

Glue Mobile 교육자료

목 차

1. [개요](#)
2. [개발환경 구성](#)
3. [모바일 UI 개발](#)
4. [모바일 장치제어](#)

『2014대한민국소프트웨어기술대상』 우수상

개요

1. 개발 배경
2. 정의 및 구성도
3. 기술 특징
4. 적용 모바일 서비스
5. 상용화

개발배경(1)

- 모바일 비즈니스 시장은 지속적 성장세, 다양한 모바일 앱이 생성될 전망
- 당사는 모바일 융합 기반의 일류화 및 기존 사업의 가치를 극대화 하는 전략

IT 트렌드

- 국내 모바일 시장 2017년까지 연평균 4.7% 성장
→ 지난해 6조 4천억, 2017년 7조 6,676억원



※ 출처: IDC, 2013

- 2014 가트너 10대 전략기술의 핵심 - Mobility
 - 1) 다양한 모바일 기기의 관리 중요
 - 2016년까지 개인별 3~5개 스마트기기 보유
 - 모바일 기기 관리와 보안위협 대응필요
 - 2) 모바일 앱(애플리케이션)
 - 사진, 동영상 활용한 역동적인 앱 증가
 - 앱 개발 방식 다양, 웹 기술 중심 발전

모바일 융합 Biz모델 확산



- 일류화 / 기존사업 가치창출
- 모바일을 통한 제조/사무혁신



- 현장 실시간관리
- 모바일 Energy Saving
- MPS 원격모니터링



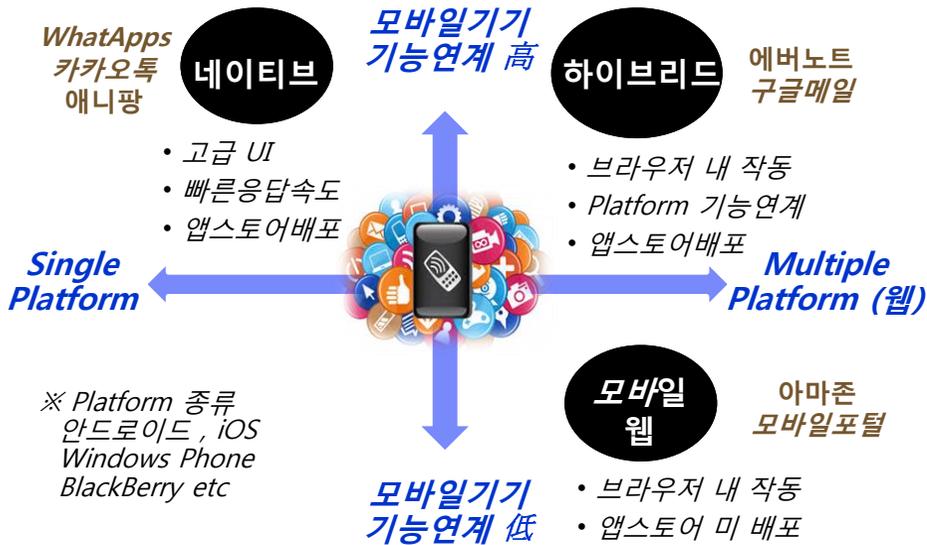
- SWP연계확대
- 기업용AppStore
- ETP B2C 확산 등

개발배경(2)

- 비즈니스 요구에 따른 세 가지 유형의 모바일 앱이 존재
- B2B 시장에서 강점을 가진 하이브리드앱 개발을 위한 공통 솔루션이 요구 됨

모바일 앱 유형과 추이

- 비즈니스 요구에 적합한 앱 유형을 선정



- 2016년까지 하이브리드앱 비중 확대

구분	네이티브	하이브리드	모바일웹
B2B	10%	60%	30%
B2C	40%	40%	20%

하이브리드앱의 장점

- 웹 기술을 활용하여 모바일앱 개발
HTML5, CSS3, JavaScript
- 모바일기기 기능연계 가능
- Multi Platform 지원

개발 프레임워크 도입효과

- 개발 프레임워크란?
SW 개발에 필요한 환경 및 공통 모듈을 미리 구현한 기반 솔루션
- 모바일 앱 개발 및 운영 비용 절감
- 개발 생산성 및 품질 향상

☞ 모바일 앱 개발 프레임워크 확보 시
B2B 모바일 앱 60%~90% 개발 시 활용 예상

개념 및 구성도(2)

■ Glue Mobile 개발 프레임워크란?

- 모바일 개발 프레임워크란? 하이브리드앱을 개발하기 위한 프레임워크와 개발도구
- 모바일UI, 하이브리드 플러그인 및 통합개발환경의 3가지로 구성됨

모바일 앱 개발프레임워크 개념도



※ 기 보유한 웹 프레임워크인 Glue FW 부분은 미 표기

모바일 UI 컴포넌트

- 모바일 앱 화면 (UI) 재사용컴포넌트
- 기술획득 : jQuery Mobile 공개SW

하이브리드 앱 플러그인

- 모바일 장치 / 센서 제어, 미디어 연결
- 기술획득 : Apache Cordova 공개SW

통합개발환경

- 모바일 앱 개발 생명주기 지원 기능
- 기술획득 : Eclipse 플러그인 개발

개념 및 구성도 - 모바일UI

■ 공개SW인 jQuery Mobile 중심으로 모바일UI 개발을 위한 라이브러리 제공

모바일 UI 컴포넌트 유형



세부 기능

기능	컴포넌트
위젯	jQuery Mobile
캘린더 (날짜/시간) 선택	DateBox
입력폼	Form Validation
그리드(테이블)	JQGrid
코드템플릿	Handlebar
선, 바,파이 차트그래프	google/am chart
사진 Swipe	Jqm-pagination
MVC 코어 라이브러리	backbone.js
의존성 관리 및 모듈화	require.js
	Underscore.js

개념 및 구성도 - 모바일장치 접근/제어

■ 모바일 앱에서 웹 기술을 활용하여 모바일 기기 내 다양한 장치와 정보를 제공

모바일 장치 접근/제어

MPS원격
모니터링

에너지
소비현황

...
BHS
수하물조회

연계 웹 모듈 장치관리 센서관리 주소록관리

공통플러그인

와이파이상태

위치정보습득

카메라연계

미디어조회

확장플러그인

난독화

푸시
연결

QR Code

Native Platform

세부 기능

기능	설명
Accelerator	가속도조회/중지
GPS	위치정보조회
Vibrator	진동알림
Camera	사진촬영
Contacts	연락처조회
Media	Media조회/재생
FileReaderWriter	파일정보조회
Network Info	네트워크조회
Device Info	메타정보조회
Secure Code	코드 난독화
Barcode Scanner	QR코드 / 바코드 스캔(?)
Push Connect	푸시 서버연계

※ 연계 웹 모듈 : 인터넷 웹 기술 활용하여 개발 (javascript)
 플러그인 : Native 별 언어활용하여 개발 (JAVA, IOS, .Net)

개념 및 구성도 - 통합개발환경

■ 모바일 프로젝트 생성에서 테스트 까지 일련의 개발 과정을 수행하도록 지원



세부 기능

기능	설명
프로젝트생성	모바일 프로젝트 Wizard
팔레트	UI 컴포넌트 생성 지원
로직 설계	UI 화면 및 비즈니스 로직구현
DB 설계	DB 설계 연계
테스트	테스트 도구
에뮬레이터	모바일 앱 직관적인 확인
소스관리	모바일 앱의 소스버전관리
빌드가이드	모바일 Native OS 별 빌드
배포	앱 스토어 배포

주요 특징

■ 모바일 앱 개발 프레임워크의 주요 특징은 아래와 같음

1 개방형표준 준수

- 공개SW기반의 범용화된 기술
- 특정사업자에 대한 종속성 배제
- 표준기반 상호운영성보장

2 재사용성 향상

- 프레임워크 기반 재사용증대
- 모듈화 기반 품질향상
- 지속적인 자산화

3 멀티플랫폼 지원

- One Source Multi Use
- 디바이스 별 웹 API 제공
- 개발 원가절감 기여

4 사용편의성/확장성

- 통합개발환경지원
- 웹 프레임워크와 동일패턴 개발
- 모바일통합솔루션확장



■ 당사 대내외 프로젝트 및 솔루션의 모바일 서비스 개발에 활용

프로젝트

- 기업 모바일 시스템
 - GroupWare, MES, ERP 모바일
 - 현장 모바일 관리



솔루션

- 일류화/에너지 솔루션
 - BESS 원격모니터링
 - Smart Home, 에너지 Saving



■ 안전방재 작업현장 위험인식 모바일 서비스 시범프로젝트 적용



작업현장 위험인식 요구사항

- 제조분야 공장 안전 모니터링
 - 안전현황
 - 안전설비의 상태모니터링
- 안전 정보관리 시스템 연동
 - 작업계획 및 현황
 - 재해상황 속보

개발 환경 구성

1. [JDK 설치 및 환경구축](#)
 2. [Eclipse 설치](#)
 3. [안드로이드 플러그인 ADT설치](#)
 4. [안드로이드 SDK 설치](#)
 5. [안드로이드 AVD 생성](#)
 6. [Glue Mobile 환경 설정](#)
 7. [Glue Mobile Project 생성 기능](#)
 8. [jQuery Mobile & Palette 활용 방법](#)
- ◆ **부록 - Genymotion 설치 (AVD 대체)**

JDK 설치 및 환경구축 (1)

■ JDK 다운로드

1. Oracle 에 접속하여 JAVA 개발도구(JDK)를 설치해야 합니다.
<http://www.oracle.com/technetwork/java/index.html>

ORACLE
Sign In/Register Help Country Communities I am a... I want to... Search
Products Solutions Downloads Store Support Training Partners About OTN

Oracle Technology Network > Java

Java

ARTICLE
JavaFX Integration Strategies
Expert Adam Bien shows you how with lambdas and support for asynchronous communication, JavaFX introduces new integration possibilities for back-end services.
Posted 11/20/13 | Tags: java, JavaFX, Lambdas | Headlines Archive

What's New
Java in the Cloud: Rapidly develop and deploy Java business applications in the cloud. Try it FREE for 30 days.

Essential Links
About Us/Become a Member
Java APIs
Technical Articles
New to Java
Java Certification & Training
Java Bug Database
"The Java Source" Blog @Java
Java Developer Newsletter
Java Magazine
Demos and Videos

Developer Spotlight
Want JDK8 to be Great?
Coding at Internet of Things (IoT) Hack Fest
Java Magazine: The Internet of Things for Java Developers
Fifty New Features of Java EE 7
New ADF Mobile Patch Release
A JavaOne Success Story: The Java Embedded Challenge

Blogs
HttpClient and NetBeans Platform
Posted: Dec 6
Happy Internet of Things (IoT) Hanutkah from Java SE Embedded
Posted: Dec 5
Video: Arun Gupta and Java EE
Posted: Dec 5
First Enhancement Request from TestFairy Team
Posted: Dec 5
Global Day of Coderetreat 2013
Posted: Dec 4

Technologies
Java SE
Java SE Support
Java SE Advanced & Suite
Java Embedded
Java EE
Java ME
JavaFX
Java Card
Java TV
Java DB
Developer Tools

Software Downloads View All Downloads

Top Downloads
Java SE
Java EE and GlassFish
JavaFX
Java ME
JDeveloper 11g and ADF
Enterprise Pack for Eclipse
NetBeans IDE
Pre-Built VM for Java Devs

New Downloads
Java SE 7 Update 46
Released 10/15/13
Java SE Embedded 7 Update 46
Released 10/15/13
Java ME 3.3
Released 7/23/13
Java EE 7
Released 6/12/13

Overview Downloads Documentation Community Technologies Training

Java SE Downloads

Latest Release Next Release (Early Access) Embedded Use Previous Releases

Java Platform (JDK) 7u17 JavaFX 2.2.7 JDK 7 + NetBeans

Here are the Java SE downloads in detail:

Java Platform, Standard Edition

Java SE 7u17
This release includes important security fixes. Oracle strongly recommends that all Java SE 7 users upgrade to this release. Learn more >

JDK 7u17
DOWNLOAD

JRE
DOWNLOAD

JDK 7 Docs
Installation Instructions
ReadMe
Release Notes
Oracle License
Java SE Products
Third Party Licenses
Certified System Configurations

JRE 7 Docs
Installation Instructions
ReadMe
Release Notes
Oracle License
Java SE Products
Third Party Licenses
Certified System Configurations

JDK 설치 및 환경구축 (2)

■ JDK 다운로드

Java SE Development Kit 7u17

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Accept License Agreement Decline License Agreement

Product / File Description	File Size	Download
Linux x86	106.65 MB	jdk-7u17-linux-i586.rpm
Linux x86	92.97 MB	jdk-7u17-linux-i586.tar.gz
Linux x64	104.78 MB	jdk-7u17-linux-x64.rpm
Linux x64	91.71 MB	jdk-7u17-linux-x64.tar.gz
Mac OS X x64	143.78 MB	jdk-7u17-macosx-x64.dmg
Solaris x86 (SVR4 package)	135.39 MB	jdk-7u17-solaris-i586.tar.Z
Solaris x86	91.67 MB	jdk-7u17-solaris-i586.tar.gz
Solaris SPARC (SVR4 package)	135.92 MB	jdk-7u17-solaris-sparc.tar.Z
Solaris SPARC	95.32 MB	jdk-7u17-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	22.97 MB	jdk-7u17-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	17.59 MB	jdk-7u17-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	22.61 MB	jdk-7u17-solaris-x64.tar.Z
Solaris x64	15.02 MB	jdk-7u17-solaris-x64.tar.gz
Windows x86	88.75 MB	jdk-7u17-windows-i586.exe
Windows x64	90.42 MB	jdk-7u17-windows-x64.exe

Java SE Development Kit 7 Update 17 - Setup



ORACLE

Welcome to the Installation Wizard for Java SE Development Kit 7 Update 17

This wizard will guide you through the installation process for the Java SE Development Kit 7 Update 17.

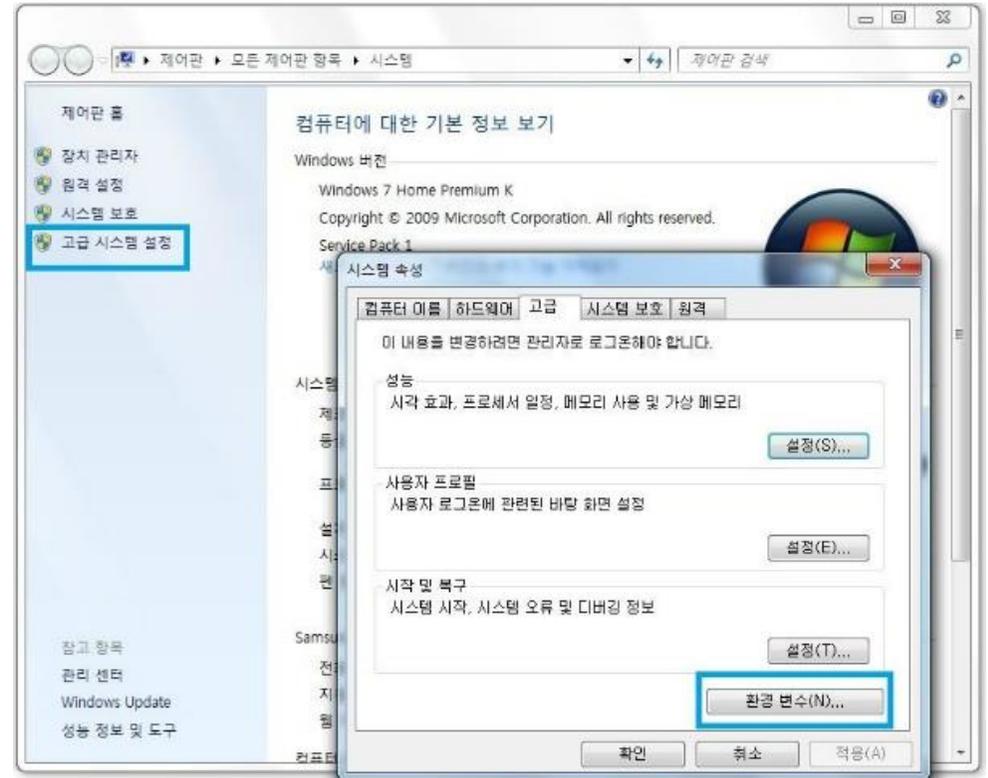
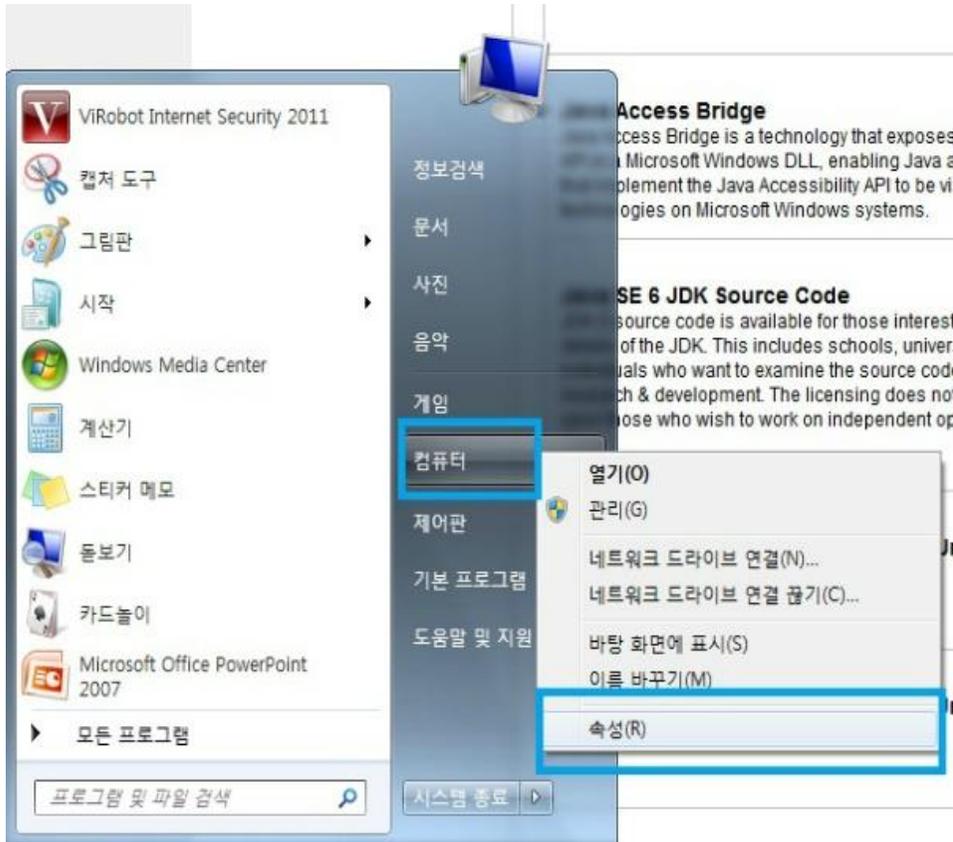
The JavaFX SDK is now included as part of the JDK.

고려사항

- 자신이 사용할 운영체제에 맞춰 설치한다.
- 보통 일반사용자들은 Windows x86(32bit) , Windows x64(64bit) 를 주로 사용한다.

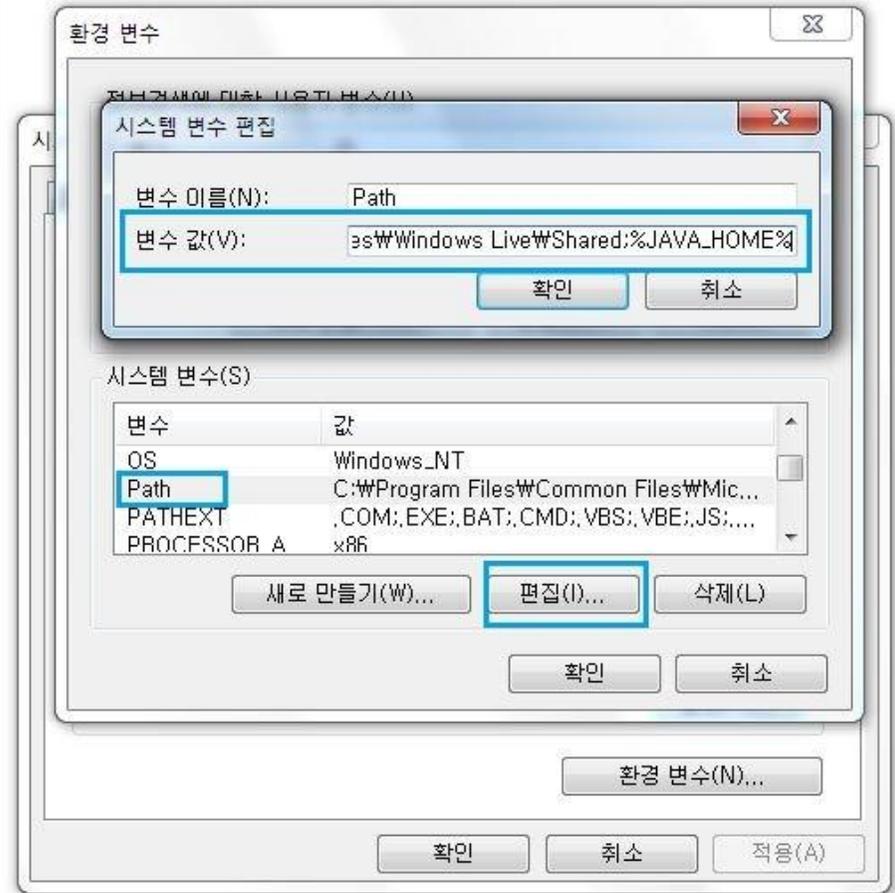
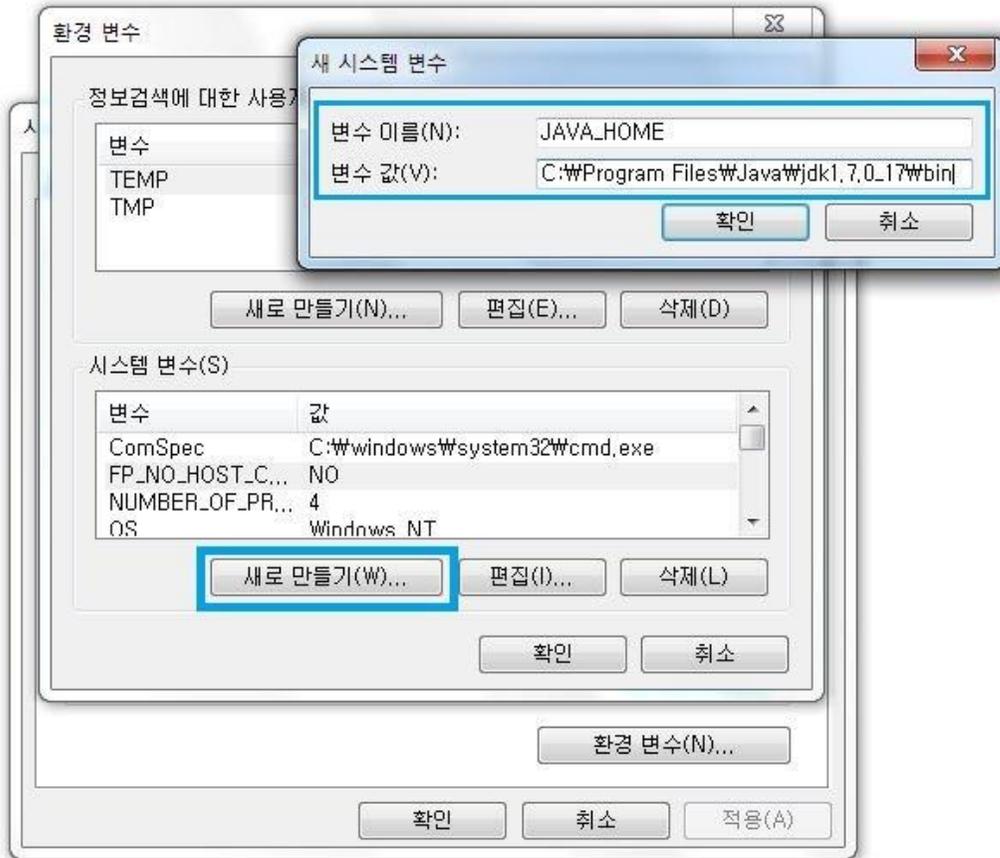
JDK 설치 및 환경구축 (3)

■ JDK 환경변수 설정



JDK 설치 및 환경구축 (4)

■ JDK 환경변수 설정



변수값은 방금 JDK를 설치한 경로입니다.
경로는 다를 수 있지만 꼭 bin 폴더까지
지정을 해주어야 합니다.

JDK 설치 및 환경구축 (5)

■ JDK 설치 확인

시작-cmd 검색 - cmd 창에서 javac 명령어 수행 시 아래와 같은 메시지를 확인하면 정상 설치



```
C:\windows\system32\cmd.exe
C:\Users\정보검색>javac
Usage: javac <options> <source files>
where possible options include:
  -g                Generate all debugging info
  -g:none           Generate no debugging info
  -g:<lines,vars,source> Generate only some debugging info
  -nowarn           Generate no warnings
  -verbose          Output messages about what the compiler is doing
  -deprecation      Output source locations where deprecated APIs are used
  -classpath <path> Specify where to find user class files and annotation processors
  -cp <path>        Specify where to find user class files and annotation processors
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path> Override location of bootstrap class files
  -extdirs <dirs>    Override location of installed extensions
  -endorseddirs <dirs> Override location of endorsed standards path
  -proc:<none,only> Control whether annotation processing and/or compilation is done.
  -processor <class1>[,<class2>,<class3>...] Names of the annotation processors to run; bypasses default discovery process
  -processorpath <path> Specify where to find annotation processors
  -d <directory>     Specify where to place generated class files
  -s <directory>     Specify where to place generated source files
  -implicit:<none,class> Specify whether or not to generate class files for implicitly referenced files
  -encoding <encoding> Specify character encoding used by source files
  -source <release>  Provide source compatibility with specified release
  -target <release>  Generate class files for specific VM version
  -version           Version information
  -help             Print a synopsis of standard options
  -Akey[=value]     Options to pass to annotation processors
  -X                Print a synopsis of nonstandard options
  -J<flag>          Pass <flag> directly to the runtime system
  -Werror           Terminate compilation if warnings occur
  @<filename>       Read options and filenames from file

C:\Users\정보검색>
```

Eclipse 설치 (1)

■ 이클립스 다운로드

<1.이클립스(Eclipse)설치>

1.Eclipse를 다운받기 위해 아래 링크주소로 이동

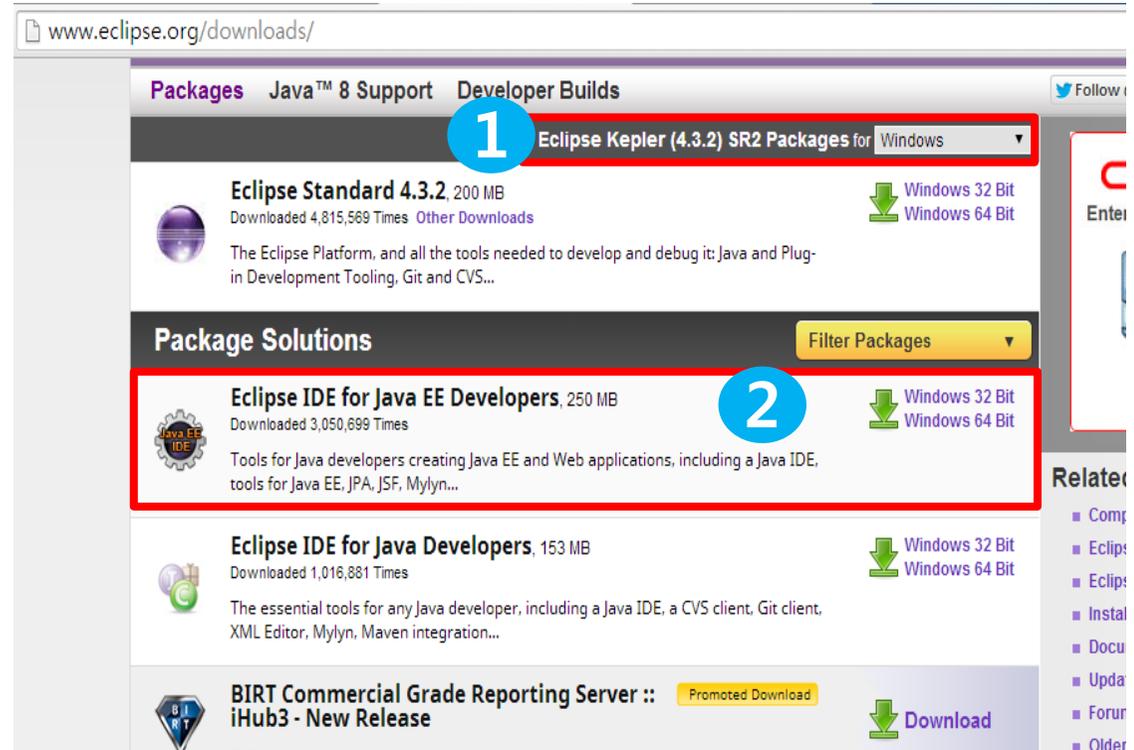
<http://www.eclipse.org/downloads/>

JAVA EE (Java Enterprise Edition)

자바로 구현되는 웹프로그래밍에서 가장 많이 사용되는 JSP, Servlet을 비롯하여, 데이터베이스에 연동하는 JDBC, 그 외에도 JNDI, JTA, EJB 등의 많은 기술들이 포함되어 있습니다.

Java EE는 Java SE의 API에 추가로 (lib 디렉토리에 포함되어 있는 JAR파일들)의 차이입니다.

다운받으실때 빨간 체크창에 자신의 OS확인 하시고요.



고려사항

- Glue Framework와 Glue Mobile Framework과의 Compatibility를 맞추기 위해 Eclipse Java EE Developer를 설치한다.

Eclipse 설치 (2)

■ 이클립스 다운로드

www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/kepler/SR2/eclipse-jee-kepler-SR2-win32.zip



June 18-19
Toulouse

Program is ready!
[Register Now](#)

Visit other Eclipse Sites

mp      

Home Downloads Users Members Committers Resources Projects About Us

Google™ Custom Search

Downloads Home >>

- Source code
- More Packages

Give Back to Eclipse

Donate \$35 or more and Become a Friend of Eclipse!



The Eclipse Foundation is a not-for-

Eclipse downloads - mirror selection

All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.

Download eclipse-jee-kepler-SR2-win32.zip from:

 [\[Korea, Republic Of\] KAIST \(http\)](#)

Checksums: [\[MD5\]](#) [\[SHA1\]](#) [\[SHA-512\]](#)

...or pick a mirror site below.

ORACLE®
Enterprise Pack for Eclipse



Download

Other options for this file

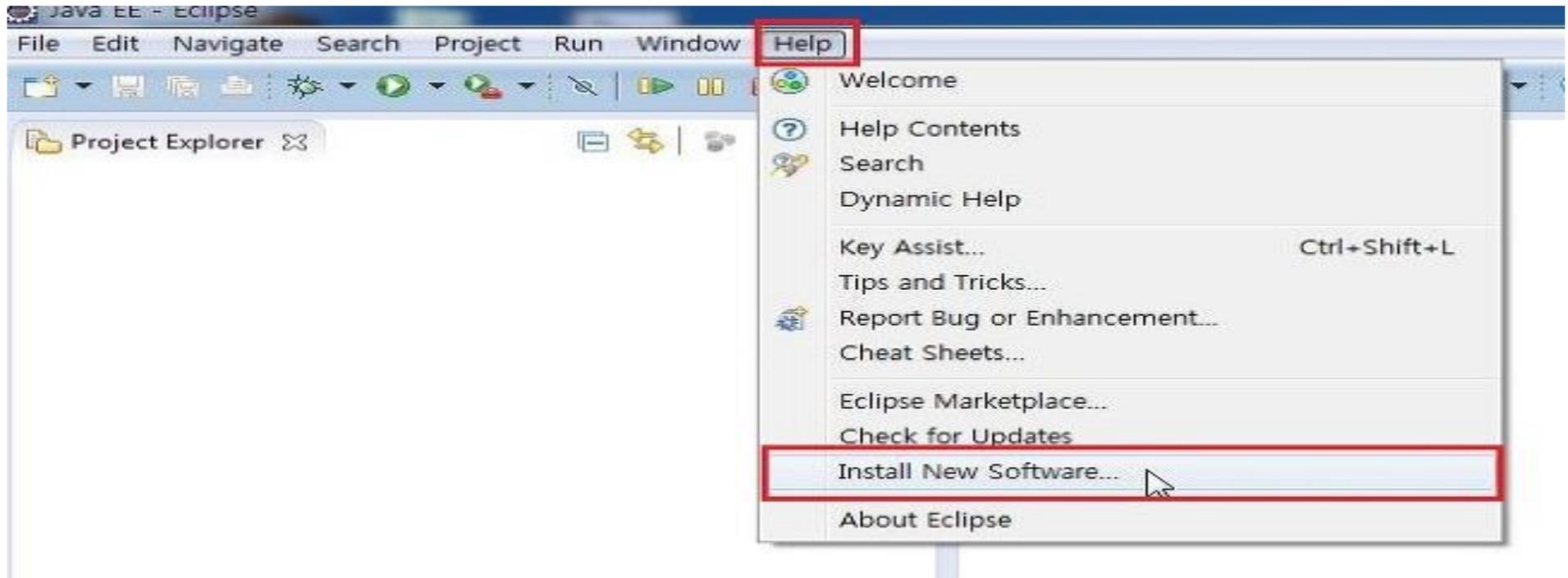
Get It Faster Here

[BLU AGE](#)
Free and fast direct Eclipse downloads. Get more BLU AGE Eclipse

안드로이드 플러그인 ADT설치 (1)

■ 안드로이드 ADT 다운로드

안드로이드 프로그래밍을 하기 위해 플러그인 ADT(Android Development Tool)를 이클립스에 추가 설치 한다. ADT는 이클립스와 안드로이드 SDK를 연결하는 역할을 한다.



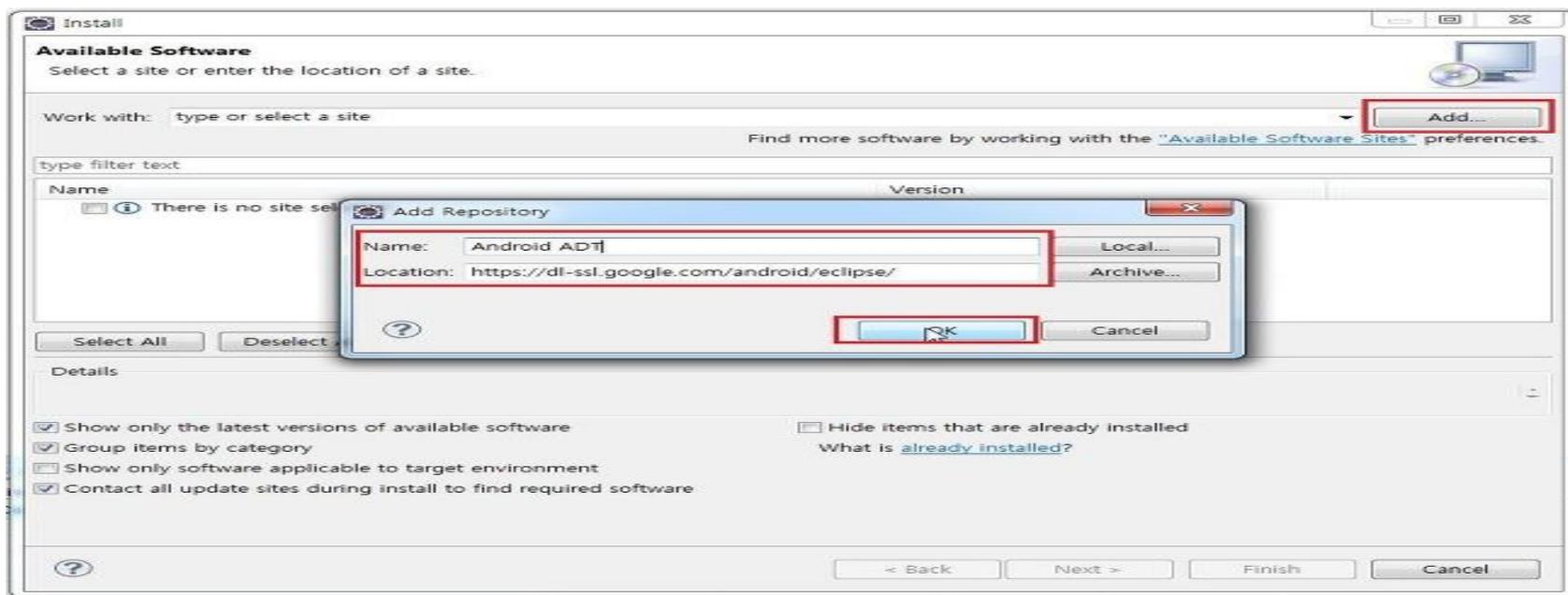
안드로이드 플러그인 ADT설치 (2)

■ 안드로이드 ADT 다운로드

Name은 ADT라는것만 알아볼 수 있게 자유 작성

Location은 이클립스에게 안드로이드 ADT가 배포되는 주소를 알려주는 것이다.

- <https://dl-ssl.google.com/android/eclipse/> 로 입력 후 OK 클릭. (띄어쓰기,영 단어에 유의)



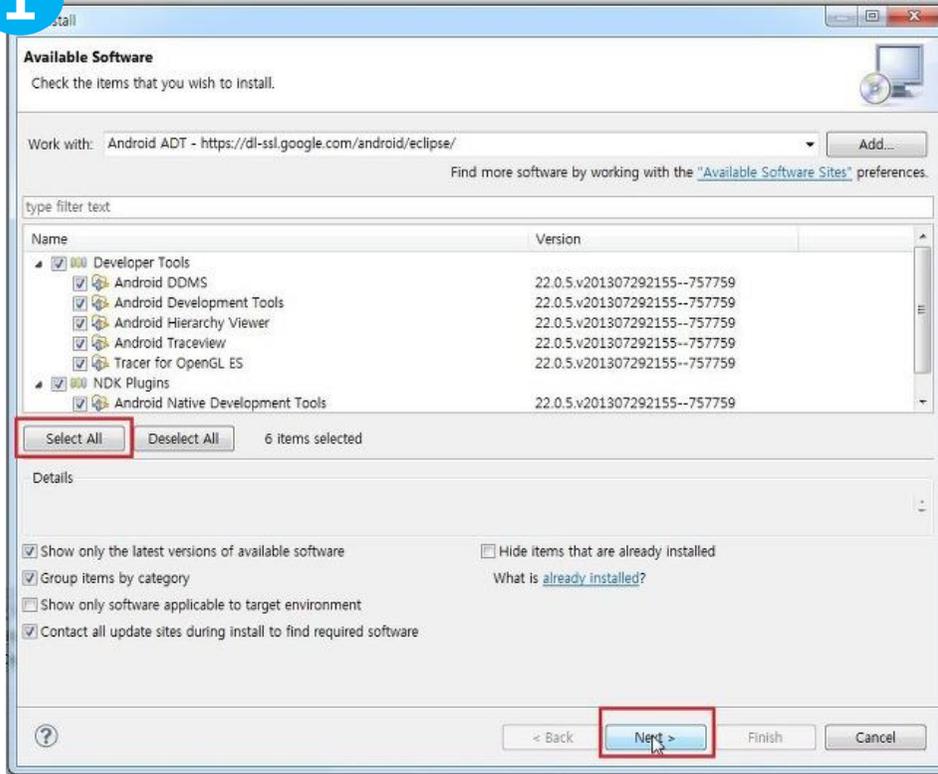
고려사항

- Java가 설치된 경로 (ex : D:\Java\jdk1.7.0_51\bin)에 POSCOICT_CA.cer를 넣어줘야 SSL통신이 가능하다.

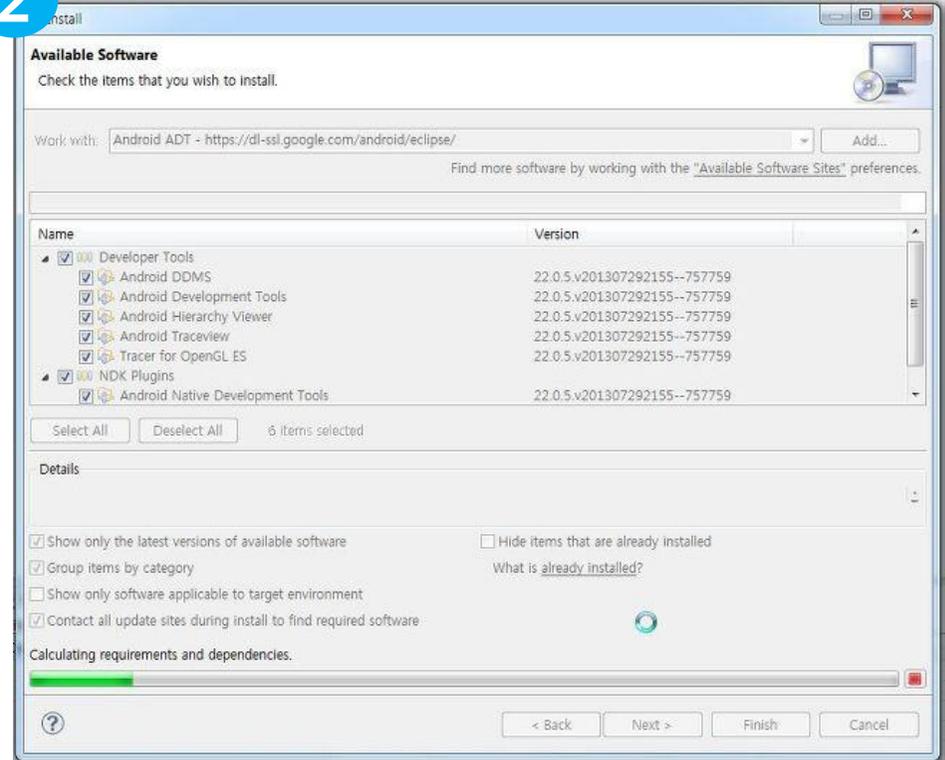
안드로이드 플러그인 ADT설치 (3)

■ 안드로이드 ADT 설치

1



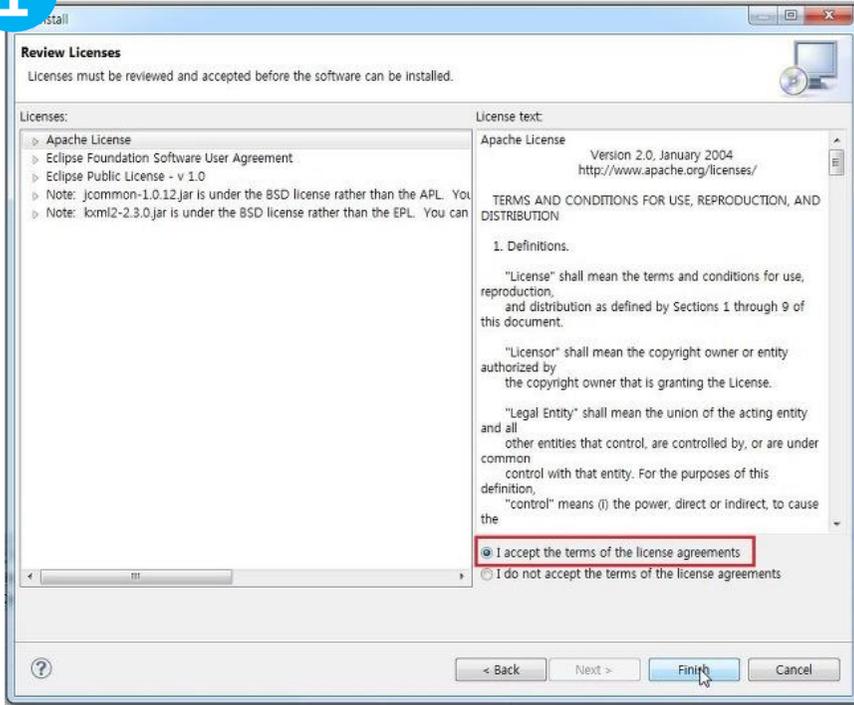
2



안드로이드 플러그인 ADT설치 (4)

■ 안드로이드 ADT 설치

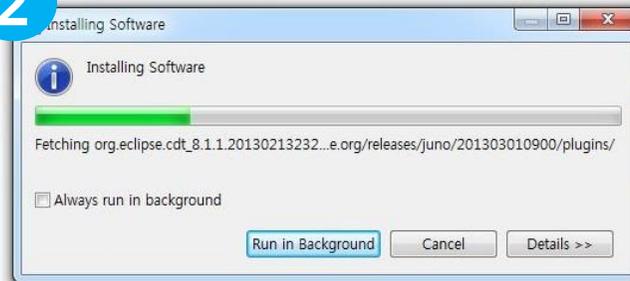
1



3



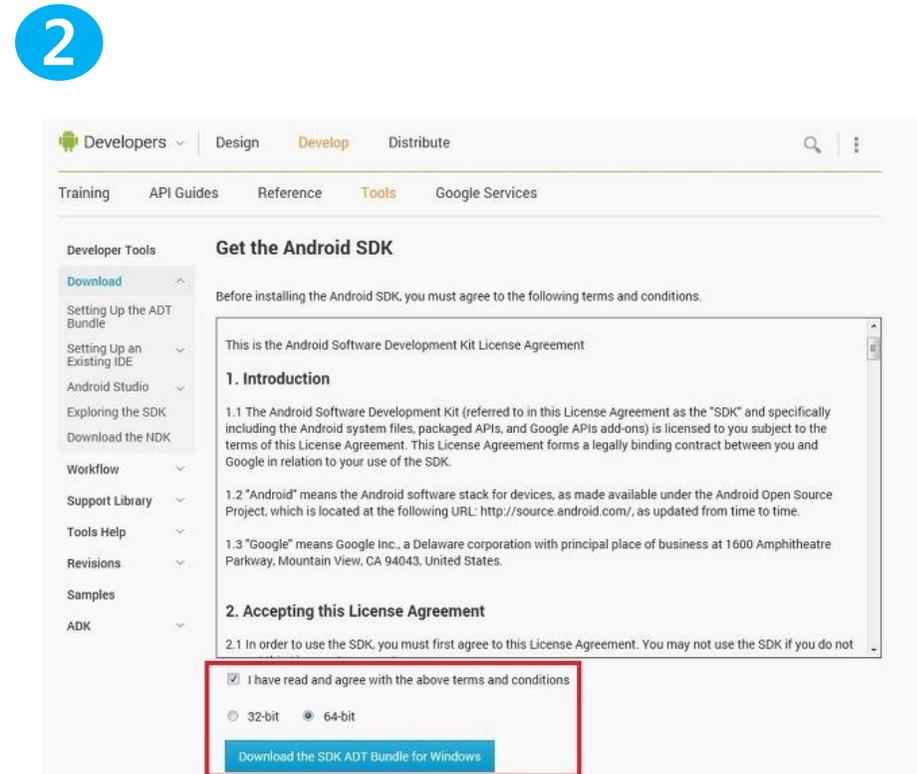
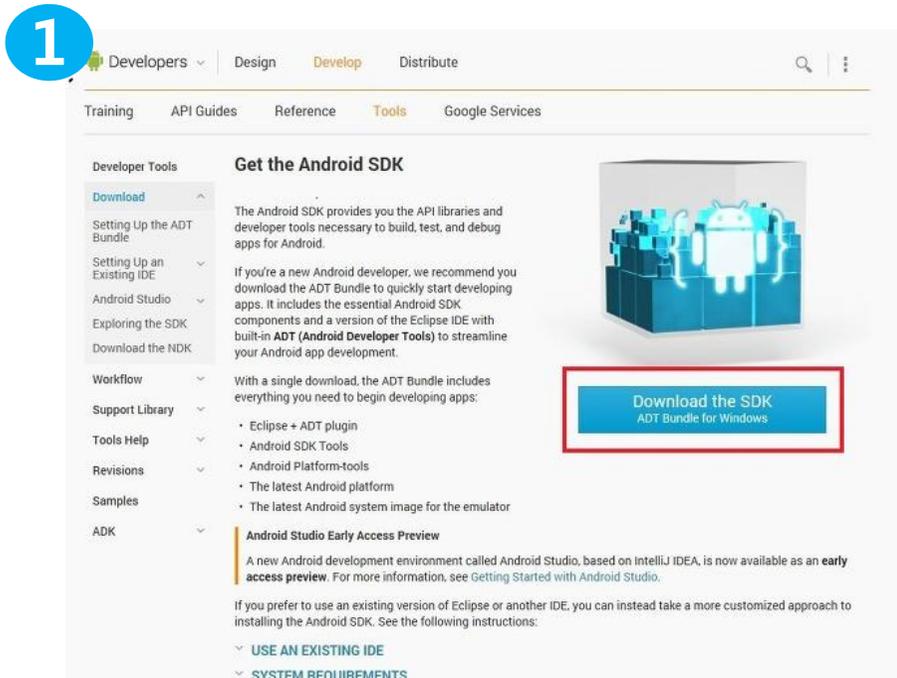
2



안드로이드 SDK 설치 (1)

■ 안드로이드 SDK 다운로드

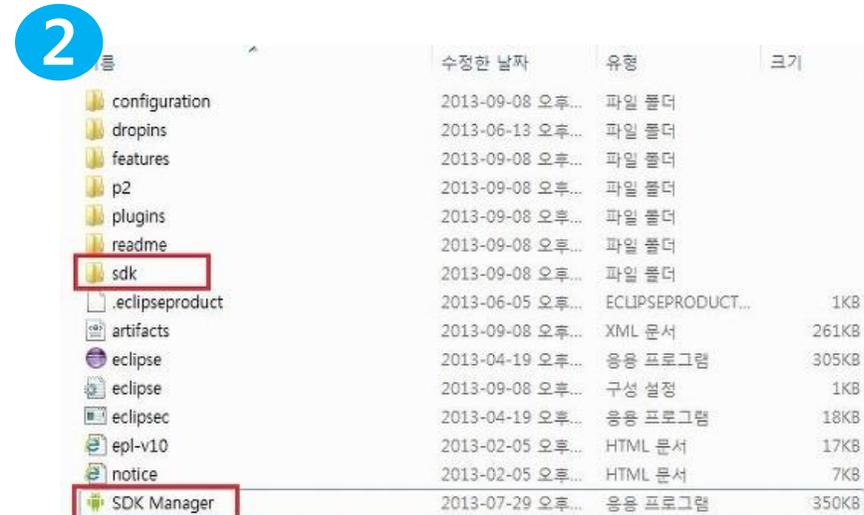
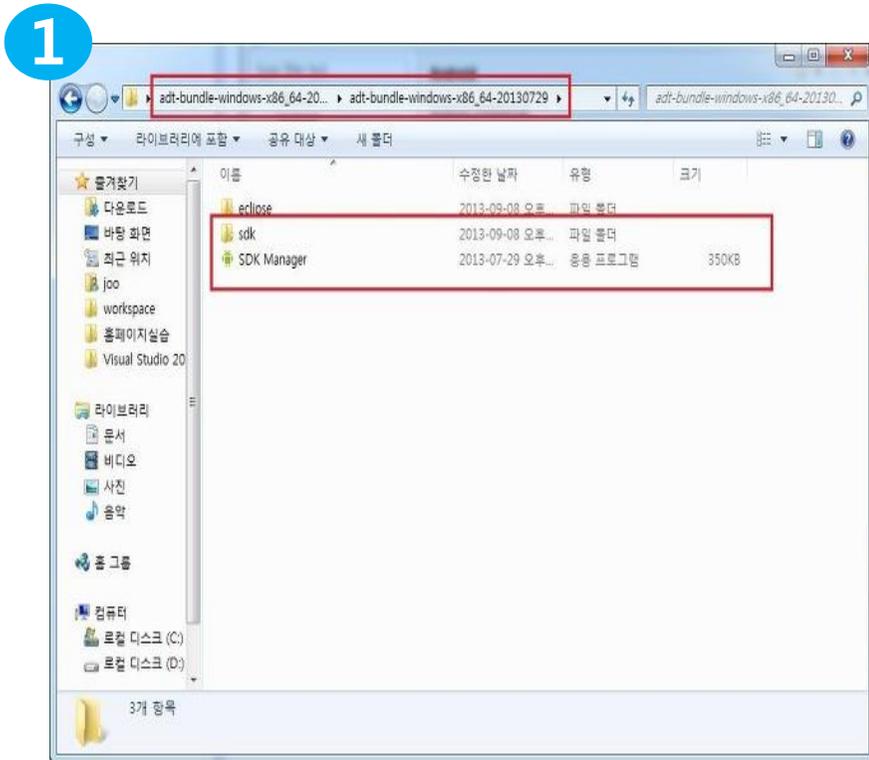
1. 아래 경로로 이동하여 SDK 다운로드 한다.
<http://developer.android.com/sdk/index.html>



안드로이드 SDK 설치 (2)

■ 안드로이드 SDK 폴더 위치이동

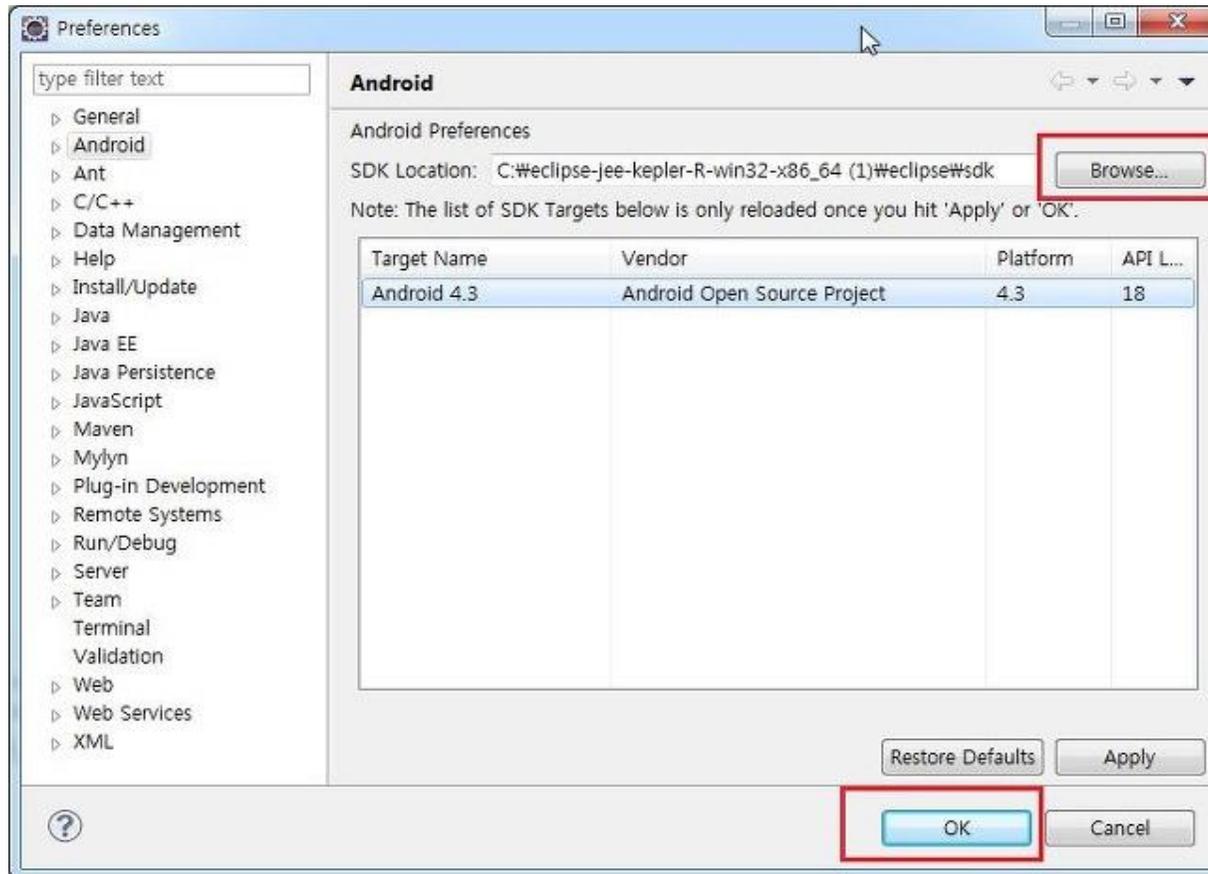
기 설치한 이클립스(JAVA EE)에 SDK 와 SDK Manager 를 복사하여 이클립스 압축 풀어 놓은 곳에 붙여넣기를 한다.



안드로이드 SDK 설치 (3)

■ 안드로이드 SDK 설정

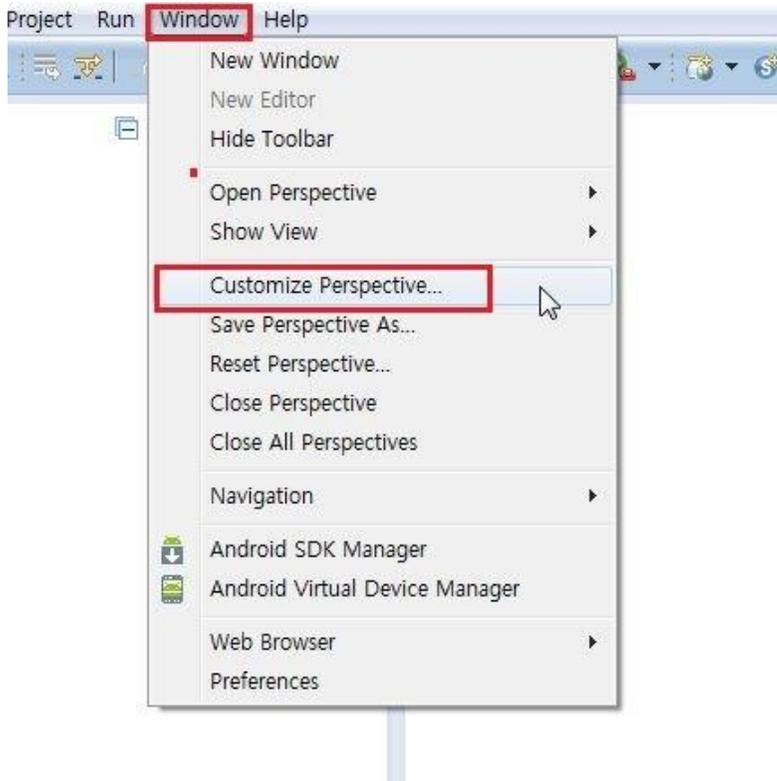
이제 Window - Preferences - Android 로 이동후 Browse 로 이클립스안에 SDK 폴더 붙여넣기 한 경로를 지정해준다. 그러면 아래 그림과 같은 것들이 나타날 것이다. 그후 OK 클릭한다.



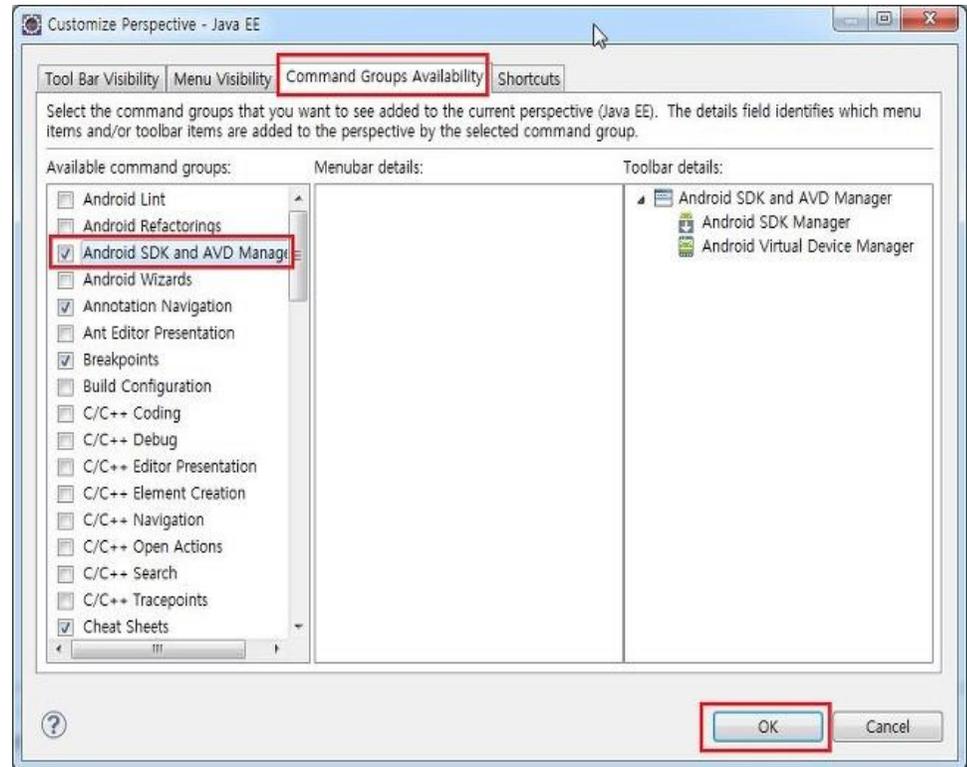
안드로이드 SDK 설치 (4)

■ 안드로이드 SDK 설정

1 Window - Customize Perspective 선택.



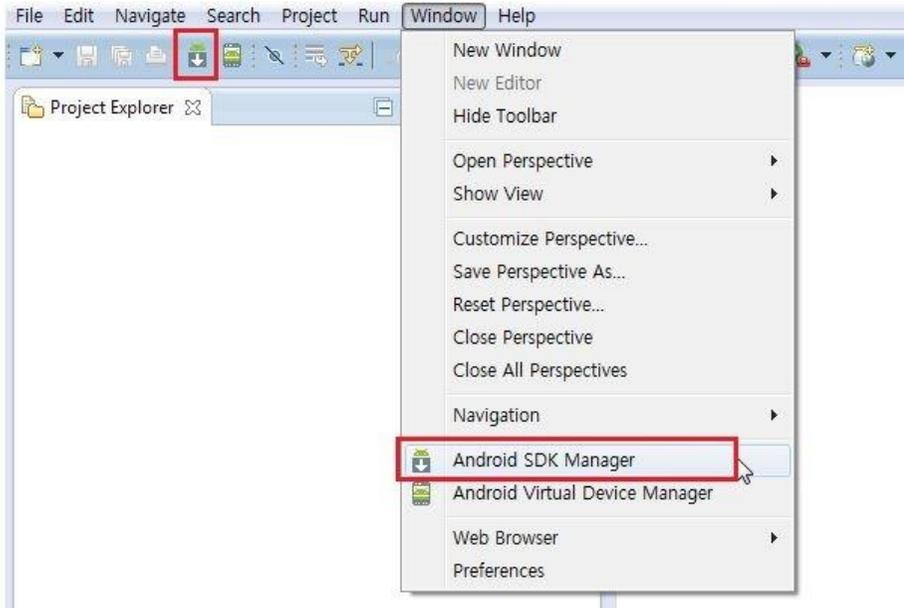
2 Command Groups Availability - Android SDK and AVD Manager 체크 - OK 클릭



안드로이드 SDK 설치 (5)

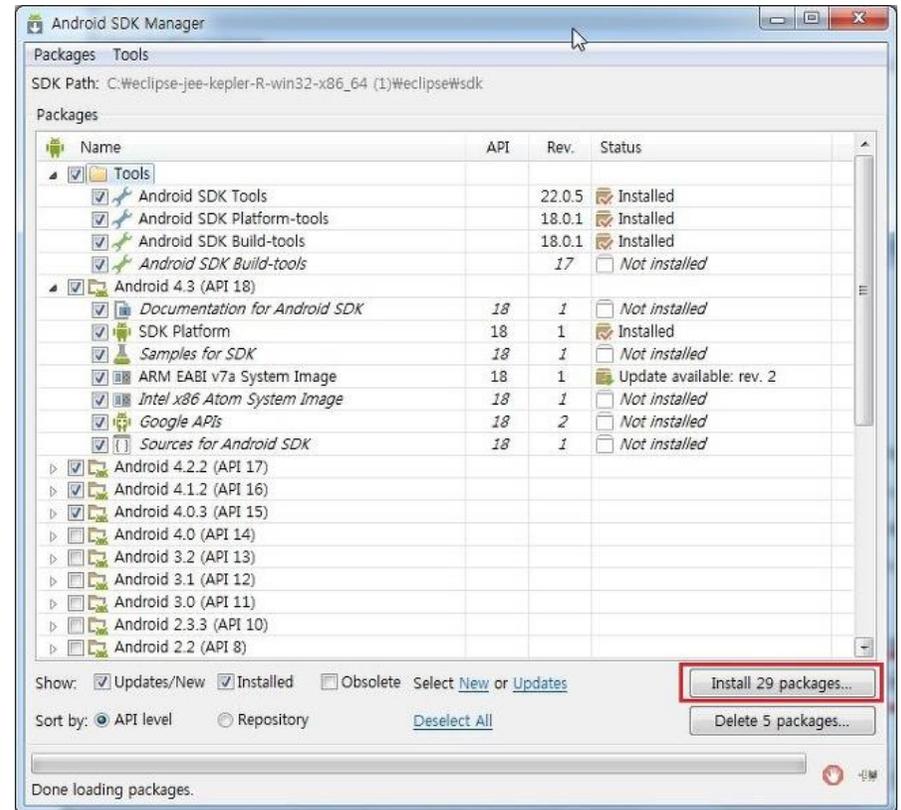
■ 안드로이드 SDK Manager의 Package 다운로드

1



2

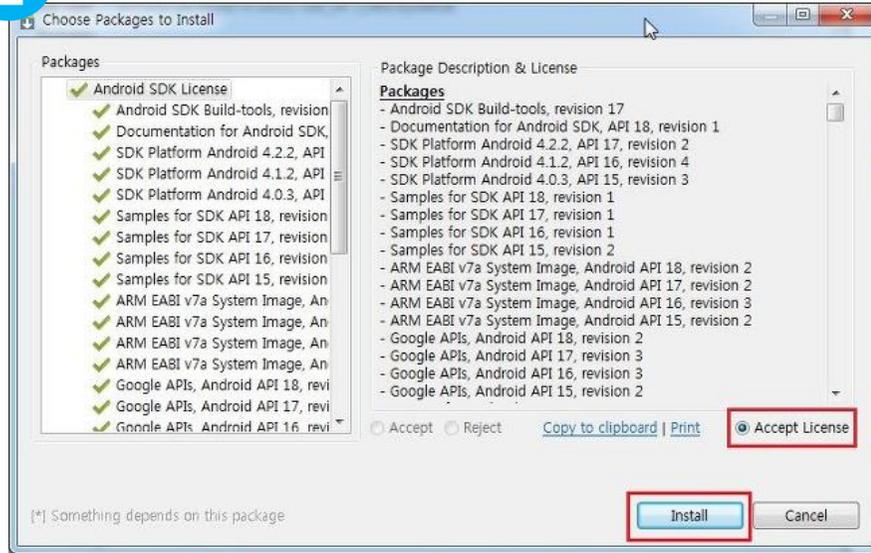
설치하고자 하는 것을 체크 하여 Install



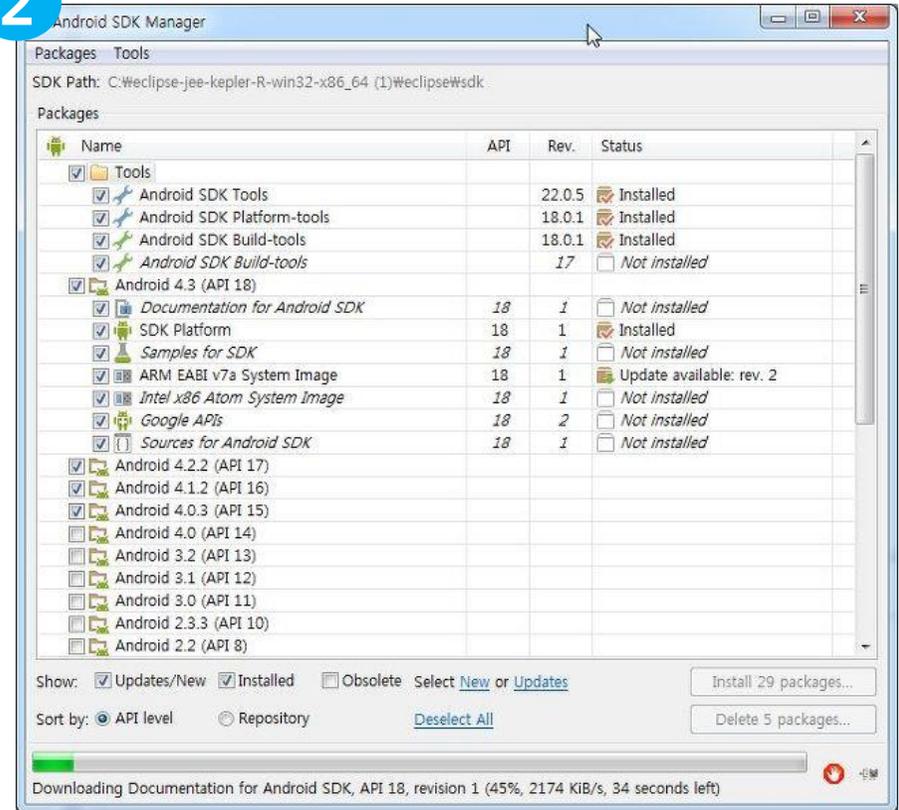
안드로이드 SDK 설치 (6)

■ 안드로이드 SDK Manager의 Package 다운로드

1

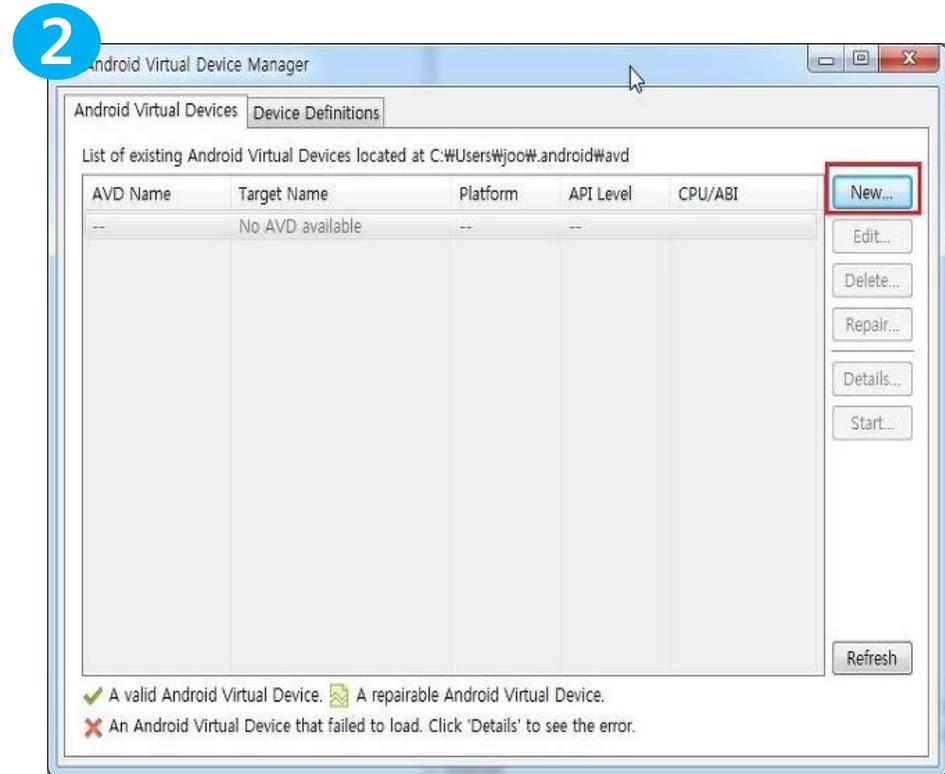
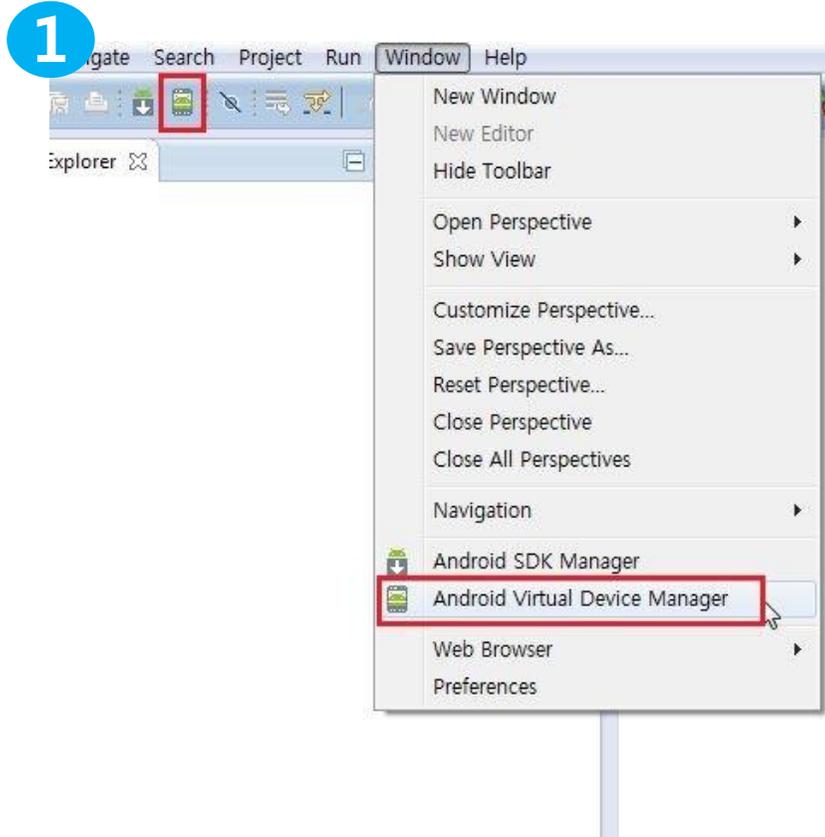


2



안드로이드 AVD(Android Virtual Device) 생성 (1)

■ 안드로이드 AVD 생성



안드로이드 AVD(Android Virtual Device) 생성 (2)

■ 안드로이드 AVD 생성

1

Create new Android Virtual Device (AVD)

AVD Name: Hmm16

Device: Nexus S (4.0", 480 x 800: hdpi)

Target: Google APIs (Google Inc.) - API Level 16

CPU/ABI: ARM (armeabi-v7a)

Keyboard: Hardware keyboard present

Skin: Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 343 VM Heap: 32

Internal Storage: 200 MiB

SD Card: Size: 128 MiB File: Browse...

Emulation Options: Snapshot Use Host GPU

Override the existing AVD with the same name

OK Cancel

2

Android Virtual Device Manager

Android Virtual Devices Device Definitions

List of existing Android Virtual Devices located at C:\Users\wjoo\android\avd

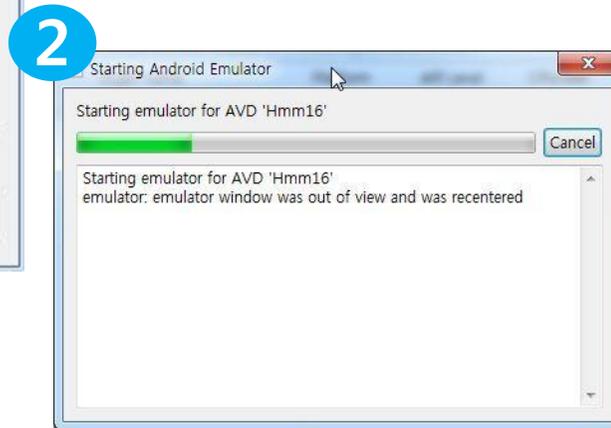
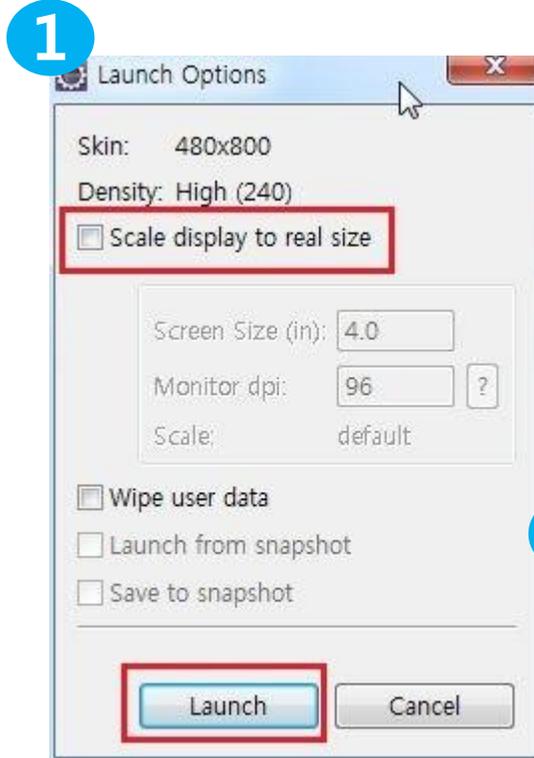
AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ Hmm16	Google APIs (Google Inc.)	4.1.2	16	ARM (armeabi-v...

New... Edit... Delete... Repair... Details... Start... Refresh

✓ A valid Android Virtual Device. ✉ A repairable Android Virtual Device.
✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

안드로이드 AVD(Android Virtual Device) 생성 (3)

■ 안드로이드 AVD 실행



Glue Mobile 환경 설정 (1)

■ 이클립스 플러그인 환경설정

1. D:\weclipse-jee-kepler-SR2-win32_android\weclipse\dropins 경로에 붙여 넣기
 - ▶ com.poscoict.mobileframework.sdk_v.1.0.0_201410011447.jar
 - ▶ com.poscoict.mobileframework.htmleditor_v.1.0.0_201412051435.jar

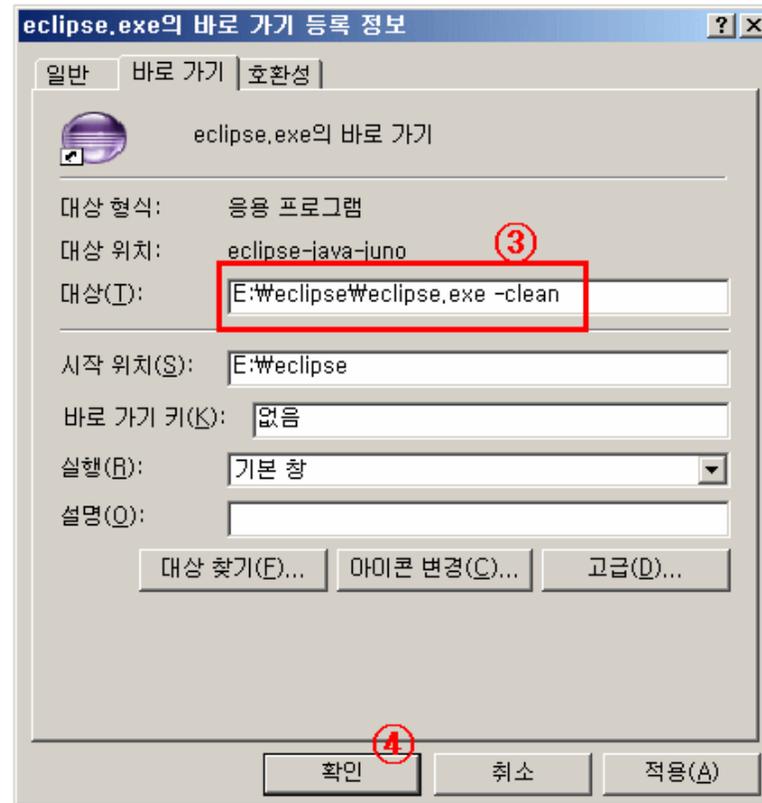
Glue Mobile Plug-In의 패치/업그레이드는 2가지 방법을 선택해서 수행

- "-clean" 옵션 이용
- 재설치(삭제후 설치)

첫번째 방법인 Clean 옵션을 이용하는 방법은 바로가기를 생성해서,
그림과 같이 바로가기 등록정보의 대상 프로그램에 -clean 옵션을 주는 것이다.
ECLIPSE_HOME / dropins 에서 2개 파일을 변경하고, "-clean" 옵션이 포함된 바로가기를 통
해 Eclipse를 실행

Glue Mobile 환경 설정 (2)

■ 이클립스 clean 속성 설정



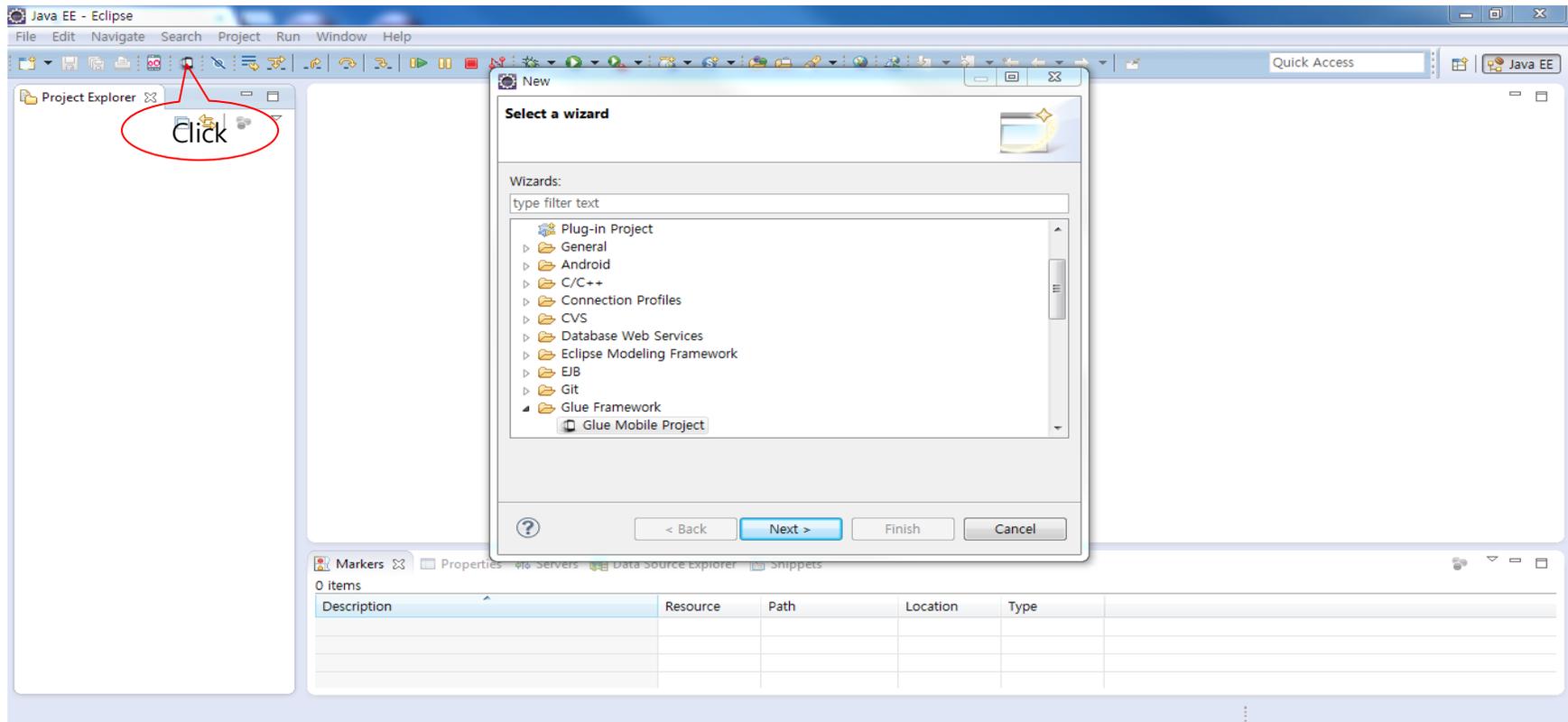
Glue Mobile Project 생성 기능 (1)

■ GlueMobile 프로젝트 생성

2가지 방안 중 선택

1> ProjectExplorer에서 우측 마우스 클릭 -> New -> Others -> Glue Mobile Project Click

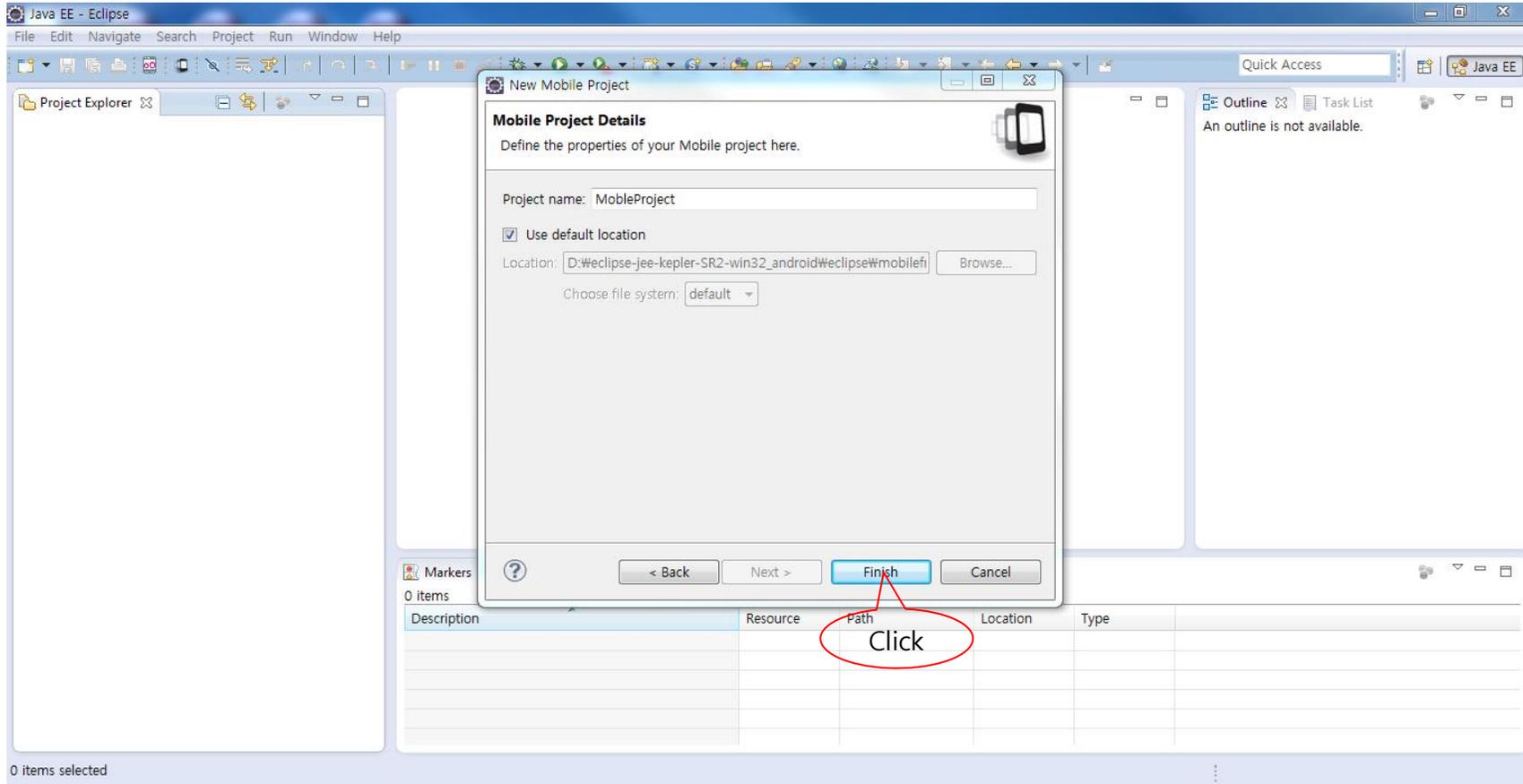
2> 상단에 ICON Click (하기 화면 참조)



Glue Mobile Project 생성 기능 (2)

■ GlueMobile 프로젝트 생성

Project 생성 Wizard PopUp -> Project name 입력 -> Finish 버튼 Click



■ 기능 활용을 위한 시나리오

Palette의 기능을 활용하는 방법을 알아보기 위해, 간단한 시나리오를 설정한다.

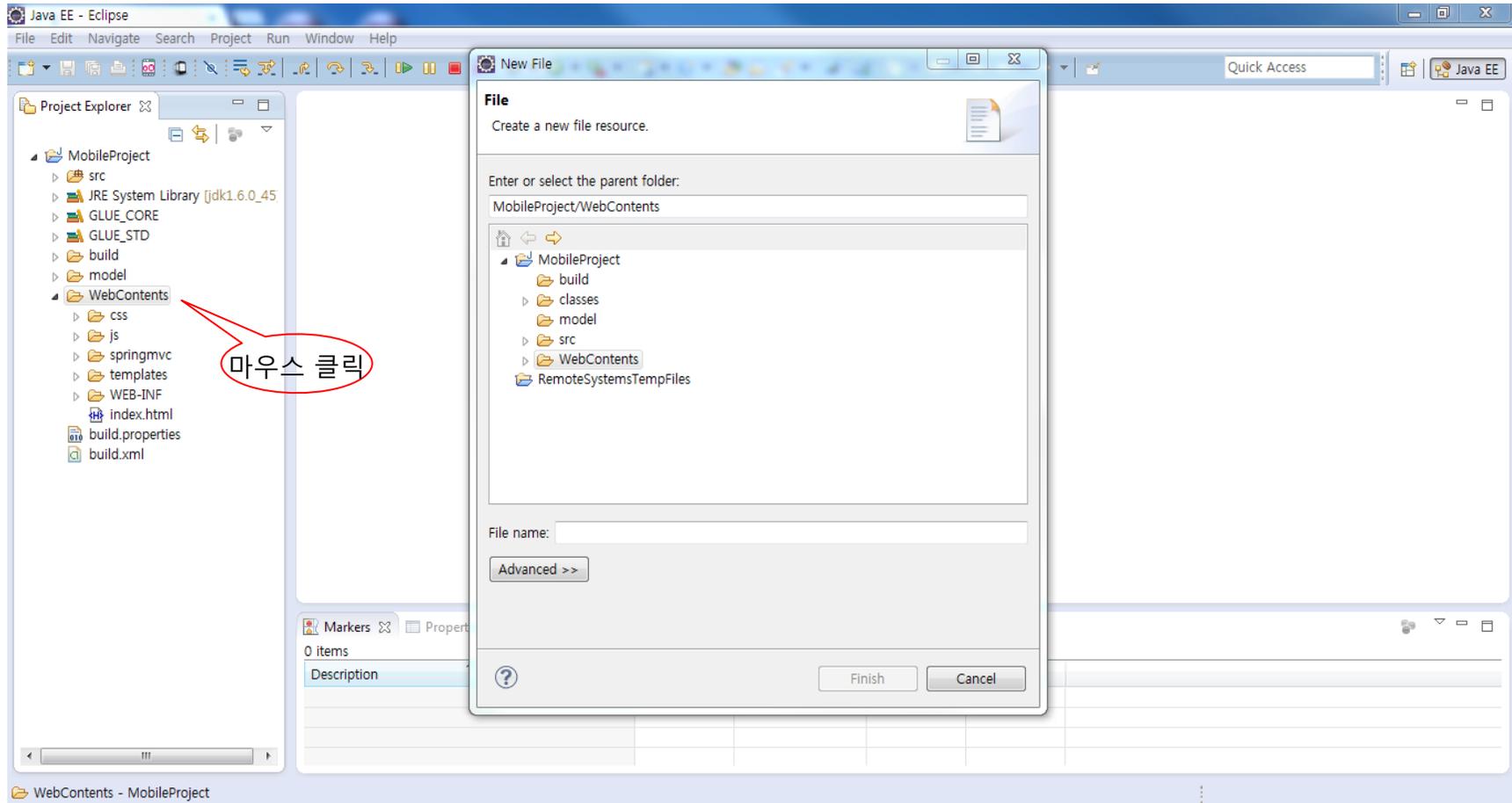
1. Html 페이지 생성
2. jQuery Mobile의 JS/CSS 기능을 사용하여 html페이지의 기본내용 구성
3. jQuery Mobile의 Page 기능을 사용하여 코드를 생성
4. Page의 Content 안에 HTML의 Text 기능을 사용하여 코드 생성
5. 추가한 Text 밑에 jQuery Mobile의 Button 기능을 사용하여 코드 추가.

※ 해당 기능들의 올바른 동작 여부 판단을 위해 매 기능 수행 후 Preview로 확인한다.

HTML & jQuery Mobile Palette 기능 활용 방법 (2)

■ HTML 파일 생성

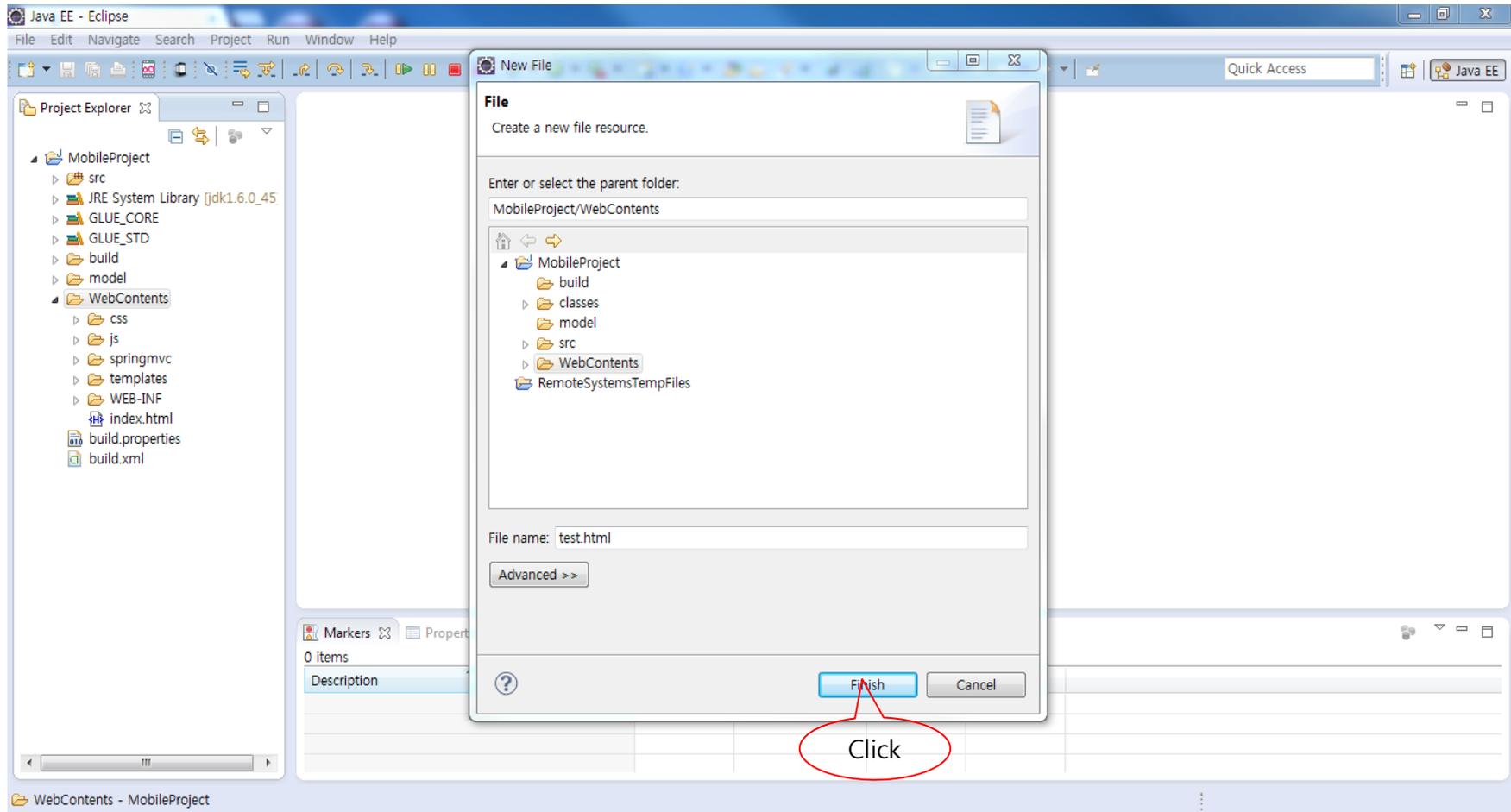
1) WebContents폴더에서 우측 마우스 Click -> New -> File



HTML & jQuery Mobile Palette 기능 활용 방법 (3)

■ HTML 파일 생성

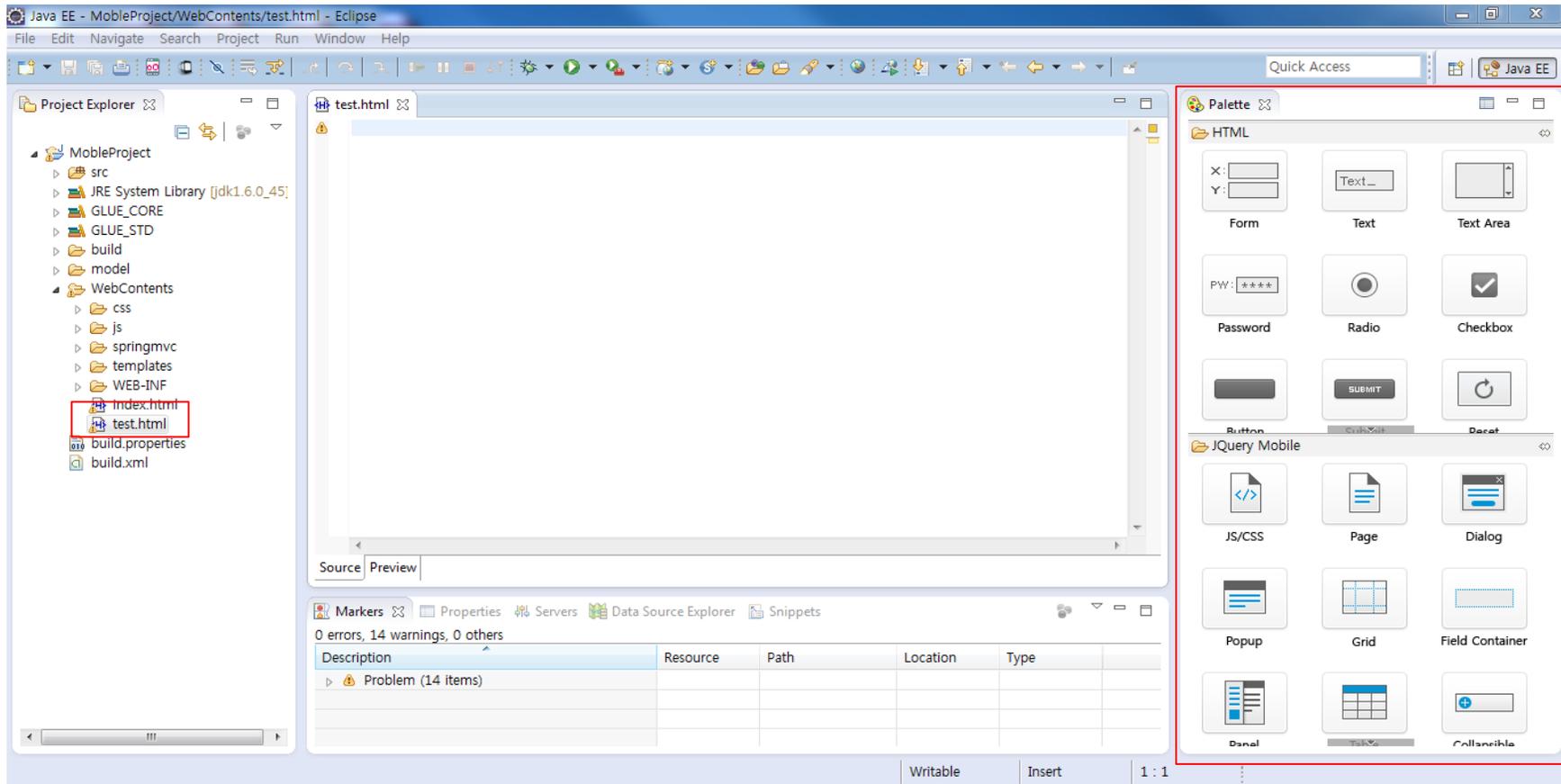
2) File name에 test.html 입력 -> finish Button Click



HTML & jQuery Mobile Palette 기능 활용 방법 (4)

■ Palette View 확인

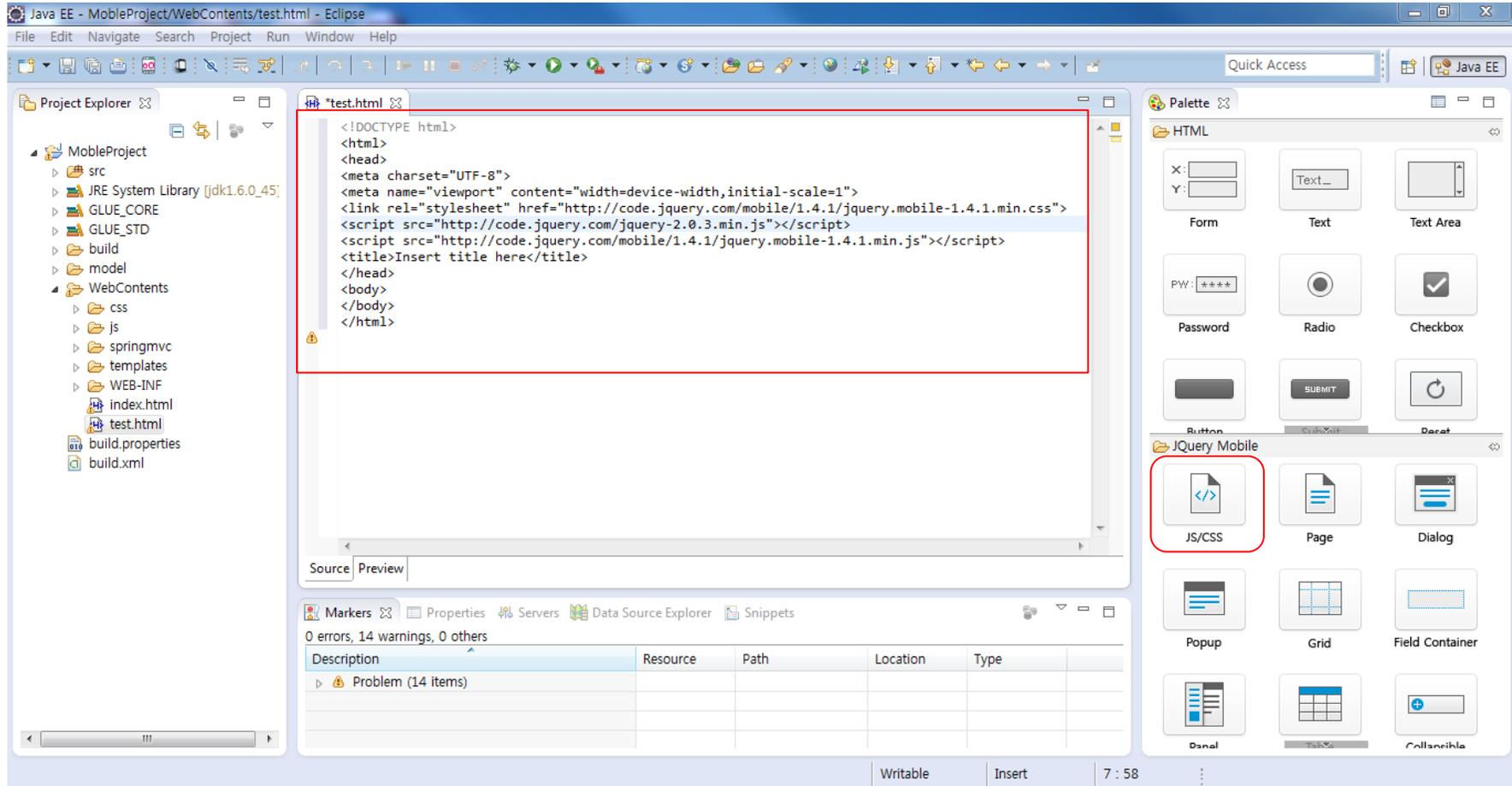
3) test.html 파일 생성 및 Palette view 확인



HTML & jQuery Mobile Palette 기능 활용 방법 (5)

■ Palette를 이용한 JS/CSS 생성

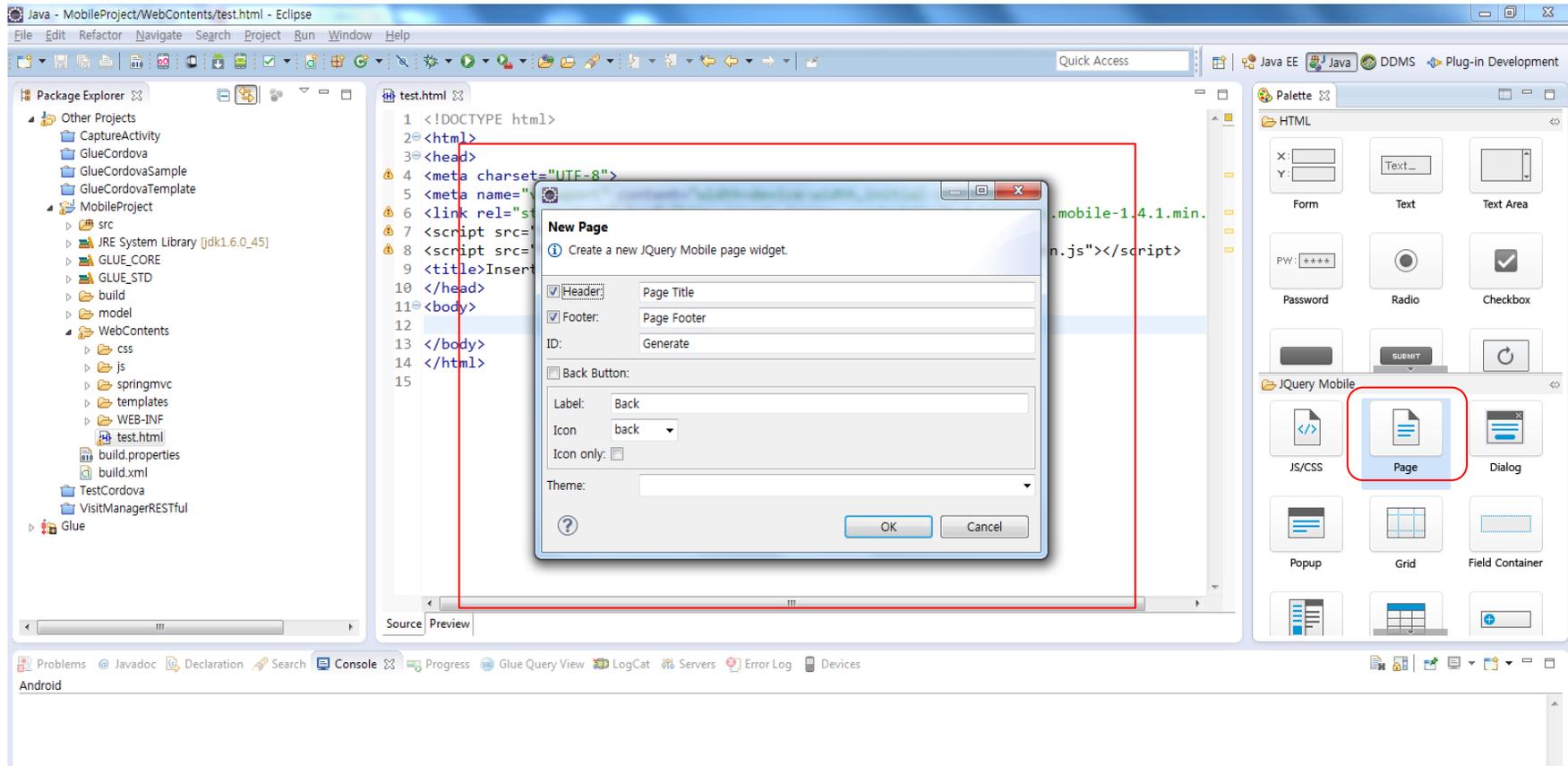
4) jQuery Mobile 기능을 사용하기 위해서는 JS/CSS 기능을 활용하여 필수 요소들을 불러온다.



HTML & jQuery Mobile Palette 기능 활용 방법 (6)

■ Palette를 이용한 Page 생성

5) Page 생성을 위해 Body Tag 부분에서 컴포넌트를 선택 후
Click -> PopUp Wizard생성 -> 입력 값 세팅 -> OK Click



※ 입력 값 없이 OK버튼 누를 시에는 Default값으로 출력

HTML & jQuery Mobile Palette 기능 활용 방법 (7)

■ Palette를 이용한 Page 생성

6) Body태그 안에 page 태그와 설정한 속성들이 생성되었는지 코드를 확인한다.

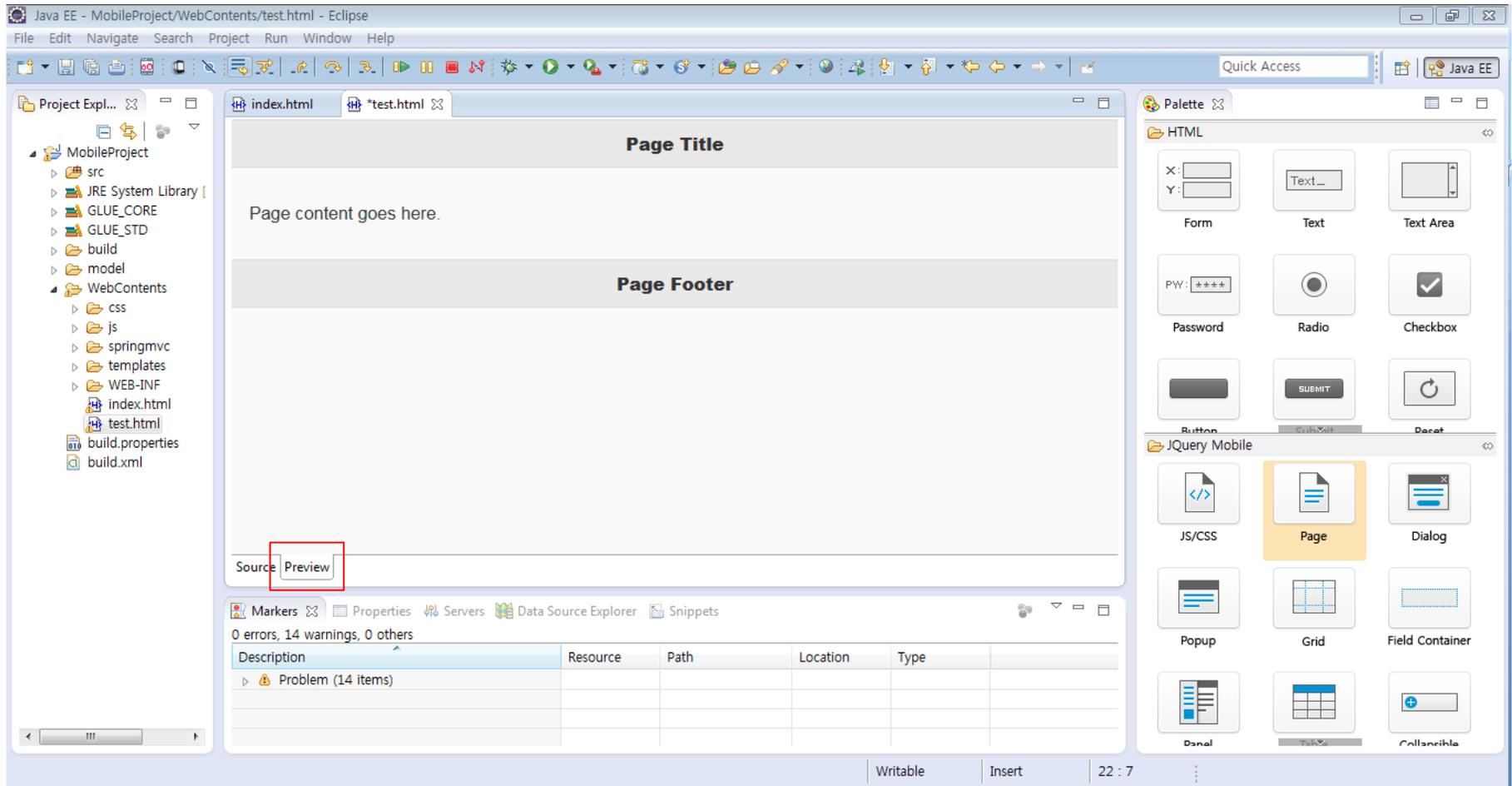
```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width,initial-scale=1">
<link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.1/jquery.mobile-1.4.1.min.css">
<script src="http://code.jquery.com/jquery-2.0.3.min.js"></script>
<script src="http://code.jquery.com/mobile/1.4.1/jquery.mobile-1.4.1.min.js"></script>
<title>Insert title here</title>
</head>
<body>
<div data-role="page" id="page-1">
<div data-role="header">
<h1>Page Title</h1>
</div>
<div data-role="content">
<p>Page content goes here.</p>
</div>
<div data-role="footer">
<h4>Page Footer</h4>
</div>
</div>
</body>
</html>
```

The screenshot shows the Eclipse IDE interface. The Project Explorer on the left shows the project structure. The main editor displays the HTML code for 'test.html'. A red box highlights the generated page structure in the body. The Palette on the right shows the 'Page' widget selected under the 'jQuery Mobile' category. The bottom status bar shows 'Writable', 'Insert', and '22:7'.

HTML & jQuery Mobile Palette 기능 활용 방법 (8)

■ Preview를 이용한 Page 레이아웃 확인

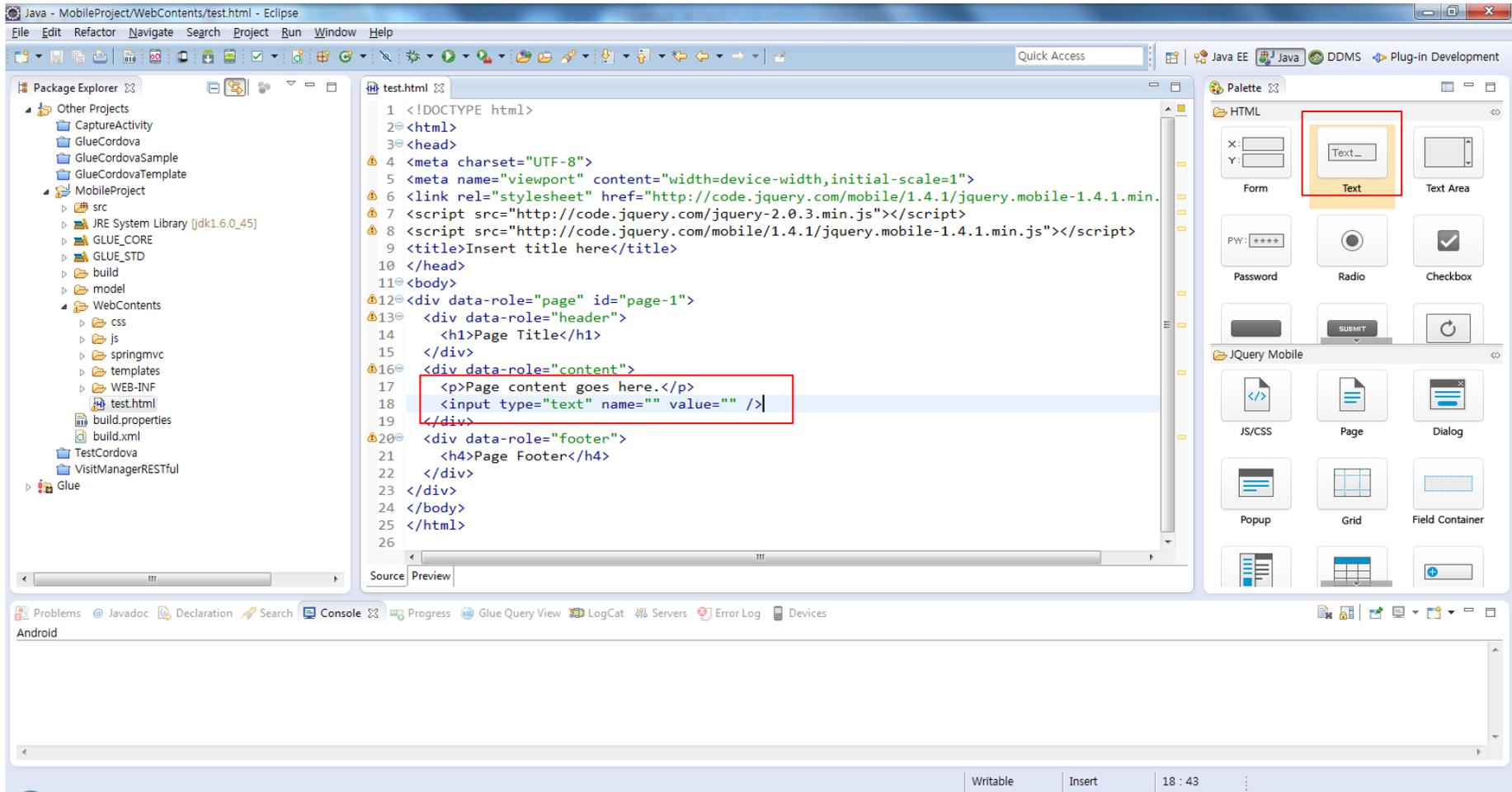
7) Preview 기능을 이용해 생성한 코드들의 Page 레이아웃을 확인한다.



HTML & jQuery Mobile Palette 기능 활용 방법 (9)

■ Palette를 이용한 Text form 생성

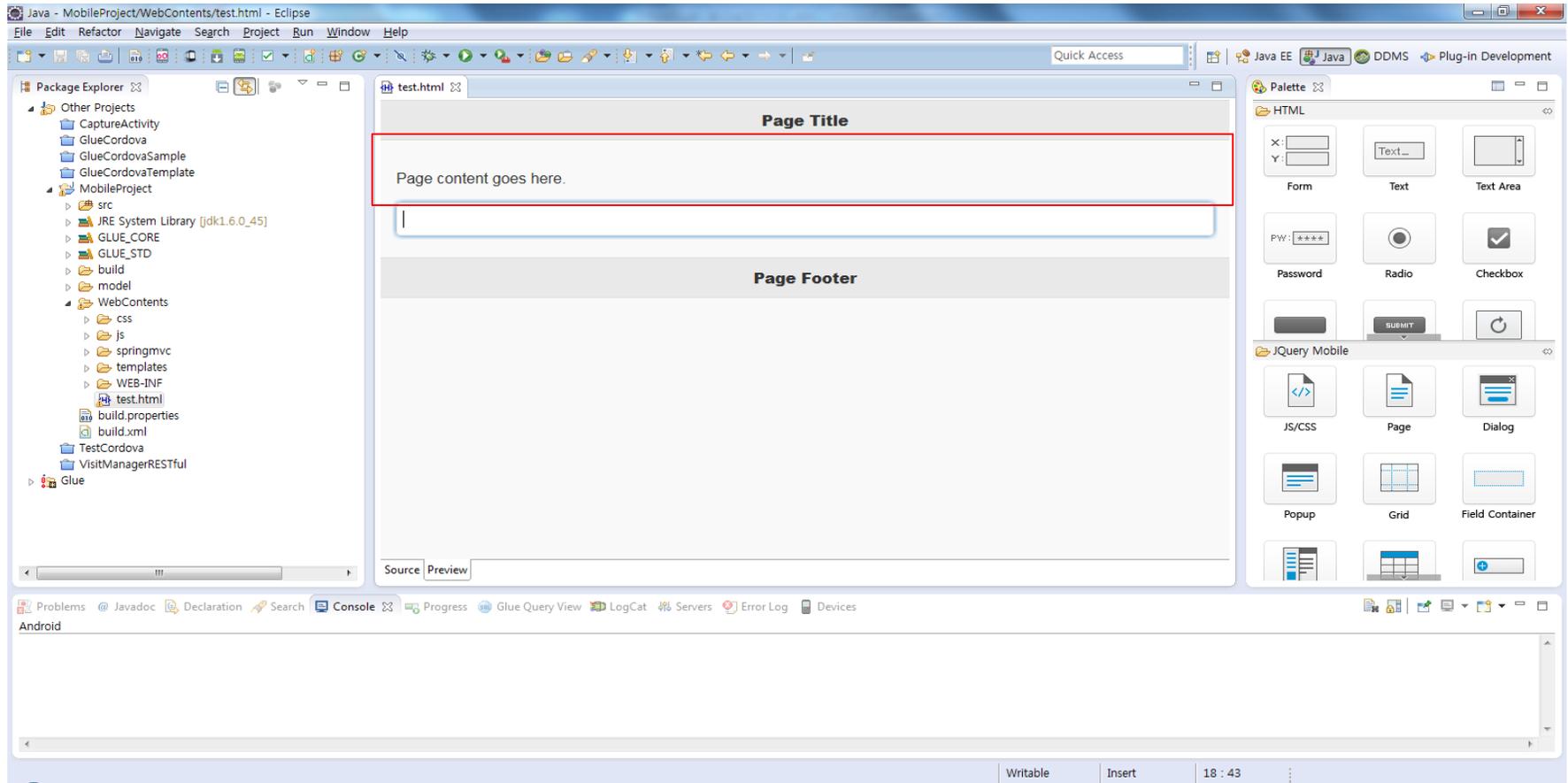
8) Palette의 HTML에서 Text 기능을 이용하여 Text Form을 생성한다.



HTML & jQuery Mobile Palette 기능 활용 방법 (10)

■ Preview를 이용한 레이아웃 확인

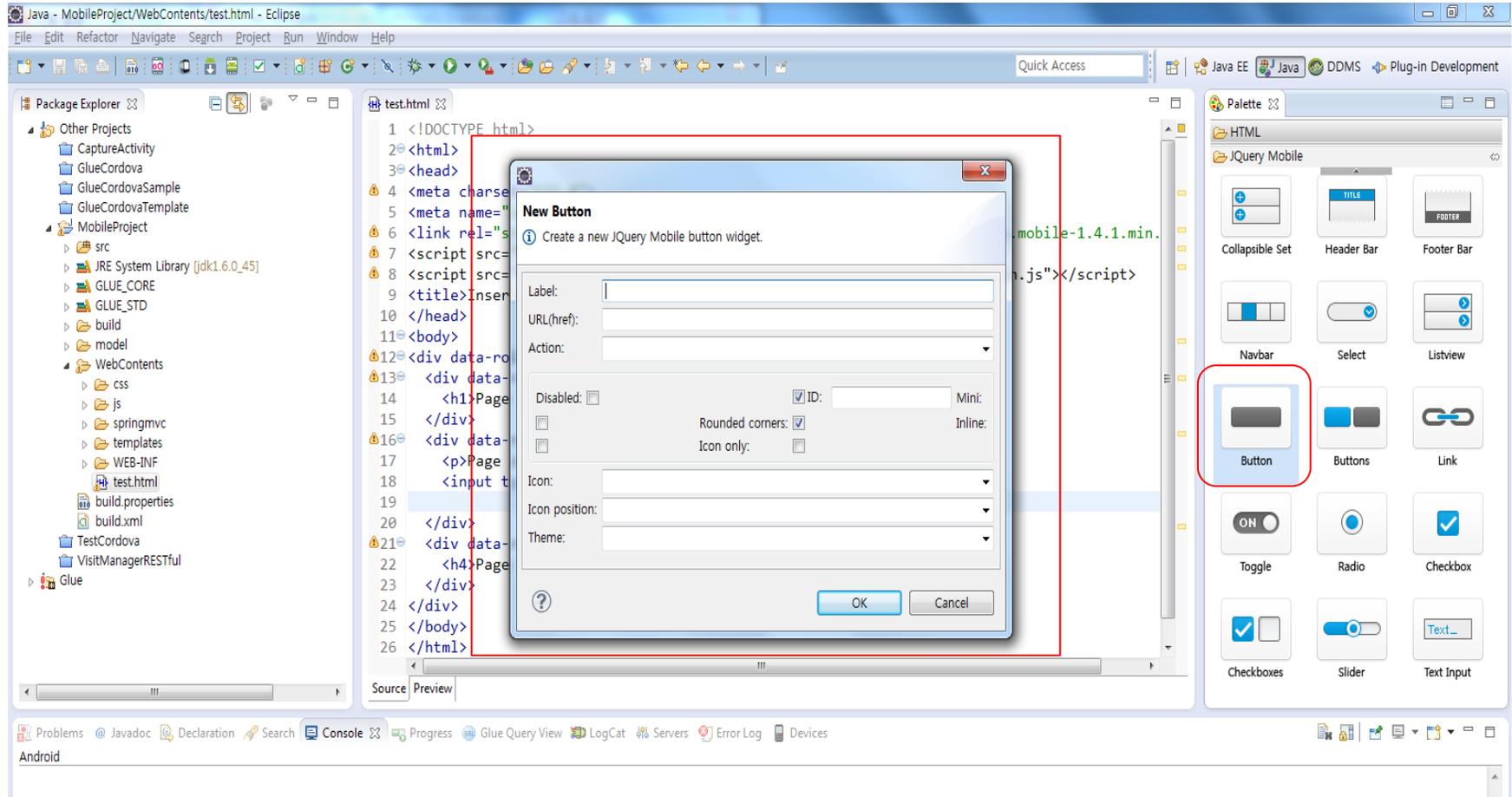
9) Preview 기능을 이용해 Page의 Content 속성 안에 Text Form이 생성되었는지 확인한다.



HTML & jQuery Mobile Palette 기능 활용 방법 (11)

■ Palette를 이용한 Button 생성

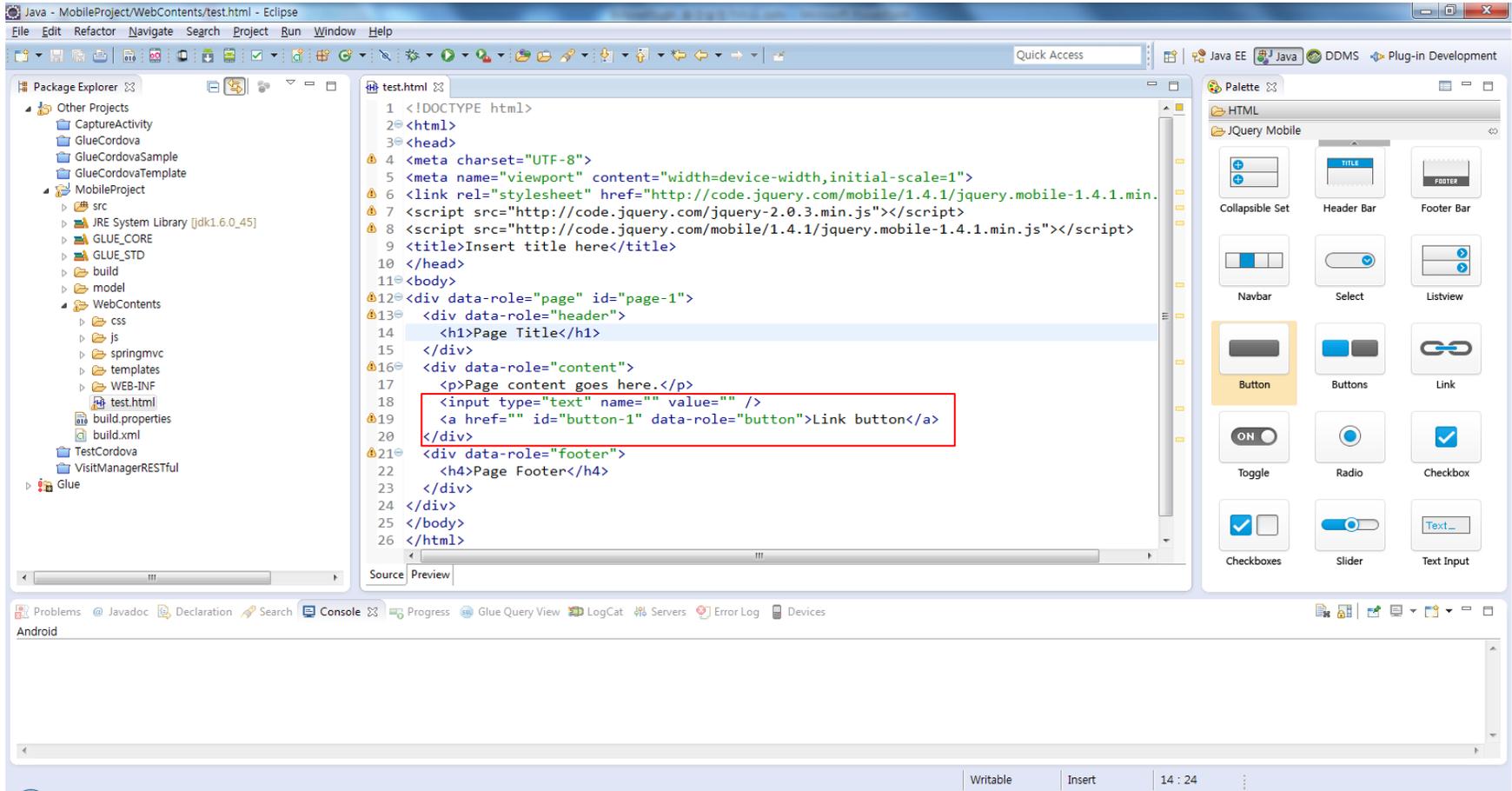
10) Palette의 jQuery Mobile에 있는 Button 기능을 이용하여 Text Form 밑에 버튼을 생성한다.



HTML & jQuery Mobile Palette 기능 활용 방법 (12)

■ Palette를 이용한 Page 생성

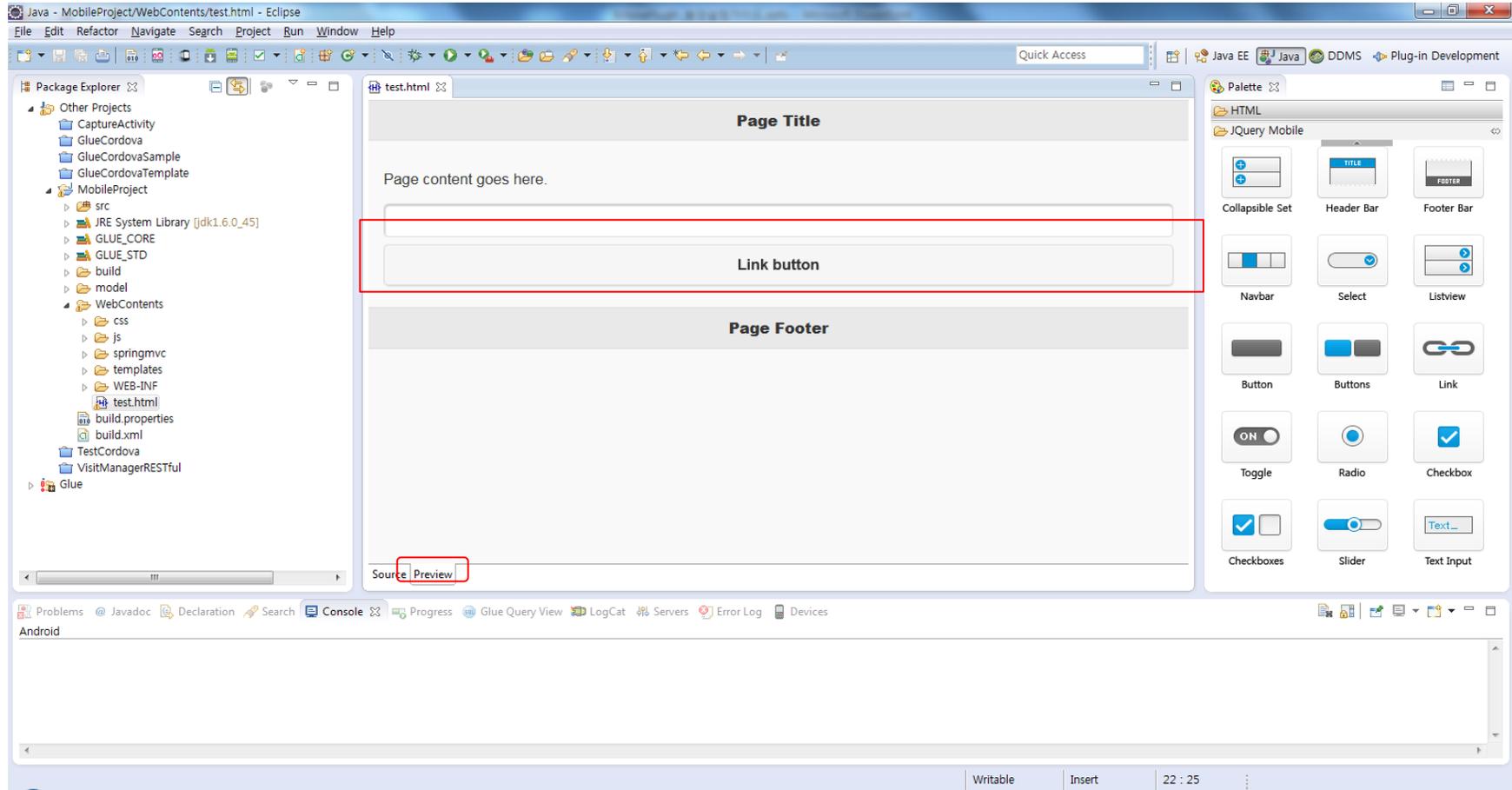
11) Text form 태그 아래에 button 태그가 생성되었는지 확인한다.



HTML & jQuery Mobile Palette 기능 활용 방법 (13)

■ Preview를 이용한 레이아웃 확인

12) Preview 기능을 이용하여 버튼이 Text form 밑에 생성 되었는지 확인한다.



◆ 부록 – Genymotion 설치 (AVD 대체)

■ Genymotion 사이트 접속

1. 이 경로로 이동하여 <http://www.genymotion.com/>에 접속 – 회원 가입

1

Team Features GENYMOTION Pricing **Sign in | Sign up**

THE FASTER ANDROID EMULATOR

Genymotion is the next generation of the AndroidVM open source project, already trusted by 450,000 developers. It's even easier to use and has lots more functionalities.

GET GENYMOTION

WIZARD ZARD <http://wizard.tistory.com/>

GENYMOTION EMULATOR

Is pure simplicity and performance. Genymotion makes your life easier!

Watch video

IN FEW WORDS

- Demo**
Do justice to your hard work. Demo the true quality of your applications on a powerful virtual device.
In 3 just clicks, your device will be ready to use.
- Test**
Control simulated sensors like battery, GPS, and accelerometer with a the user-friendly interface.
Powerful command line tools allow you to build complex tests.
- Performance**
Genymotion is an emulator using x86 architecture virtualization, making it much more efficient!
Taking advantage of OpenGL hardware acceleration, it allows you to test your applications with amazing 3D performance.
- Simplicity**
Perfectly integrates in your development environment with Eclipse plugin.
User-friendly interface.
Packaged for Windows, Mac and Linux.

2

GENYMOTION Discover Genymotion User Guide FAQ Support/Contact **Get Genymotion**

Login

Username or email address

*** Password

forgot your password?

LOG IN

Sign up

Username

email@domain.com

*** Password

*** Password confirmation

Allow Genymotion to send me emails about new releases

I accept terms of privacy statements

User profile (optional)

Enterprise size
Select a value

Usage type
Select a value

SIGN UP

WIZARD ZARD <http://wizard.tistory.com/>

Copyright © Genymobile 2013. All rights reserved.

◆ 부록 – Genymotion설치 (AVD 대체)

■ Genymotion 다운로드



Get a license: to use Genymotion for commercial use, or to get access to all features.

Release note

Download ready-to-run Genymotion installer for Windows

This version includes Oracle VirtualBox 4.2.12 dependency, so that you don't need to download and install VirtualBox manually

Windows 32/64 bits (with VirtualBox) v2.2.2

	Free	Trial	Indie	Business
Android 4.1 & 4.2 These are our main Android flavours.	✓	✓	✓	✓
Android 2.3 & 4.3 These are our new Android Flavours. Enjoy them! NEW	✓	✓	✓	✓
Android 4.4 We are finishing baking this hot new Flavour... Be ready! SOON	✓	✓	✓	✓
Battery Widget Make sure your apps are power efficient.	✓	✓	✓	✓
GPS Widget Easily build location enhanced apps	✓	✓	✓	✓
Camera Widget Use your computer webcam	✓	✓	✓	✓
Screencast Widget Record a video file of your app with only one click NEW		✓	✓	✓
Remote Control Widget Control Genymotion from your hand, with departed video NEW		✓	✓	✓
Accelerometer Build motion enhanced apps with a real accelerometer		✓	✓	✓
Remote Control Widget Multitouch Control Genymotion from your hand, with departed video NEW		✓	✓	✓
Standard support Get help from our support team about any question or problem	✓	✓	✓	✓
Premium Support Priority queue, reply guaranteed in 24h max. NEW				✓
	Free!	0 € ⁽¹⁾	99 € ⁽²⁾ VAT per year per user	299 € ⁽²⁾ VAT per year per user
	DOWNLOAD	DOWNLOAD FOR FREE	BUY	BUY

Indie License is restricted to only one licence per company and only for companies with less than 3 employees.

⁽¹⁾ The Trial license will allow you to test the Indie features until December 15th, 2013. This license is not renewable.
⁽²⁾ Conversions to US Dollars: Indie: \$ 136 Business: \$ 412
All prices in US dollars are indicative. The final transaction will be in Euros. Your bank may charge you transaction fees for currency conversions.

◆ 부록 – Genymotion 설치 (AVD 대체)

■ Genymotion 설치

1. 1-2-3번 수행 후 재부팅을 수행

1

GENYMOTION ^{OO} Discover Genymotion User Guide Contact

WIZARD _{zard} ^{Center}
<http://wizdard.tistory.com/>

Get a license to use Genymotion for commercial use, or to get access to all features, you need to get a license.

Release note

Download ready-to-run Genymotion installer for Windows

This version includes Oracle VirtualBox 4.2.12 dependency, so that you don't need to download and install VirtualBox manually

Windows 32/64 bits (with VirtualBox)	v2.0.1	↓
--------------------------------------	--------	---

Other platforms and versions

These versions needs to have Oracle VirtualBox installed to run virtual devices.

Windows 32/64 bits	v2.0.1	↓
Mac OS X 64 bits	v2.0.1	↓
Linux 32 bits (Ubuntu, Fedora, Redhat)	v2.0.1	↓
Linux 64 bits (Ubuntu, Fedora, Redhat)	v2.0.1	↓
Linux Debian 64 bits	v2.0.1	↓

The standard Linux 32/64 bits installers packages available on this download page are not compatible with Debian systems.
So, this package is specially designed for Debian systems (Debian Squeeze, Debian Wheezy, Debian Mint, ...) and only for 64 bits architecture.
Warning: for an optimal renderer, we recommend you to install the following package: libpngX (X depends of your Debian version)

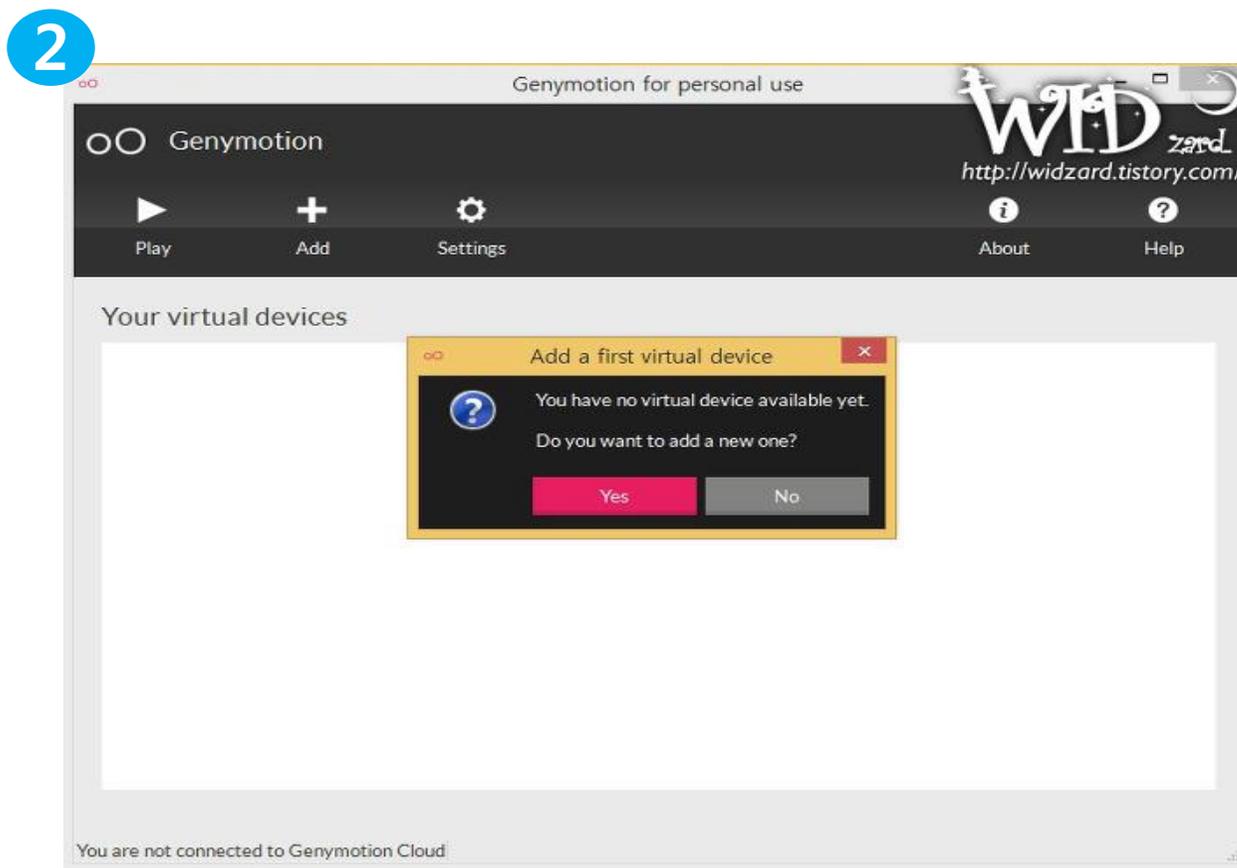
2

3

◆ 부록 – Genymotion설치 (AVD 대체)

■ Genymotion 실행

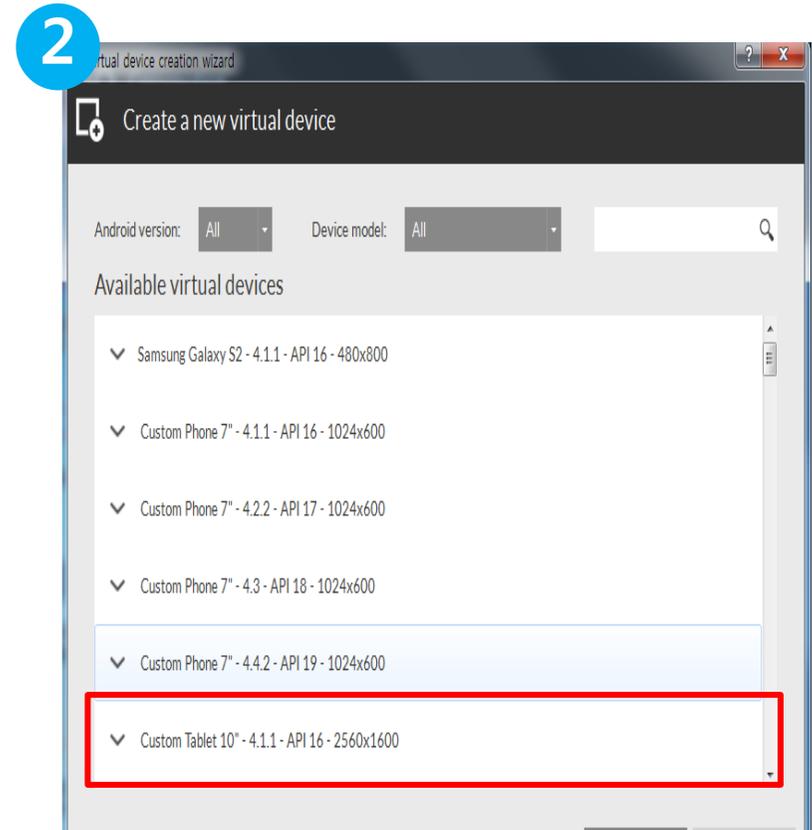
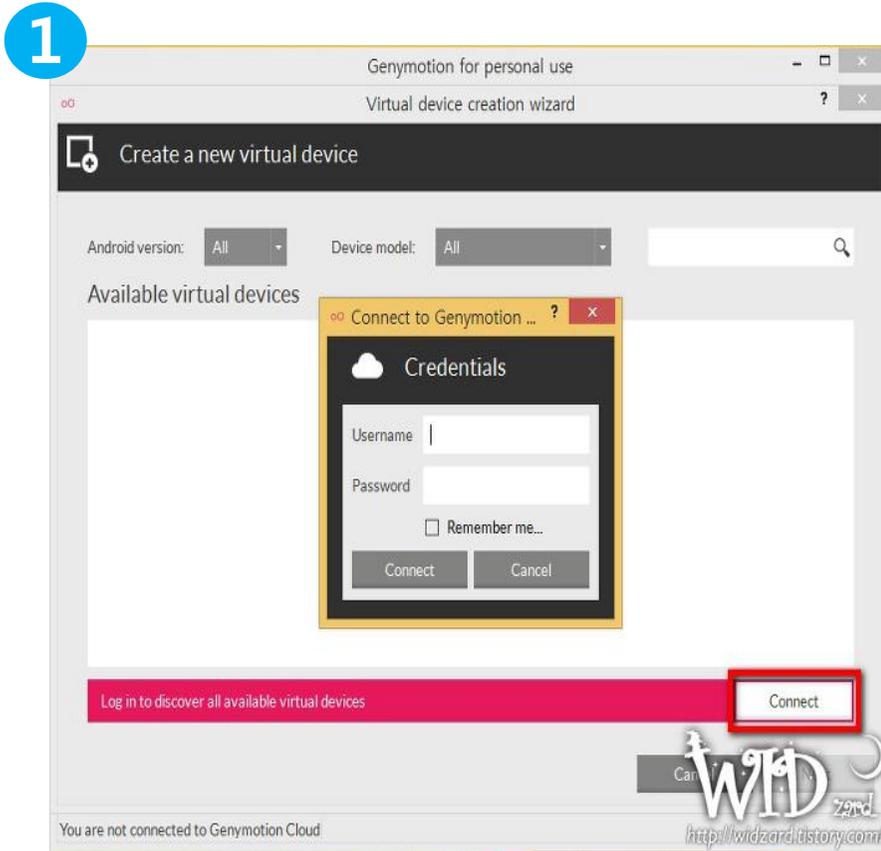
1. 바탕화면에 아이콘 클릭 후 2번 수행



◆ 부록 – Genymotion 설치 (AVD 대체)

■ Genymotion Virtual device 추가

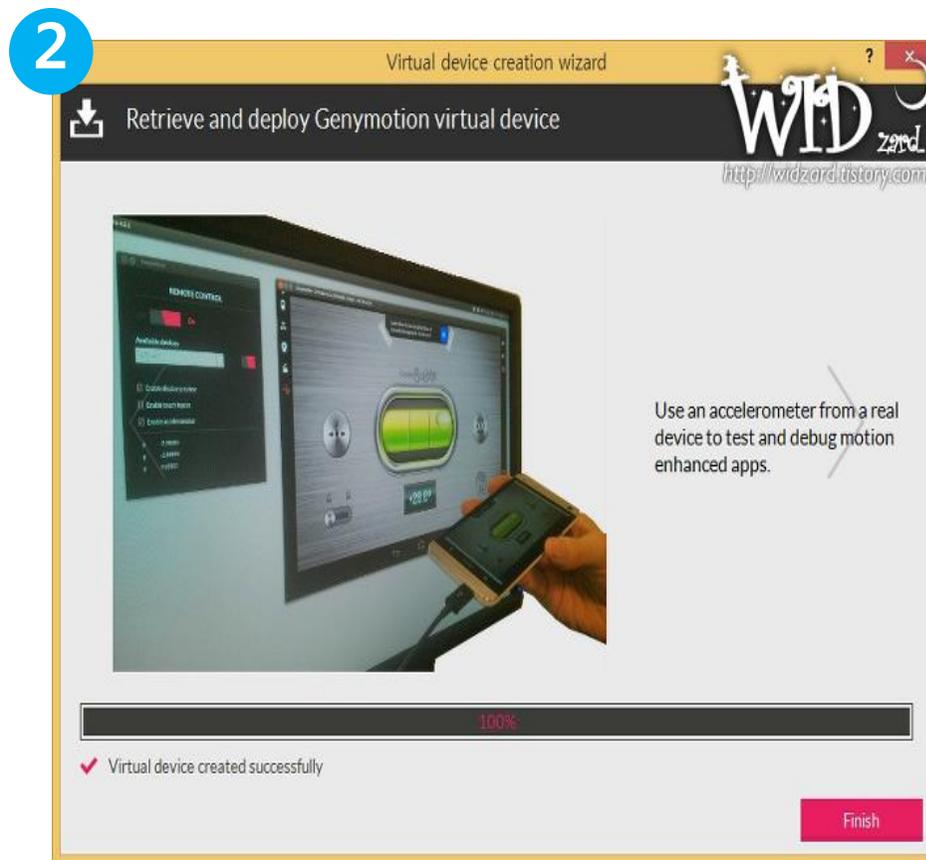
1. Connect 버튼을 누르면 다음과 창이 뜹니다. 최초 Genymotion 가입했던 Email / Password를 입력 후 4.4.2 선택



◆ 부록 – Genymotion설치 (AVD 대체)

■ Genymotion Virtual device 추가

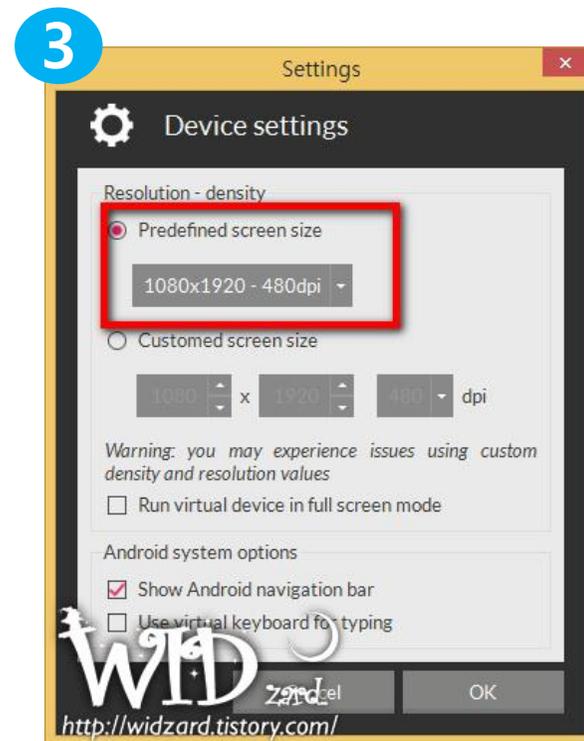
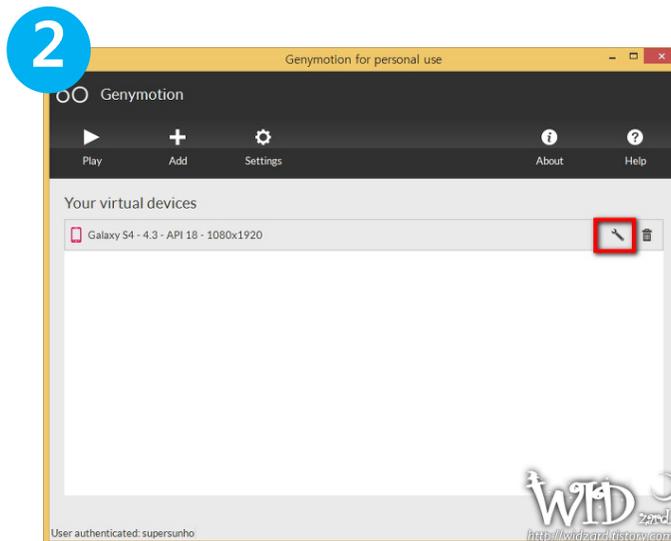
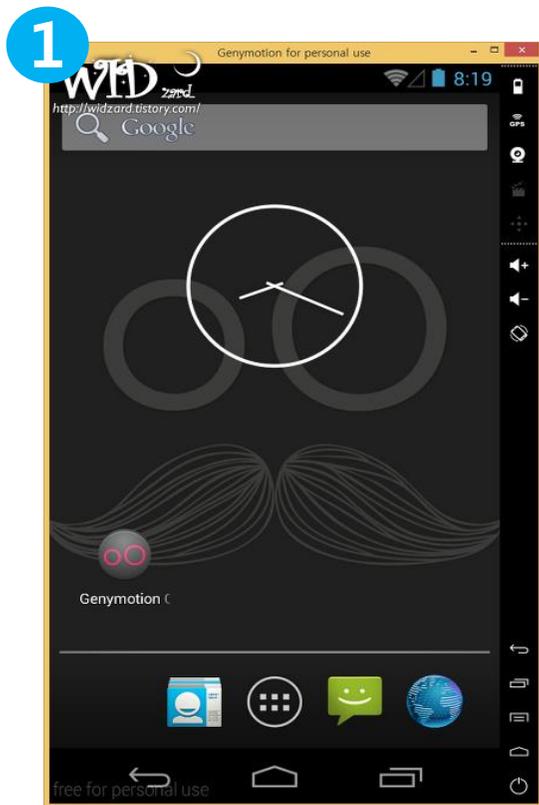
1. 1,2번 완료 후 Finish를 눌러 설치를 끝냄



◆ 부록 – Genymotion설치 (AVD 대체)

■ Genymotion Virtual device 설정

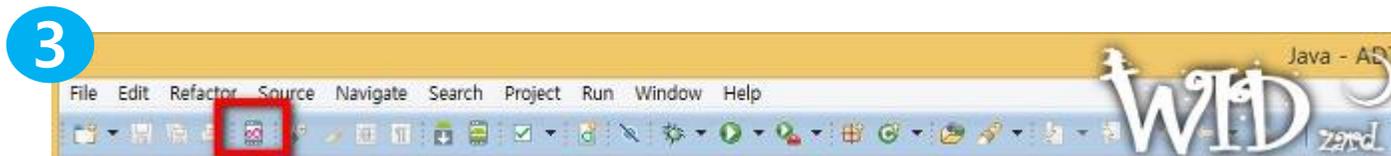
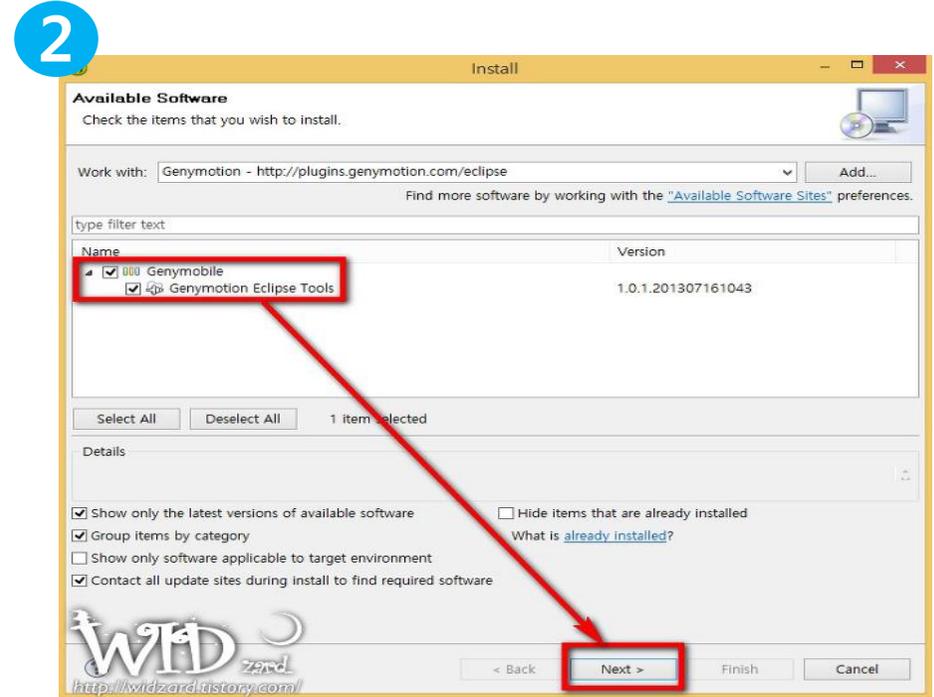
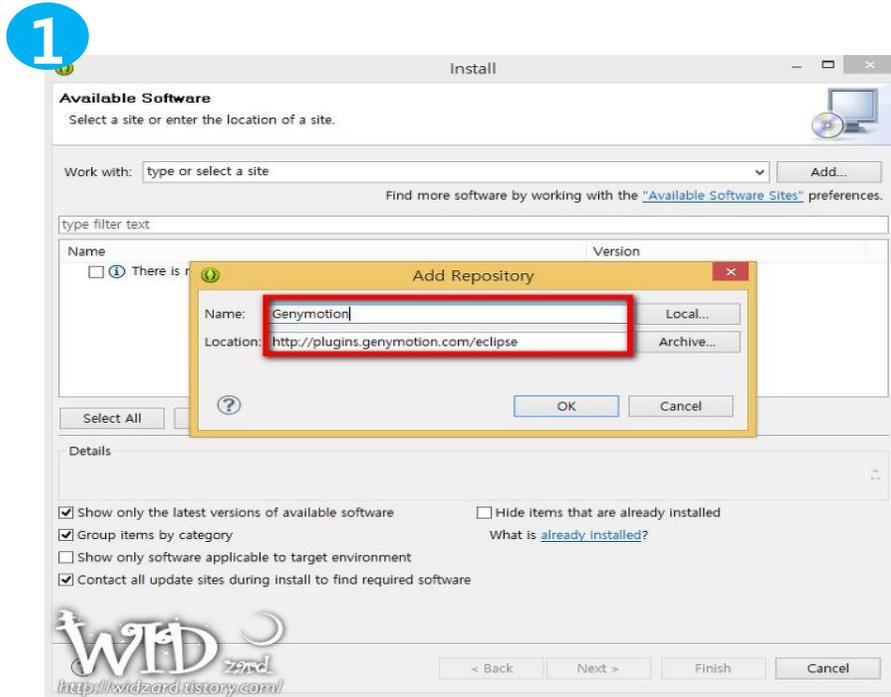
1. Play버튼을 눌러 실행 시 사이즈 조절이 안되어 큰 화면이 뜬
2. 설정 버튼 클릭
3. 사이즈 조절 수행



◆ 부록 – Genymotion 설치 (AVD 대체)

■ 이클립스 Genymotion 플러그인 설치

1. 이클립스 실행하여 상단에 Help 누른 뒤 Install New Software 클릭
2. Next Click



◆ 부록 – Genymotion설치 (AVD 대체)

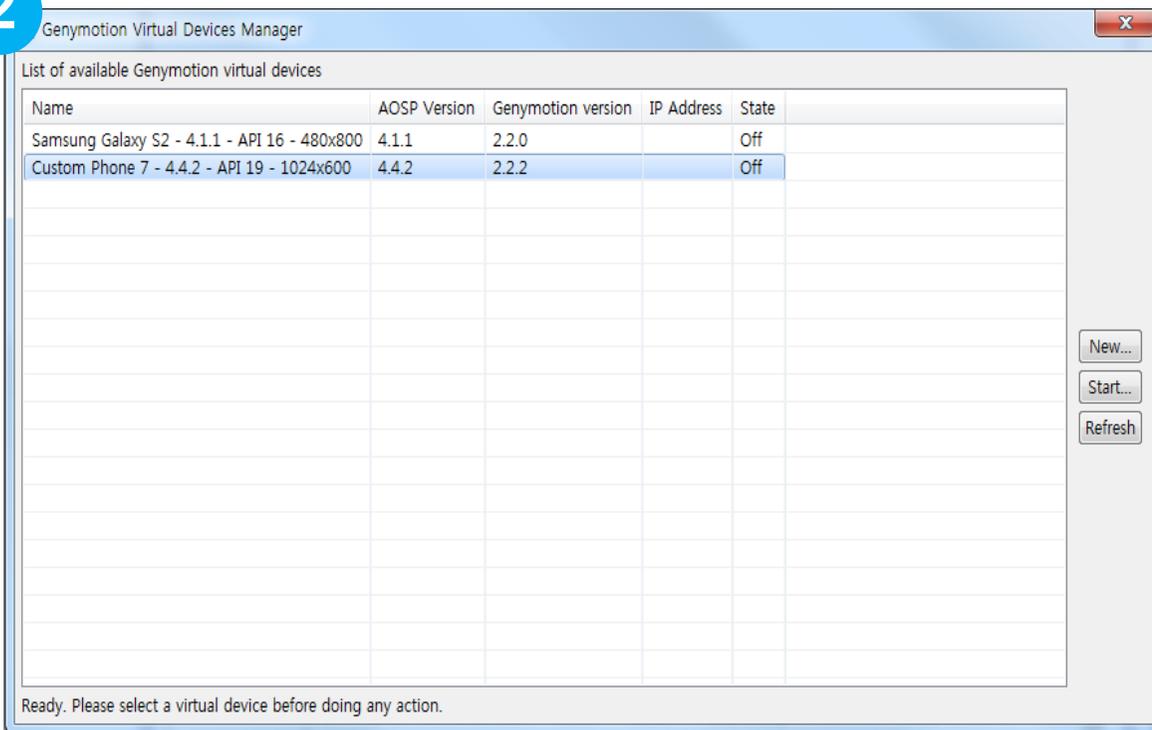
■ 이클립스 Genymotion 플러그인 설치

1. 1-2-3수행 후 Eclipse 창에서 Ctrl + F11 누르면 Device선택 되어 Genymotion과 연동됨

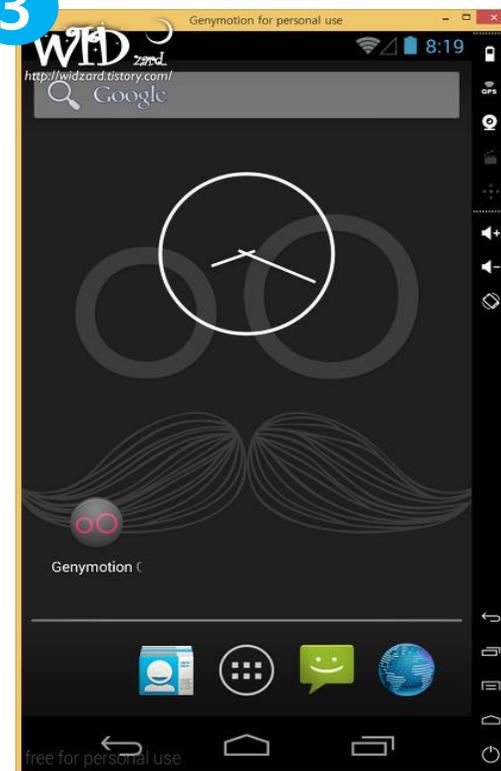
1



2



3



모바일UI 개발

1. *모바일UI 개요*
2. *모바일UI 시작하기*
3. *페이지 이동 하기*
4. *서버 연동 하기*
5. *차트 그리기*

개요 - GlueJS란?

■ GlueJS 란?

모바일 사용자 인터페이스를 개발하기 위한 자바스크립트 라이브러리로
MVC 설계 패턴을 적용하여 구조적 설계와 모듈화를 지원하며
모바일에 최적화된 다양한 사용자 위젯을 제공한다.

■ 기술 특징

- **Single Page Application**

클라이언트 렌더링과 라우팅을 수행하며 좋은 사용자 경험과 성능향상 제공

- **MVC 설계 패턴**

기본적으로 Model-View-Controller 설계 패턴을 적용하여 로직을 분리

- **코드 템플릿 적용**

HTML 태그를 템플릿 형태로 Model과 바인딩 하여 화면 생성

- **모바일 최적화된 UI 위젯 제공**

모바일에 최적화된 터치 기반의 사용자 인터페이스 제공 및 다양한 플랫폼 지원

- **모듈화 및 비동기식 로딩**

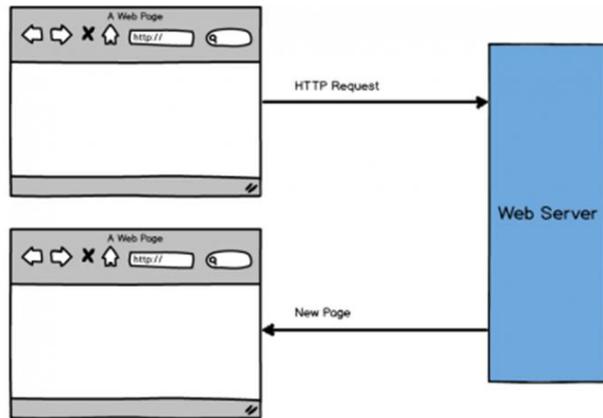
Model, View 등의 논리 모듈 단위로 자바스크립트 파일을 물리적으로 분리

개요 - 기술 특징

■ Single Page Application 란?

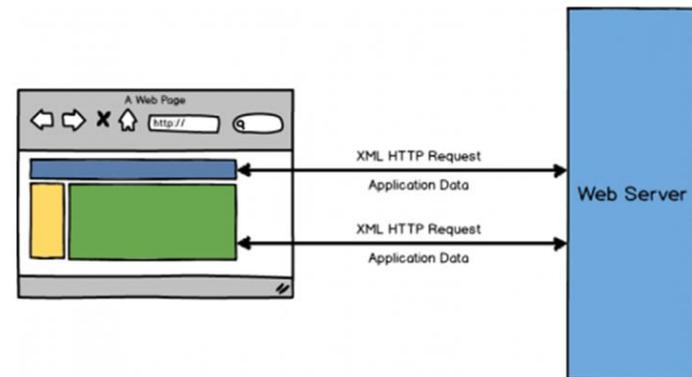
- 정의 : 사용자가 브라우저에서 사용하고 있는 동안 페이지 재로딩 없이 동작
- 특징 : 사용자 상호작용 및 응용 프로그램의 상태 변화가 단일 웹 도큐먼트에서 처리
예시) Gmail, Google Maps, Google Drive

일반 웹의 문제점



- 나쁜 사용자 경험
- 성능 저하
- 오프라인 지원의 부족 (??)

Single Page App의 특징



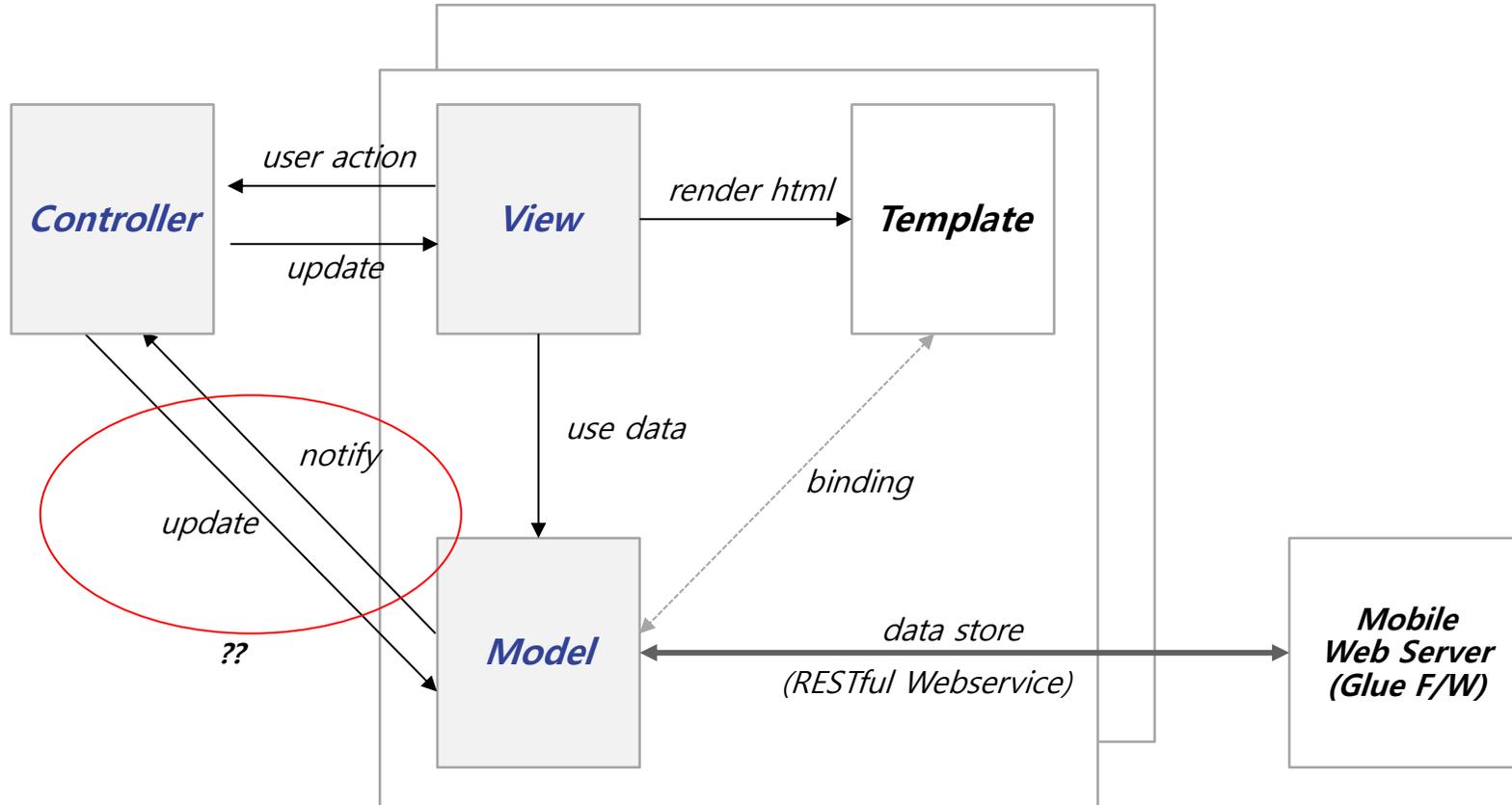
- 사용자 경험 및 성능 향상 ↑



- Single Page Application 참고 사이트 : <http://todomvc.com/examples/backbone/#/>

개요 - 기술 특징

■ MVC 설계 패턴



■ MVC 설계 패턴

Model

- ✓ *Represents knowledge or data*
- ✓ *Talks to server using RESTful architecture*
- ✓ *Isolated from controllers and views*
- ✓ *Can be gathered into groups (**collections**)*

View

- ✓ *Considered to be specific elements of UI (table record)*
- ✓ *is generated from template*
- ✓ *One model might have multiple views*

Controller

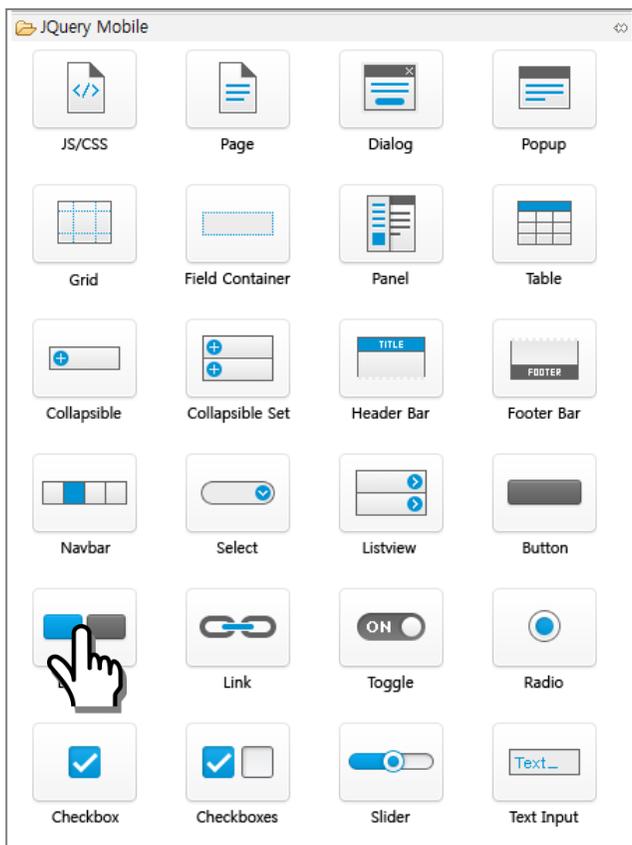
- ✓ *Glue between Model and View*
- ✓ *Handles user interactions*
- ✓ *Might perform business logic role*

개요 - 기술 특징

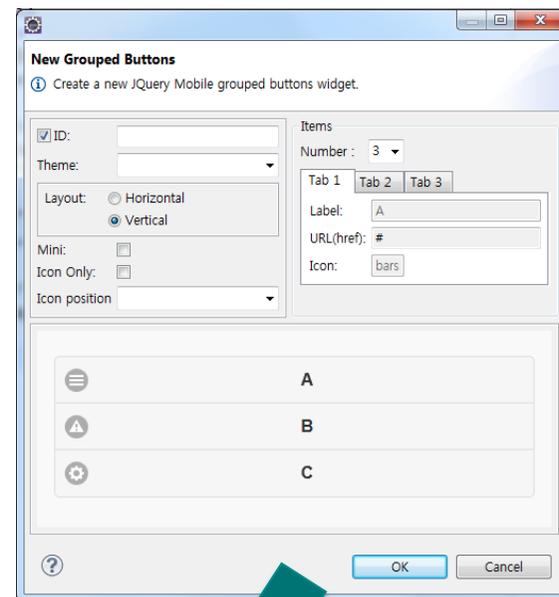
■ 모바일 최적화된 UI 위젯 제공 - 이클립스 파레뜨

- 이클립스 파레뜨 기반으로 모바일 사용자 인터페이스를 개발하도록 지원

1 파레뜨 클릭



2 소스 코드 편집창



3 소스코드 생성

```
<div data-role="controlgroup">  
<a href="#" data-role="button" data-icon="bars">A</a>  
<a href="#" data-role="button" data-icon="alert">B</a>  
<a href="#" data-role="button" data-icon="gear">C</a>  
</div>
```

■ UI 화면 개발 생산성을 올려주는 자바스크립트 라이브러리 - jQuery

• jQuery란?

DOM 요소 조작을 더 수월하게 하고, 이벤트 처리 / 애니메이션 / Ajax Call 과 같은 기능을 더 쉽게 수행할 수 있도록 여러 API들이 내장되어 있는 자바스크립트 라이브러리이다. 다양한 브라우저에서 사용할 수 있는 장점을 지니고 있다.

▪ jQuery 맛보기

`$(document)` :

- 1) \$ -> 선택자
- 2) document -> 접근할 요소

`$("#p")` :

- 1) \$ -> 선택자
- 2) p -> html p 태그

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
<script>
$(document).ready(function(){
  $("#p").click(function(){
    $(this).hide();
  });
});
</script>
</head>
<body>

<p>If you click on me, I will disappear.</p>
<p>Click me away!</p>
<p>Click me too!</p>

</body>
</html>
```

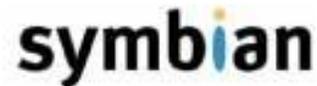
jQuery 라이브러리 로딩

코드결과 확인 : http://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_hide

■ 모바일 최적화된 UI 위젯 제공 - jQuery Mobile

- jQuery Mobile 이란?

모바일 웹 앱을 만들기 위한 터치 이벤트 처리에 최적화된 모바일 웹 프레임워크이며 모든 스마트폰과 태블릿에서 작동한다. jQuery Mobile을 사용하기 위해서는 HTML5, CSS3, JavaScript & jQuery 에 대한 기초적 지식이 필요하다.



■ 모바일 최적화된 UI 위젯 제공 - jQuery Mobile

- jQuery Mobile 이란?



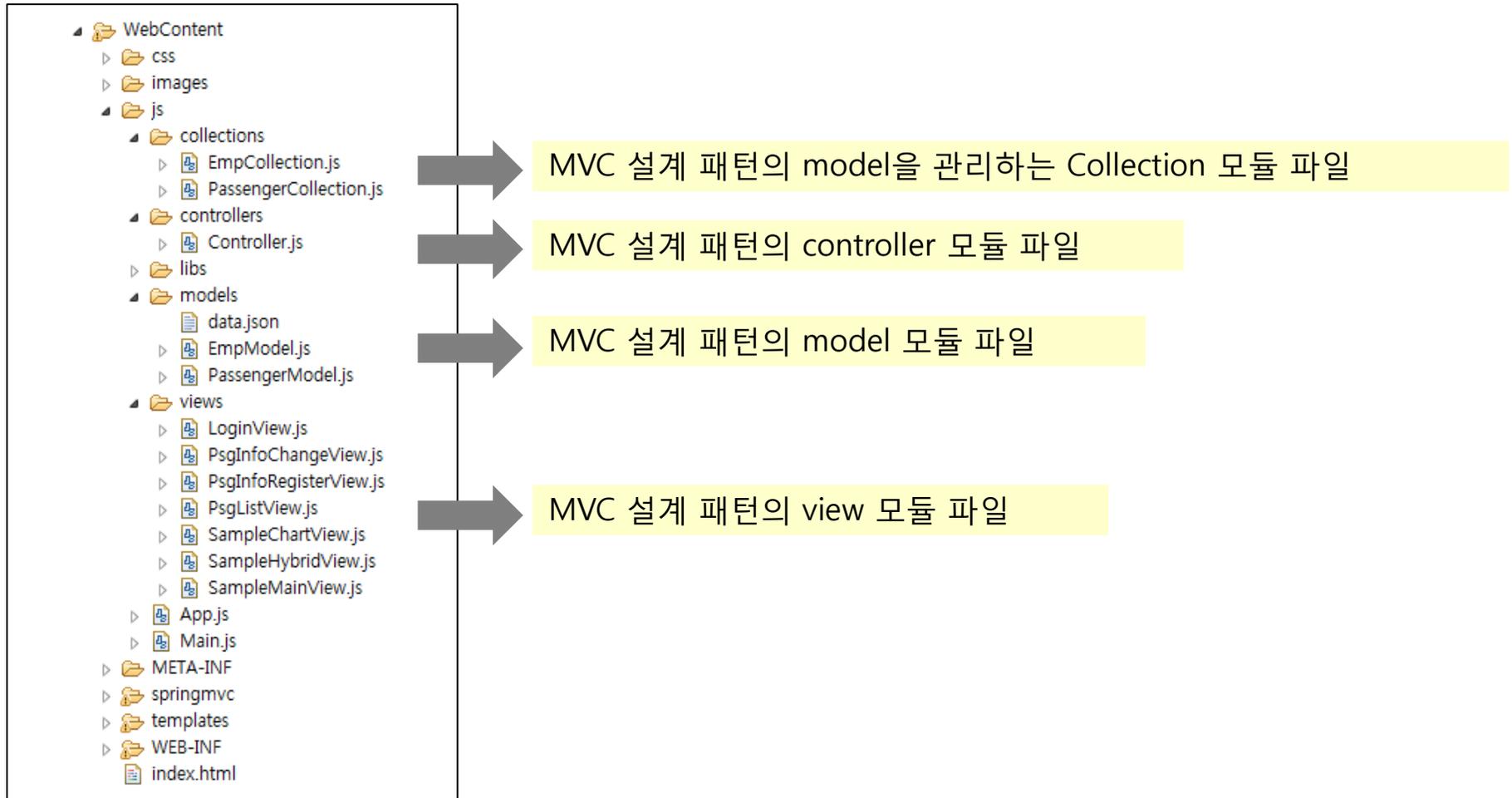
※ 참조 : jQuery Mobile 컴포넌트 데모 사이트

<http://demos.jquerymobile.com/1.4.5/>

개요 - 기술 특징

■ 모듈화 및 비동기식 로딩

- Model-View-Controller 모듈 단위로 자바스크립트 파일이 물리적으로 분리하여 관리



■ 모듈화 및 비동기식 로딩

• RequireJS 란?

JavaScript 파일과 모듈을 로딩하고 의존관계를 관리하는 라이브러리로 GlueJS에서 기본적으로 사용되기 때문에 간단한 설정 및 사용방법을 알아본다.

기타 자세한 사항은 RequireJS 홈페이지 <http://requirejs.org> 를 참고한다.

• index.html 설정 하기

개발하는 Application의 index.html파일에 script태그로 다음과 같이 설정한다.

RequireJS가 모두 로딩되고 해당 경로에 main.js를 자동으로 로딩한 뒤 실행한다.

```
01 <!DOCTYPE html>
02 <html>
03 <head>
04   <meta charset="UTF-8">
05   <title> </title>
06 </head>
07 <body>
08 </body>
09 <script data-main="js/main" src="js/libs/require.js"> </script>
```

■ 모듈화 및 비동기식 로딩

• Main.js

require.js는 여러 설정 옵션들을 제공한다.

```
01  require.config({
02      baseUrl: 'js',
03      paths: {
04          'jquery': 'libs/jquery',
05          'jquerymobile': 'libs/jquerymobile',
06          'underscore': 'libs/underscore',
07          'backbone': 'libs/backbone',
08          'handlebars': 'libs/handlebars',
09          'glue': 'libs/glue'
10      },
11      shim: {
12          'glue': {
13              deps: ['jquery', 'underscore', 'backbone', 'handlebars'],
14              exports: 'GlueJS'
15          }
16      }
17  });
```

- require.config : require.js 설정 선언, require.js가 로딩되면 자동으로 읽어 반영
- baseUrl : 모듈의 기본 경로 위치를 지정
- paths : 모듈의 단축 경로 지정 및 이름에 대한 alias를 정의
- shim : dependencies, exports 등을 설정하는 속성, 특정 라이브러리들의 의존성을 설정할 수 있다.
ex) glue의 경우 deps에 설정된 라이브러리 4개가 모두 먼저 로딩이 되어야, 로딩된다.

■ 모듈화 및 비동기식 로딩

• 사용방법

require.js 를 이용 모듈을 정의하여 사용하는 기본형태는 아래와 같다.

```
01 define([
02     'jquery',
03     'glue',
04     'text!templates/template.html'
05 ], function($, GlueJS, template) {
06     console.log('callback function 수행');
07 });
```

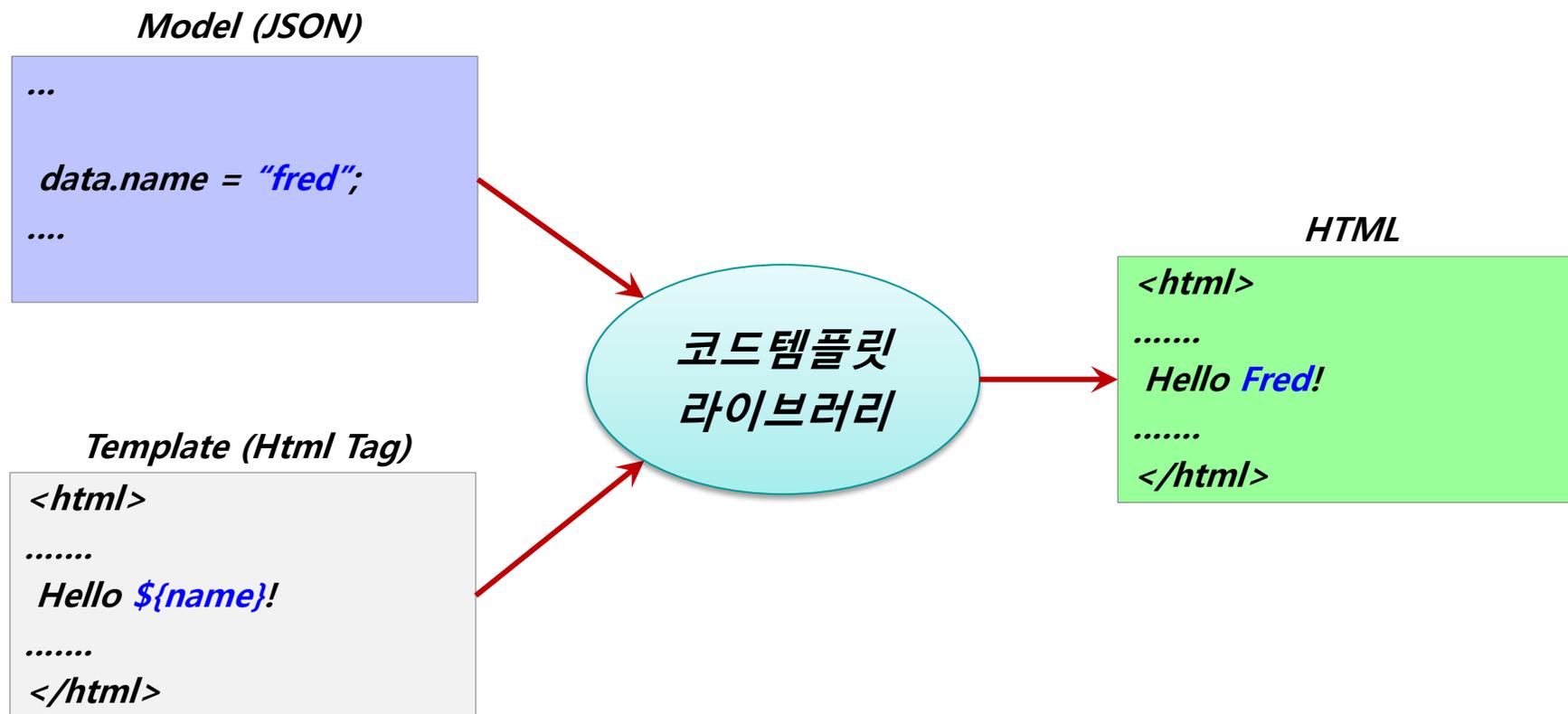
- define : require.js에서 모듈을 정의 할때 사용한다.
 - 첫번째 인자는 사용하고자 하는 라이브러리 또는 모듈 배열
 - 두번째 인자는 모듈 로딩 후에 수행되어야 하는 함수가 정의 된다.
- function : 로딩을 완료한 모듈 및 파일을 function에서 사용
 - function에 전달 받은 인자는 define의 첫번째 인자에 정의 되어 있는 모듈 배열을 순서대로 전달
- text! : CSS와 HTML같은 텍스트 파일을 불러오기 위해 require.js 의 text플러그인을 사용

개요 - 기술 특징

■ 코드 템플릿의 기본 개념

코드 템플릿 엔진을 사용하면 모델(데이터)과 뷰를 분리 할 수 있다.

즉 순수한 정보와 표현정보를 분리하고 이 둘을 합쳐 결과물을 만들어 낼 수 있다.



■ 코드 템플릿 적용

• Underscore.js 란?

흔한 프로그래밍 로직 처리를 위해 여러 기능들을 제공하는 자바스크립트 라이브러리.
기타 자세한 사항은 Underscore 홈페이지 <http://underscorejs.org/> 을 참고한다.

• Underscore 자주 쓰이는 functions

[each]

- each(argument1 , argument2);

```
_.each([1, 2, 3], alert);  
=> alerts each number in turn...  
_.each({one: 1, two: 2, three: 3}, alert);  
=> alerts each number value in turn...
```

argument1에 위치하는 배열이나 객체 숫자 만큼, argument2(일반적으로 함수가 들어감)를 실행시킨다.

[where]

- where(list , properties);

```
_.where(listOfPlays, {author: "Shakespeare", year: 1611});  
=> [{title: "Cymbeline", author: "Shakespeare", year: 1611},  
    {title: "The Tempest", author: "Shakespeare", year: 1611}]
```

list 안의 데이터를 모두 살펴, properties 와 일치하는 데이터들을 리턴해준다.

■ 코드 템플릿 적용

• Handlebars 란?

GlueJS는 Handlebars 템플릿 모듈을 지원한다. Handlebars의 기본적인 사용법을 알아본다.
기타 자세한 사항은 Handlebars 홈페이지 <http://handlebarsjs.com> 을 참고한다.

• Handlebars 표현

Handlebars는 `{{content}}`와 같은 형태로 표현한다.

```
01 <div class="user">
02   <h1>
03     {{username}}
04   </h1>
05   <div class="status">
06     {{status}}
07   </div>
08 </div>
```

• Template 정의

`<script>`태그를 사용하여 Template을 정의한다.

```
01 <script id="user-template" type="text/x-handlebars-template">
02   // template content
03 </script>
```

■ Handlebars 사용 예제

• Handlebars 사용

```
01 var context = {  
02     username: "Thomas",  
03     status: "GOOD"  
04 };  
05  
06 var source = $("user-template").html();  
07 var template = Handlebars.compile(source);  
08 var html = template(context);
```

- var context : Template을 채우는데 사용되는 JavaScript 객체를 정의
- \$("user-template").html() : 앞서 정의한 Template을 정의
- Handlebars.compile : Handlebars.compile을 사용하여 Template을 컴파일
- template(context) : context를 이용 template을 실행

• HTML 최종 결과

```
01 <div class="user">  
02     <h1>  
03         Thomas  
04     </h1>  
05     <div class="status">  
06         GOOD  
07     </div>  
08 </div>
```

■ 반복처리

- 반복 처리를 위해 each 를 사용하면 for 또는 while loop와 같은 작업을 처리 함.

▪ each 블록 사용 Template

```
01 <ul class="people_list">
02   {{#each people}}
03   <li>{{this}}</li>
04   {{/each}}
05 </ul>
```

▪ each 블록 context

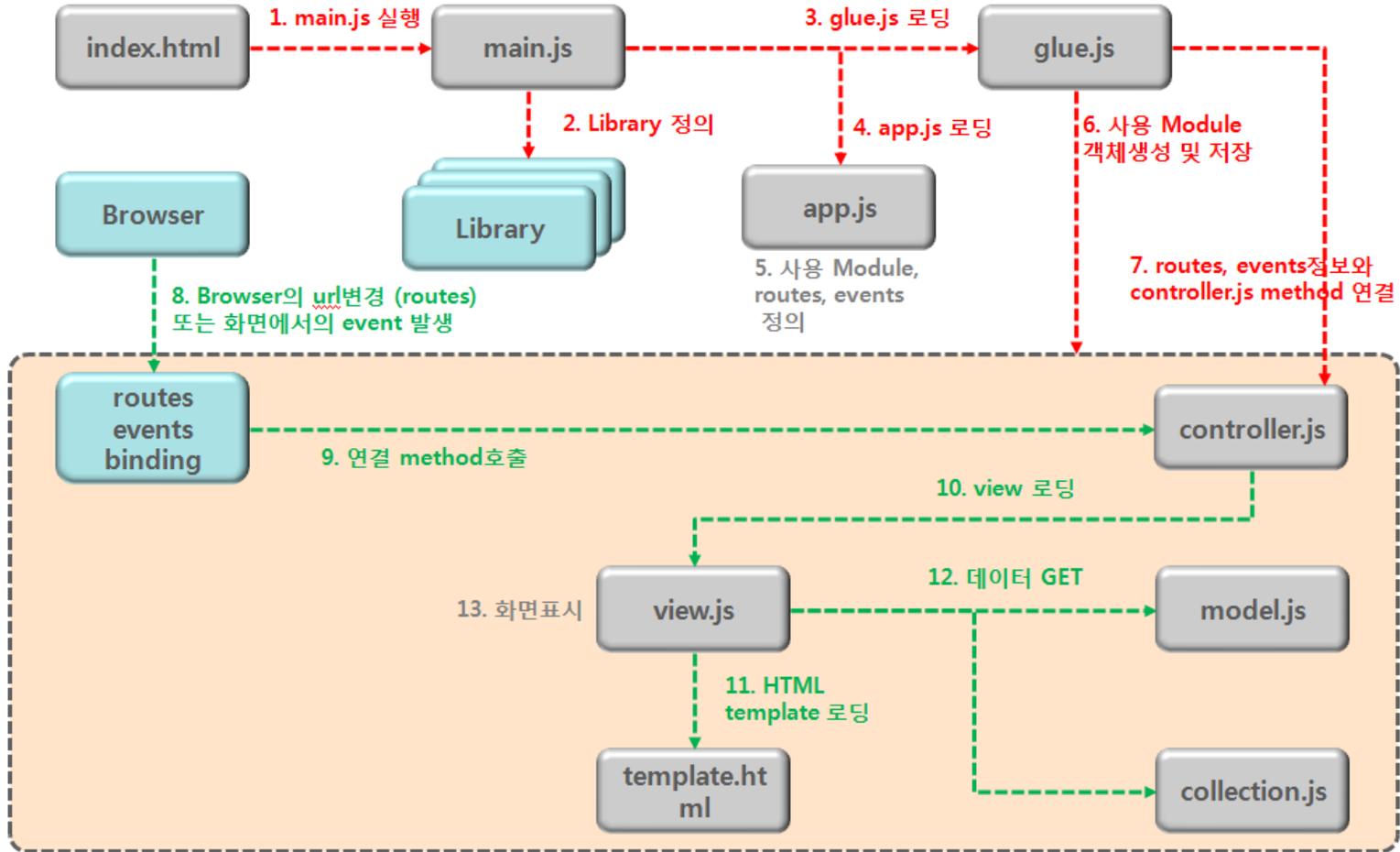
```
01 {
02   people: [
03     "Yehuda Katz",
04     "Alan Johnson",
05     "Charles Jolley"]
06 }
```

▪ each 블록 사용 결과 HTML

```
01 <ul class="people_list">
02   <li>Yehuda Katz</li>
03   <li>Alan Johnson</li>
04   <li>Charles Jolley</li>
05 </ul>
```

개요 - 처리흐름도

■ 처리 흐름도



개요 - 구성요소

■ 라이브러리 목록

라이브러리명	Version	설명	라이선스
jQuery	2.0.3	JavaScript 코드를 단순하고 간결한 상태로 개발 가능하게 도와준다.	MIT
jQuery Mobile	1.4.1	Mobile Web Application을 개발하기 위한 각종 UI 컨트롤과 Event 처리, 애니메이션 효과 및 자동 네비게이션, Ajax 통신 등 Mobile에 최적화된 기능을 제공한다.	MIT
RequireJS	2.1.9	JavaScript 파일과 모듈을 로딩하고 의존관계를 관리한다.	MIT or new BSD
text.js	2.0.10	text파일을 로딩하는 RequireJS의 플러그인이다.	MIT or new BSD
Backbone.js	1.1.0	Client Side Web Application 개발에 MVC패턴 적용을 가능하게 해주는 JavaScript Framework로 underscore.js Library에 의존성을 갖는다.	MIT
Underscore.js	1.5.2	Arrays, Functions, Objects에 대한 유틸리티 함수를 제공하고 함수형 프로그래밍을 지원한다.	MIT
Handlebars	1.3.0	Mustache를 기반으로 구현한 JavaScript 템플릿 엔진으로 두 개의 브레이스 {{ }}를 사용해 변환하며, 'Helper' 개념으로 자유롭게 값을 제어 할 수 있다.	MIT

개요 - 라이브러리

■ 라이브러리 목록

라이브러리명	Version	설명	라이선스
DataTables	1.10.0	Datatables는 동적으로 테이블을 만들어주는 jQuery JavaScript 플러그인이며 Paging, Filtering, Sorting, FixedColumn 등 많은 기능들을 제공한다.	MIT
DataTables. FixedColumns	3.0.1	테이블의 FixedColumn 기능을 제공하는 DataTables 플러그인이다.	MIT
DataTables.ColVis	1.1.0	테이블의 컬럼을 숨기거나 보이게 하는 기능을 제공하는 DataTables의 플러그인이다.	MIT
amCharts	3.10.0	JavaScript와 HTML5를 기반으로 하고 다양한 기능과 성능을 제공하는 차트 라이브러리이다.	Free (무료 버전의 경우 차트에 작은 amCharts 링크가 포함된다.)
Backbone. Validation	0.9.1	Backbone의 Model과 form input을 검증하는 플러그인이다.	MIT
Backbone.Stickit	0.6.3	Backbone View 요소에 Model 속성의 데이터를 바인딩하는 플러그인이다.	MIT

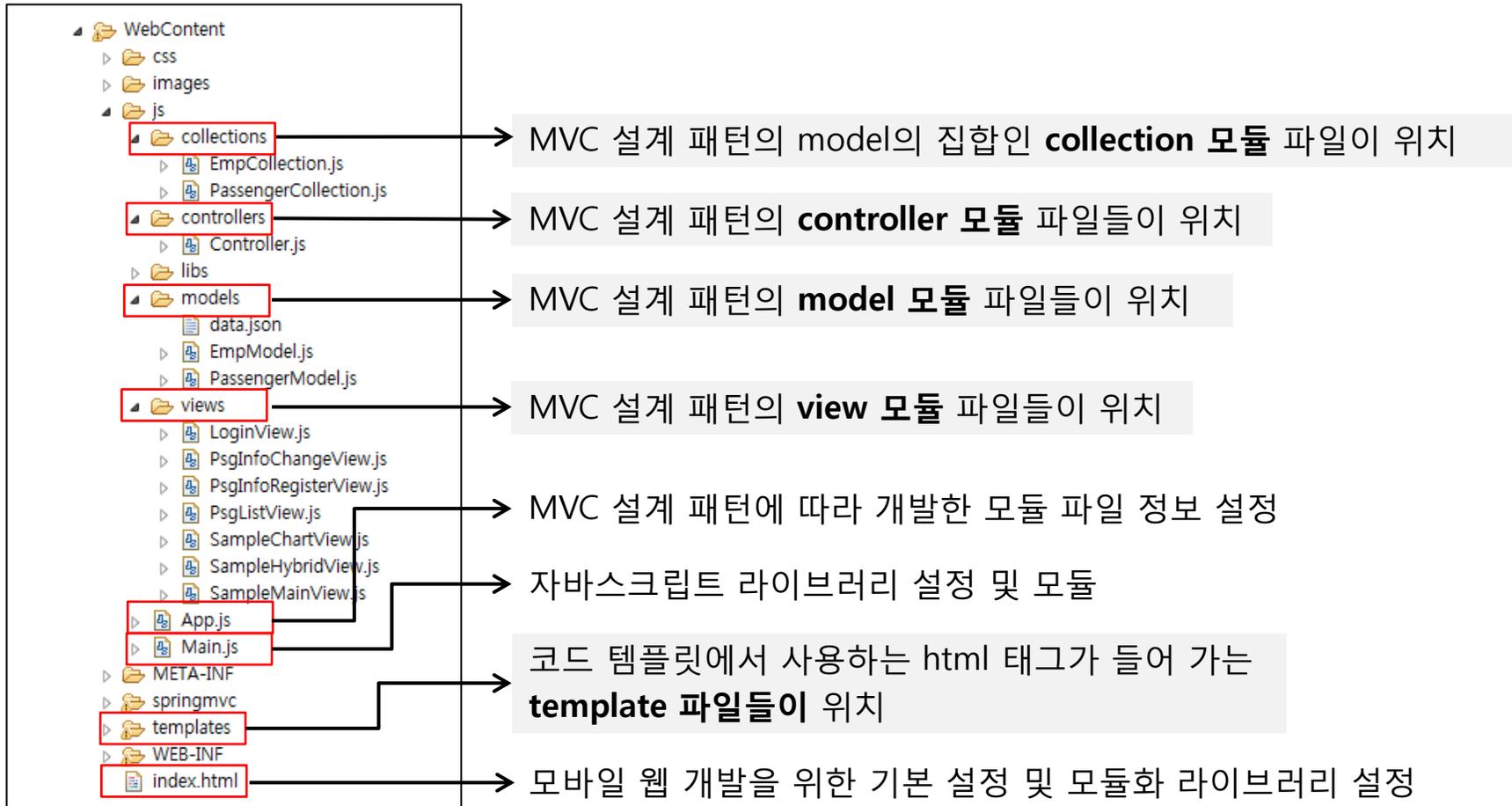
모바일UI 개발

1. 모바일UI 개요
2. 모바일UI 시작하기
3. 페이지 이동 하기
4. 서버 연동 하기
5. 차트 그리기

GlueJS시작하기-HelloWorld Application

■ 디렉토리 폴더구조

- GlueJS를 사용하여 Application을 개발 할 경우 아래 폴더구조를 권장한다.
- ※ 해당 용어들에 대한 설명은 앞에 개요 부분을 참고한다.



GlueJS시작하기-HelloWorld Application

■ index.html 설정하기

- 모바일 웹 시작에 필요한 설정을 하고 모듈 로드를 위한 require.js 설정을 한다.

```
01 <!DOCTYPE html>
02 <html>
03 <head>
04   <meta charset="UTF-8">
05   <meta name="viewport" content="user-scalable=no, initial-scale=1, maximum-scale=1, minimum-scale=1, width=device-width, height
06 =device-height, target-densitydpi=high-dpi" />
07   <title>Hello World</title>
08   <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.css" />
09 </head>
10 <body>
11   <div id="home" data-role="page" data-theme="a" data-title="Glue Mobile Sample">
12     </div>
13 </body>
14 <script data-main="js/Main" src="js/libs/require.js"></script>
```

require.js를 사용하기 위한 기본 설정으로
data-main에 처음 로드하는 자바스크립트 파일을 명시한다.

* *data-main* = "*js/Main*"

→ js 디렉토리 밑의 Main.js 파일을 로드 하도록 설정

GlueJS시작하기-HelloWorld Application

■ Main.js 설정하기

- 모바일 웹에서 사용하는 자바스크립트 모듈의 경로지정하고 각 모듈의 의존성을 정의하고 App.js를 로드 한다.

```
01 require.config({
02
03   baseUrl: 'js',
04   paths: {
05     'text': 'libs/text',
06     'jquery': 'libs/jquery',
07     'underscore': 'libs/underscore',
08     'backbone': 'libs/backbone',
09     'jquerymobile': 'libs/jquerymobile',
10     'handlebars': 'libs/handlebars-v1.3.0',
11     'glue': 'libs/glue',
12     'templates': '../templates' },
13   shim: {
14     'backbone': {
15       deps: ['jquery', 'underscore'],
16       exports: 'Backbone'},
17     'glue': {
18       deps: ['backbone', 'handlebars'],
19       exports: 'GlueJS'
20     }
21   }
22 });
23 require([
24   'glue'
25   'App'
26 ], function(GlueJS, App) {
27   App.start();
28 });
```

- **require.config** : RequireJS설정 선언, require.js가 로딩되면 자동으로 읽어 반영
- **baseUrl** : 모듈의 기본 위치 지정
- **paths** : 모듈의 단축 경로 지정 및 이름에 대한 별칭을 지정
- **shim** : AMD를 지원하지 않는 외부 라이브러리를 모듈로 사용하도록 지정
- **require** : require.js를 이용 App.js 로드하고 시작

App.js 를 로드하고 start()메소드를 실행

GlueJS시작하기-HelloWorld Application

■ app.js 설정하기

- Model, View, Controller 모듈 정보와 Route, Event를 처리하는 Controller 함수 설정

```
01 define(function() {
02     var start = function() {
03
04         var HelloWorldApp = new GlueJS.App({
05
06             controller: 'HelloWorldController',
07             models: [
08                 'HelloWorldModel' //HelloWorldModel.js 설정
09             ],
10             views: [
11                 'HelloWorldView' //HelloWorldView.js 설정
12             ],
13             templates: [
14                 'helloWorldTemplate' //helloWorldTemplate.html
15             ],
16             appRoutes: {
17
18                 'helloworld': 'helloworld',
19                 '*actions': 'defaultAction'
20             },
21             appEvents: {
22
23                 'click:#helloworld_btn': 'btnClick'
24             }
25         });
26         HelloWorldApp.start();
27     }
28     return {
29         start: start
30     };
31 });
```

- **define** : app에서 사용할 모듈을 정의
- **HelloWorldApp** : Application 변수 명
- **controller** : URL변경, 이벤트 발생 때 실행되는 method들을 정의한 Controller지정
- **models** : js/models 폴더에 로드할 Model 객체들을 정의
- **views** : js/views폴더의 로드할 View 객체들을 정의
- **templates** : templates 폴더의 로드할 템플릿 html파일들을 정의
- **appRoutes** : Application에서 hash URL이 변경될 때 실행 할 Controller의 method를 지정
- **appEvents** : Application에서 이벤트가 발생했을 때 실행 할 Controller의 method를 지정
- **HelloWorldApp.start()** : Application실행

페이지이름과 페이지전환 처리를 수행하는 Controller function 이름 설정

이벤트형태, HTML 태그 ID 와 이벤트 처리를 수행하는 Controller function 이름 설정

GlueJS시작하기-HelloWorld Application

■ HelloWorldController.js

- 사용자가 페이지 전환, 클릭 등의 이벤트를 발생 시키면 처리하는 로직을 구현 한다.

```
01 define(  
02   function() {  
03     var HelloWorldController = GlueJS.Controller.extend({  
04  
05       initialize: function(options) {  
06         this.init_navigate = '#helloworld';  
07       },  
08       helloworld: function(options) {  
09  
10         GlueJS.loadModule('HelloWorldView').show();  
11  
12       },  
13       defaultAction: function(options) {  
14  
15         if (options == null) {  
16           GlueJS.AppRouter.navigate(this.init_navigate, {  
17             trigger: true,  
18             replace: true  
19           });  
20         }  
21       }  
22       btnClick: function(event) {  
23  
24         alert("HelloWorld Button Click");  
25       }  
26     });  
27     return HelloWorldController;  
28   }  
);
```

GlueJS.Controller를 확장하여 Controller 정의

GlueJS.App에서 등록된 HelloWorldView를 로드하여 show method를 이용하여 화면을 표시
→ show() : view의 설정된 model을 fetch하고 화면 UI를 렌더링한다

- **initialize** : Controller가 생성 될 때 초기화 함수 등록
- **helloworld** : appRoutes에 정의한 helloworld Hash URL에 대한 처리 정의
- **defaultAction** : appRoutes에 정의한 *action Hash URL 요청에 대한 처리 정의 (app.js의 AppRoutes에서 정의하지 않은 Hash URL 이 요청되었을 때 실행)
- **btnClick** : appEvents에 정의한 HTML 요소의 helloworld_btn 아이디를 클릭할 시 동작하는 함수

GlueJS시작하기-HelloWorld Application

■ HelloWorldModel.js

- 데이터를 관리하고 서버와의 인터페이스를 담당한다.

```
01 define(  
02   function() {  
03  
04     var HelloWorldModel = GlueJS.Model.extend({  
05  
06       url: 'js/models/HelloWorld.json'  
07     });  
08     return HelloWorldModel;  
09   }  
10 );
```

GlueJS.Model을 확장하여 Model 정의

- url : 데이터가 정의된 JSON file url을 정의
(RESTful방식의 Server에 데이터를 요청할 수 있다.)

GlueJS시작하기-HelloWorld Application

■ HelloWorldView.js

- Model의 데이터와 HTML Template을 사용하여 화면에 표시하는 View를 생성 한다.

```
01 define(  
02   function() {  
03     var HelloWorldView = GlueJS.View.extend({  
04       el: '#home',  
05       template: 'helloWorldTemplate'  
06     });  
07     return HelloWorldView;  
08   }  
09 );  
10 }  
11 );
```

GlueJS.View를 확장하여 View 정의

- el : 데이터와 Template을 조합한 HTML이 표시되는 DOM 요소
- template: 'helloWorldTemplate' : 데이터와 조합 될 HTML 템플릿을 Loader를 이용하여 지정

GlueJS시작하기-HelloWorld Application

■ HelloWorld.json

- Model에서 사용하는 데이터를 JSON방식으로 정의하여 테스트 한다.
- 즉 실제 모바일앱 개발 시에는 필요 없는 파일이다

```
01 {  
02   "helloworld_msg": "Hello World Message"  
03 }
```

■ HelloWorldTemplate.html

- View에서 html을 생성하기위한 Template을 정의한다.
- Underscore.js Template을 사용하여 치환되어야 하는 Model 요소의 값을 나타낸다.

```
01 <div id="header" data-role="header">  
02   <h5>Hello World</h5>  
03 </div>  
04 <div id="content" data-role="content" style="text-align:center;">  
05  
06   <h3 style="height:150px;"><%= helloworld_msg%></h3>  
07  
08   <a id="helloworld_btn" class="ui-btn ui-corner-all ui-btn-inline">HelloWorld Button</a>  
09 </div>  
10 <div data-role="footer" data-position="fixed">  
11   <h5>GlueJS</h5>  
12 </div>
```

Model의 helloworld_msg속성 값을 표시

모바일UI 개발

1. 모바일UI 개요
2. 모바일UI 시작하기
3. 페이지 이동 하기
4. 서버 연동 하기
5. 차트 그리기

페이지 이동 하기

■ App.js의 appRoutes 설정 페이지 전환 작업

- appRoutes에 페이지이름과 페이지 전환 처리를 수행하는 Controller 함수 이름을 설정한다.

```
01 appRoutes: {  
02     // Routes  
03     'loginView':'loginView',  
04     'mainView':'gotoSampleMain',  
05     'visitor':'gotoVisitorInfo',  
06     'chart' : 'gotoSampleChart',  
07     'hybrid' : 'gotoSampleHybrid',  
08     '*actions': 'defaultAction'  
09 },  
10  
11
```

appRoutes에 설정된 페이지 요청 없이,
Context Root로 직접 접근 시 처리해야 하는 함수를 정의
→ 반드시 가장 마지막에 설정해야 한다.

default page를 "init_navigate"로 설정하는 코드
"init_navigate" 는 initialize 함수에서 "#loginView"로 설정
즉 default page는 #loginView로 loginView 함수가 수행

```
01 var Controller = GlueJS.Controller.extend({  
02  
03     //초기 설정 및 처리 로직을 구현한다.  
04     initialize: function(options) {  
05         this.init_navigate = '#loginView';  
06     },  
07     loginView: function(options) {  
08         loginView.show();  
09     },  
10     gotoSampleMain: function(options) {  
11         sampleMainView.show();  
12     },  
13     gotoVisitorInfo: function(options) {  
14         fetchCollection();  
15     },  
16     gotoSampleChart: function(options) {  
17         sampleChartView.show();  
18     },  
19     gotoSampleHybrid: function(options) {  
20         sampleHybridView.show();  
21     },  
22     defaultAction: function(options) {  
23         if (options == null) {  
24             GlueJS.AppRouter.navigate(this.init_navigate, {  
25                 trigger: true,  
26                 replace: true  
27             });  
28         }  
29     },  
30
```

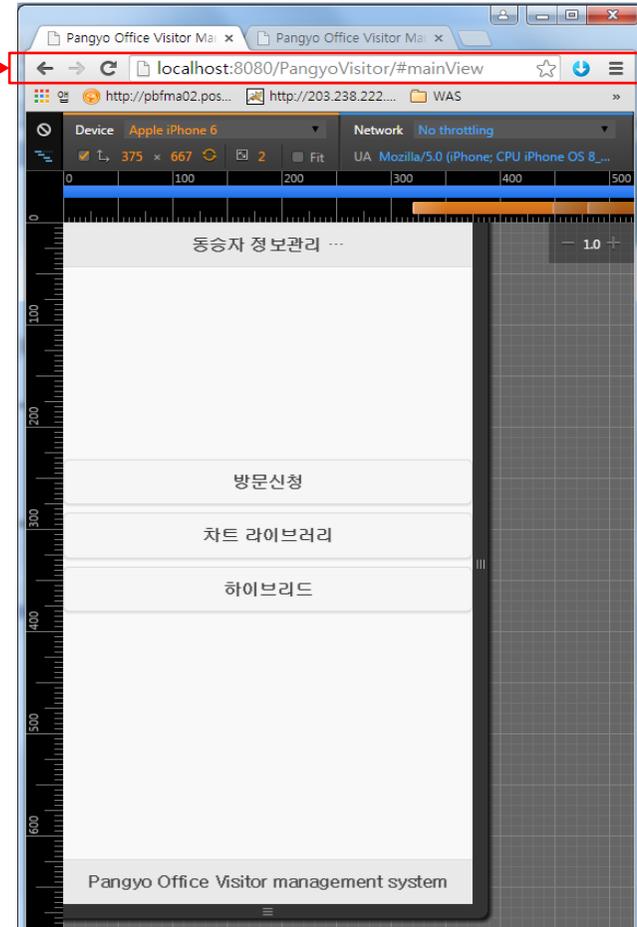
페이지 이동 하기

■ 페이지 전환 요청 URL

- appRoutes에 설정된 페이지이름(아이디)에 “#” 붙여 페이지 전환한다.
- context root url 에 “#” + 페이지이름으로 직접 접근도 가능하다.

```
01 <div data-role="content" style="text-align:center;">  
02 ....  
03 <a href="#mainView" data-role="button">로그인</a>  
04 ....  
05 ....  
06 </div>
```

<http://localhost:8080/PangyoVisitor/#mainView>



페이지 이동 하기

■ App.js의 appEvents 설정 페이지 전환 작업

- appEvents 에 이벤트를 정의하고 이벤트 발생 시 처리하는 Controller 함수 이름을 설정한다.
- 이벤트가 발생한 Div 태그 속성 등을 Controller 함수에 전달 할 수 있다.
- 페이지가 정의 되어 있지 않아 브라우저에서 url로 직접 접근이 불가능 하다.

```
01 appEvents: {  
02  
03     'click:#sampleback': 'gotoSampleMain',  
04     'click:#visitor_list': 'gotoPsgDetail',  
05     'click:#psg_info_delete': 'deletePsgInfo',  
06     'click:#psg_info_update': 'updatePsgInfo',  
07     'click:#move_psg_info': 'gotoPsgRegInfo',  
08     'click:#psg_info_reg': 'registerPsgInfo',  
09     'click:#psg_info_cancel': 'gotoVisitorInfo',  
10  
11 }  
12
```

event.target을 통해 이벤트가 발생한 태그에 정의되어 있는 속성 값을 가져 올 수 있다.
-id 속성 값을 가져온다.

```
01 var Controller = GlueJS.Controller.extend({  
02  
03     gotoPsgDetail: function(event){  
04  
05         var changeid = event.target.id;  
06         console.log("@ id : ", changeid);  
07  
08         model = passengerCollection.get(changeid);  
09         console.log("@ PassengerModel : ", model);  
10         psgInfoChangeView.model = model;  
11         model.fetch();  
12         psgInfoChangeView.render();  
13     },  
14     registerPsgInfo: function(event){  
15         console.log('register_psginfo');  
16         var data = $("#frm").serializeObject();  
17         passengerCollection.create(  
18             _extend(data, {emp_num: "297436"}),  
19             {  
20                 success: function (response) {  
21                     fetchCollection();  
22                 }  
23             }  
24         );  
25     },  
26
```

모바일UI 개발

1. 모바일UI 개요
2. 모바일UI 시작하기
3. 페이지 이동 하기
4. 서버 연동 하기
5. 차트 그리기

서버 연동 하기-RESTful 웹 서비스 API 설계

■ REST API 디자인 가이드

- URL에 동사보다는 명사를 사용한다.
- 단수(Singular) 보다는 복수(Plural)형 명사를 사용한다.

리소스	create	read	update	delete
HTTP	POST	GET	PUT	DELETE
<code>/employees</code>	새로운 employee 등록	employee 목록을 리턴	Bulk로 여러 employee 정보를 업데이트	모든 employee 정보를 삭제
<code>/employees/262094</code>	새로운 employee 등록	262094 이라는 이름의 employee 정보를 리턴	262094 이라는 이름의 employee 정보를 업데이트	262094 이라는 이름의 employee 정보를 삭제

- `GET /tickets` - Retrieves a list of tickets
- `GET /tickets/12` - Retrieves a specific ticket
- `POST /tickets` - Creates a new ticket
- `PUT /tickets/12` - Updates ticket #12
- `PATCH /tickets/12` - Partially updates ticket #12
- `DELETE /tickets/12` - Deletes ticket #12

<http://www.vinaysahni.com/best-practices-for-a-pragmatic-restful-api>

서버 연동 하기-gluejs 서버 연동 API

■ Model 서버 연동 주 API

- **fetch** : `model.fetch([options])` → *http get 호출*
서버로부터 특정 모델의 데이터를 가져 오고자 할 때 사용한다.
→ 기 설정된 url 정보를 기반으로 **http get** request를 호출한다.
- **save** : `model.save([attributes], [options])` → *create :http post , update : http put 호출*
모델의 데이터를 서버에 저장 할 때 사용한다.
신규 생성된 모델이 경우 create 되고, 서버에서 fetch 한 모델인 경우 update 된다.
→ 기 설정된 url 정보를 기반으로 **create** 시는 **http post** request를 호출한다.
update 시는 **http put** request를 호출한다.
- **destroy** : `model.destroy([options])` → *http delete request 호출*
모델의 데이터를 서버에서 삭제 할 때 사용한다.
→ 기 설정된 url 정보를 기반으로 **http delete** request를 호출한다.

서버 연동 하기-gluejs 서버 연동 API

■ Collection 서버 연동 주 API

- **fetch** : `collection.fetch([options])` → *http get 호출*
서버로부터 여러 개의 데이터를 가져 올 때 사용한다.
→ 기 설정된 url 정보를 기반으로 **http get** request를 호출한다.
- **create** : `collection.create(attributes, [options])` → *http post 호출*
신규 데이터를 모델 기반으로 생성할 때 사용하며 서버에 데이터를 추가 한다.
→ 기 설정된 url 정보를 기반으로 **create** 시는 **http post** request를 호출한다.

서버 연동 하기-서버에서 데이터 조회하기

■ Collection을 활용한 여러 개의 데이터 조회

- Collection 설정하기

```
01 define(  
02   function() {  
03     var PassengerCollection = GlueJS.Collection.extend({  
04  
05       url: "http://localhost:8080/PangyoVisitor/rest/access-control-list-service",  
06  
07       model: 'PassengerModel'  
08     });  
09     return PassengerCollection;  
10   }  
11 }  
12 );
```

RESTful 웹 서비스 URL을 url property 값에 설정한다.

collection 내에서 관리하는 데이터 model 값을 설정 한다.

서버 연동 하기-서버에서 데이터 조회하기

■ Collection을 활용한 여러 개의 데이터 조회

- Collection fetch 함수를 통한 서버 데이터 가져오기

```
01  
02 passengerCollection.fetch({  
03  
04   data : { emp_id : 297436 },  
05  
06   processData: true,  
07  
08   success : function(){  
09     psgListView.render();  
   }  
});
```

서버에서 필요로 하는 검색 조건을 파라미터로 전달한다.

위에서 설정한 파라미터를 serialize 하기 위해 반드시 필요한 설정 이다

서버에서 데이터 가져오기가 성공하면 view를 호출하여 html 화면을 렌더링 한다.

- Collection fetch 함수 수행 시의 http get request

http://localhost:8080/PangyoVisitor/rest/access-control-list-service/visitor?emp_id=297436

서버 연동 하기-서버에서 데이터 조회하기

■ Collection을 활용한 여러 개의 데이터 조회

- Glue F/W 서버 측 RestFul 웹서비스 구현 Controller.java의 GET 메소드

```
01  
02 @RequestMapping (value = "/", method = RequestMethod.GET, produces = "application/json")  
03  
04 public @ResponseBody Object getPassenger(@PathVariable("serviceName") String serviceName, @RequestParam("emp_id") String emp_id)  
05 {  
06     return doGlueService(serviceName, GlueWebConstants.GET, emp_id, null);  
07 }
```

Gluejs의 collection fetch 함수 호출 시에
검색조건으로 넘긴 파라미터를
@RequestParam을 이용하여 전달 받는다.

@RequestMapping 을 사용하여 웹서비스를 설정한다.
- value : 웹서비스 접근 url
- method : get, put, post, delete
- produces : json 등의 데이터 타입)

서버 연동 하기-신규 데이터 생성하기

■ Collection을 활용한 신규 데이터 생성하기

- URL에 동사보다는 명사를 사용한다.

```
01 registerPsgInfo: function(event){
02     console.log('register_psginfo');
03
04     var data = $("#frm").serializeObject();
05
06     passengerCollection.create(
07
08         _extend(data, {emp_num: "297436"}),
09
10         {
11             success: function(response) {
12                 psgListView.render();
13             }
14         }
15     );
16 }
```

Html 태그 id frm에서 신규로 생성해야 할 데이터 가져오기

Data array에 emp_num key/value 을 추가한다.

서버에 신규 데이터 생성이 성공하면 view를 호출하여 html 화면을 refresh 한다

서버 연동 하기-신규 데이터 생성하기

■ Collection을 활용한 신규 데이터 생성하기

- 신규 데이터 등록 처리를 위한 html form

```
01 <form id="frm">
02 <div class="ui-field-contain">
03 <label for="text-basic">이름 :</label>
04 <input type="text" name="passenger_name" id="passenger_name" placeholder="주민등록상 이름 기재" autocomplete="on">
05 </div>
06 <div class="ui-field-contain">
07 <label for="text-basic">나이 :</label>
08 <input type="text" name="passenger_age" id="passenger_age" placeholder="숫자만 입력 가능" autocomplete="on">
09 </div>
10 <div class="ui-field-contain">
11 <label for="text-basic">소속 :</label>
12 <input type="text" name="passenger_org" id="passenger_org" placeholder="소속되어 있는 회사 및 단체" autocomplete="on">
13 </div>
14 <div class="ui-field-contain">
15 <label for="text-basic">분류 :</label>
16 <input type="text" name="passenger_orgtype" id="passenger_orgtype" placeholder="회사 분류" autocomplete="on">
17 </div>
18 <div class="ui-field-contain">
19 <label for="text-basic">사번 :</label>
20 <input type="text" name="passenger_emp_num" id="passenger_emp_num" placeholder="사번을 입력하세요" autocomplete="on">
21 </div>
22 </form>
```

서버 연동 하기-신규 데이터 생성하기

■ Collection을 활용한 신규 데이터 생성하기

- Glue F/W 서버 측 RestFul 웹서비스 구현 Controller.java의 POST 메소드

```
01
02 @RequestMapping(method = RequestMethod.POST, produces = "application/json")
03
04 public @ResponseBody Object insertPassenger(@PathVariable("serviceName") String serviceName, @RequestBody Map<String, String>
05 map) {
06
07     Object glueResult = doGlueService(serviceName, GlueWebConstants.POST, null, map);
08
09     Map<String, String> returnMap = new HashMap<String, String>();
10
11     returnMap.put("result", glueResult.toString());
12
13     return returnMap;
14
15 }
16
```

@RequestMapping 을 사용하여 웹서비스를 설정한다.
- method : GET, PUT, **POST**, DELETE
- produces : json 등의 데이터 타입)

서버 연동 하기-데이터 수정하기

■ Model를 사용한 데이터 수정하기

```
01
02 updatePsgInfo: function(event){
03
04     var name = $("#passenger_name").val();
05     var age = $("#passenger_age").val();
06     var org = $("#passenger_org").val();
07
08     console.log(name, age, org);
09     var model = psgInfoChangeView.model ; //기 설정된 model
10     model.save({
11
12         passenger_name: name,
13         passenger_age: age,
14         passenger_org: org
15
16     }, {
17
18         success: function (model, response) {
19             console.log("success");
20         },
21         error: function (model, response) {
22             console.log("error");
23         }
24     });
25     console.log("업데이트가 끝났습니다.");
26 },
27
```

Model에 업데이트 정보를 key/value 형태로 설정한다.

서버 업데이트 성공과 실패 시의 수행 함수를 설정한다.
success : 성공 시 수행 함수
Error : 실패 시 수행 함수

서버 연동 하기-데이터 수정하기

■ REST API 디자인 가이드

- Glue F/W 서버 측 RestFul 웹서비스 구현 Controller.java 함수

```
01
02 @RequestMapping(value = "/{id}", method = RequestMethod.PUT, produces = "application/json")
03
04 public @ResponseBody Map<String, String> updatePassenger(@PathVariable("serviceName") String serviceName, @PathVariable("id")
05 String id, @RequestBody Map<String, String> map) {
06
07     Object glueResult = doGlueService(serviceName, GlueWebConstants.PUT, null, map);
08
09     Map<String, String> returnMap = new HashMap<String, String>();
10     returnMap.put("result", glueResult.toString());
11
12     return returnMap;
}
```

@RequestMapping 을 사용하여 웹서비스를 설정한다.

- value : 웹서비스 접근 url, {id} 경로의 마지막 값을 id로 받는다.
- method : GET, **PUT**, POST, DELETE
- produces : json 등의 데이터 타입)

서버 연동 하기-데이터 삭제하기

■ Model destroy 함수를 활용한 데이터 삭제하기

```
01 deletePsgInfo: function(event){
02
03     var delmodel = psgInfoChangeView.model;
04
05     console.log("delmodel", delmodel);
06
07     delmodel.destroy({
08
09         success: function(){
10             psgListView.show();
11         }
12     },
13     {wait : true}
14 );
15 }
```

서버에서 데이터 삭제 성공 시 수행 함수 설정
: 정상 삭제 후 view를 통해 화면 Refresh 한다.

서버 연동 하기-데이터 삭제하기

■ Model destroy 함수를 활용한 데이터 삭제하기

```
01  
02 @RequestMapping(value = "/{id}", method = RequestMethod.DELETE, produces = "application/json")  
03  
04 public @ResponseBody Object deletePassenger(@PathVariable("serviceName") String serviceName, @PathVariable("id") String id) {  
05  
06     Object glueResult = doGlueService(serviceName, GlueWebConstants.DELETE, id, null);  
07  
08     return glueResult.toString();  
09  
10 }  
11
```

@RequestMapping 을 사용하여 웹서비스를 설정한다.
- value : 웹서비스 접근 url, {id} 경로의 마지막 값을 id로 받는다.
- method : GET, PUT, POST, **DELETE**
- produces : json 등의 데이터 타입)

모바일UI 개발

1. 모바일UI 개요
2. 모바일UI 시작하기
3. 페이지 이동 하기
4. 서버 연동 하기
5. 차트 그리기

차트 그리기 - 파이 차트

■ amChart 란?

- amCharts는 JavaScript와 HTML5를 기반으로 다양한 기능을 제공하는 차트 라이브러리이다
GlueJS와 연동하여 사용하는 방법에 대해 알아본다.
- 기타 자세한 사항은 amCharts 홈페이지 <http://www.amcharts.com>을 참고한다.

■ index.html

- 기존 index.html 파일과 동일하게 main.js 파일을 위치를 설정한다.

```
01 <!DOCTYPE html>
02 <html>
03 <head>
04   <meta charset="UTF-8">
05   <meta name="viewport" content="user-scalable=no, initial-scale=1, maximum-scale=1, minimum-scale=1, width=device-width, height=device-height, target-densitydpi=middle-dpi" />
06   <title>Chart GlueJS</title>
07   <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.css" />
08 </head>
09 <body>
10   <div id="home" data-role="page" data-theme="a" data-title="Chart GlueJS">
11     </div>
12 </body>
13 <script data-main="js/main" src="js/libs/require.js"></script>
```

■ RequireJS 설정 (Main.js)

- 'amcharts': 'libs/amcharts/amcharts' : path에 amCharts 모듈명 및 위치를 정의
- 'amcharts.funnel' : 사용 할 차트의 의존관계와 export설정

```
01 require.config({
02   baseUrl: 'js',
03   paths: {
04     'text': 'libs/text',
05     'jquery': 'libs/jquery',
06     'underscore': 'libs/underscore',
07     'backbone': 'libs/backbone',
08     'routefilter': 'libs/backbone.routefilter',
09     'jquerymobile': 'libs/jquerymobile',
10     'handlebars': 'libs/handlebars-v1.3.0',
11     'datatables': 'libs/jquery.dataTables',
12     'datatables-fixedcolumns': 'libs/dataTables.fixedColumns
13     'amcharts': 'libs/amcharts/amcharts',
14     'amcharts.funnel': 'libs/amcharts/funnel',
15     'amcharts.gauge': 'libs/amcharts/gauge',
16     'amcharts.pie': 'libs/amcharts/pie',
17     'amcharts.radar': 'libs/amcharts/radar',
18     'amcharts.serial': 'libs/amcharts/serial',
19     'amcharts.xy': 'libs/amcharts/xy',
20     'amcharts.amexport': 'libs/amcharts/exporting/amexport',
21     'glue': 'libs/glue',
22     'templates': '../templates'
23   },
```

Amcharts 사용을 하기 위한 라이브러리 설정

차트 그리기 - 파이 차트

■ RequireJS 설정 (Main.js)

- amcharts가 지원하는 funnel, gauge, pie, radar, serial 차트 라이브러리를 사용하도록 설정

```
01 shim: {
02   'backbone': {
03     deps: ['jquery', 'underscore'],
04     exports: 'Backbone'
05   },
06   'glue': {
07     deps: ['backbone', 'handlebars'],
08     exports: 'GlueJS'
09   },
10   'amcharts.funnel': {
11     deps: ['amcharts'],
12     exports: 'AmCharts',
13     init: function() {
14       AmCharts.isReady = true;
15     }
16   },
17   'amcharts.gauge': {
18     deps: ['amcharts'],
19     exports: 'AmCharts',
20     init: function() {
21       AmCharts.isReady = true;
22     }
23   },
24   'amcharts.pie': {
25     deps: ['amcharts'],
26     exports: 'AmCharts',
27     init: function() {
28       AmCharts.isReady = true;
29     }
30   },
```

```
01   'amcharts.radar': {
02     deps: ['amcharts'],
03     exports: 'AmCharts',
04     init: function() {
05       AmCharts.isReady = true;
06     }
07   },
08   'amcharts.serial': {
09     deps: ['amcharts'],
10     exports: 'AmCharts',
11     init: function() {
12       AmCharts.isReady = true;
13     }
14   },
15   'amcharts.xy': {
16     deps: ['amcharts'],
17     exports: 'AmCharts',
18     init: function() {
19       AmCharts.isReady = true;
20     }
21   },
22   'amcharts.amexport': {
23     deps: ['amcharts'],
24     exports: 'AmCharts'
25   }
26 }
27 }
28 });
29
30
```

■ app.js 설정하기

```
01 define(  
02   function() {  
03     var start = function() {  
04       GlueJS.isDevelopment = true;  
05       GlueJS.debugpoint.timeunit = 'ms';  
06       var App = new GlueJS.App({  
07  
08         controller: 'ChartController',  
09         models: [  
10           'PieChartModel'  
11         ],  
12         views: [  
13           'SimplePieChartView'  
14         ],  
15         templates: [  
16           'chartTemplate'  
17         ],  
18         viewSetModel: {  
19           'SimplePieChartView': 'PieChartModel'  
20         },  
21         appRoutes: {  
22           'pie_chart': 'pieChart',  
23           '*actions': 'defaultAction'  
24         }  
25       });  
26       App.start();  
27     }  
28     return {  
29       start: start  
30     };  
31   }  
21 );
```

- controller : 차트 처리를 담당할 Controller 지정
- models : 데이터를 담당할 Model 지정
- views : 차트를 화면에 그려줄 View 지정
- template : 차트가 나타나는 화면의 HTML을 지정
- viewSetModel : View에 Model속성을 지정
- appRouter : Hash URL과 처리할 controller의 method지정

■ ChartController.js

- init_navigate : 지정한 Hash URL이 아닐 때 이동할 URL을 지정
- pieChart : SimplePieChartView 모듈을 로드하고 show method를 호출

```
01 define(  
02     function() {  
03         var ChartController = GlueJS.Controller.extend({  
04  
05             initialize: function(options) {  
06                 this.init_navigate = '#pie_chart';  
07             },  
08  
09             defaultAction: function(param) {  
10                 GlueJS.AppRouter.navigate(this.init_navigate, {  
11                     trigger: true,  
12                     replace: true  
13                 });  
14             },  
15             pieChart: function() {  
16                 GlueJS.loadModule('SimplePieChartView').show();  
17             }  
18         });  
19  
20         return ChartController;  
21     }  
22 );
```

차트 그리기 - 파이 차트

■ PieChartModel.js

- Model에 데이터 파일의 위치를 지정한다.

```
01 define(  
02   function() {  
03     var PieChartModel = GlueJS.Model.extend({  
04       url: 'js/models/PieChartData.json'  
05     });  
06     return PieChartModel;  
07   }  
08 );
```

기본적으로 서버 URL 정보를 설정한다.

PieChartData.json : 테스트용 파일

→서버 앱이 구현되어 있지 않은 경우, 상기와 같이 json 파일을 서버에 업로드하여 테스트를 할 수 있다. 실제 서버 앱이 구현되면 URL을 설정하고 해당 json 파일은 삭제한다.

- 차트에서 사용할 JSON데이터를 정의한다.

```
01 [{  
02   "country": "Ireland",  
03   "litres": 201.1  
04 }, {  
05   "country": "Germany",  
06   "litres": 165.8  
07 }, {  
08   "country": "Australia",  
09   "litres": 139.9  
10 }, {  
11   "country": "Austria",  
12   "litres": 128.3  
13 }, {  
14   "country": "UK",  
15   "litres": 99  
16 }, {  
17   "country": "Belgium",  
18   "litres": 60  
19 }]
```

■ ChartTemplate.html

- 차트가 그려지는 DIV(chartdiv)를 포함한 HTML 템플릿을 정의한다.

```
01 <div id="header" data-role="header">
02   <a href="#amchart_menu" data-icon="back">메뉴 </a>
03   <h5><%=type%> Chart</h5>
04 </div>
05 <div id="content" data-role="content" style="text-align:center;">
06
07   <div id="chartdiv" style="width: 100%; height: 400px;"></div>
08
09 </div>
10 <div data-role="footer" data-position="fixed">
11   <h5>GlueJS</h5>
12 </div>
```

차트가 그려지는 div 태그이다 view에서
"chartdiv" 값으로 렌더링 한다.

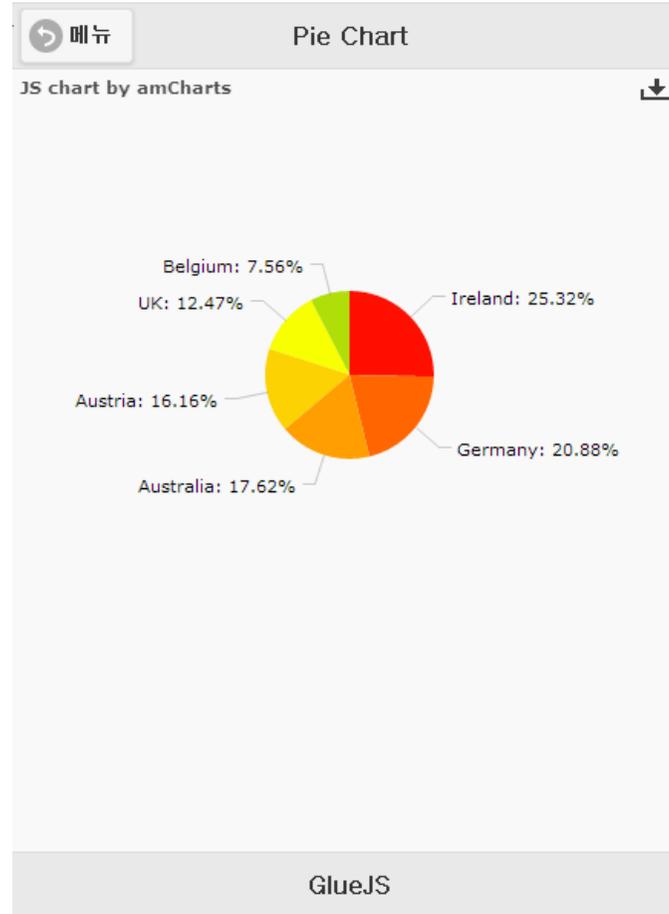
차트 그리기 - 파이 차트

SimplePieChartView.js

```
01 define(['amcharts.pie'],
02     function(AmCharts) {
03         var SimplePieChartView = GlueJS.View.extend({
04             el: '#home',
05             template: 'chartTemplate',
06             onBeforeRender: function() {
07                 this.data = {
08                     type: 'Pie'
09                 };
10             },
11             onAfterRender: function() {
12                 this.chartRender();
13             },
14             chartRender: function() {
15                 var chartData = _.toArray(this.model.attributes);
16                 this.chart = AmCharts.makeChart('chartdiv', {
17                     'type': 'pie',
18                     'dataProvider': chartData,
19                     'titleField': 'country',
20                     'valueField': 'litres',
21                     'exportConfig': {
22                         'menuItems': [{
23                             'icon': 'js/libs/amcharts/images/export.png',
24                             'format': 'png'
25                         }]
26                     }
27                 });
28             }
29         });
30         return SimplePieChartView;
31     }
32 );
```

- define(['amchart.pie'] : 사용할 차트유형의 라이브러리를 지정
- function(AmCharts) : 로딩을 완료한 라이브러리를 사용할 매개변수 지정
- template : 차트가 그려질 HTML 템플릿을 로드
- this.data : 템플릿에 사용할 값을 정의
- onAfterRender : GlueJS내부적으로 render를 실행 후에 자동으로 호출
- chartRender : 차트를 그리는 함수 정의
- var chartData : Model의 데이터를 배열로 변환하여 정의
- AmCharts.makeChart : 지정한 DIV에 차트를 생성하는 함수
- type : 차트의 유형을 정의
- dataProvider : 차트에 사용할 데이터를 지정
- valueField : dataProvider에서 값으로 사용할 필드명
- titleField : dataProvider에서 제목으로 사용할 필드명
- exportConfig : 차트를 이미지로 Export하는 기능구성을 정의
- menuItems : 이미지를 Export하는 메뉴설정
- icon : 메뉴 아이콘 지정
- format : Export하는 포맷 지정

■ Pie Chart 예제 실행화면



■ Column Chart 예제 (SimpleColumnChartView.js)

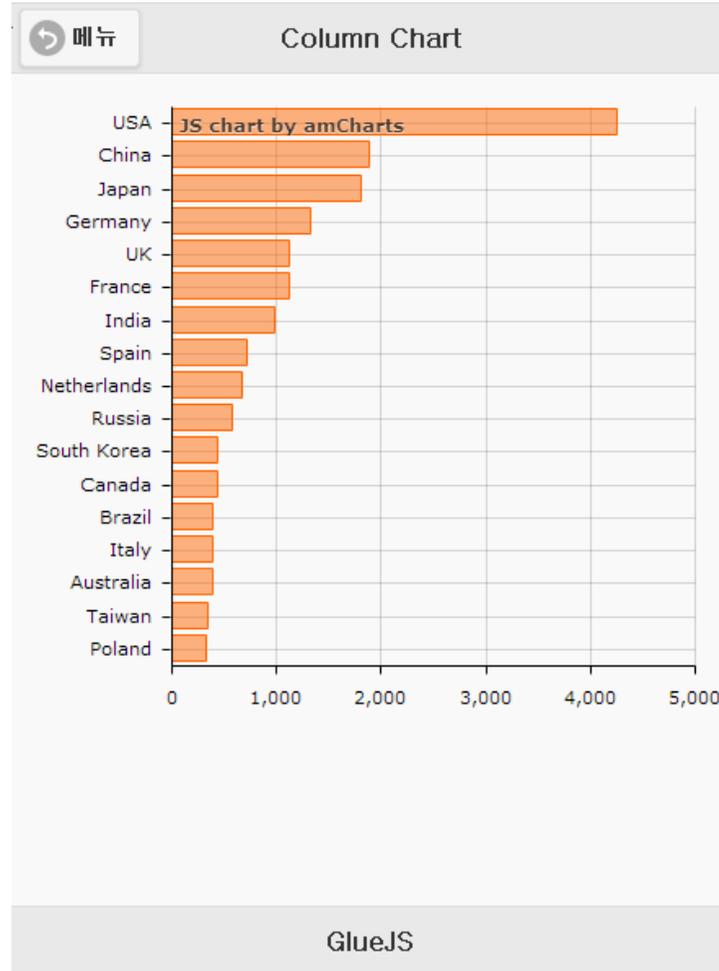
- Column Chart는 Pie Chart의 기본적인 구조와 동일하다.
- 이 단락에서는 amCharts의 Column Chart를 구현하기 위한 옵션을 위주로 설명한다.

```
01 define(['amcharts.serial'],
02 function(AmCharts) {
03     var SimpleColumnChartView = GlueJS.View.extend({
04         el: '#home',
05         template: GlueJS.loadModule('chartTemplate'),
06         onBeforeRender: function() {
07             this.data = {
08                 type: 'Column'
09             };
10         },
11         onAfterRender: function() {
12             this.chartRender();
13         },
14         chartRender: function() {
15             var chartData = _.toArray(this.model.attributes);
16             this.chart = AmCharts.makeChart("chartdiv", {
17                 "type": "serial",
18                 "dataProvider": chartData,
19                 "categoryField": "country",
20                 "graphs": [{
21                     "type": "column",
22                     "balloonText": "[[category]]: <b>[[value]]</b>",
23                     "fillAlphas": 0.5,
24                     "valueField": "visits"
25                 }],
26                 "rotate": true
27             });
28         }
29     });
30     return SimpleColumnChartView;
31 }
32
```

- define(['amchart.serial'] : 사용할 차트라이브러리를 지정 (serial 은 line, area, column, bar chart 지원.)
- type : 차트의 유형을 정의
- dataProvider : 차트에 사용할 데이터를 지정
- categoryField : dataProvider에서 축값으로 사용할 필드명
- graphs : 차트에 속하는 그래프를 정의
- type : 그래프 유형을 정의 (line, column, step, smoothedLine, candlestick, ohlc)
- balloonText : 텍스트 풍선의 형식을 지정 (value, description, percents, open, category)
- fillAlphas : 그래프를 채우는 불투명도를 지정
- valueField : dataProvider에서 값으로 사용할 필드명
- rotate : 차트를 90도로 회전

차트 그리기 - 컬럼 차트

■ Column Chart 예제 실행화면



■ Line Chart 예제 (SimpleLineChartView.js)

Line Chart는 amCharts연동 (Pie Chart)의 기본적인 구조를 참고하여 구현할 수 있으며 이 단락에서는 amCharts의 Line Chart를 구현하기 위한 옵션을 위주로 설명한다.

```
01 define(['amcharts.serial'],
02   function(AmCharts) {
03     var SimpleLineChartView = GlueJS.View.extend({
04       el: '#home',
05       template: GlueJS.loadModule('chartTemplate'),
06       onBeforeRender: function() {
07         this.data = {
08           type: 'Line'
09         };
10       },
11       onAfterRender: function() {
12         this.chartRender();
13       },
14       chartRender: function() {
15         var chartData = _.toArray(this.model.attributes);
16         this.chart = AmCharts.makeChart("chartdiv", {
17           "type": "serial",
18           "autoMargins": true,
19           "dataProvider": chartData,
20           "categoryField": "date",
21           "valueAxes": [{
22             "axisAlpha": 0,
23             "inside": true
24           }],
```

- define(['amchart.serial']): 사용할 차트라이브러리를 지정
- type: 차트의 유형을 정의
- autoMargins: 차트의 여백을 자동으로 지정
- dataProvider: 차트에 사용할 데이터를 지정
- categoryField: dataProvider에서 축값으로 사용할 필드명
- valueAxes: 값축을 표시하는 배열을 지정
- axisAlpha: 축의 불투명도를 지정
- inside: 값축의 기준값이 안쪽에 위치 할지를 지정

■ Line Chart 예제 (SimpleLineChartView.js)

```
01         "graphs": [{
02             "balloonText": "[[category]]<br>
03                 <b>value: [[value]]</b>",
04             "bullet": "round",
05             "bulletBorderAlpha": 1,
06             "bulletBorderColor": "#FFFFFF",
07             "hideBulletsCount": 50,
08             "lineColor": "#67b7dc",
09             "valueField": "visits"
10         }],
11         "categoryAxis": {
12             "parseDates": true,
13             "axisAlpha": 0,
14             "minHorizontalGap": 60
15         }
16     });
17 }
18 });
19 return SimpleLineChartView;
20 }
```

graphs : 차트에 속하는 그래프를 지정

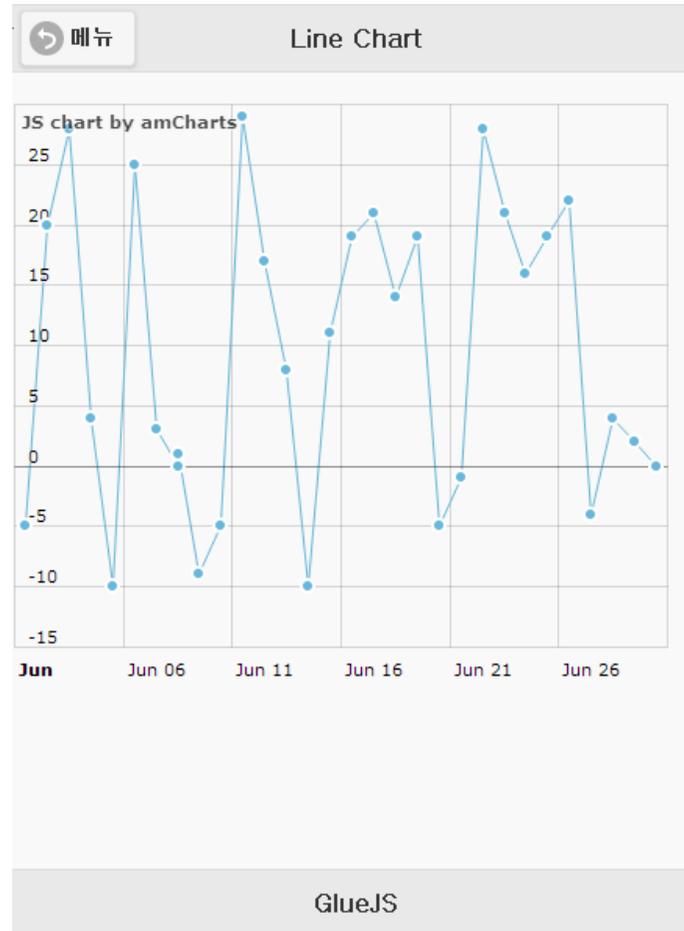
- balloonText : 텍스트 풍선의 형식을 지정 (value, description, percents, open, category)
- bullet : 점의 유형을 지정 (none, round, square, triangleUp, triangleDown, triangleLeft, triangleRight, bubble, diamond)
- bulletBorderAlpha : 점 border의 불투명도를 지정
- bulletBorderColor : 점 border의 색을 지정
- hideBulletsCount : 점이 나타나는 최소값을 지정
- lineColor : 선의 색을 지정
- valueField : dataProvider에서 선의 값으로 사용할 필드명

categoryAxis : 카테고리 축 설정

- parseDates : 카테고리 축의 값이 날짜일 경우 구문 분석 여부를 지정
- axisAlpha : 카테고리 축의 불투명도를 지정
- minHorizontalGap : 카테고리 축의 최소 넓이를 지정

차트 그리기 - 컬럼 차트

■ Line Chart 예제 실행화면



모바일장치 API

1. [개요](#)
2. [GlueCordova 프로젝트 생성 및 설정](#)
3. [GlueCordova Plugin API](#)

1. 개요

- ① 하이브리드 앱
- ② Apache Cordova
- ③ Glue Cordova

■ 하이브리드 앱

기존 웹 어플리케이션 개발자가 Android나 IOS 같은 네이티브 언어를 배우지 않고도, 모바일 장치 API를 이용하여 개발할 수 있는 모바일 어플리케이션

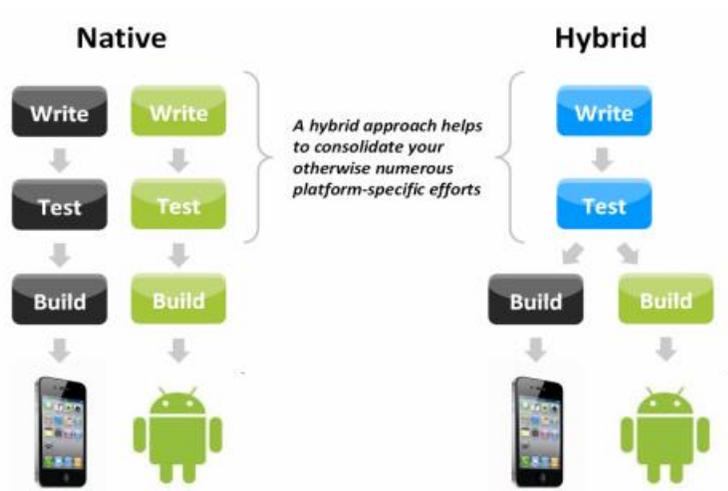


그림 1. 하이브리드 앱 개발방식

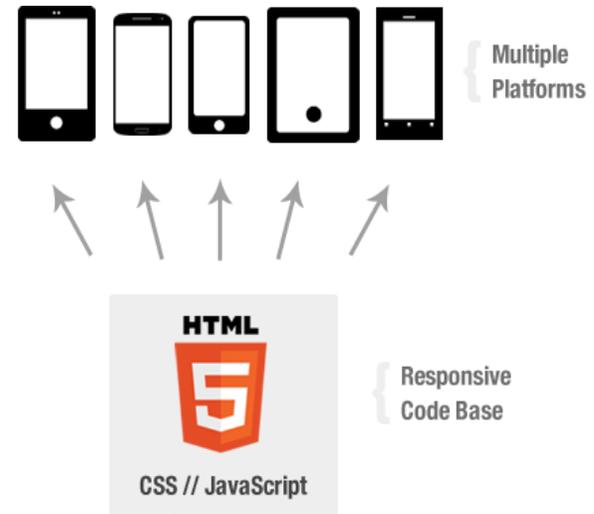


그림 2. 하이브리드 앱 개발 장점

■ Apache Cordova

Javascript & HTML5 & CSS3 를 이용하여 모바일 어플리케이션을 만들 수 있는 모바일 개발 프레임워크. 모바일 장치 API를 사용하여 개발할 수 있다.

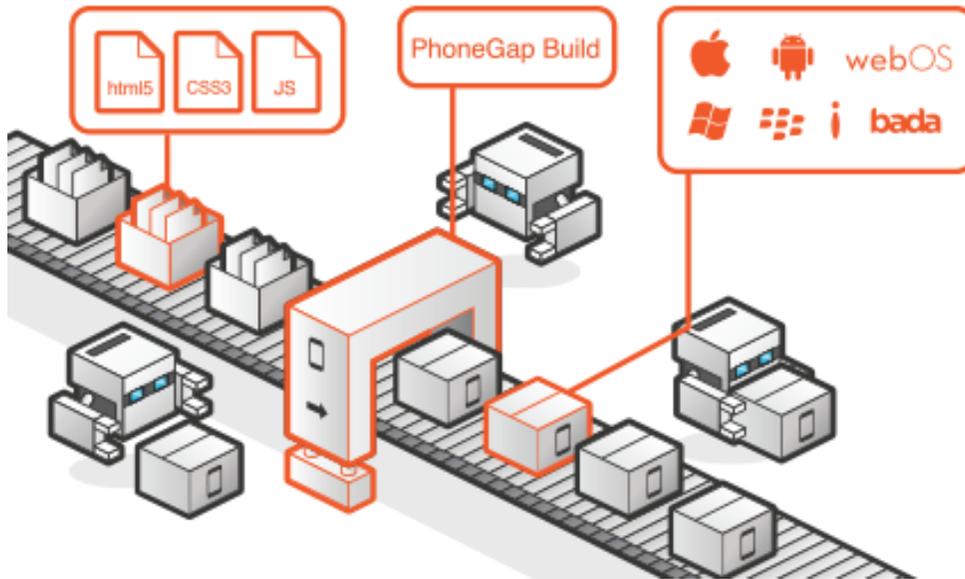


그림 4. Apache Cordova 이해



그림 5. Apache Cordova 이해

■ Glue Cordova

Cordova 플랫폼을 기반으로 제작된 하이브리드 앱으로써 Cordova에서 공식적으로 제공하는 기능 뿐만 아니라 추가적으로 Mobile Push Service, Barcode Scanner, 음성인식, NFC 기능을 제공

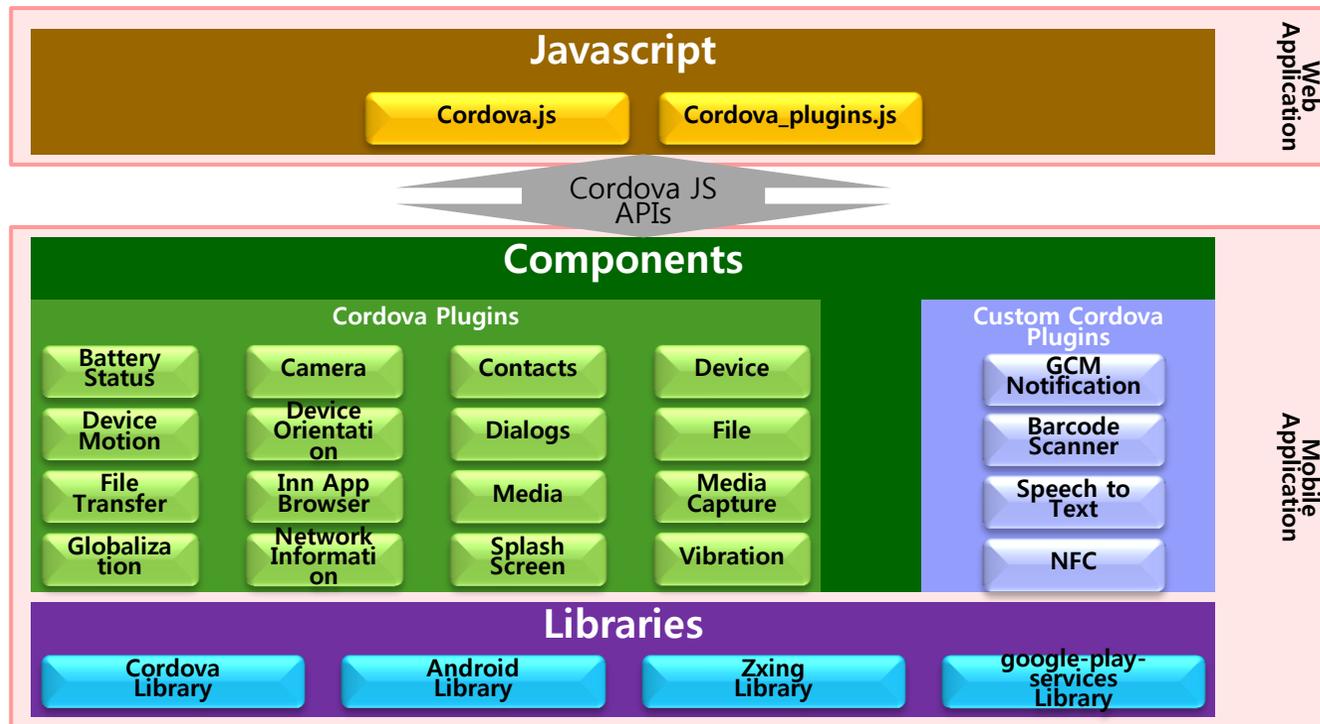


그림 3. GlueCordova 구조

2. GlueCordova 프로젝트 생성 및 설정

- ① GlueCordovaActivity 템플릿 폴더
- ② GlueCordova Code Template 설정
- ③ GlueCordova 프로젝트 생성

■ GlueCordova Activity 템플릿 폴더 설정

GlueCordovaActivity 템플릿 폴더

이클립스에서 GlueCordova 프로젝트를 위자드로 생성하기 위해서는 GlueCordovaActivity 폴더를 Android SDK templates 폴더 안에 위치시킨다.

GlueCordova CodeTemplate 설정

- Android SDK 폴더의 위치로 이동한다.

(ex) C:\weclipse\sdk

- SDK 폴더 안의 tools/templates/activities 에 GlueCordovaActivity 폴더를 위치시킨다.

(ex) C:\weclipse\sdk\tools\templates\activities

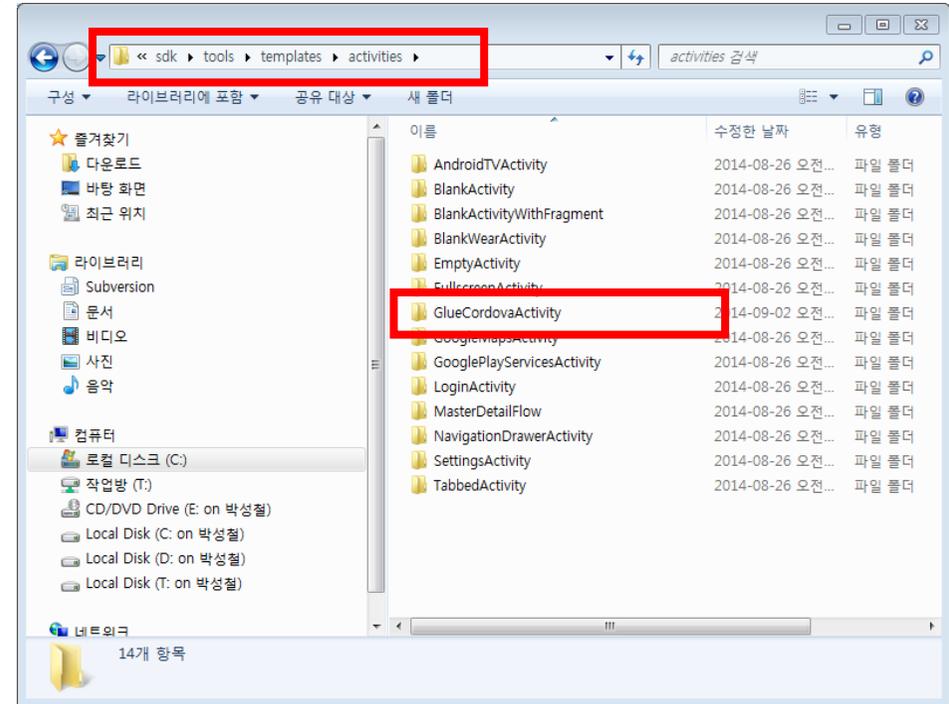
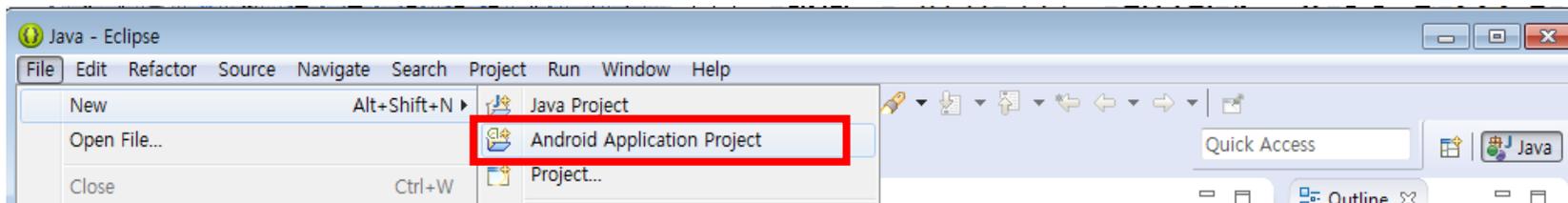


그림6. GlueCordova Code Template 설정 후 폴더화면

■ GlueCordova 안드로이드 프로젝트 생성

- Eclipse 의 File -> New -> Android Application Project 클릭



- 마법사에서 세부항목을 입력하고 Next 버튼을 클릭

그림7. Android Application 선택 화면

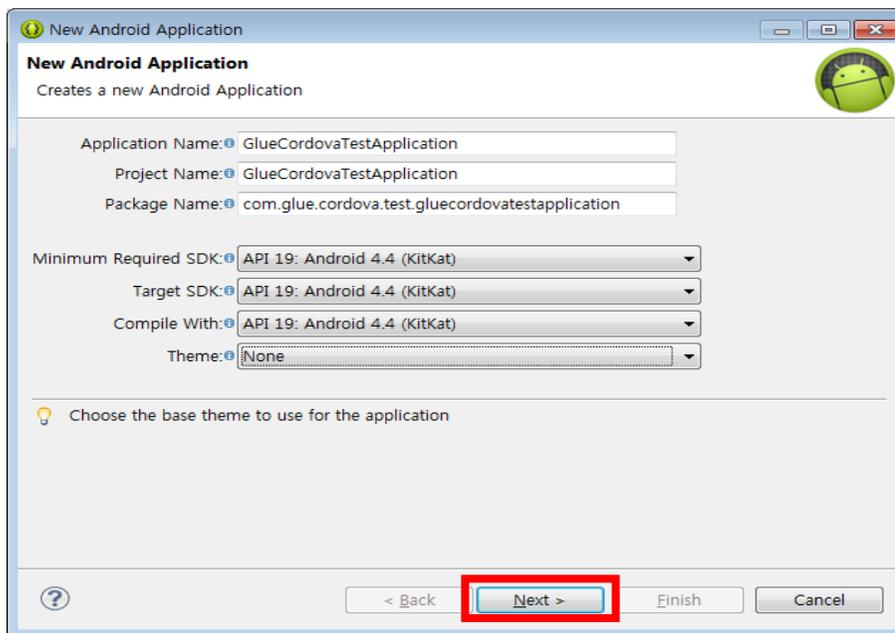


그림8. Android Application 생성 화면

■ GlueCordova 안드로이드 프로젝트 생성

- Create Activity 화면에서 Glue Cordova Activity 선택하고 Next 버튼을 클릭

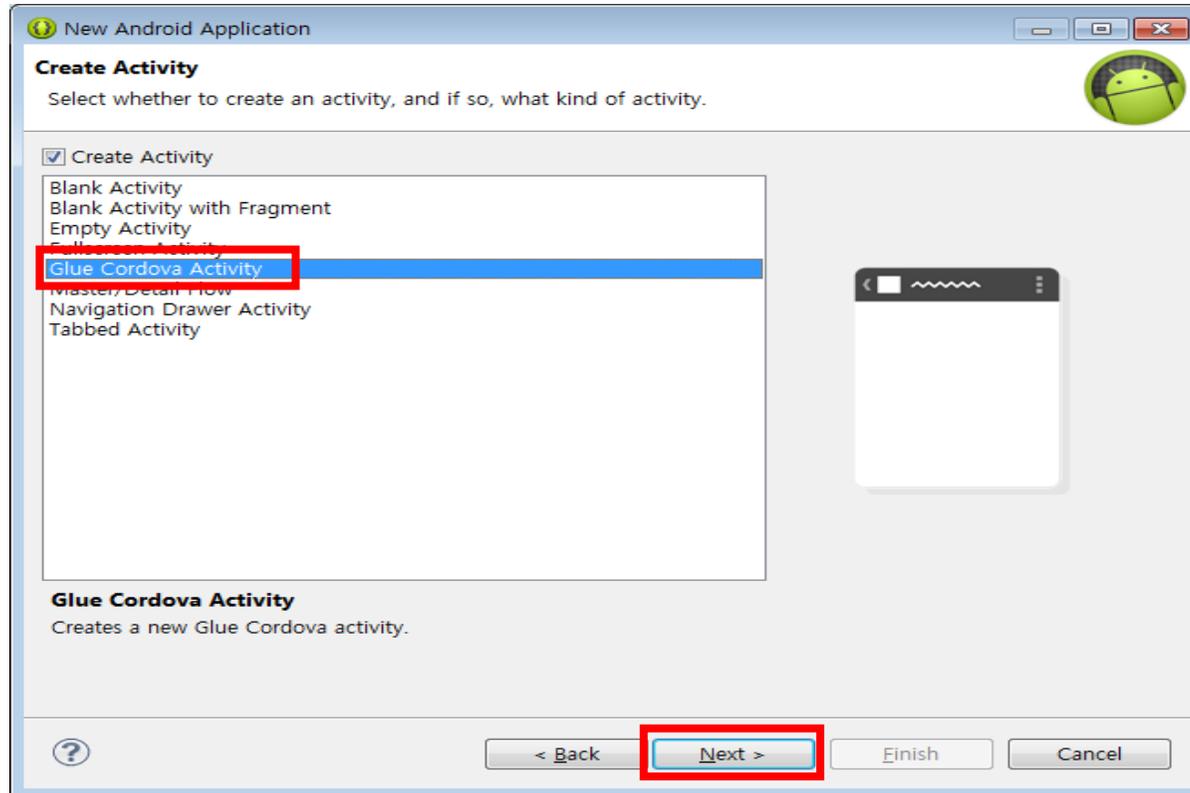


그림9. Glue Cordova Activity 선택 화면

■ GlueCordova 안드로이드 프로젝트 생성

- Main Activity 명을 입력하고 Finish 버튼 클릭

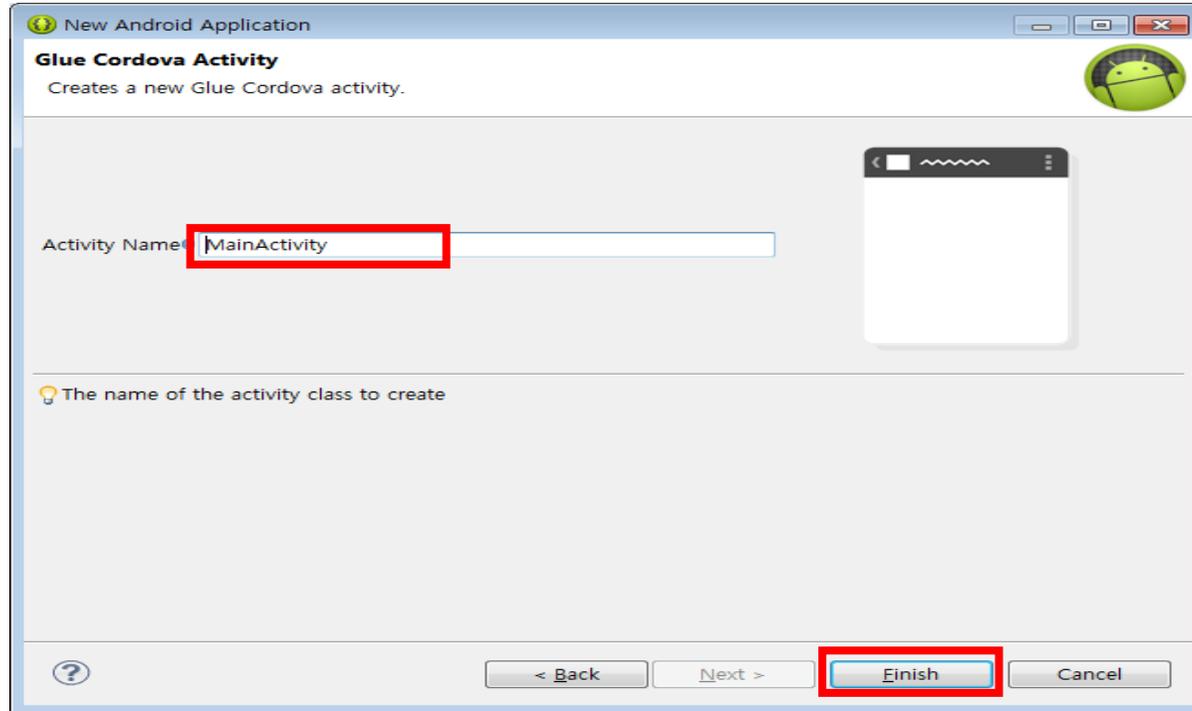


그림10. Glue Cordova Activity 이름 설정 화면

■ GlueCordova 안드로이드 프로젝트 생성

- Main Activity 명을 입력하고 Finish 버튼 클릭

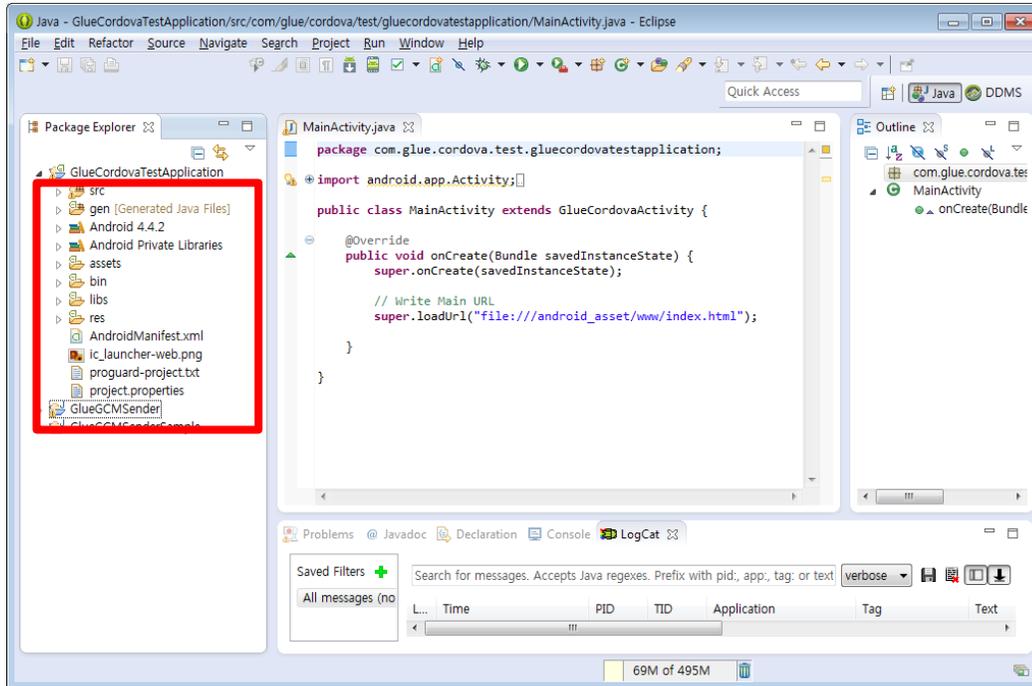


그림11. Glue Cordova Android 프로젝트 구조

- res 디렉토리

GlueCordova 어플리케이션 동작에 필요한 데이터 파일 (config.xml 위치)

- AndroidManifest.xml

GlueCordova 어플리케이션 동작에 필요한 설정 파일

- Activity 클래스

GlueCordova 기본 단위인

GlueCordovaActivity를 상속받는다

load.Url() 메서드를 이용하여 웹 페이지에 접근한다

- assets 디렉토리

GlueCordova 어플리케이션이 동작하기 위한 Javascript 파일 및 테스트를 위한 샘플 페이지

- libs 디렉토리

GlueCordova 어플리케이션 동작에 필요한 라이브러리

-cordova-3.6.4.jar : Cordova 플랫폼 코어

-gluecordova.jar : GlueCordova 코어

-google-play-services.jar : GCM 서비스

-core-3.0.1.jar : QR코드 (Zxing)

-gluecaptureactivity.jar : QR코드 (Zxing)

- config.xml 파일

Cordova 어플리케이션 설정 파일

플러그인을 정의하는 곳

3. GlueCordova Plugin API

- ① GlueCordova Plugin API 구성
- ② GlueCordova Plugin API – 기본 API 실습
- ③ GlueCordova Plugin API – Customized API 실습
 - a. QRCode
 - b. STT
 - c. NFC

GlueCordova Plugin API

■ GlueCordova Plugin 개요

GlueCordova Plugin API

Javascript API를 활용하여 모바일 장치 제어 및 접근 가능
= Cordova Plugins(16종) + GlueCordova Plugins(4종)으로 구성

Cordova Plugins

- Apache Cordova에서 기본적으로 제공하는 플러그인 16종
- Battery, Camera, Contacts, GPS, Device, FileSystem.. 등등

GlueCordova Plugins

- 추가적으로 제작된 플러그인 4종 (Mobile Push, QRCode, 음성인식, NFC)

플러그인명	설명
GCM Notification	구글의 GCM Notification 서비스를 이용할 수 있게 하고 Push 메시지를 받았을 때 단말기에 Notification 이벤트를 발생시킨다.
Barcode Scanner	QR코드 및 바코드를 인식하여 값을 반환한다. 또한 데이터를 QR코드 이미지로 생성한다.
Speech to Text	구글 음성인식 모듈을 이용하여 음성을 텍스트로 변환한다.
NFC	NFC 스캔 및 쓰기 기능을 제공한다.

표 1. Glue Cordova Plugins 목록

GlueCordova Plugin API

■ Cordova Plugins API

자바스크립트를 이용해 모바일 장치에 접근 및 제어할 수 있도록 해준다.

The screenshot shows the 'Supported Platforms' section of the Cordova website. On the left, a list of plugins is shown, with a red box highlighting the first few: Accelerometer, Camera, Capture, Compass, Contacts, File, GeoLocation, Media, Network, Notification, and Storage. On the right, a table shows the compatibility of these plugins across various platforms: iOS, Android, BlackBerry, Firefox OS, HTML5, Java, Node.js, PhoneGap, and Windows. A red box highlights the top row of the table, which contains the platform names and their icons. Below the table, there is a link to the 'Updated list' at <http://wiki.apache.org/cordova/PlatformSupport>.

지원되는
API 목록

지원되는
플랫폼 목록

그림12. Apache Cordova 플러그인 목록 & 호환성 체크

http://cordova.apache.org/docs/en/4.0.0/cordova_plugins_pluginapis.md.html#Plugin%20APIs

GlueCordova Plugin API

■ Cordova Plugins 기본 API 활용

Code Example

```
<!DOCTYPE html>
<html>
<head>
<title>Device Properties Example</title>
<script type="text/javascript" charset="utf-8" src="cordova-2.0.0.js"></script>
<script type="text/javascript" charset="utf-8">
// Wait for Cordova to load
document.addEventListener("deviceready", onDeviceReady, false);

// Cordova is ready
function onDeviceReady() {
    navigator.geolocation.getCurrentPosition(onSuccess, onError);
}

// onSuccess Geolocation
function onSuccess(position) {
    var element = document.getElementById('geolocation');
    element.innerHTML = 'Latitude: ' + position.coords.latitude
    + 'Longitude: ' + position.coords.longitude
    + 'Altitude: ' + position.coords.altitude
    + 'Accuracy: ' + position.coords.accuracy
    + 'Altitude Accuracy: ' + position.coords.altitudeAccuracy;
}

// onError Call
function onError(position) {
    alert('code: ' + position.code);
}
</script>
</head>
<body>
<p id="geoloca">
</body>
</html>
```

document.addEventListener("deviceready", onDeviceReady, false);

Cordova가 로딩이 될 때, onDeviceReady 함수가 동작

function onDeviceReady() {

navigator.geolocation.getCurrentPosition(onSuccess, onError);

}

플러그인 파일에 정의된 navigator.~~~.~~~ 를 이용하여 로직 구현

APACHE CORDOVA
PLUGINS REGISTRY

Find a plugin... [Browse all 659 plugins](#)

[← Back to search results](#)

Plugin ID	0.3.11
org.apache.cordova.geolocation	
Description	Total downloads
Cordova Geolocation Plugin	316701
Maintainers	Version
anis, stevegill, maxw, agrieve, purplecabbage, ian, cmarcelk	0.3.11
	Last

Methods

- navigator.geolocation.getCurrentPosition
- navigator.geolocation.watchPosition
- navigator.geolocation.clearWatch

Plugins API 활용 방법에 대해서는 Apache Cordova 공식 사이트를 활용한다.

<http://plugins.cordova.io/#/package/org.apache.cordova.camera>

그림13. Apache Cordova Plugin Registry

GlueCordova Plugin API

■ Glue Mobile에서 Cordova Plugin API 구성하기

- Main.js 에서 cordova 라이브러리 정의

```
1 require.config({
2   baseUrl: 'js',
3   paths: {
4     'text': 'libs/text',
5     'jquery': 'libs/jquery',
6     'underscore': 'libs/underscore',
7     'backbone': 'libs/backbone',
8     'routefilter': 'libs/backbone.routefilter',
9     'jquerymobile': 'libs/jquerymobile',
10    'handlebars': 'libs/handlebars-v1.3.0',
11    'datatables': 'libs/jquery.dataTables',
12    'datatables-fixedcolumns': 'libs/dataTables.fixedColumns',
13    'glue': 'libs/glue',
14    'cordova': 'libs/cordova',
```

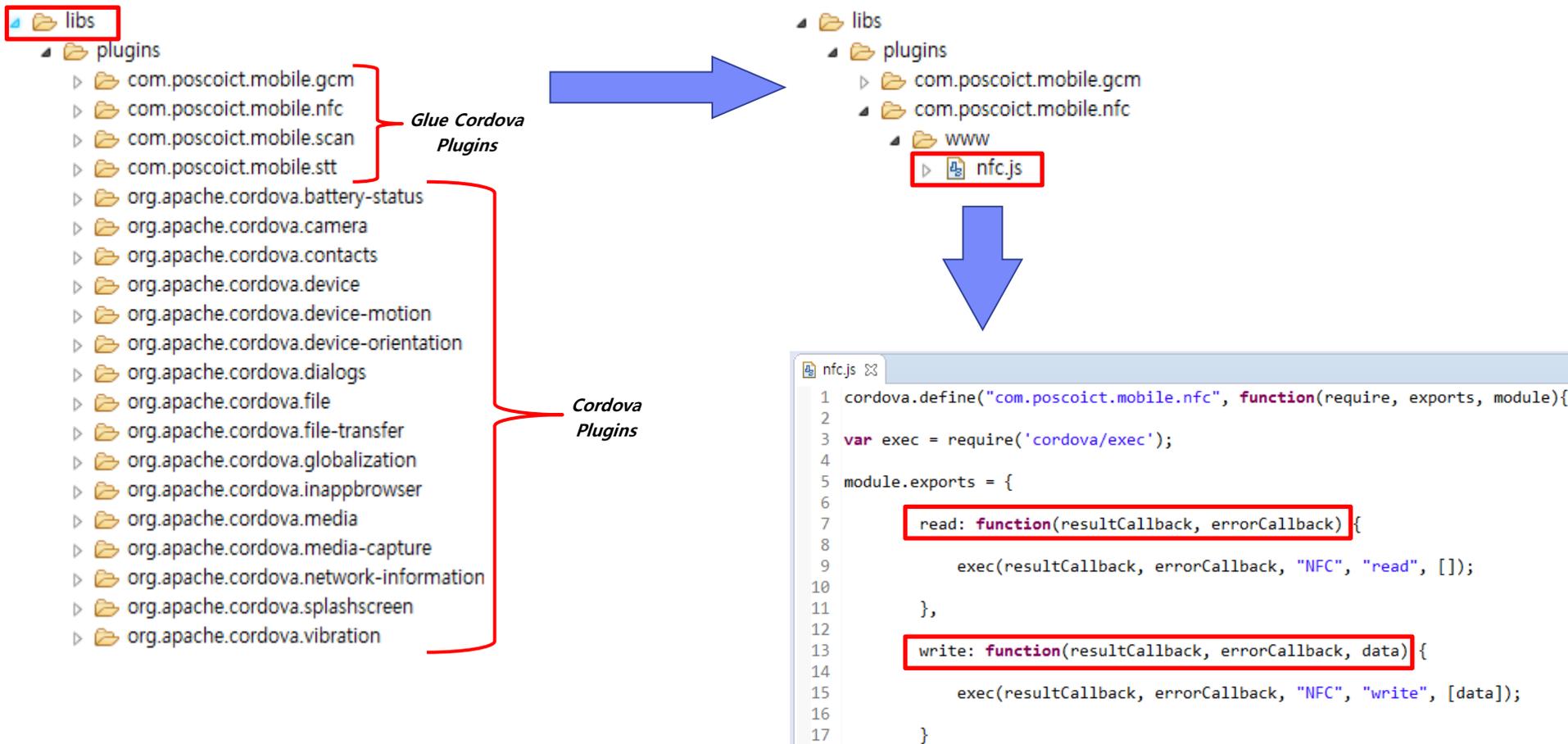
- Controller.js 에서 cordova 호출

```
1 define([
2   'cordova'
3 ], function() {
```

GlueCordova Plugin API

■ Glue Mobile에서 Cordova Plugin API 사용법

- Cordova API 참조하는 방법



■ Cordova Plugins API 실습

Contacts API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView를 추가하고, Palette를 이용해 버튼을 생성
2. 버튼을 클릭할 시 이벤트를 수행할 수 있도록 app.js에 등록, 이벤트는 Cordova Contacts API를 등록
※ navigator.contacts.pickContact (successCallback, errorCallback, options)
3. 버튼을 클릭했을 때, 해당 단말기의 연락처 기능이 동작
4. 연락처 기능을 통해서 원하는 연락처를 검색

■ Cordova Plugins API 실습

Camera API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView를 추가하고, Palette를 이용해 버튼을 생성
2. 버튼을 클릭할 시 이벤트를 수행할 수 있도록 app.js에 등록, 이벤트는 Cordova Camera API를 등록
※ navigator.camera.getPicture(successCallback, errorCallback, options)
3. SampleHybrid 템플릿의 아래에 img 요소를 추가하고, 카메라의 찍은 데이터가 해당 img에 나타나게 구현
4. 사진을 찍은 후 저장
5. 화면에 표시

Barcode Scanner (QRCode)

QR코드의 생성 및 스캔 기능을 제공하는 Zxing 라이브러리로 제작한 API

```
01 <head>
02 <script type="text/javascript" charset="UTF-8" src="cordova.js"></script>
03 <script type="text/javascript" charset="UTF-8">
04     function scan(){
05         navigator.scan.scan(scanSuccess, scanFail);
06     }
07     function scanSuccess(message){
08         alert(message);
09     }
10     function scanFail(error){
11         alert(error);
12     }
13 </script>
14 </head>
15 <body>
16     <input type="button" id="Button" onclick="scan();" value="scan"/>
17 </body>
```

Scan

1차원 바코드 및 스캔 인식이 가능

▪ navigator.scan.scan(scanSuccess, scanFail);

- scanSuccess : 스캔 성공 시 호출
- scanFail : 스캔 오류 시 호출

▪ scanSuccess(message) : 스캔 성공시 호출되는 function

- message : 스캔된 값

▪ scanFail(message) : 스캔 오류 시 호출되는 function

- message : 오류 메시지

그림14. QRCode scan API 소스

■ GlueCordova Plugins API 실습

QRCode 스캔 API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView에 Palette를 이용해 버튼을 생성
2. 버튼을 클릭 시 발생 이벤트를 app.js에 등록, 이벤트는 GlueCordova QRCode scan API를 등록
※ navigator.scan.scan(successCallback, errorCallback)
3. 버튼을 클릭하여 QR 코드 인식

Barcode Scanner (QRCode)

QR코드의 생성 및 스캔 기능을 제공하는 Zxing 라이브러리로 제작한 API

```
01 <head>
02 <title> </title>
03 <script type="text/javascript" charset="UTF-8" src="../../cordova.js"> </script>
04 <script type="text/javascript" charset="UTF-8">
05
06 function generate(){
07 //Test Data
08 var data = {"id":"123","dept":"arch"}
09 navigator.scan.generate(generateSuccess, generateFail, data);
10 }
11
12 function generateSuccess(img){
13 document.getElementById("aaa").src = "data:image/png;base64," + img;
14 document.getElementById("aaa").height = 200;
15 document.getElementById("aaa").width = 200;
16 }
17
18 function generateFail(message){
19 alert(message);
20 }
21
22 </script>
23 </head>
24 <body>
<input type="button" id="generateButton" onclick="generate(); />
<img id="aaa" src="" />
</body>
```

Generate

바코드 및 QR코드 생성

- navigator.scan.generate(generateSuccess, generateFail, data);
 - generateSuccess : QR코드 생성 성공 시 호출
 - generateFail : QR코드 생성 실패 시 호출
 - data : QR코드 이미지로 인코딩 할 데이터

- generateSuccess(img) : 이미지 생성 성공 시 호출되는 function
 - message : 스캔된 값

- generateFail(message) : 이미지 생성 실패 시 호출되는 function
 - message : 오류 메시지

※ **Base64** : 2진 데이터를 ASCII 형태의 텍스트로 변환하거나
그 반대로 변환하는 인코딩 방식

그림15. QRCode generate API 소스

■ GlueCordova Plugins API 실습

QRCode 생성 API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView를 추가하고, Palette를 이용해 버튼을 생성
2. 버튼을 클릭할 시 이벤트를 수행할 수 있도록 app.js에 등록, 이벤트는 GlueCordova QRCode generate API 를 등록

※ navigator.scan.generate(successCallback, errorCallback, data)

3. SampleHybrid 템플릿의 아래에 img 요소를 추가하고, 카메라의 찍은 데이터가 해당 img에 나타나게 구현
4. 사진을 찍은 후 저장
5. 화면에 표시

GlueCordova Plugin API

■ STT (Speech To Text)

사용자의 음성 인식을 지원하는 API

```
01 <head>
02 <title> </title>
03 <script type="text/javascript" charset="UTF-8" src="../cordova.js"> </script>
04 <script type="text/javascript" charset="UTF-8">
05     function speechToText(){
06         navigator.stt.startSTT(onSuccess, onFail);
07     }
08     function onSuccess(message){
09         alert(message);
10     }
11     function onFail(error){
12         alert(error);
13     }
14 </script>
15 </head>
16 <body>
17 <input type="button" id="sstButton" onclick="speechToText();" />
18 </body>
19 </html>
```

그림16. STT API 소스

STT sample

사용자의 음성을 인식한 결과를 리스트로 출력하고
리스트에 있는 값 중 하나를 선택하여 해당 값을 반환

- `navigator.stt.startSTT(onSuccess, onFail, '<language>','<prompt>');`
 - `onSuccess` : 음성인식 성공 시 호출되는 function명
 - `onFail` : 음성인식 오류 시 호출되는 function명
 - `<language>`: 음성인식 언어 (default : ko-KR)
 - `<prompt>` : 음성인식 화면 Prompt (default : Say!)
- `onSuccess(message)` : 음성인식 성공 시 호출되는 function
 - `message` : 음성인식 결과 값
- `onFail(message)` : 음성인식 오류 시 호출되는 function
 - `message` : 오류 메시지

■ GlueCordova Plugins API 실습

STT API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView에 Palette를 이용해 버튼을 생성
2. 버튼을 클릭 시 발생 이벤트를 app.js에 등록, 이벤트는 GlueCordova STT API를 등록
※ navigator.stt.startSTT(successCallback, errorCallback, 'language', 'prompt')
3. 버튼을 클릭하여 음성인식을 수행
4. 음성인식 후 인식된 데이터를 처리

■ NFC

NFC 칩 읽기 / 쓰기 기능을 지원하는 API

```
01 <head>
02 <title></title>
03 <script type="text/javascript" charset="UTF-8" src="../cordova.js"></script>
04 <script type="text/javascript" charset="UTF-8">
05     function readNFC(){
06         navigator.nfc.read(onSuccess, onFail);
07     }
08
09     function onSuccess(message){
10         alert(message);
11     }
12     function onFail(error){
13         alert(error);
14     }
15
16 </script>
17 </head>
18 <input type="button" id="readButton" onclick="readNFC();" />
19 </body>
```

NFC Reading

NFC 태그 칩의 데이터를 읽어온다.

▪ navigator.nfc.read(onSuccess, onFail);

- onSuccess : NFC 읽기 성공 시 호출되는 function명
- onFail : NFC 읽기 오류 시 호출되는 function명

▪ onSuccess(message) : NFC 읽기 성공 시 호출되는 function

- message : NFC 읽기 결과 값

▪ onFail(message) : NFC 읽기 오류 시 호출되는 function

- message : 오류 메시지

그림17. NFC Reading API 소스

■ GlueCordova Plugins API 실습

NFC 스캔 API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView에 Palette를 이용해 버튼을 생성
2. 버튼을 클릭 시 발생 이벤트를 app.js에 등록, 이벤트는 GlueCordova NFC read API를 등록
※ navigator.nfc.read(successCallback, errorCallback)
3. 버튼을 클릭하여 NFC 인식

■ NFC

NFC 칩 읽기 / 쓰기 기능을 지원하는 API

```
01 <head>
02 <title> </title>
03 <script type="text/javascript" charset="UTF-8" src="../cordova.js"> </script>
04 <script type="text/javascript" charset="UTF-8">
05     function readNFC(){
06         var data = '{"name": "테스트", "age": "30", "dept": "Arch"}';
07         navigator.nfc.write(onSuccess, onFail, data);
08     }
09     function onSuccess(message){
10         alert(message);
11     }
12     function onFail(error){
13         alert(error);
14     }
15 </script>
16 </head>
17 <input type="button" id="writeButton" onclick="writeNFC"/>
18 </body>
```

그림18. NFC Writing API 소스

NFC Writing

NFC 태그 칩에 해당 데이터를 입력하고 저장

- navigator.nfc.write(onSuccess, onFail, data);
 - onSuccess : NFC 쓰기 성공 시 호출되는 function명
 - onFail : NFC 쓰기 오류 시 호출되는 function명
 - data : NFC Tag에 저장될 데이터

- onSuccess(message) : NFC 쓰기 성공 시 호출되는 function
 - message : NFC 쓰기 결과 값

- onFail(message) : NFC 쓰기 오류 시 호출되는 function
 - message : 오류 메시지

■ GlueCordova Plugins API 실습

NFC 입력 API를 이용한 샘플 예제 작성

1. 기존 샘플 예제에 SampleHybridView에 Palette를 이용해 버튼을 생성
2. 버튼을 클릭 시 발생 이벤트를 app.js에 등록, 이벤트는 GlueCordova NFC write API를 등록

※ navigator.nfc.write(successCallback, errorCallback, data)

3. 버튼을 클릭하여 NFC 데이터 입력

데이터는 임의로 지정하여 입력

End of Document