

# 헬로, 안드로이드

2주차 - 주요 개념

강대기

동서대학교 컴퓨터정보공학부

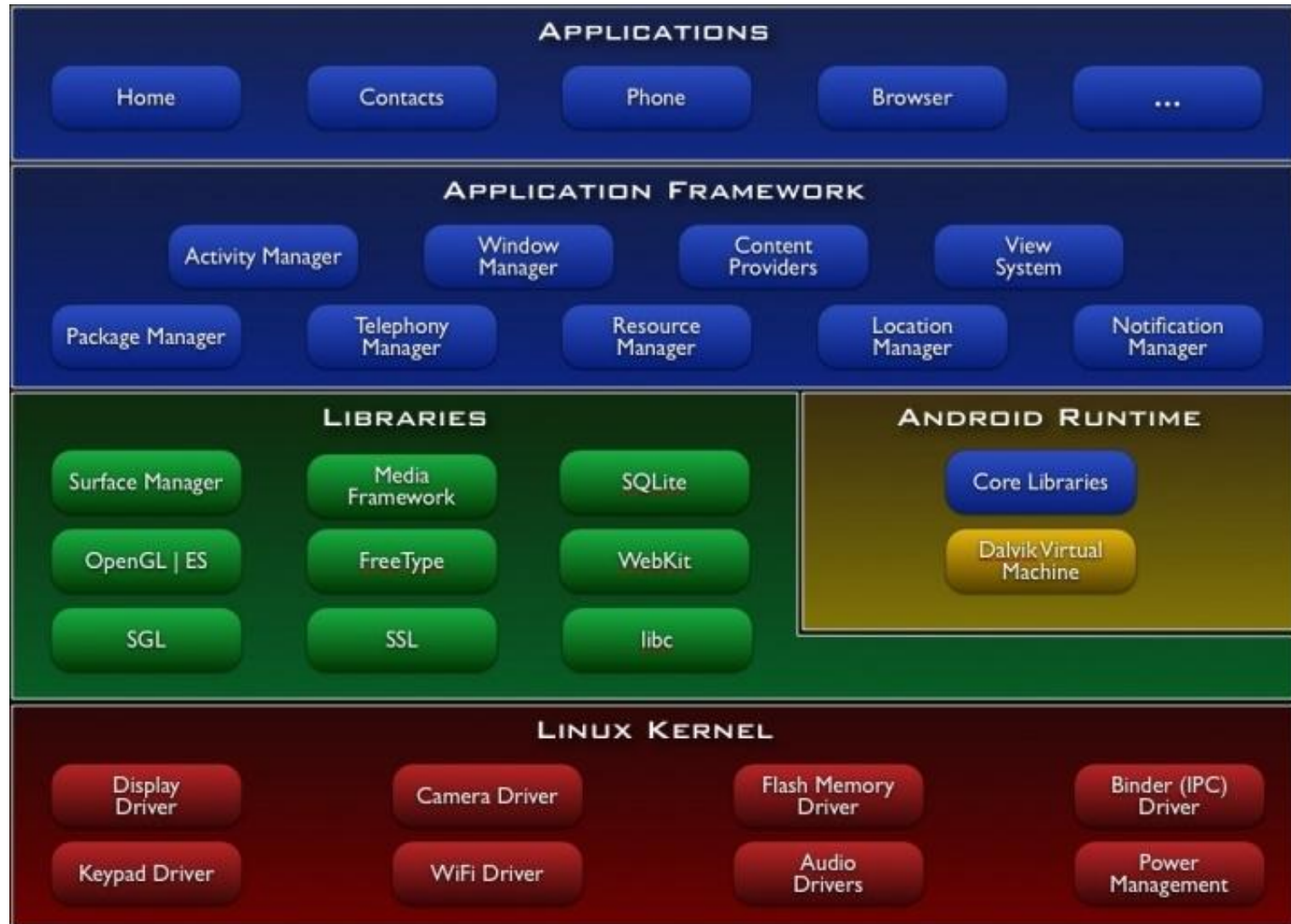
# 학습 목표

- 안드로이드의 전체 구조를 설명할 수 있다.
- 안드로이드의 내장 라이브러리들을 설명할 수 있다.
- 안드로이드의 애플리케이션 프레임워크를 설명할 수 있다.
- 안드로이드의 리소스 사용의 기본적인 사항을 이해한다.
- 안드로이드의 안전과 보안을 보장하기 위한 기본적인 사항을 이해한다.

# 차례

- 전체 구조
- 액티비티 생명 주기
- 구성 요소
- 리소스 사용
- 요약
- 퀴즈
- 연습문제

# 안드로이드의 전체 시스템 아키텍처



# 리눅스 커널

- 리눅스 토발즈가 구현한 리눅스 커널 사용
- 리눅스 커널은 대부분의 기능이 커널 내에 같이 구현되는 모노리딕 커널 (monolithic kernel 또는 integrated kernel) 구조임
- 안드로이드의 리눅스 커널도 모노리딕 커널임
- 모노리딕 커널은 파일 크기가 크므로 실은 임베디드 버전에 적합하지 않음
- 몇몇 임베디드 리눅스는 마이크로 커널 구조로 구현되거나, 모듈화로 이러한 단점을 극복함

# Android Debug Bridge (adb)

- 안드로이드 디버그 브리지를 통해 실행 중인 장치나 에뮬레이터 내부의 리눅스 셸로 들어갈 수 있음
- 에뮬레이터를 실행한 후, adb shell 실행
  - `cd C:\Android\android-sdk-windows\tools`
  - `start emulator -avd MyAVD`
  - `adb shell`
  - 일반 유닉스처럼 `ls` 명령이나 `ps` 명령 사용
  - `exit`로 빠져 나옴

# 내장 라이브러리

- C/C++로 구현되었으며, 휴대폰 하드웨어에 특화되어 컴파일되어 업체에 의해 미리 설치됨
- 중요 라이브러리들
  - 서피스 매니저 - Vista 나 Compiz와 비슷하지만 훨씬 간단한 윈도우 매니저. 오프-스크린 비트맵 사용
  - 2D/3D 그래픽 - 2차원 요소와 3차원 요소가 하나의 사용자 인터페이스에서 결합됨
  - 미디어 코덱 - 동영상 재생, 오디오 녹음과 재생 지원.
  - SQL 데이터베이스 - SQLite 데이터베이스 엔진. 아이폰 및 아이폰에도 사용되는 데이터베이스 엔진임
  - 브라우저 엔진 - WebKit 라이브러리 사용. 구글 크롬, 애플 사파리, 애플 아이폰, 노키아 S60 등에 동일하게 사용되는 엔진임

# 안드로이드 런타임

- 달빅 가상 머신 (Dalvik virtual machine)
- 구글의 댄 본스타인이 만든 VM
- 달빅이란 이름은 댄 본스타인의 조상이 살던 아이슬란드의 어느 마을 이름
- 모바일 장비의 낮은 메모리에 최적화된 VM
- .class와 jar가 변환되어 더 콤팩트해진 .dex 파일을 실행시킴
- 안드로이드 런타임은 코어 자바 라이브러리를 가지는 데, J2SE나 J2ME와 다르나 상당히 중복됨



# 애플리케이션 프레임워크

- 액티비티 매니저 - 애플리케이션의 생명주기를 제어하고 사용자 내비게이션을 위한 백스택(back-stack)을 유지함
- 콘텐츠 제공자 - 주소록과 같이 애플리케이션 사이에서 공유되는 정보를 요약함
- 리소스 매니저 - 리소스란 프로그램에서 코드 외의 모든 부분임
- 위치 매니저 - 안드로이드 폰은 항상 자신의 위치를 파악하고 있음
- 알림 매니저 - 도착 메시지, 약속, 근접성 알림 등을 사용자에게 방해되지 않는 방식으로 전달함
- 포용(embrace)과 확장(extend) - 안드로이드에선 모든 애플리케이션이 평등한 수준으로 실행됨. 이는 안드로이드 기본 애플리케이션 대신 사용자가 만든 것을 대신 사용할 수 있음을 의미

# 애플리케이션

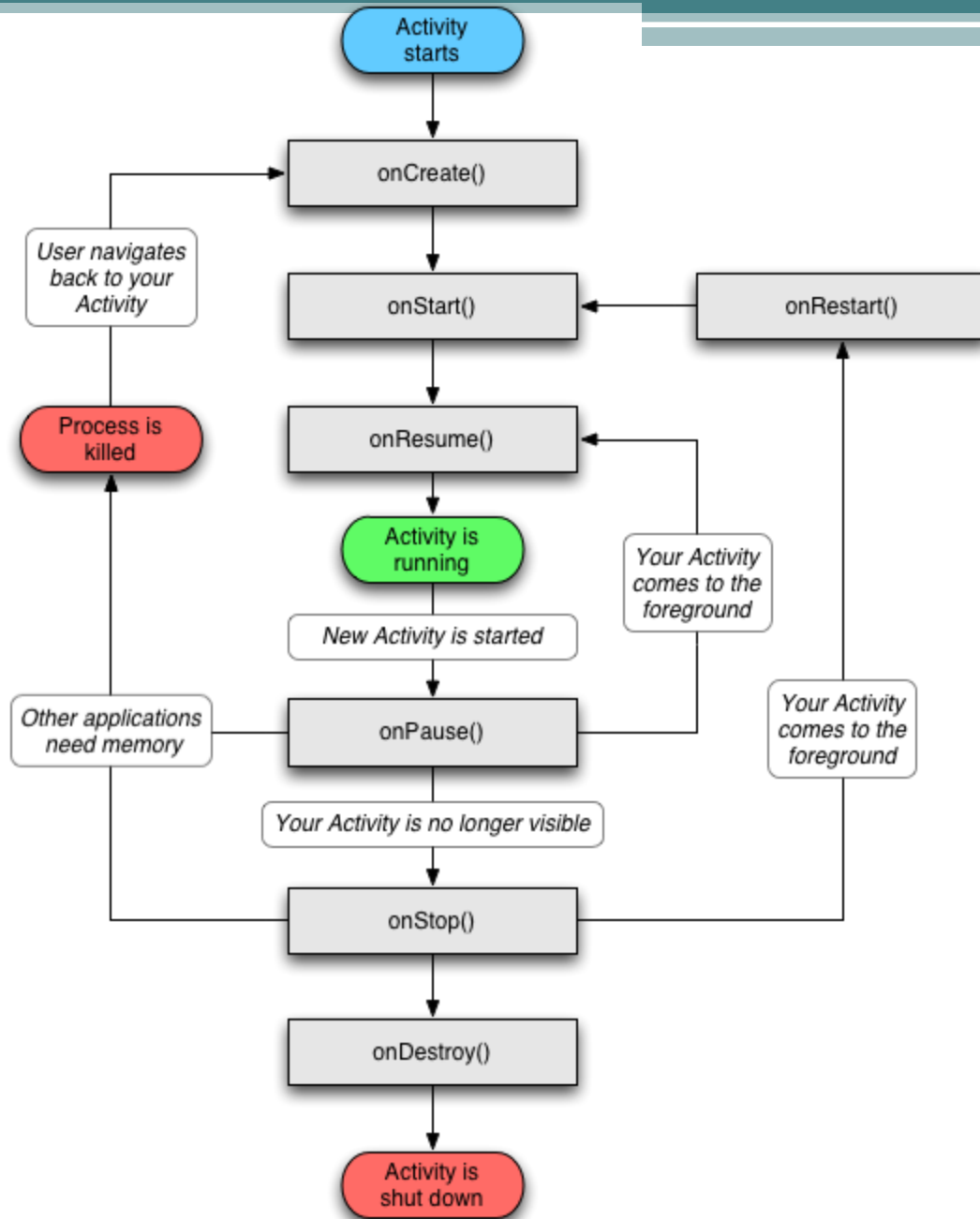
- 이미 설치되어 제공되는 어플리케이션
  - 전화 다이얼 장치
  - 이메일
  - 주소록
  - 웹 브라우저
  - 안드로이드 마켓

# 대부분의 스마트폰의 작동 방식

- 하나의 포어그라운드 애플리케이션이 화면 전체를 차지함
- 사용자가 전원을 켜올 때, 가장 먼저 보는 것은 홈 (Home) 화면 - 시간, 배경 이미지, 사용 가능한 애플리케이션 목록 등을 보여줌
- 사용자가 애플리케이션을 실행하면, 그 프로그램이 포어그라운드가 됨
- 모든 프로그램과 화면은 시스템의 액티비티 매니저에 의해 애플리케이션 스택에 기록됨 - 이전 (back) 버튼으로 이전 화면으로 이동함

# 액티비티

- 개별 사용자의 인터페이스 화면은 Activity 클래스로 표현됨
- 액티비티는 생명 주기를 가지고 있음
- 여러 개의 액티비티에 하나의 리눅스 프로세스를 합쳐서, 하나의 애플리케이션이 됨
- 액티비티가 특정 상태로 바뀌면, 그에 상응하는 콜백 메서드가 (onXXX 라는 이름) 실행되며, 이를 오버라이드하여 개발자가 자신의 코드를 실행시킬 수 있음
- 새 액티비티를 위해 포어그라운드의 액티비티나 리눅스 프로세스가 중지될 수 있음
- 다음 번에 필요한 데이터는 onPause()에서 저장



# 덮개 젖히기

- 상태 저장용 코드가 잘 작동하는지를 알아보려면, 폰을 열거나, 가로 세로 방향을 바꿔보면 됨.
- 가로, 세로 방향이 바뀌면, “시스템 일시 중지, 정지, 소멸, 액티비티 새 인스턴스 생성”의 상태 변화 과정을 거침
- 에뮬레이터에선 CTRL-F11이나 키패드 키의 7 또는 9를 누르면 됨

# 구성요소

- 액티비티 - 응용 프로그램에서 하나의 스크린. 각 액티비티는 Activity 기본 클래스를 확장한 하나의 클래스로 구현됨. 각 화면은 하나의 액티비티로 구현됨
- 인텐트 - 우리 말로 의도, 의지 라는 의미임. 하나의 화면에서 다른 화면으로 넘어가기 위해서는 인텐트라는 특별한 클래스를 사용함. 인텐트 필터는 액티비티 또는 브로드캐스트리시버(BroadcastReceiver)가 어떤 인텐트를 다룰 수 있는지를 표현한 것임
- 서비스 - 유닉스/리눅스의 데몬(daemon)과 같은 것으로 사용자 인터페이스 없이 오랜 시간 동안 실행되는 프로그램 (윈도우에서도 서비스라는 용어를 사용함)
- 콘텐츠 제공자 - 데이터의 읽기와 쓰기를 위해 데이터가 사용자 API로 포장된 것으로, 사용자의 응용 프로그램이 다른 응용 프로그램들과 데이터를 공유하게 하는 데 매우 유용함

# 리소스

- 프로그램에 필요한 지역화된 텍스트, 비트맵 등의 코드 외의 정보
- 프로젝트 내의 res 디렉터리에서 리소스를 만들고 저장함
- 안드로이드 리소스 컴파일러 (aapt)는 파일이 어느 하위 디렉터리에 있는지와 파일 형식에 따라 처리함
- 리소스 컴파일러는 리소스를 압축하고 묶어서 R이라는 클래스를 생성하는 데, R 클래스는 프로그램에 있는 리소스 참조에 사용되는 식별자를 가짐



# 안전과 보안

- 안드로이드에서 애플리케이션은 개별 리눅스 프로세스에서 실행됨
- 각 애플리케이션은 다른 애플리케이션의 데이터를 읽거나 쓸 수 없음
- 애플리케이션이 일부 중요한 작업에 접근하려면, `AndroidManifest.xml` 에서 허가를 받아야 함
  - 인터넷(`INTERNET`)
  - 사용자 주소록 읽기(`READ_CONTACTS`)
  - 사용자 주소록 쓰기(`WRITE_CONTACTS`)
  - 수신 메시지 모니터링(`RECEIVE_SMS`)
  - 하등의 위치 제공자 사용(`ACCESS_COARSE_LOCATION`)
  - 정밀 위치 제공자 사용(`ACCESS_FINE_LOCATION`)

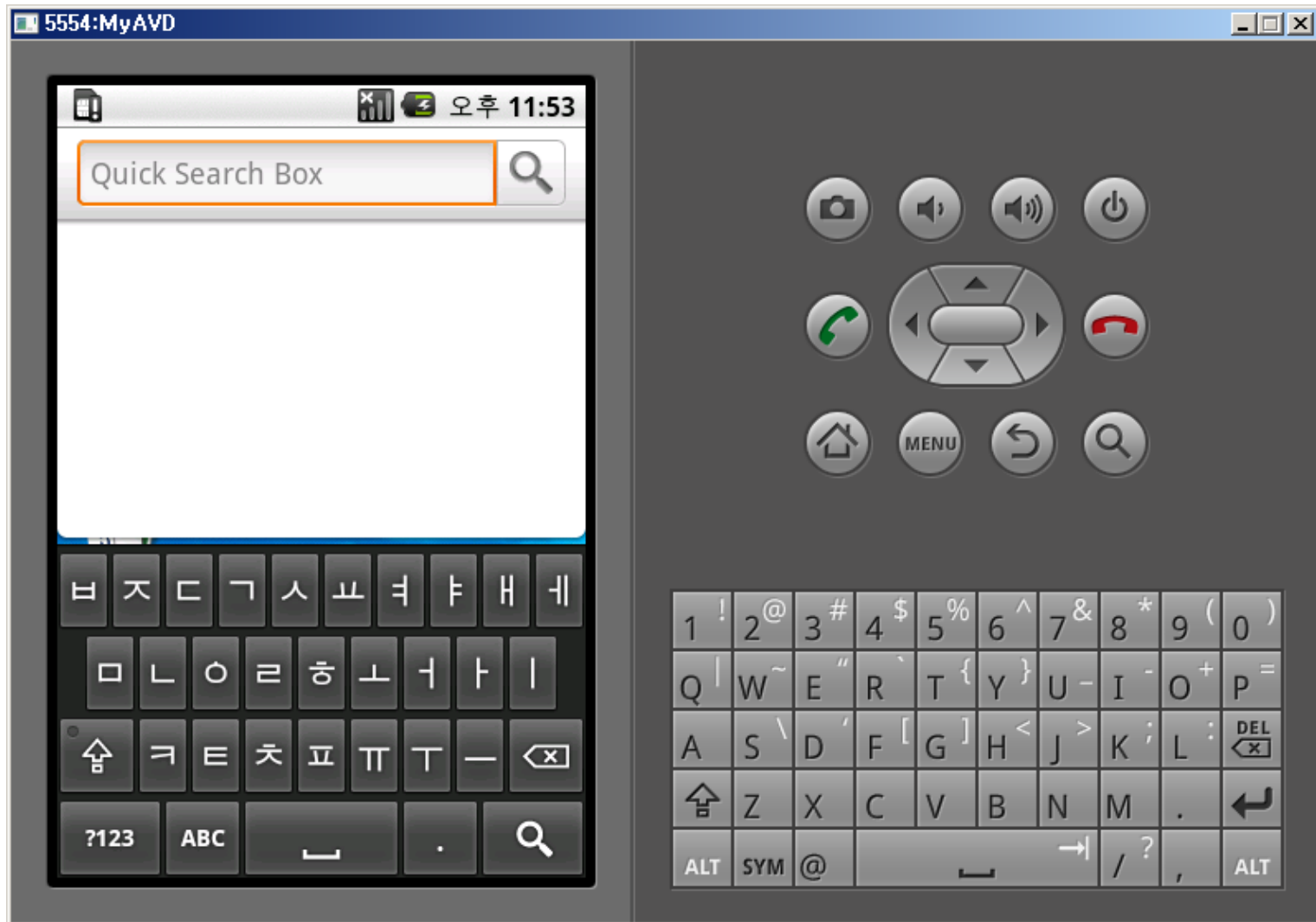
# 안드로이드 한글화

- 이미 앞에서 배웠던 내용 복습
- 에뮬레이터를 실행한 후, Settings 를 실행하고, Locale & text 를 실행하고, Select locale에서 Korean을 선택함
- 화면을 스크롤할 때에는, 터치폰을 사용하듯 마우스를 클릭한 채로 드래깅(dragging)함

# 안드로이드 한글 자판 설치

- 현재 몇 개의 한글 자판을 인터넷을 통해 구할 수 있음 (안드로이드 프로그래밍에 관심이 있다면 한글 자판을 스스로 개발할 수도 있음)
- 박성서씨가 개발한 접촉식 한글 자판을 설치
- 우선 최신 자판의 apk 파일을 받음
  - <http://www.androidpub.com/keyboard>
- 최신 버전의 글을 선택하여 바이너리를 받음
  - <http://www.androidpub.com/keyboard/198992>
- 받은 파일 HanguKeyboard.apk 을 C:\Android\android-sdk-windows\tools 디렉토리로 옮기고 cd 명령으로 그 디렉토리로 변경함
- start emulator -avd MyAVD 로 에뮬레이터 실행
- adb install HanguKeyboard.apk 로 한글 자판 설치
- 에뮬레이터에서 설정 > 로케일 및 텍스트로 가서 “한글 접촉식 키보드”만 남기고 나머지 키보드는 체크를 지움

# 한글 자판이 설치된 모습



# 요약

- 안드로이드의 구조를 설명하였다.
- 안드로이드의 내장 라이브러리들을 설명하였다.
- 안드로이드의 애플리케이션 프레임워크를 설명하였다.
- 안드로이드의 리소스 사용의 기본적인 사항을 설명하였다.
- 안드로이드의 안전과 보안을 보장하기 위한 기본적인 사항을 설명하였다.

# 퀴즈

- 안드로이드의 중요한 내장 라이브러리들은 무엇이 있는가?
- 안드로이드의 애플리케이션 프레임워크에는 어떠한 것들이 있는가?
- 액티비티를 설명하라.
- 인텐트란 무엇인가?
- 서비스란 무엇인가?
- 콘텐츠 제공자란 무엇인가?
- 리소스란 무엇인가?

# 연습문제

- 이클립스 내의 Hello Android 프로그램에서 Activity 를 찾아보자.
- 이클립스 내의 Hello Android 프로그램에서 AndroidManifest.xml 을 찾아보자.
- 이클립스 내의 Hello Android 프로그램에서 리소스를 위한 R 을 찾아보자.
- 에뮬레이터에서 안드로이드에 이미 설치되어 기본적으로 제공되는 애플리케이션들을 찾아보자.
- 에뮬레이터에 한글 자판을 설치해 보자.