"/>
49 |
50 | </tomcat-users>
```

- 2) pom.xml 하단 부분 <plugins> 태그 내부에 maven 배포를 위한 플러그인 추가

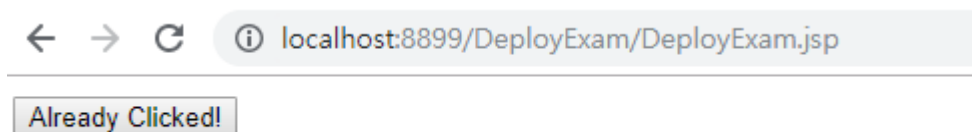
```
<plugin>
  <groupId>org.apache.tomcat.maven</groupId>
  <artifactId>tomcat7-maven-plugin</artifactId>
  <version>2.2</version>
  <configuration>
    <url>http://127.0.0.1:8899/manager/text</url>
    <username>admin</username>
    <password>admin</password>
    <path>/DeployExam</path>
  </configuration>
</plugin>
```

- 3) 프로젝트 우클릭 → Run as → Run Configuration → Maven Build 선택  
4) Goals 입력상자에 tomcat7:deploy 작성 후 Apply, Run 실행

```
[INFO] tomcatManager status code:200, ReasonPhrase:
[INFO] OK - 컨텍스트 경로 [/DeployExam]에 애플리케이션을 배치했습니다.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.753 s
[INFO] Finished at: 2019-10-04T17:41:15+09:00
[INFO] -----
```

10. 애플리케이션 배포 후 정상 작동 하는지 확인.

- 1) <http://localhost:8899/DeployExam/DeployExam.jsp> 으로 접속



- 2) 버튼을 클릭하여 알림창이 출력되는지 확인.

