

# CH 13

---

## 하이브리드앱과 폰갭

1. 하이브리드앱 개요
2. 폰갭 개요
3. 폰갭 안드로이드 개발 환경
4. 폰갭 프로젝트 생성 및 실행 요약

---

>> 학습목표 <<

- ❖ 하이브리드앱 개발 방식의 특성을 이해한다.
- ❖ 폰갭 프레임워크 구성과 개발 절차에 대해 알아본다.
- ❖ 폰갭 안드로이드 개발 환경을 구축한다.
- ❖ 실제 폰갭 프로젝트를 생성하고 패키징하여 실행해본다.

# 1.1 하이브리드앱

## ● 하이브리드앱(hybrid app)

- 화면 등의 사용자 인터페이스는 웹 기술을 이용하고 모바일 장치의 고유 기능은 자바스크립트 API를 이용하여 개발한 앱
- 자바스크립트 API는 하이브리드 프레임워크에 의해 제공
- 다양한 스마트폰 환경과 플랫폼에 웹 방식으로 대응함으로써 비용과 인력을 줄이고자 하는 시장의 요구에 의해 등장

## ● 앱 개발 방식

앱 종류	특징
네이티브앱	특정 모바일 플랫폼에 최적화된 애플리케이션 개발 가능 모든 장치 제어 가능 호환성과 생산성 낮음
웹앱	호환성과 생산성 높음 장치 제어가 어렵고 실행 속도가 느림
하이브리드앱	네이티브앱의 성능과 웹앱의 호환성을 절충 원소스 멀티플랫폼(생산성 향상) 지원 장치 제어 가능

# 네이티브앱과 하이브리드앱 개발 방식 비교

## ● 하이브리드앱 <-> 웹앱

- 웹앱에 비교해서 장치 관련 다양한 네이티브 기능을 사용할 수 있고 플러그인 기능을 통해 확장이 가능
- 모바일 플랫폼마다 네이티브앱처럼 패키징해서 배포해야 하고 앱 마켓에 등록해야 유통시킬 수 있어 웹앱에 비해서 불편

## ● 하이브리드앱 <-> 네이티브앱

- 앱 기능의 대부분을 웹 기술을 사용하므로 네이티브앱보다 쉽게 구현
- 네이티브앱과는 다르게 개발, 시험, 배포 전체 과정을 한 번에 수행



네이티브앱 방식

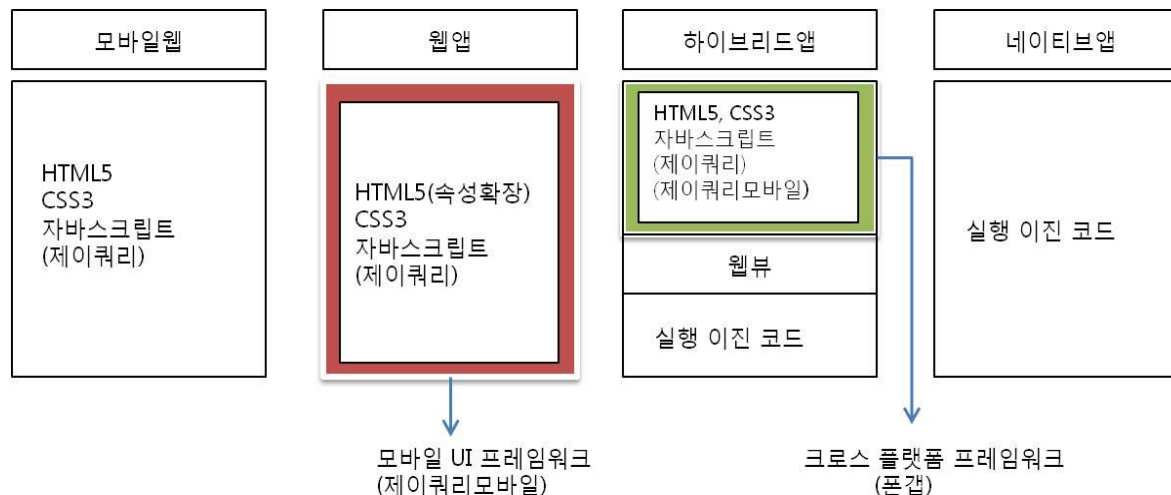


하이브리드앱 방식

# 하이브리드 개발 유형

## ● 모바일 환경에서의 앱 개발 유형

- 네이티브앱 - 컴파일을 통해 실행 가능한 이진 코드로 생성
- 웹앱 - 텍스트와 미디어 파일로 구성, 배포가 쉽도록 압축 파일로 생성
- 하이브리드앱 - 네이티브앱과 똑같이 플랫폼별로 생성된 이진 실행 파일
  - 하이브리드 프레임워크가 모바일 장치에 접근할 수 있는 자바스크립트 API를 제공하고 이진 코드 파일로 변환
  - 자바스크립트 등의 텍스트와 미디어 파일이 합쳐진 패키지화된 웹 애플리케이션
  - 웹뷰(WebView) : 화면에 웹 내용(HTML5 문서)을 표시하기 위한 앱의 구성 요소



## 1.2 하이브리드 프레임워크

### ● 특성

- 원소스 멀티플랫폼(one-source multi-platform) 앱 개발 방식을 지원
- 하나의 코드를 다양한 모바일 플랫폼에 맞는 여러 네이티브앱으로 패키징하여 등록과 배포 지원
- 크로스플랫폼(cross-platform) 앱 개발 방식을 지원
- 다양한 플랫폼과 개발 언어를 모르더라도 표준 웹 기술만으로 앱을 개발할 수 있고 애플의 앱스토어나 안드로이드의 플레이스토어에 등록 가능

### ● 하이브리드 앱을 만들면 유용한 경우

- 표준 웹 기술로 독립적으로 실행 가능한 앱을 만들고자 할 경우
- 모바일 장치를 활용하는 앱을 만들고자 할 경우
- 다양한 플랫폼에 동작하는 크로스플랫폼 앱을 쉽게 만들고자 할 경우
- 마켓에 등록하여 배포하고자 할 경우
- 내용이 빈번하게 갱신되는 앱을 만들고자 할 경우

# 하이브리드 프레임워크

## ● 하이브리드 플랫폼 종류

플랫폼	특성
폰갭(PhoneGap)	대표적인 공개 소스 하이브리드 플랫폼 다양한 플랫폼 지원 장치 제어용 자바스크립트 API 제공 제이쿼리, 센차터치 UI 프레임워크를 주로 사용
티타늄(Titanium)	프로그래밍에 친숙한 사용자들이 많이 사용 자바스크립트 언어 중심의 UI와 코드 개발 방식 자바스크립트 API 사용
앱스프레소(Appspresso)	국내에서 개발한 자바스크립트 기반 언어 방식 플랫폼 실행 속도가 느림

## 2.1 폰갭(1)

### ● 폰갭

- 이름은 'phone'과 'gap'의 합성어(다양한 모바일 플랫폼의 차이를 해결한다는 의미)
- 여러 플랫폼의 SDK를 통합하여 하나의 SDK로 개발하도록 지원 -> 크로스플랫폼
- 공개 소스인 폰갭은 iOS, 안드로이드, 블랙베리, WebOS, 윈도우폰7, 심비안, 바다 등의 스마트폰 플랫폼을 지원하는 유일한 프레임워크
- 어도브사에 인수되어 버전 1.5.0부터는 '코르도바'(Cordova)로 이름이 변경(자바 스크립트 파일 명칭도 변경) 상업용 솔루션 이름은 폰갭을 그대로 사용
- 통합 변환기' 역할을 수행
  - HTML5와 CSS3, 제이쿼리, 자바스크립트(폰갭 API)로 작성된 하나의 소스 코드를 각 각의 스마트폰 플랫폼에 맞는 언어로 변환하여 앱을 생성

### ● 웹킷(WebKit) 엔진

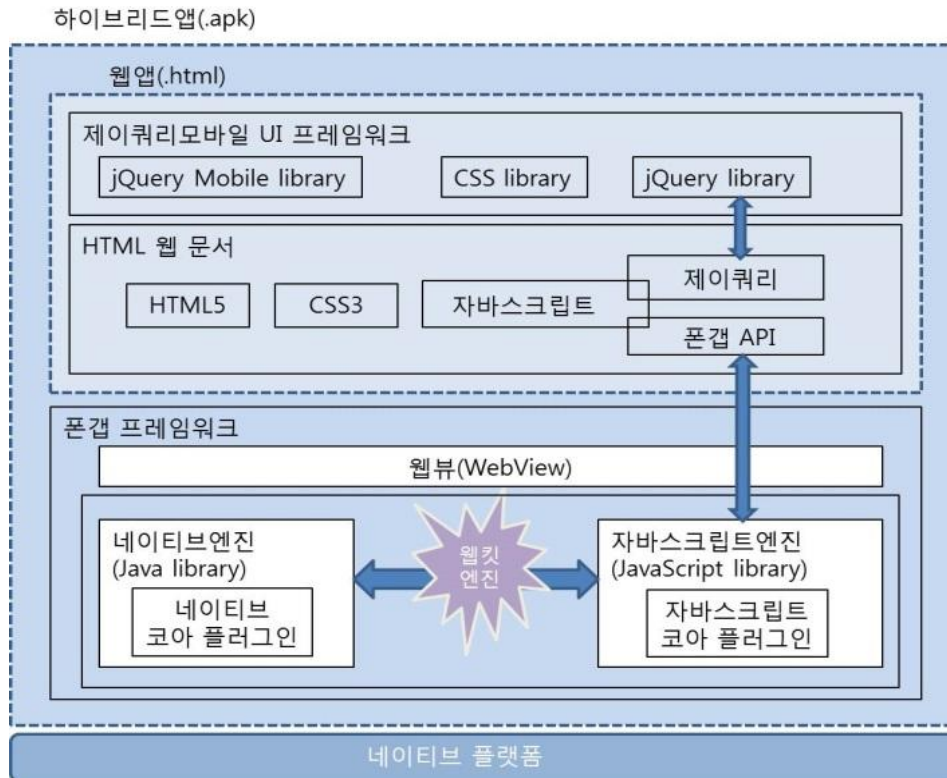
- 폰갭 프레임워크는 웹 콘텐츠를 볼 수 있는 웹뷰(WebView) 컴포넌트를 포함하는데 이는 웹킷 엔진을 사용하여 구현
- 웹 콘텐츠를 웹 브라우저에 해석하여 표현해주는 공개소스 프레임워크
- 사파리, 크롬 등 많은 웹 브라우저의 기본 엔진으로 사용
  - 웹 코어(WebCore) : DOM 파서, 레이아웃 및 렌더링 엔진
  - 자바스크립트 코어(JavaScriptCore) : 자바스크립트 엔진
  - 드로세라(Drosera) : 자바스크립트 디버거



# 폰갭(2)

## ● 폰갭 프레임워크

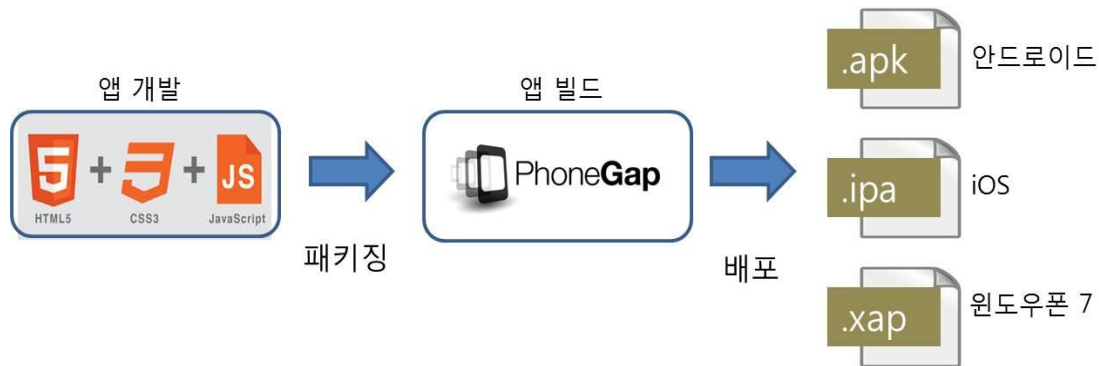
- cordova-1.x.x.js 자바스크립트 라이브러리와 cordova-1.x.x.jar(x.x는 버전) 네이티브 라이브러리로 구성(자바스크립트 엔진과 네이티브 엔진을 구성)
- 폰갭 하이브리드 앱 구성



## 2.2 폰갭 개발 절차

### ● 폰갭으로 하이브리드앱을 만드는 과정

- 웹 표준을 따르는 웹앱을 개발 -> 웹앱을 폰갭 라이브러리를 이용해 패키징 -> 웹킷을 제공하는 브라우저가 탑재된 장치에 배포 -> 배포된 앱은 웹 브라우저 안의 웹뷰 컴포넌트를 기반으로 실행
- 개발된 하나의 앱은 플랫폼별로 각각 다르게 변환(빌드)되어야 함
  - 안드로이드의 경우, 이클립스와 안드로이드 SDK 개발 환경을 사용하여 패키징
  - 모바일 장치 별로 폰갭 API 지원 범위가 다르므로 고려가 필요
- 폰갭 앱의 개발 과정

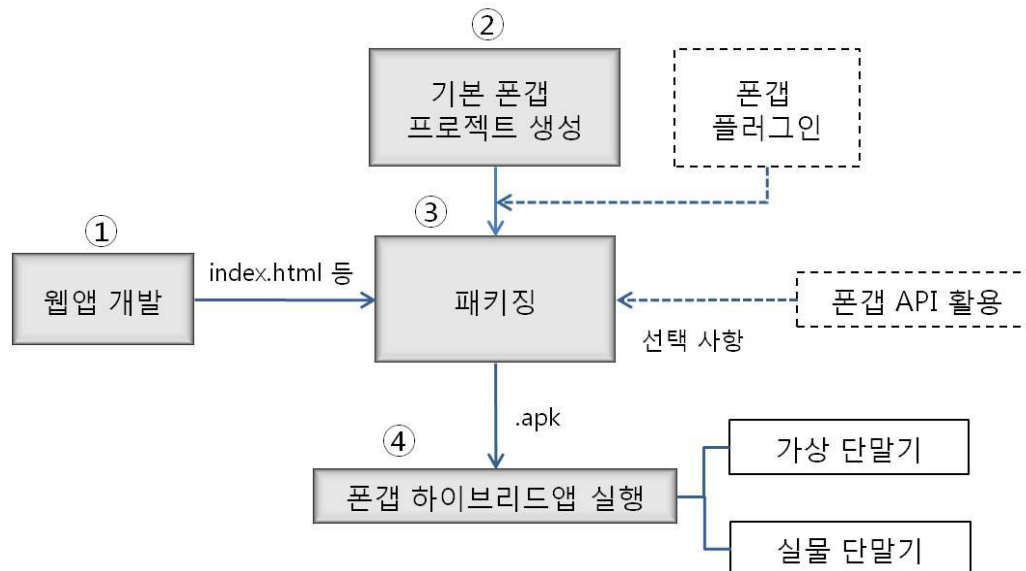


## 3.1 안드로이드 개발 환경 구축

- 안드로이드용 폰갭 앱을 개발하려면 먼저, 안드로이드 애플리케이션을 개발하는데 필요한 개발 환경이 설치되어야 함
- [부록C]의 안드로이드 개발 환경 설치하기 내용을 참고하여 개발 환경을 구축
  - Java SE Development Kit(JDK) 7 설치
  - 안드로이드 통합 개발 환경(IDE)인 이클립스(Eclipse) 설치
  - ADT(Android Development Tools) 이클립스 플러그인 설치
  - 안드로이드 개발 도구인 Android SDK(Software Development Kit) 설치
  - 안드로이드 플랫폼 추가 설치
  - 가상 단말기 AVD(Android Virtual Device) 설치

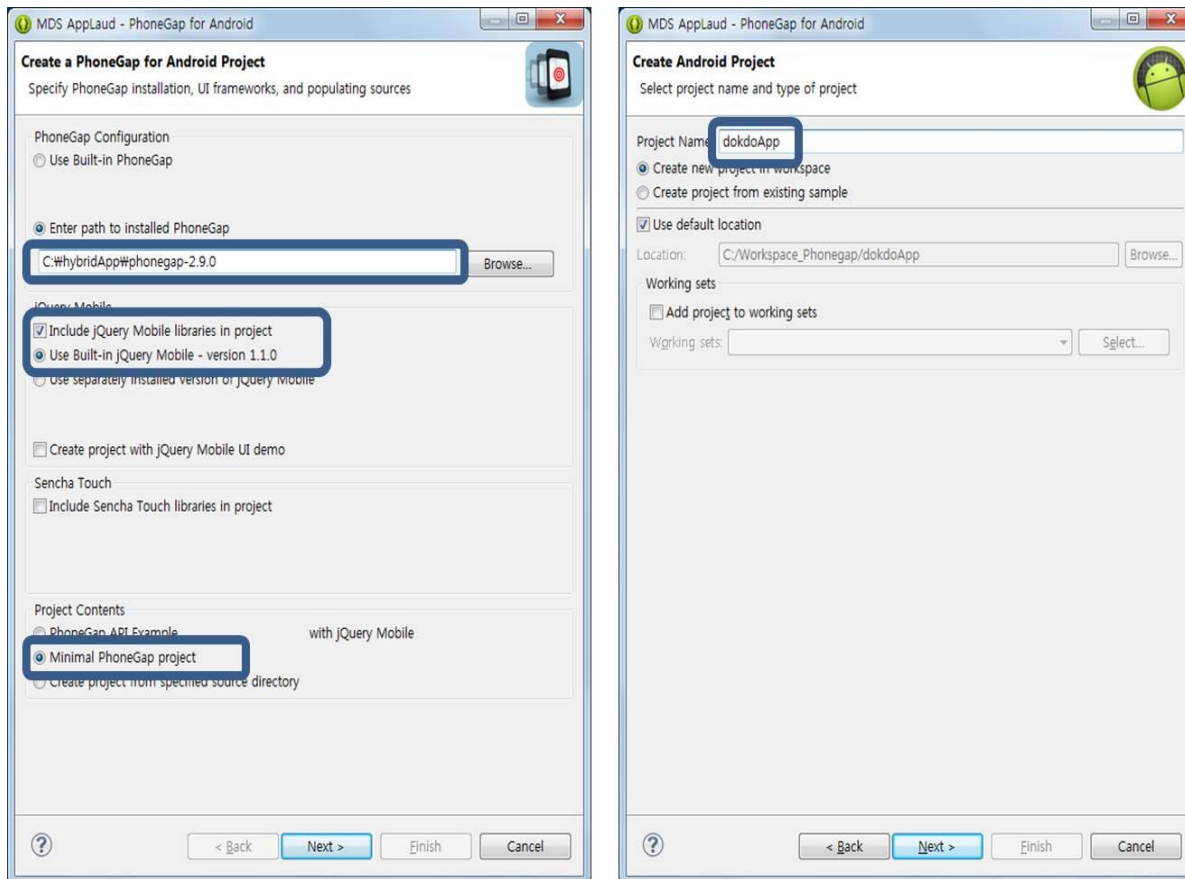
## 3.2 폰갭 프로젝트 개발 환경 구축

- 추가로 폰갭 프레임워크 설치되어 있어야 함
- [부록D]의 폰갭 설치하기 내용을 참고, 이클립스에서 폰갭 프로젝트를 개발할 수 있는 환경을 구축
  - 이클립스 폰갭 플러그인 설치
- 폰갭 하이브리드앱의 개발 과정



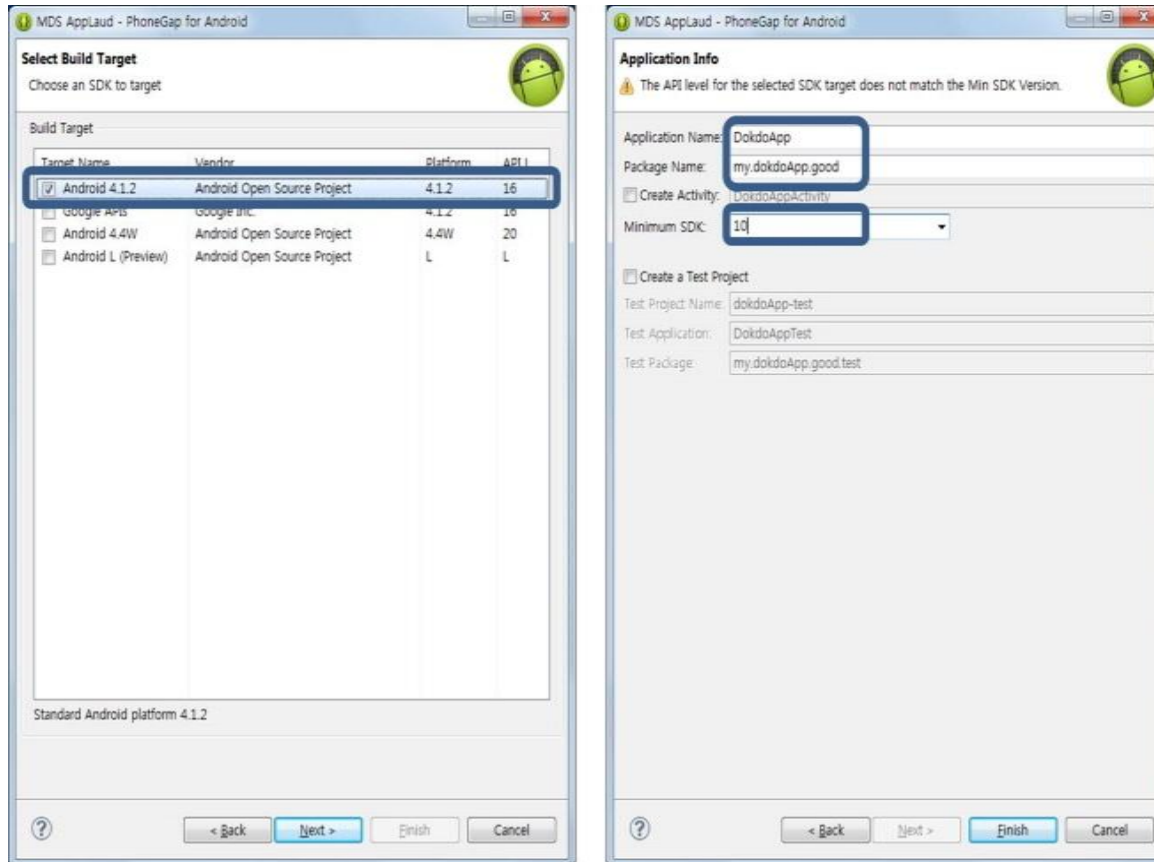
## 4.1 폰갭 기본 프로젝트 생성하기(1)

- 부록D에서 설치한 MDS AppLaud 마법사를 이용
- [그림 13-6] 안드로이드 폰갭 프로젝트 생성(1)



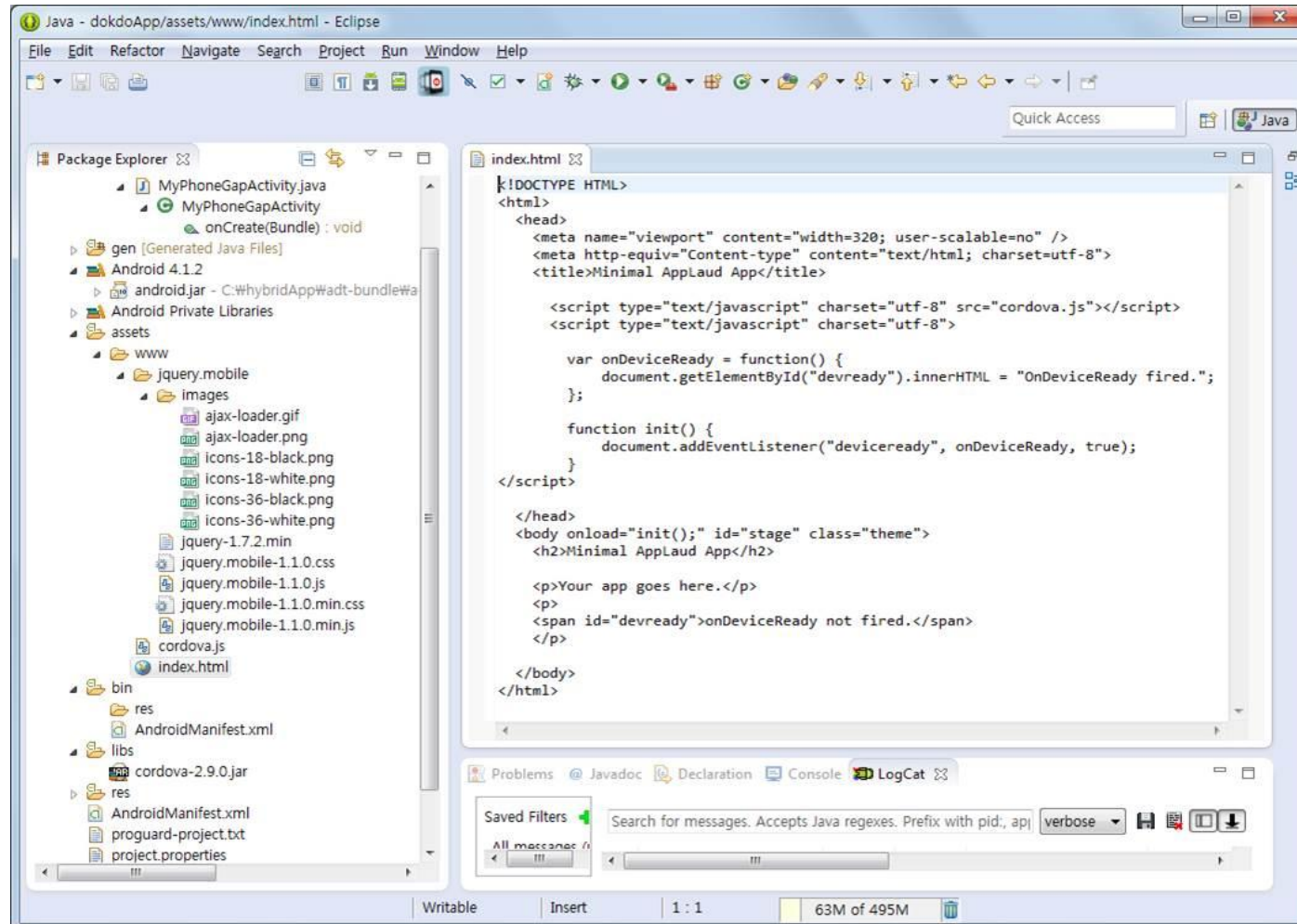
## 폰갭 기본 프로젝트 생성하기(2)

- [그림 13-7] 안드로이드 폰갭 프로젝트 생성(2)



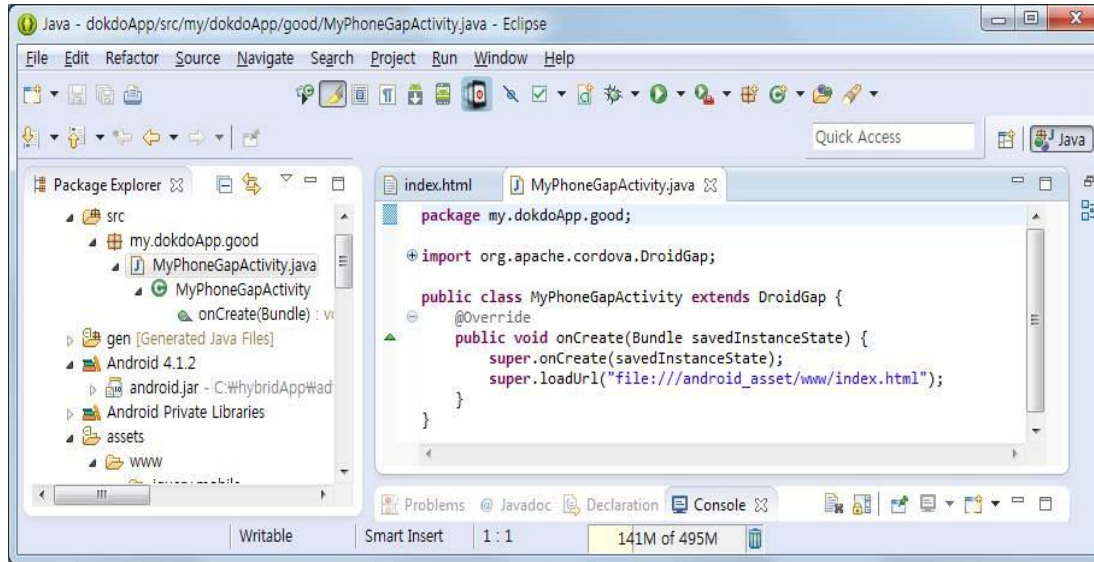
# 폰갭 기본 프로젝트 생성하기(3)

- [그림 13-8] dokdoApp 폰갭 프로젝트 초기 생성 내용



# 폰갭 기본 프로젝트 생성하기(4)

## • [그림 13-9] MyPhoneGapActivity.java 래퍼



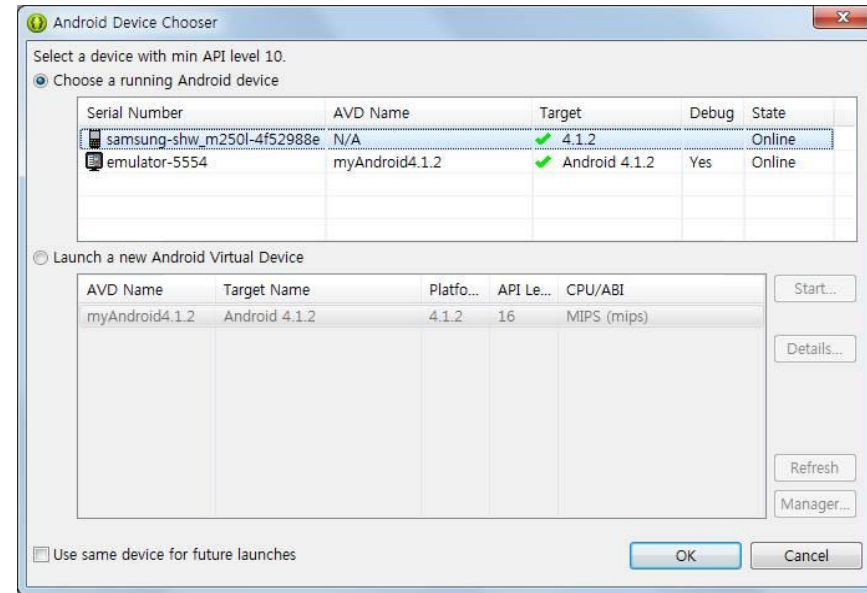
- 안쪽의 내부 코드는 index.html 웹 문서로 구성되지만 외부는 안드로이드 자바 프로그램으로 포장하기 위해 필요한 코드
- 'import org.apache.cordova.DroidGap;' 코드 변경
  - cordova-2.9.0.jar 파일 안의 폰갭 라이브러리 함수를 사용하기 위해
- MyPhoneGapActivity 클래스의 상위 클래스가 'Activity'에서 'DroidGap'으로 변경
  - 안드로이드 자바 프로젝트를 안드로이드 폰갭 프로젝트로 변환하기 위해





## 4.3 폰갭 프로젝트 실행하기

- [그림 13-11] 실행 단말기 선택



- [그림 13-12] dokdoApp 실물 단말기 실행 결과

