

Introduction to JUnit, Eclipse, Build Environment

200611494 원스타
200810047 김성원
200811466 허태경

Index

1. Junit
2. Eclipse
3. Build Environment
 1. ANT
 2. Jenkins

1. Junit

The background features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the right side of the page, creating a modern, layered effect against the white background.

1. Junit

- ▶ Java를 위한 Unit Testing Framework
- ▶ 자동화된 테스트 도구
- ▶ 소스 코드의 특정 모듈이 의도된 대로 정확히 작동하는지 검증하는 작업
- ▶ Annotation을 통해 테스트 제어
- ▶ Method에 대한 테스트 케이스(Test case)를 작성하고 Test
- ▶ 문제를 초기에 발견할 수 있게 해준다

The logo for JUnit, featuring the letter 'J' in green and 'Unit' in red, both in a serif font.

1. JUnit



Overview ▾ Project Documentation ▾

JUnit

JUnit / About Version: 4.12 | Last Published: 2014-12-04

JUnit is a simple framework to write repeatable tests. It is an instance of the xUnit architecture for unit testing frameworks.

```
@Test
public void newArrayListsHaveNoElements() {
    assertThat(new ArrayList().size(), is(0));
}

@Test
public void sizeReturnsNumberOfElements() {
    List instance = new ArrayList();
    instance.add(new Object());
    instance.add(new Object());
    assertThat(instance.size(), is(2));
}
```

Annotations
Start by marking your tests with `@Test`.

[Let's take a tour »](#)

Welcome

- [Download and install](#)
- [Getting started](#)
- [Release Notes](#)
 - [4.12](#)
 - [4.11](#)
 - [4.10](#)
 - [4.9.1](#)

Usage and Idioms

- [Assertions](#)
- [Test Runners](#)
- [Aggregating tests in Suites](#)
- [Test Execution Order](#)
- [Exception Testing](#)
- [Matchers and assertThat](#)
- [Ignoring Tests](#)

Third-party extensions

- [Custom Runners](#)
- [net.trajano.commons.commons-testing-for-UtilityClassTestUtil](#) per #646
- [System Rules](#) – A collection of JUnit rules for testing code that uses java.lang.System.
- [JUnit Toolbox](#) - Provides runners for

<http://junit.org/>
(v4.12 사용)

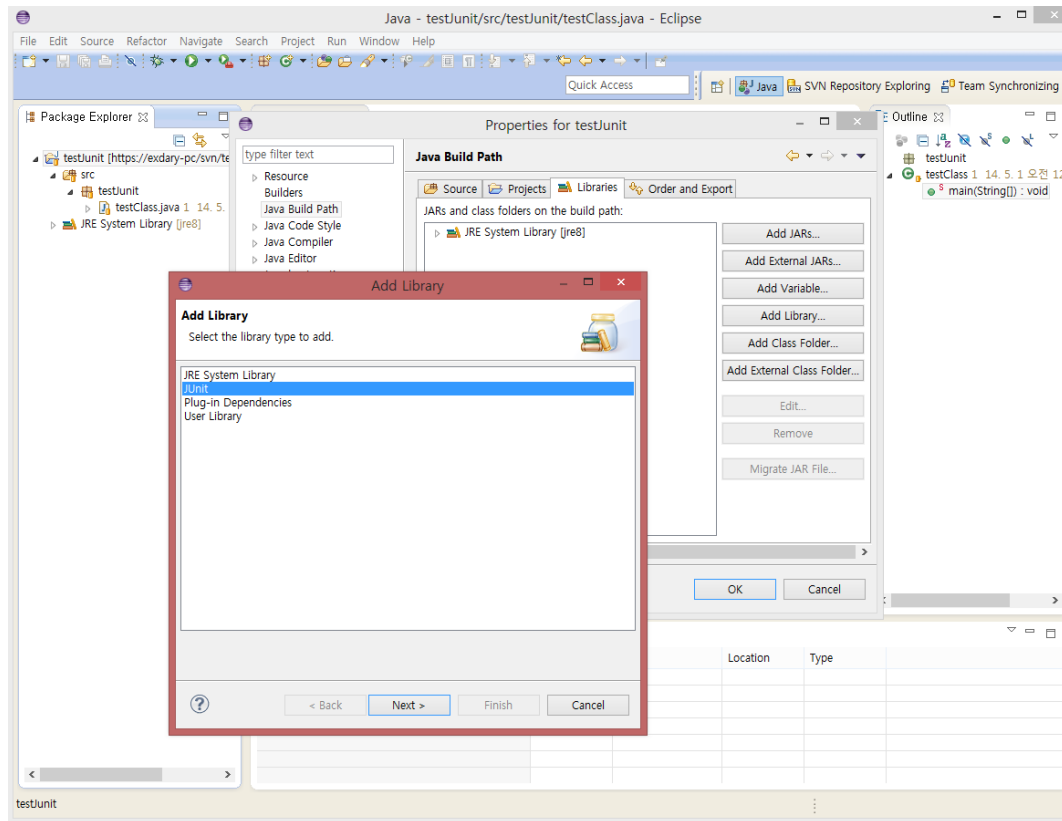
1. Junit

Annotation	Description
@BeforeClass	Class의 단위테스트 수행 전 작업
@AfterClass	Class의 단위테스트 수행 후 작업
@Before	Method의 단위테스트 수행 전 작업
@After	Method의 단위테스트 수행 후 작업
@Ignore	단위 테스트를 하지 않을 Method
@Test	단위 테스트를 수행할 Method

1. Junit

Method	Description
<code>assertEquals</code>	기대값과 실제값이 동일한지 판단
<code>assertFalse</code> / <code>assertTrue</code>	실제값이 거짓/참인지 판단
<code>assertNotNull</code> / <code>assertNull</code>	실제값이 <code>Null</code> 또는 <code>Null</code> 이 아닌지 판단
<code>assertNotSame</code> / <code>assertSame</code>	두 객체가 동일한 객체 또는 다른 객체인지 판단
<code>fail</code>	실패처리

1. Junit



Add Junit Library

1. Junit

The screenshot shows the Eclipse IDE with a 'New JUnit Test Case' dialog box open. The background window displays the Package Explorer on the left and the source code of 'testClass.java' in the center. The source code is as follows:

```
package testJUnit;  
  
public class testClass {  
    public static void main(String[] args){  
    }  
}
```

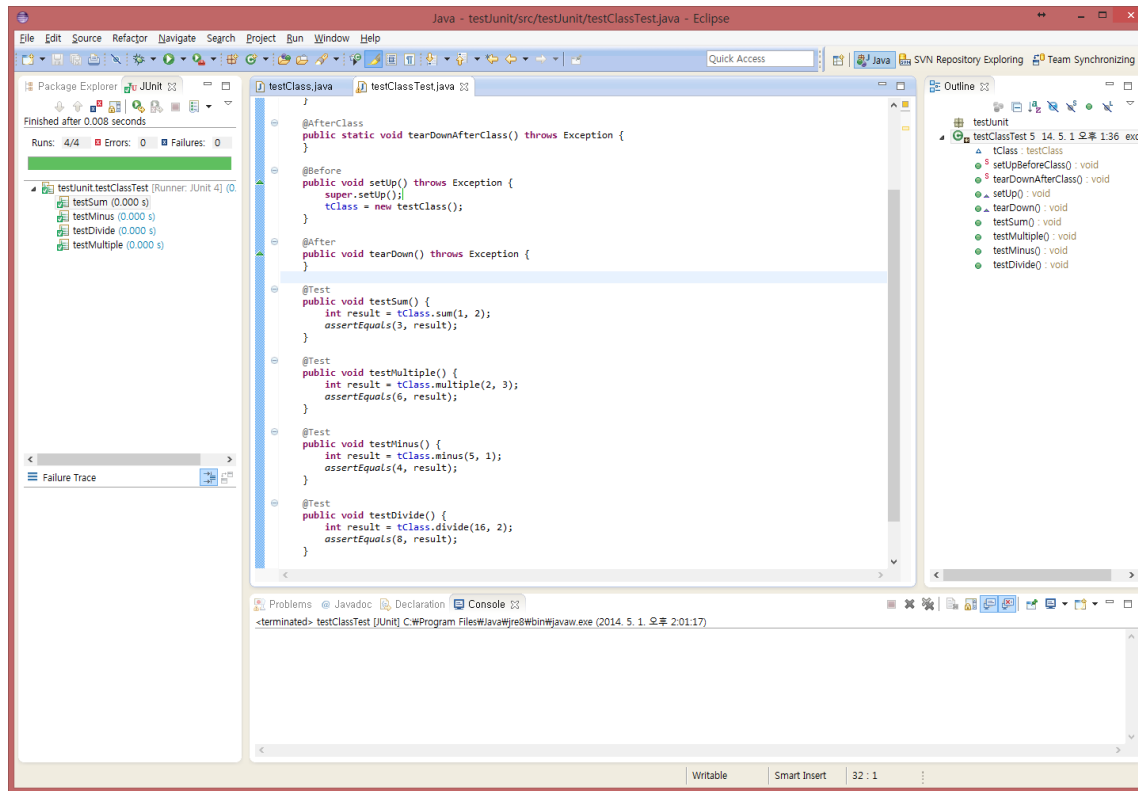
The 'New JUnit Test Case' dialog box contains the following fields and options:

- JUnit Test Case**: Select the name of the new JUnit test case. You have the options to specify the class under test and on the next page, to select methods to be tested.
- New JUnit 3 test New JUnit 4 test
- Source folder: testJUnit/src (Browse...)
- Package: testJUnit (Browse...)
- Name: testClassTest
- Superclass: java.lang.Object (Browse...)
- Which method stubs would you like to create?
 - setUpBeforeClass() tearDownAfterClass()
 - setUp() tearDown()
 - constructor
- Do you want to add comments? (Configure templates and default value [here](#))
 - Generate comments
- Class under test: testJUnit.testClass (Browse...)

At the bottom of the dialog, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'.

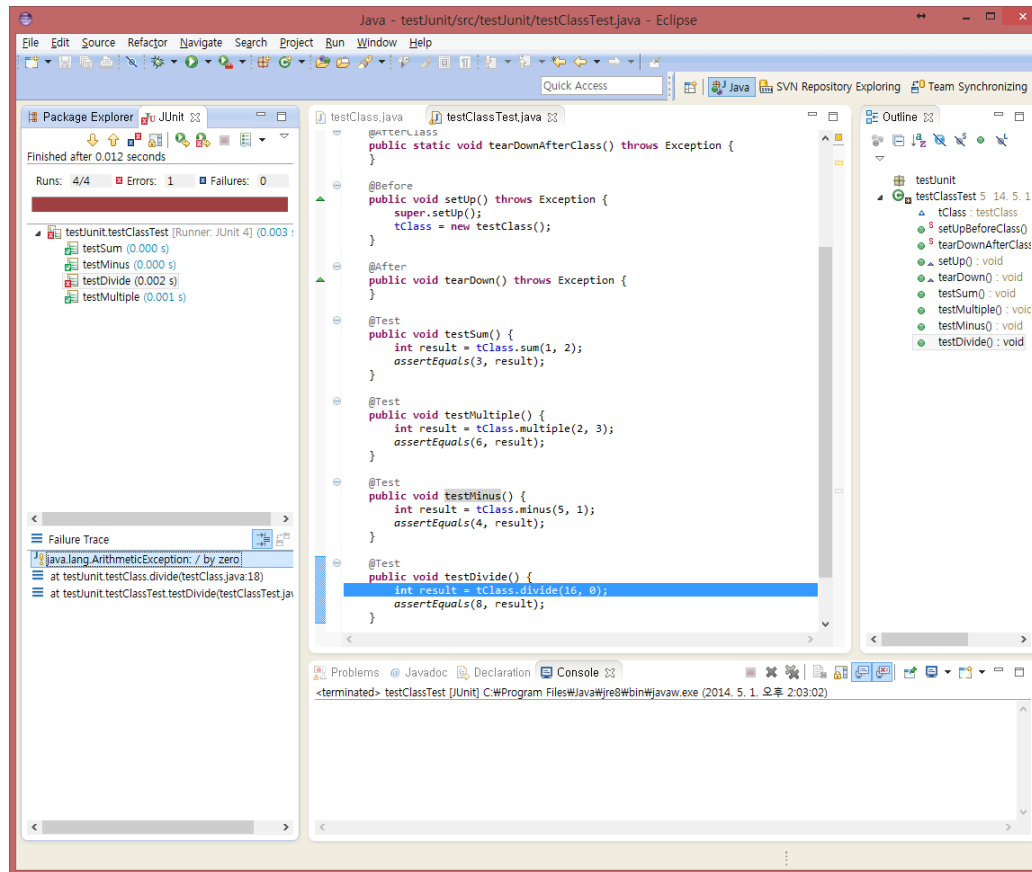
Create Junit Testcase Class

1. Junit



Success Case

1. Junit



Fail Case

2. Eclipse

The background features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the right side of the slide, creating a modern, layered effect against the white background.










2. Eclipse

- ▶ 오픈 소스 IDE
- ▶ 비영리 공개 소프트웨어 커뮤니티에서 제공되는 개방형 개발 플랫폼
- ▶ JAVA, Android, C, C++ 등등 다양한 언어를 지원하며 플러그인을 통해 추가적인 기능을 얻을 수 있다.



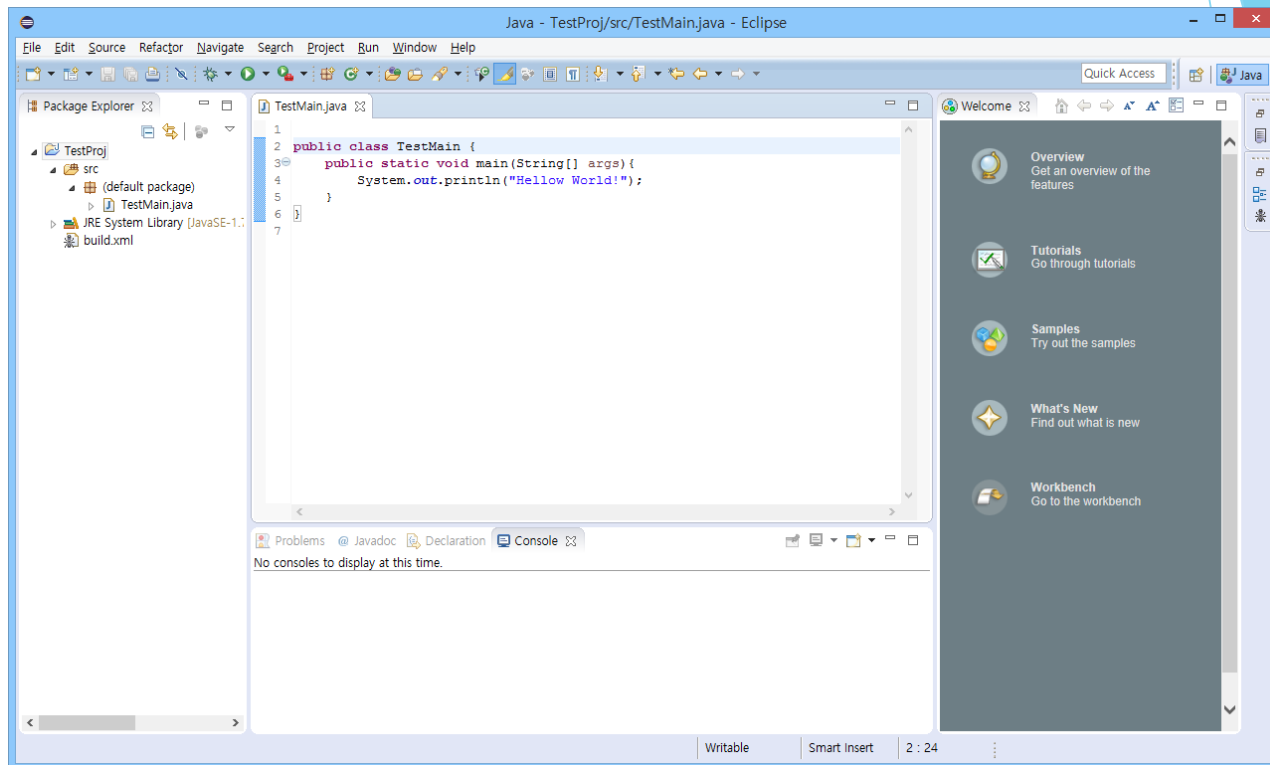
2. Eclipse

Eclipse Luna SR2 (4.4.2) Release for Windows

	Eclipse IDE for Java Developers , 155 MB Downloaded 763,032 Times The essential tools for any Java developer, including a Java IDE, a CVS client, Git client, XML Editor, Mylyn, Maven integration...		Windows 32 Bit Windows 64 Bit
<hr/>			
Filter Packages			
	Eclipse IDE for Java EE Developers , 254 MB Downloaded 358,129 Times Tools for Java developers creating Java EE and Web applications, including a Java IDE, tools for Java EE, JPA, JSF, Mylyn...		Windows 32 Bit Windows 64 Bit
	Enterprise-Class BIRT Report Server (Free, Not a Trial) Instantly enhance your Eclipse BIRT reports with interactive, real-time analytics that can be seamlessly embedded into web and mobile apps.		 Promoted Download
	Eclipse IDE for C/C++ Developers , 165 MB Downloaded 98,876 Times An IDE for C/C++ developers with Mylyn integration.		Windows 32 Bit Windows 64 Bit

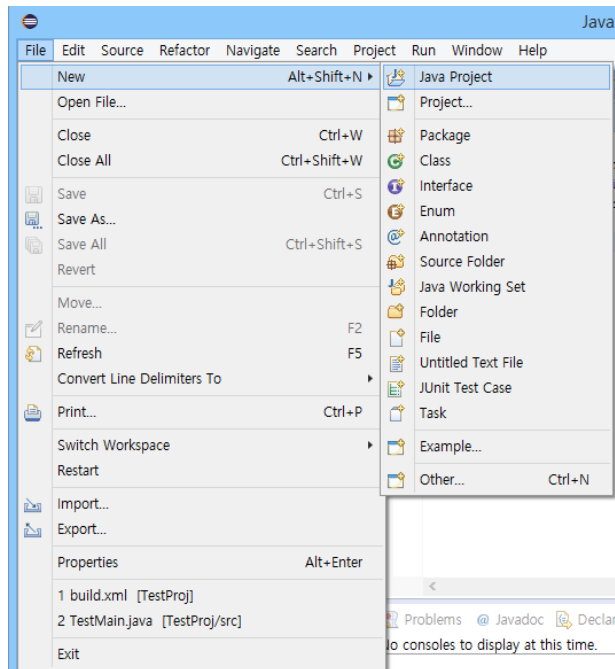
<http://www.eclipse.org/>
(Luna SR2 4.4.2 사용)

2. Eclipse



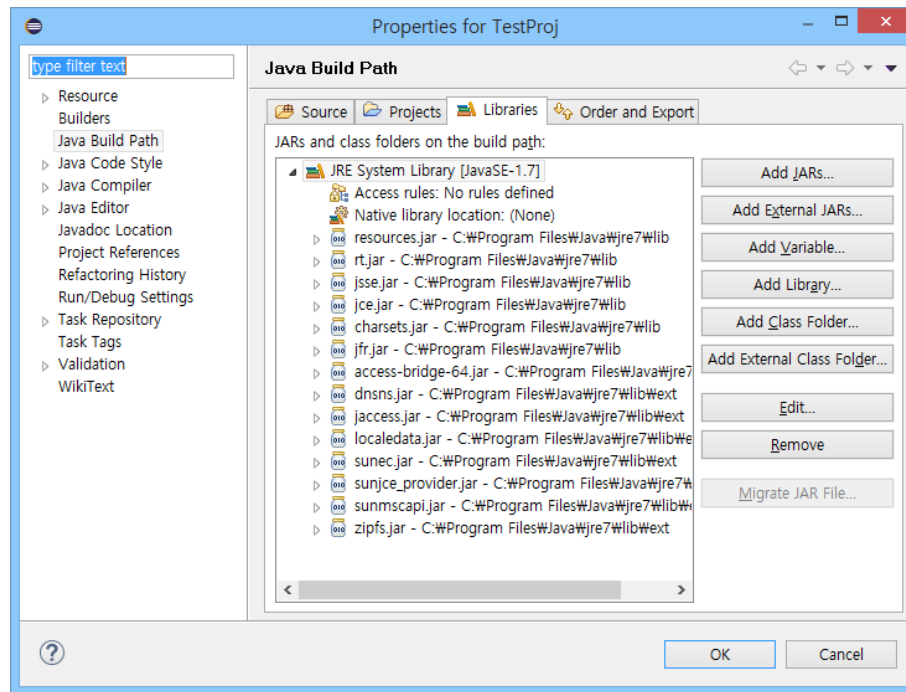
기본 레이아웃

2. Eclipse



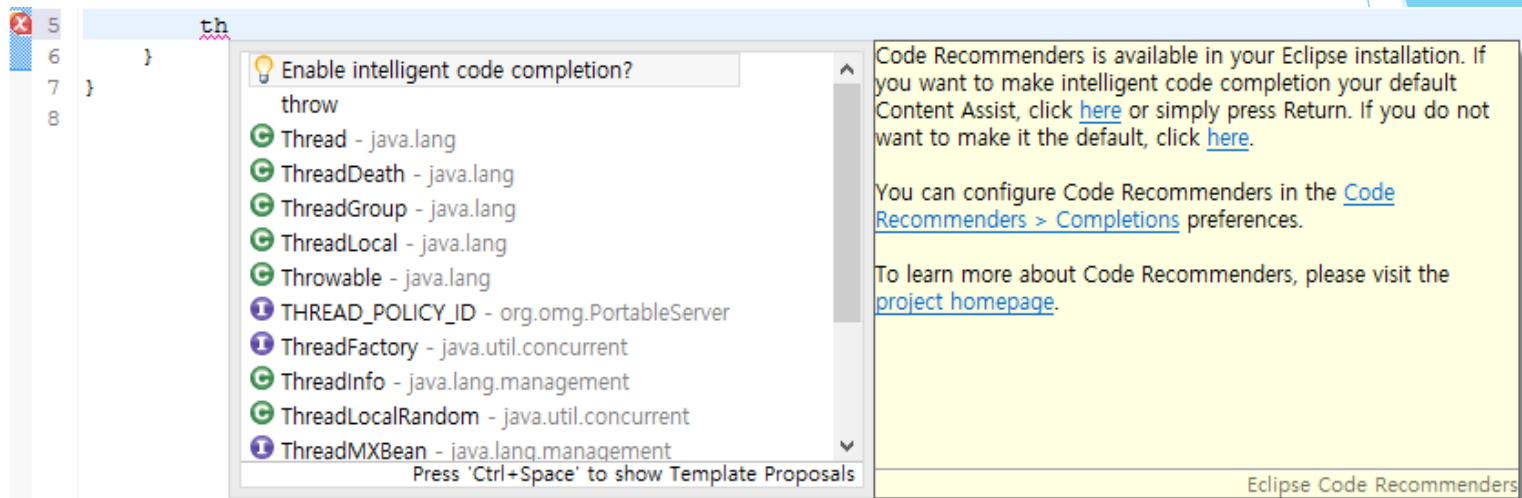
프로젝트 생성 및 Class, interface, Annotation 등 개발에 필요한 요소 생성 가능

2. Eclipse



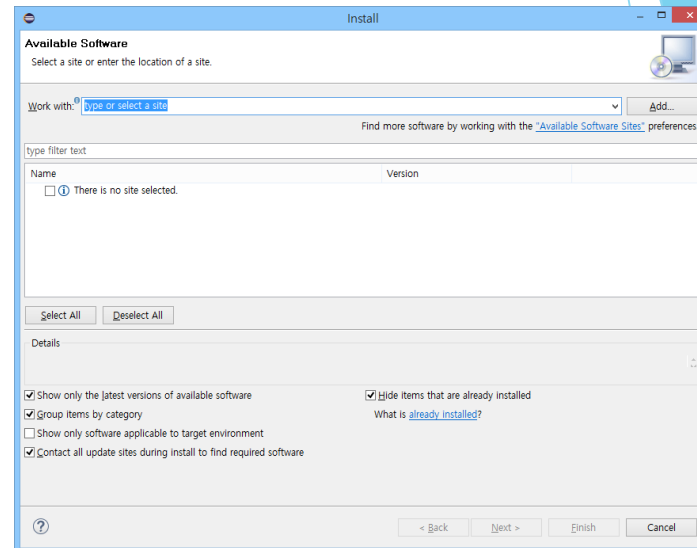
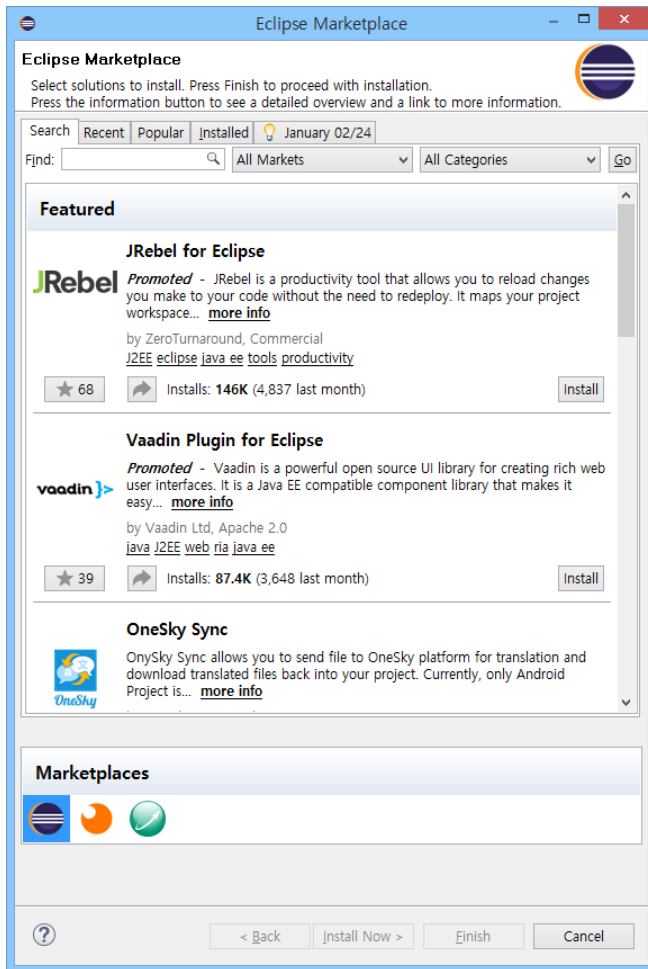
Build Path 수정을 통해 외부 라이브러리 참조 가능

2. Eclipse



Quick Fix를 통한 Code Intelligence 지원

2. Eclipse



Marketplace와 url을 통한
plug-in 설치 지원

3. Build Environment

ANT, Jenkins

3. Build Environment

ANT

- ▶ Another Neat Tool의 약자
- ▶ Apache Software재단에서 만든 JAVA에서 사용되는 자동 소프트웨어 빌드 도구.
- ▶ XML을 통해 환경구성을 함.
- ▶ JVM 위에서 동작.
- ▶ 배포와 유닛 테스트 기능까지 포함.
- ▶ Eclipse에 기본 내장.



3. Build Environment

ANT

Current Release of Ant

Currently, Apache Ant 1.9.4 is the best available version, see the [release notes](#).

Note

Ant 1.9.4 was released on 05-May-2014 and may not be available on all mirrors for a few days.

Tar files may require gnu tar to extract

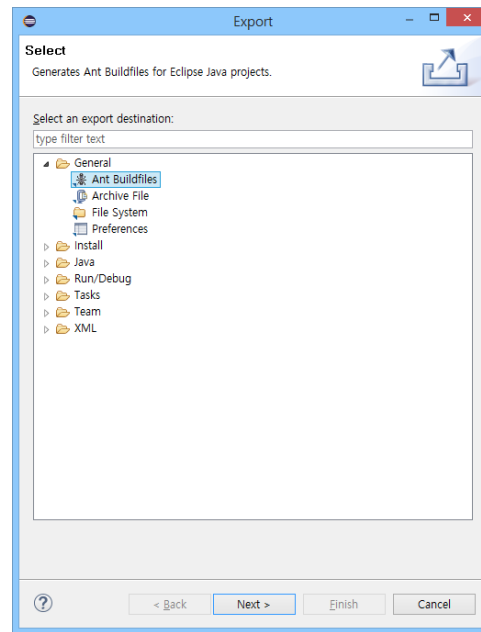
Tar files in the distribution contain long file names, and may require gnu tar to do the extraction.

- .zip archive: [apache-ant-1.9.4-bin.zip](#) [PGP] [SHA1] [SHA512] [MD5]
- .tar.gz archive: [apache-ant-1.9.4-bin.tar.gz](#) [PGP] [SHA1] [SHA512] [MD5]
- .tar.bz2 archive: [apache-ant-1.9.4-bin.tar.bz2](#) [PGP] [SHA1] [SHA512] [MD5]

<http://ant.apache.org/>
(v1.9.4 사용)

3. Build Environment

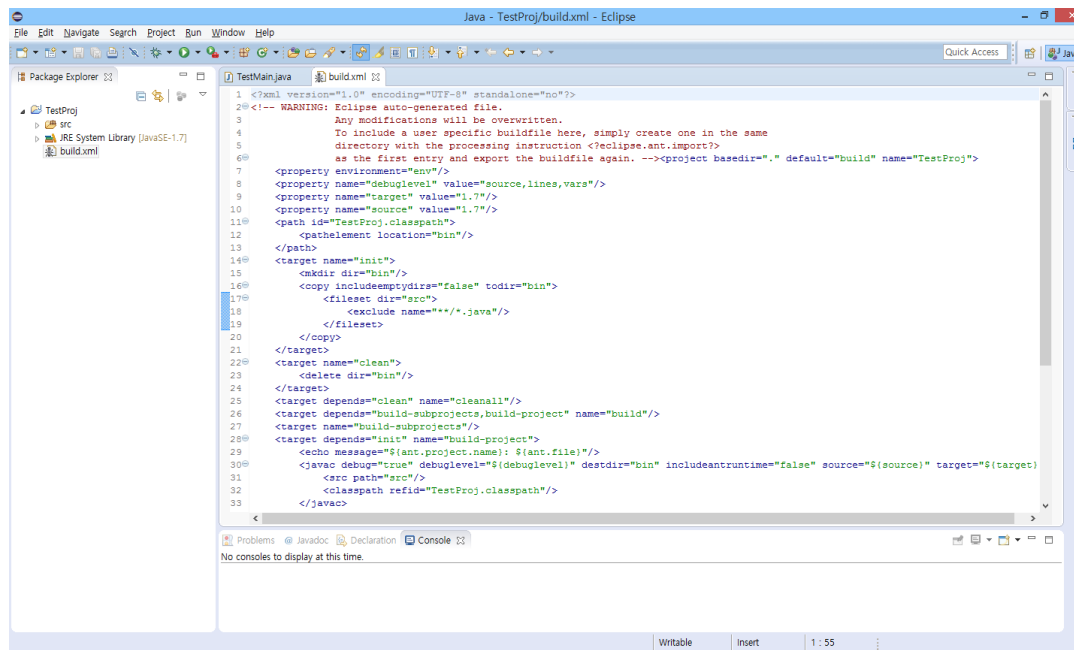
ANT



Export를 통해 Build.xml을 생성할 수 있음

3. Build Environment

ANT

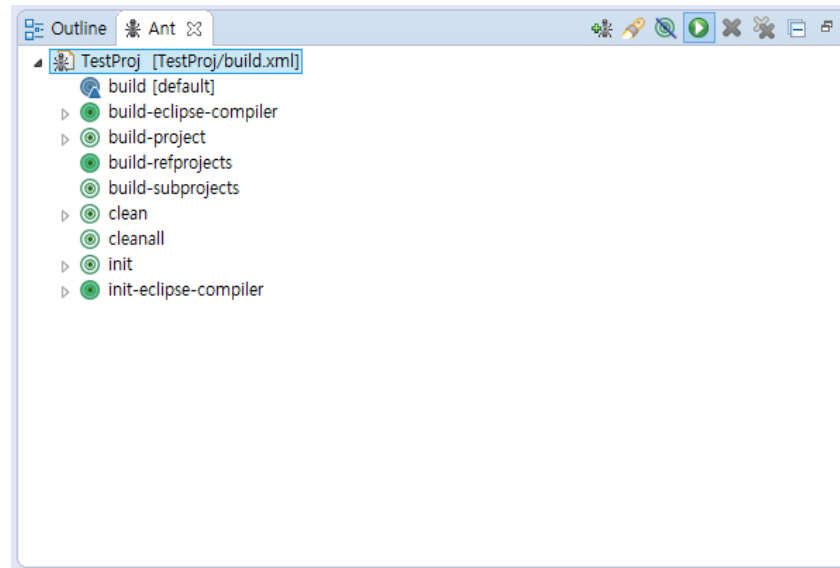


```
1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2@ <!-- WARNING: Eclipse auto-generated file.
3 Any modifications will be overwritten.
4 To include a user specific buildfile here, simply create one in the same
5 directory with the processing instruction <!doctype eclipse.ant.import>
6 as the first entry and export the buildfile again. --><project basedir="." default="build" name="TestProj">
7 <property environment="env"/>
8 <property name="debuglevel" value="source,lines,vars"/>
9 <property name="target" value="1.7"/>
10 <property name="source" value="1.7"/>
11@ <path id="TestProj.classpath">
12 <pathelement location="bin"/>
13 </path>
14@ <target name="init">
15 <mkdir dir="bin"/>
16@ <copy includeemptydirs="false" todir="bin">
17@ <fileset dir="src">
18 <exclude name="**/*.java"/>
19 </fileset>
20 </copy>
21 </target>
22@ <target name="clean">
23 <delete dir="bin"/>
24 </target>
25 <target depends="clean" name="cleanall"/>
26 <target depends="build-subprojects,build-project" name="build"/>
27 <target name="build-subprojects"/>
28@ <target depends="init" name="build-project">
29 <echo message="${ant.project.name}: ${ant.file}"/>
30@ <javac debug="true" debuglevel="${debuglevel}" destdir="bin" includeantruntime="false" source="${source}" target="${target}"
31 <src path="src"/>
32 <classpath refid="TestProj.classpath"/>
33 </javac>
```

Build.xml을 통한 자동 빌드 환경 조정

3. Build Environment

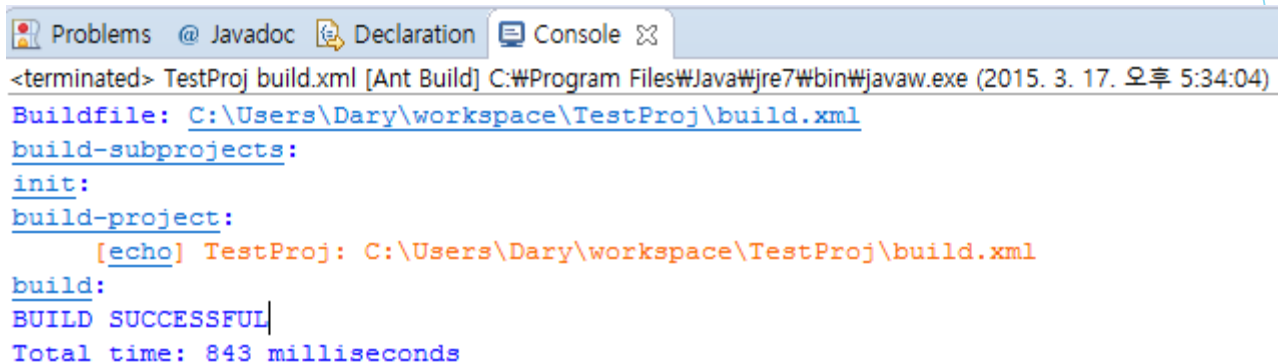
ANT



ANT 윈도우를 출력해 Build 가능

3. Build Environment

ANT



```
Problems @ Javadoc Declaration Console ✕
<terminated> TestProj build.xml [Ant Build] C:\Program Files\Java\jre7\bin\javaw.exe (2015. 3. 17. 오후 5:34:04)
Buildfile: C:\Users\Dary\workspace\TestProj\build.xml
build-subprojects:
init:
build-project:
    [echo] TestProj: C:\Users\Dary\workspace\TestProj\build.xml
build:
BUILD SUCCESSFUL
Total time: 843 milliseconds
```

ANT를 통한 Build 결과 출력물

3. Build Environment

Jenkins

- ▶ 오픈 소스 Continuous Integration Tool
- ▶ JAVA기반 빌드 자동화 서비스
- ▶ Hudson으로부터 파생 됨
- ▶ SVN, CVS, Git 등과 연결하여 소스 관리
- ▶ ANT, Maven을 통하여 Build
- ▶ Shell Script, Window Batch Commands를 통해 수행 지정 가능



Jenkins

3. Build Environment

Jenkins

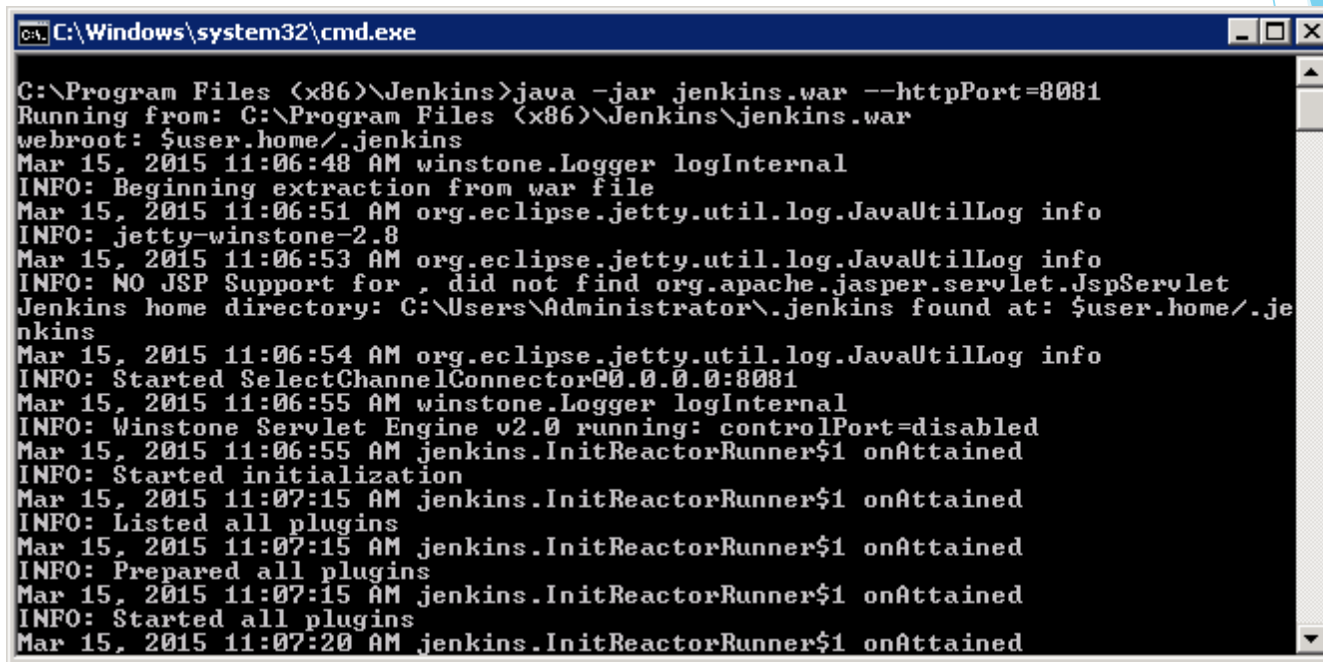


The screenshot shows the Jenkins website homepage. At the top left is the Jenkins logo, a cartoon character in a suit, followed by the word "Jenkins" in a large, bold, black font. Below the logo is the tagline "An extensible open source continuous integration server". To the right of the logo is a red ribbon with the text "Fork me on GitHub". Below the logo and tagline is a navigation bar with links: BLOG, CONNECT, BUG TRACKER, WIKI, CI, TUTORIALS, ARCHIVES, DONATION, ABOUT. Below the navigation bar is a banner celebrating the 100th anniversary of Jenkins, with the text "CELEBRATE 100K CELEBRATE" and a small image of the Jenkins character. Below the banner are three main sections: "Meet Jenkins" with a small image of the Jenkins character and the text "Find out what Jenkins is and get started.", "Use Jenkins" with an image of three people and the text "See how to get more out of your Jenkins.", and "Customize Jenkins" with a puzzle piece icon and the text "Choose from 1009 plugins to customize Jenkins exactly to your needs." To the right of these sections is a "Download Jenkins" section with tabs for "Release" and "Long-Term Support Release". Under "Release" is the "Java Web Archive (.war)" section, which includes the text "Latest and greatest (1.605)", "changelog | past releases | RC", and "upgrading from Hudson?". Below this is the "Or native package" section, which lists operating systems with their respective icons: Windows, Ubuntu/Debian, Red Hat/Fedora/CentOS, Mac OS X, openSUSE, and FreeBSD.

<http://jenkins-ci.org/>

3. Build Environment

Jenkins

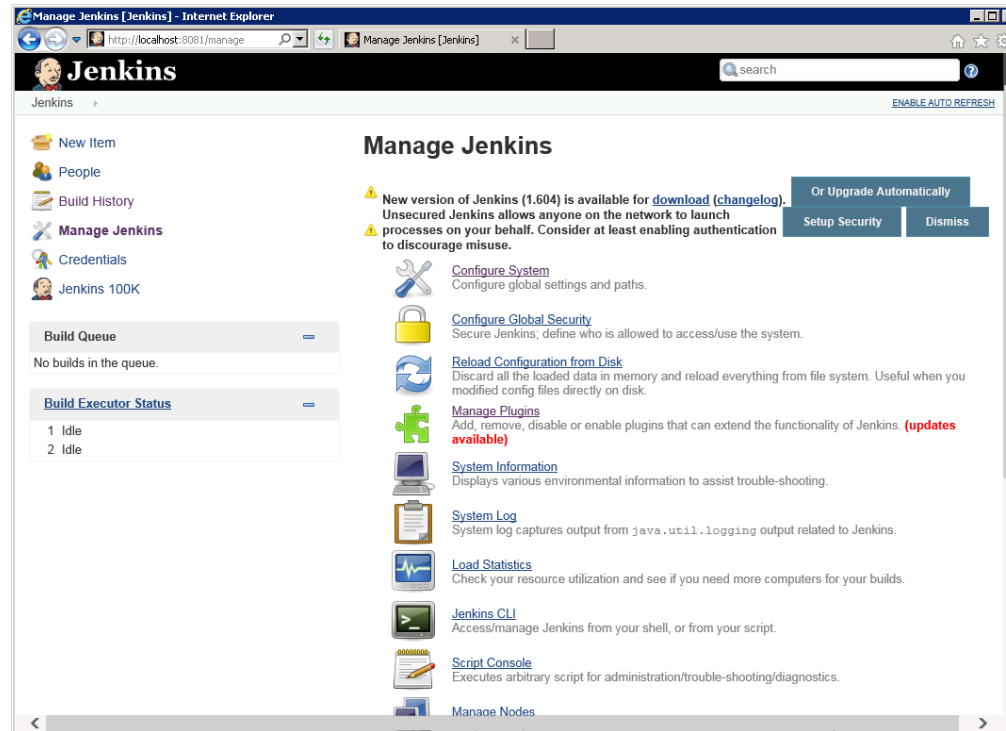


```
C:\Windows\system32\cmd.exe
C:\Program Files (x86)\Jenkins>java -jar jenkins.war --httpPort=8081
Running from: C:\Program Files (x86)\Jenkins\jenkins.war
webroot: %user.home%\.jenkins
Mar 15, 2015 11:06:48 AM winstone.Logger logInternal
INFO: Beginning extraction from war file
Mar 15, 2015 11:06:51 AM org.eclipse.jetty.util.log.JavaUtilLog info
INFO: jetty-winstone-2.8
Mar 15, 2015 11:06:53 AM org.eclipse.jetty.util.log.JavaUtilLog info
INFO: NO JSP Support for , did not find org.apache.jasper.servlet.JspServlet
Jenkins home directory: C:\Users\Administrator\.jenkins found at: %user.home%\.jenkins
Mar 15, 2015 11:06:54 AM org.eclipse.jetty.util.log.JavaUtilLog info
INFO: Started SelectChannelConnector@0.0.0.0:8081
Mar 15, 2015 11:06:55 AM winstone.Logger logInternal
INFO: Winstone Servlet Engine v2.0 running: controlPort=disabled
Mar 15, 2015 11:06:55 AM jenkins.InitReactorRunner$1 onAttained
INFO: Started initialization
Mar 15, 2015 11:07:15 AM jenkins.InitReactorRunner$1 onAttained
INFO: Listed all plugins
Mar 15, 2015 11:07:15 AM jenkins.InitReactorRunner$1 onAttained
INFO: Prepared all plugins
Mar 15, 2015 11:07:15 AM jenkins.InitReactorRunner$1 onAttained
INFO: Started all plugins
Mar 15, 2015 11:07:20 AM jenkins.InitReactorRunner$1 onAttained
```

Jenkins 구동(v1.605 사용)

3. Build Environment

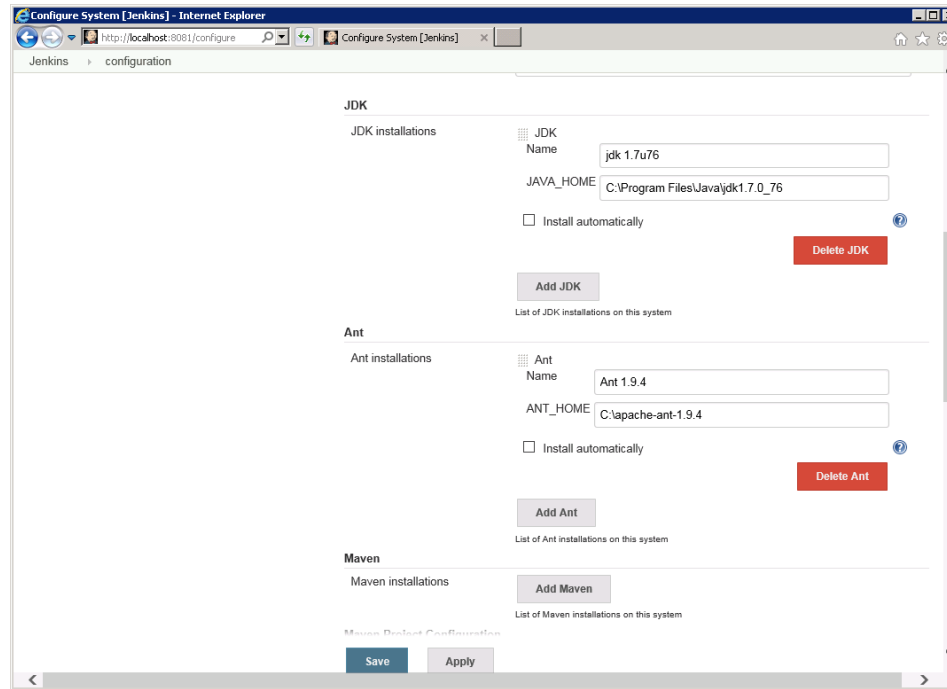
Jenkins



Manage Jenkins 화면

3. Build Environment

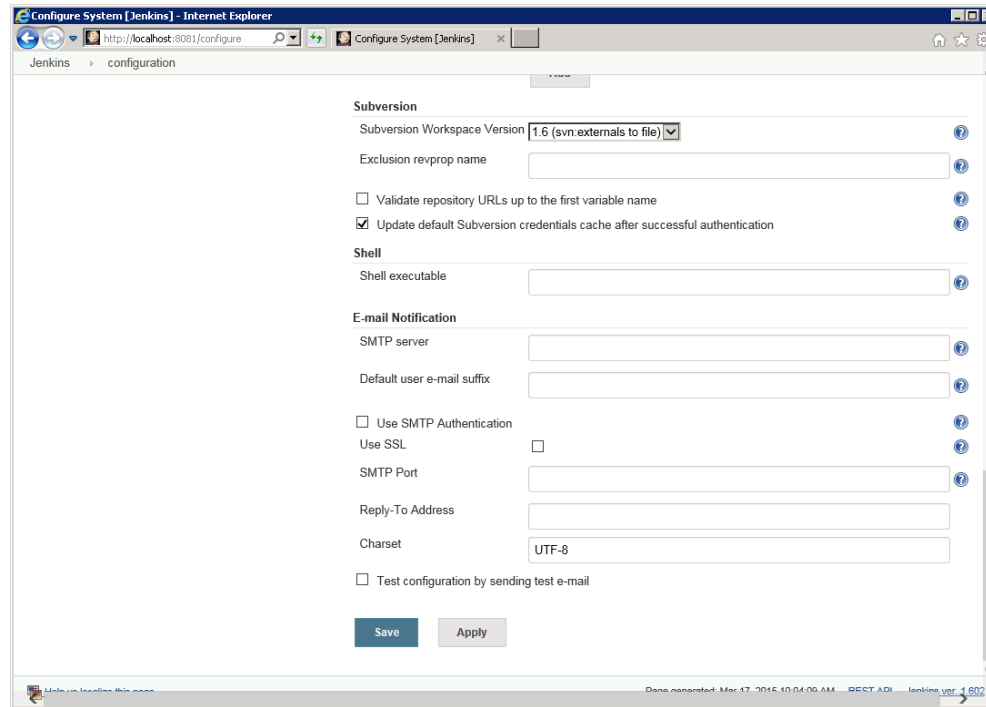
Jenkins



시스템 환경설정에서
JDK, ANT, Maven의 경로를 설정

3. Build Environment

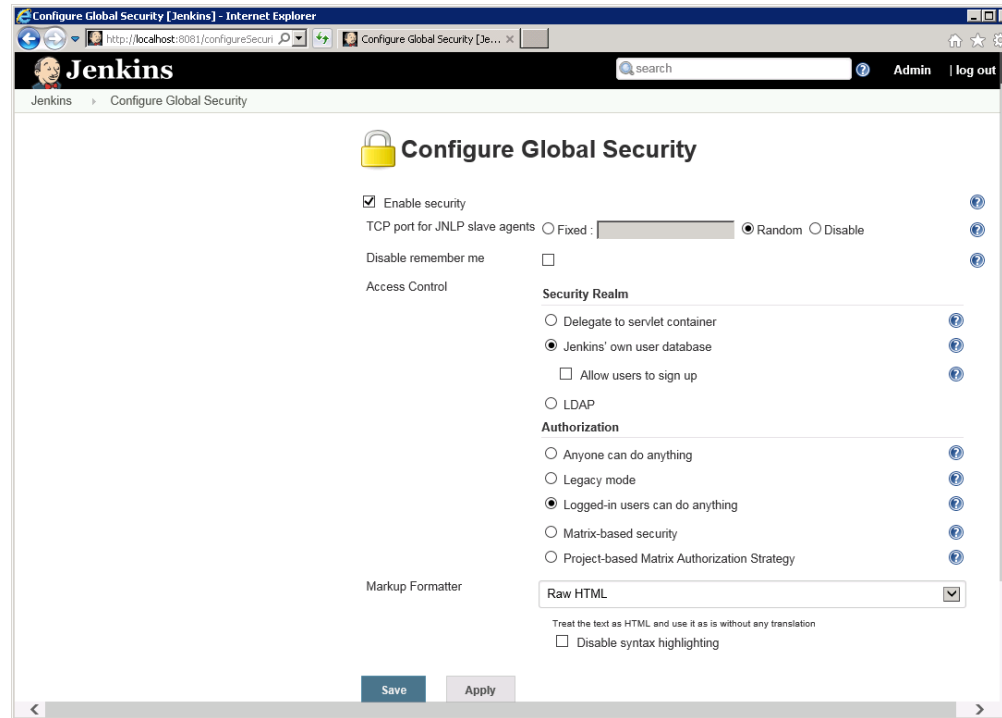
Jenkins



SVN Workspace 버전, SMTP Server 지정

3. Build Environment

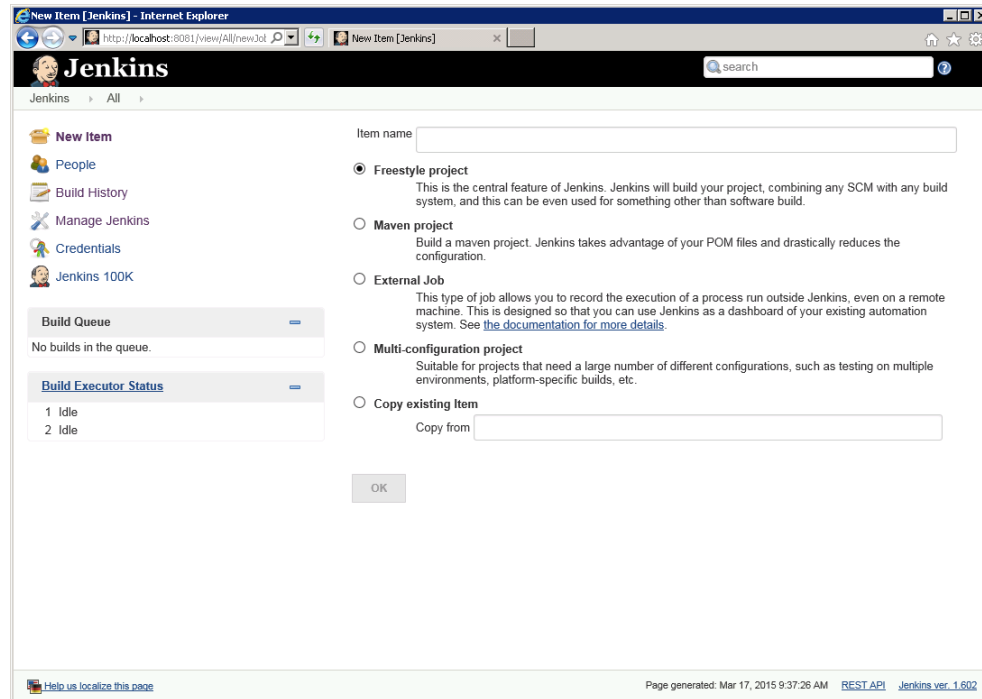
Jenkins



보안 설정을 통해 로그인 시스템 생성

3. Build Environment

Jenkins



새로운 프로젝트 생성

3. Build Environment

Jenkins

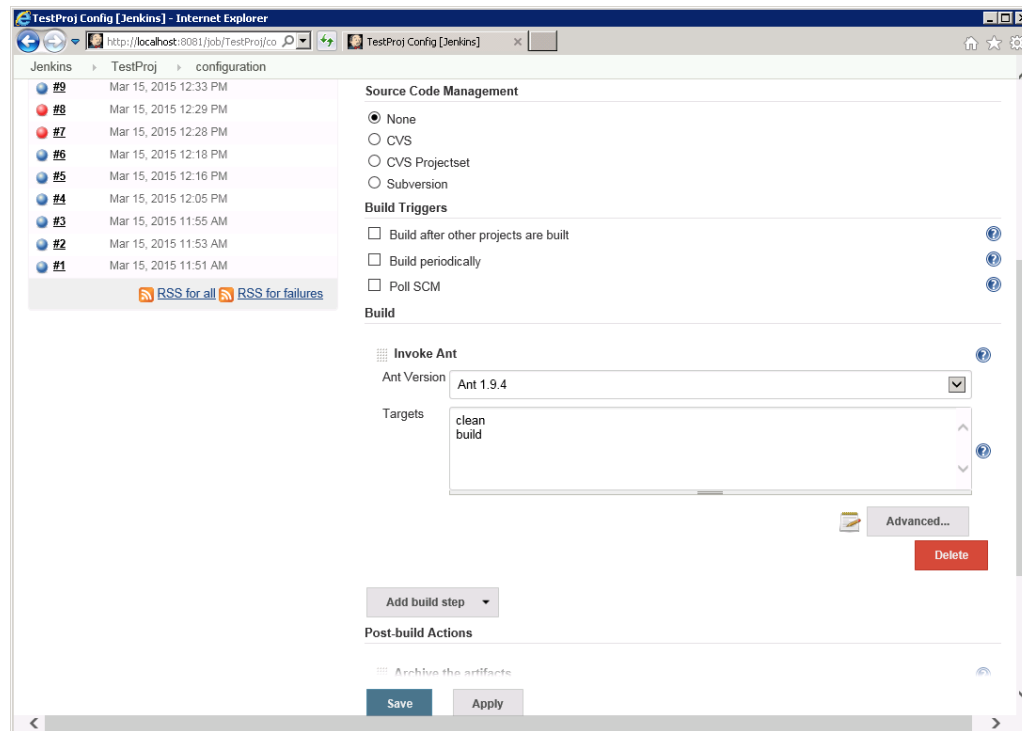
The screenshot shows the Jenkins web interface for a project named 'TestProj'. The browser address bar indicates the URL is <http://localhost:8081/job/TestProj/>. The page features a navigation sidebar on the left with options like 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build Now', 'Delete Project', and 'Configure'. The main content area is titled 'Project TestProj' and includes a search bar, an 'add description' link, and a 'Disable Project' button. Below these are links for 'Workspace', 'Last Successful Artifacts' (showing 'TestMain.class' with 529 B view), and 'Recent Changes'. A 'Build History' table is displayed, listing builds from #1 to #12 with their respective dates and times. The table includes a 'trend' link. At the bottom, there are 'Permalinks' for various build types and RSS feeds for all builds and failures. The footer shows the page was generated on Mar 17, 2015 at 9:41:01 AM, with links for 'Help us localize this page', 'BEST API', and 'Jenkins ver. 1.602'.

Build Number	Date and Time
#12	Mar 15, 2015 12:45 PM
#11	Mar 15, 2015 12:42 PM
#10	Mar 15, 2015 12:39 PM
#9	Mar 15, 2015 12:33 PM
#8	Mar 15, 2015 12:29 PM
#7	Mar 15, 2015 12:28 PM
#6	Mar 15, 2015 12:18 PM
#5	Mar 15, 2015 12:16 PM
#4	Mar 15, 2015 12:05 PM
#3	Mar 15, 2015 11:55 AM
#2	Mar 15, 2015 11:53 AM
#1	Mar 15, 2015 11:51 AM

프로젝트 메인 화면

3. Build Environment

Jenkins




프로젝트 환경 설정에서
소스 코드 관리, 빌드 트리거, 빌드 도구 지정

3. Build Environment

Jenkins

Post-build Actions

Archive the artifacts 

Files to archive





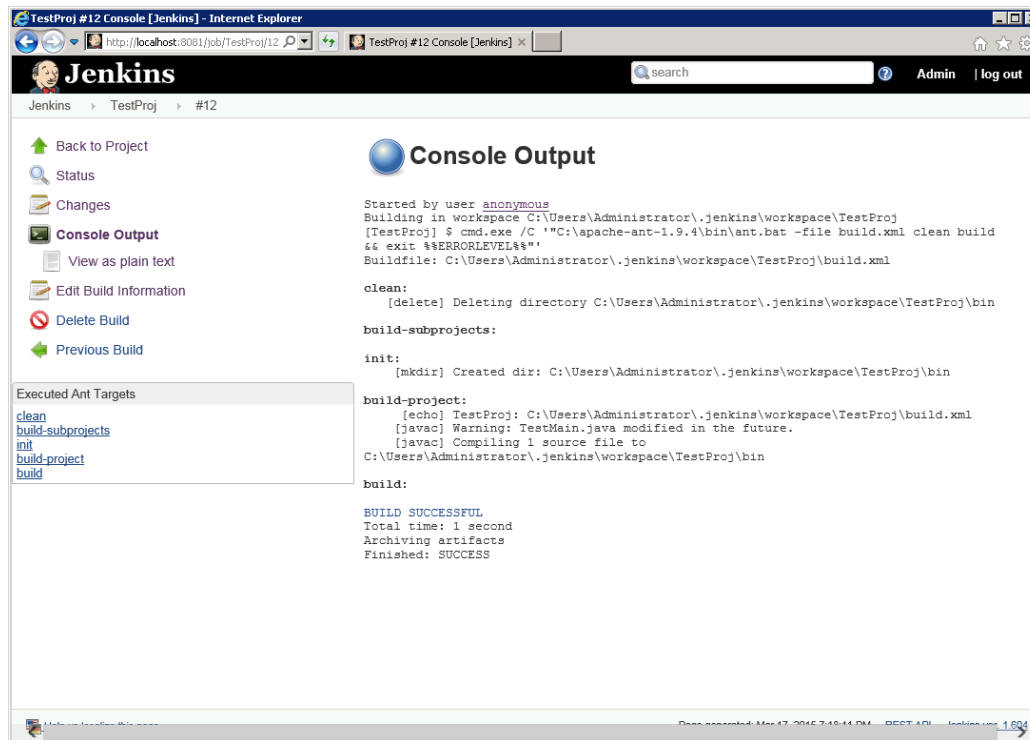
Advanced...

Delete

프로젝트 환경 설정에서
Post-build 동작 지정

3. Build Environment

Jenkins



Console Output, Changes를 통해 빌드 정보 출력

Thank you

The background features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the right side of the frame, creating a modern, dynamic feel. The rest of the background is a clean, solid white.