1. 개요

- 최신 버전의 안드로이드 스튜디오와 안드로이드(SDK)사용해서 안드로이드 애플리케이션을 개발하는데 필요한 기술을 배울 수 있다.

안드로이드 스튜디오의 장점.

- 유연성이 좋은 그래들 기반의 자동화 프로젝트 빌드 시스템을 사용한다.

- 하나의 프로젝트 코드 여러 안드로이드 장치용 어플리케이션을 구현할 수 있다.

- 코드의 성능이나 버전 호환성 및 기타 문제점을 잡아내는 lint의 기능이 강화되었음.

2. 안드로디으 스튜디오 개발 환경 구성하기

- 안드로이드 SDK는 자바 프로그래밍 언어를 사용해서 개발되었다.

2.1 자바JDK 를 설치한다.

2.2 안드로이드 스튜디오 패키지를 다운로드 및 설치한다.

2.3 각 OS별 안드로이드 스튜디오 패키지를 설치하는 방법을 배운다.

3. 안드로이드 스튜디오로 첫 번째 애플리케이션 만들기

- 간단한 애플리케이션을 하나 만들어서 빌드하고 실행하는 법을 배운다.

3.1 신규 안드로이드 프로젝트를 생성한다.

3.2 프로젝트와 SDK 설정을 정의한다.

3.3 액티비티 생성하기

3.4 레이아웃과 리소스 파일의 위치와 사용법에 관련하여 배운다.

4. 안드로이드 스튜디오 UI 둘러보기

안드로이드 스튜디오의 주된 요소는 웰컴 스크린과 메인창이다. 열린 프로젝트는 각각 자신의 메인 창을 가진다. 안드로이드 스튜디오의 거의 모든 작업은 단축키를 사용해서 할 수 있다. 메인창의 메뉴에서 help -> Default Keymap Reference을 선택하면 언제든지 모든 단축키 내역을 조회할 수 있다.

5. 안드로이드 스튜디오에서 AVD 생성하기

- 안드로이드 애플리케이션은 실제 장치 또는 안드로이드 가상장치(AVD, android Virtual Device) 에뮬레이터 환경에서 설치 및 실행할 수 있다. AVD는 AVD 매니저 도구를 통해 생성되고 관리된다. AVD매니저는 명령 행 도구 또는 그래픽 사용자 인터페이스를 통해 사용 될 수 있다. 특정 안드로이드 장치 모델로 모의 실행하기 위해서는 AVD가 실제 장치의 하드웨어 명세와 일치하도록 구성하는 것이 중요하다.

6. 안드로이드 스튜디오 버전2, AVD 에뮬레이터 사용과 구성하기

- 안드로이드 스튜디오 버젼2에서는 새롭고 기능이 개선된 AVD 에뮬 레이터 환경을 포함하고 있으며, 실제 안드로이드 장치에서 앱을 실행하지 않고도 쉽게 테스트할 수 있다.

7. 안드로이드 장치에서 애플리케이션 테스트 하기.

- 실행 중인 AVD 인스턴스와 안드로이드 장치 모두와의 통신은 ADB(Android Debug Bridge)에 의해 처리된다. ABD는 개발 컴퓨터 시스템의 백그라운에서 실행되는 클라이언트와 서버 프로세스, 그리고 폰, 테블릿과 같은 실제 안드로이드 장치나 AVD에서 실행되는 데몬 백그라운드 프로세스로 구성된다.

8. 안드로이드 스튜디오 코드 편집기

- 안드로이드 스튜디오 코드 편집기는 코드 작성에 필요한 입력 부담을 줄여주고 코드를 읽고 이동하면서 작업하기 쉽게 해준다. 코드 자동완성, 코드 자동생성, 편집기 창 나누기, 코드 접어 감추기, 코드 형식 변환 빠른 문서 검색 등이 있다.

9. 안드로이드 아키텍쳐 개요

- 안드로이드 소프트웨어 스택 아키텍처의 형태로 구현 되어있다. 리눅스 커널, 런타임 환경과 그에 부합되는 라이브러리, 애플리케이션 프레임워크, 애플리케이션등의 스택으로 구성된다. 애플리케이션은 자바로 작성되고 빌드 환경에서 바이트 코드로 컴파일된다. 나중에 실행될 때, 런타임 바이트 코드를 장치의 프로세서가 필요하는 네이티브 명령어로 일괄 변경한다.

10. 액티비티와 인텐트 개요

- 액티비티와 인텐트의 형태로 독립적으로 실행 가능한 모듈을 생성함으로써 재사용성과 상호운용성이 극대화된다. 콘텐츠 제공자를 구현하면 애플리케이션 간의 데이터의 공유가 가능하다. 액티비티에서는 사용자가 애플리케이션과 상호작용하는 데 초점을 두는 반면, 백그라운드 프로세싱은 서비스와 브로드 캐스트 수신자에 의해 처리된다.

* 애플리케이션을 구성하는 컴포넌트들은 매니페스트의 파일에 정의되어 안드로이드 런타임 시스템에 의해 참조된다