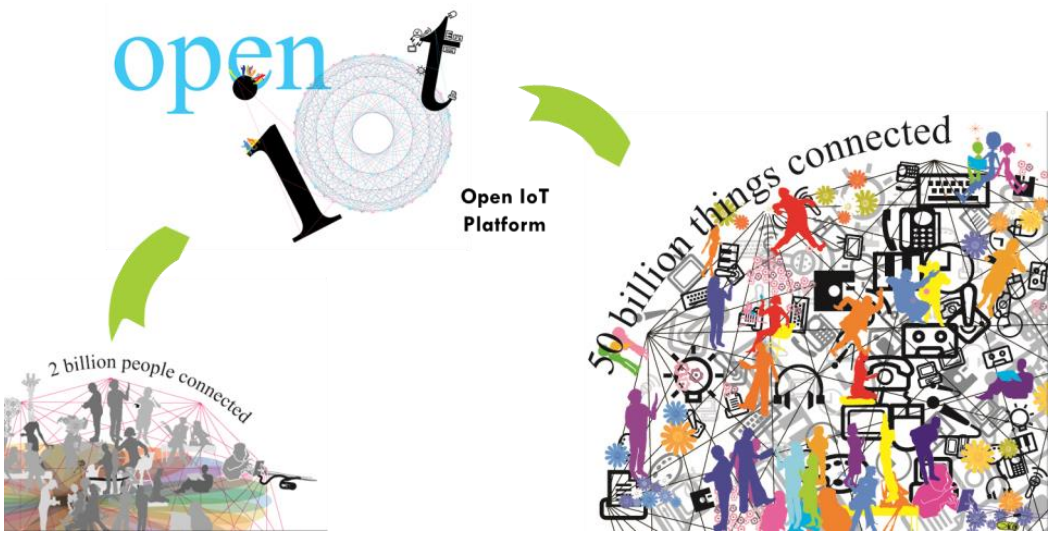
- Mobius 기반 사물인터넷 응용서비스 개발 방법 -



전 자 부 품 연 구 원

박동하 책임

(araha@keti.re.kr)

Agenda

1	Android 소개
2	Android Studio 개발 환경 구성
3	Web View App 실습
4	개방형 IoT Platform 연동 App Review
5	App Code Review
6	Generate APK

Android 4대 컴포넌트

1. Activity
   - ✓ 모든 화면이 있는 어플리케이션이라면 Activity가 존재하며 Activity가 존재해야만 화면 구성을 할 수 있음.
   - ✓ 화면 UI를 구성하는 View를 담을 수 있는 빈 테이블이라고 보면 됨.

2. Service
   - ✓ 서비스는 백그라운드 서비스를 의미. (예: 카카오톡 알림)

3. Broadcast Receiver
   - ✓ Android OS 자체와 설치된 App에서 수많은 이벤트가 발생됨.

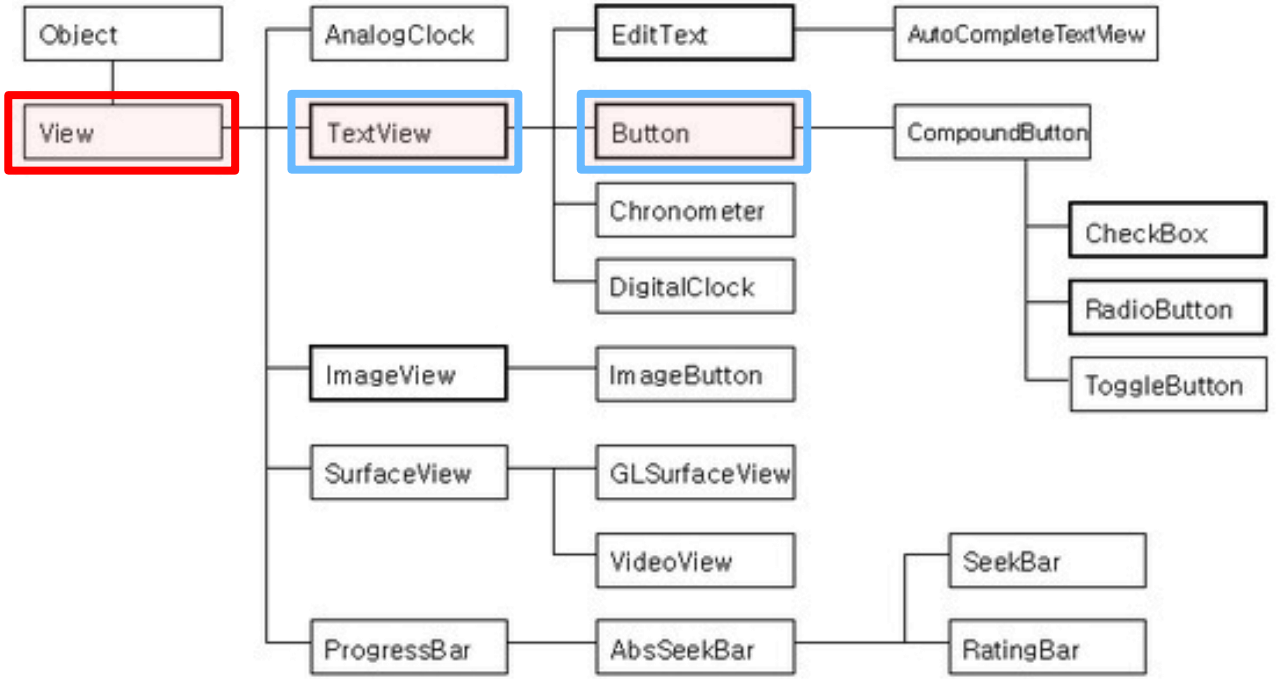
     (Wifi 켜짐, 꺼짐 등 여러 이벤트를 받을 수 있는 역할 수행)

4. Content Provider
   - ✓ Android 데이터 공유 인터페이스.
   - ✓ Android 시스템의 각종 설정 값, SD카드 내의 미디어 등에 접근 및 타 어플 접근.

     (단 타 어플 접근시 해당 App에서 정의한 수준만 가능)

1 Android 소개

View



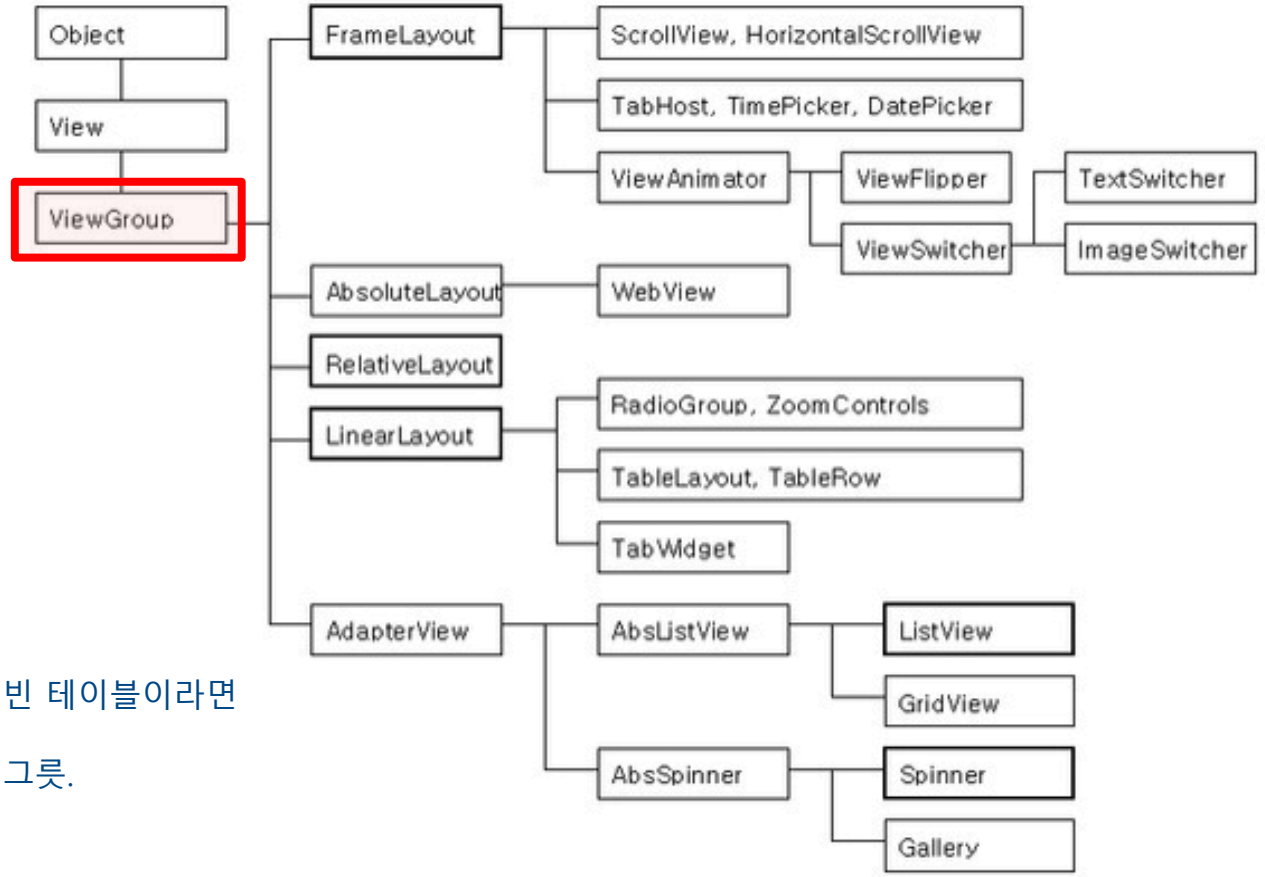
- ✓ Activity만으로 화면을 보여줄 수 없음
- ✓ View가 실질적인 화면임.
- ✓ Java에는 최상위에는 Object Class가 있고 하위 View들로 파생됨.
- ✓ 안드로이드는 계층 구조로 구성되어 있음.

(예: Button은 Text가 표시될 수 있으므로 TextView를 상속받고 있다.

즉 Button의 Text관련 기능은 TextView를 상속받아서 구현한다고 보면 됨)

1 Android 소개

ViewGroup



- ✓ Activity가 이 모든 것들을 얻을 수 있는 빈 테이블이라면

  ViewGroup 은 일부분의 View들을 담는 그릇.
- ✓ ViewGroup을 Layout이라고 부름.
- ✓ ViewGroup은 또 다른 ViewGroup을 담을 수 있음.
- ✓ ViewGroup안에 여러 ViewGroup을 이중, 삼중으로 ViewGroup을 포함할 수 있음.

Linear Layout(리니어 레이아웃)

- ✓ Linear Layout은 Child View를 일렬로 배치하는 Layout.
- ✓ 안에 배치되는 View는 Child View가 됨.
- ✓ Layout중에 Relative Layout과 함께 가장 많이 사용하는 레이아웃.
- ✓ Linear Layout의 핵심 속성은 orientation.

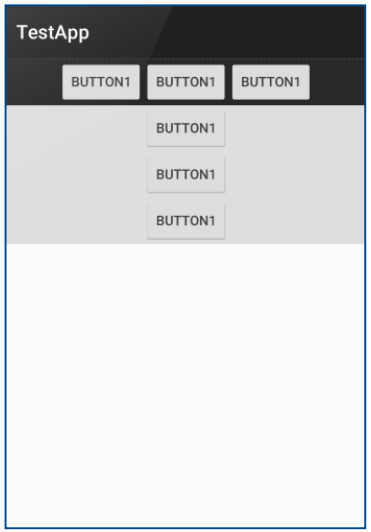
속성명	값	
layout_width, layout_height	fill_parent, match_parent	fill_parent**와** match_parent**는 현재 뷰의 크기를 부모의 크기와 맞춘다는 뜻**. fill_parent**는 안드로이드에서 권장하지 않음**. **따라서** match_parent**를 사용하면 됨**. wrap_content**는 내부 요소에 따라서 크기를 조절함**.
	wrap_content	
orientation	vertical	Child View**를 위에서 아래로 배치**
	horizontal	Child View**를 왼쪽에서 오른쪽으로 배치**
gravity	center	Child View**를 정 중앙에 배치**
	center_horizontal	Child View**를 수평 중앙배치**
	center_vertical	Child View**를 수직 중앙배치**
layout_weight	number	**화면을 비율로 나눔**

Linear Layout(리니어 레이아웃)

최상위 xmlns:android="http://schemas.android.com/apk/res/android"는

해당 View Group에서 android: 와 같은 android 속성을 사용하겠다고 정의한 스키마

1. LinearLayout으로 전체 선언.
2. android:orientation="vertical" 로 하위 Child View를 수직배치.
3. LinearLayout에 LinearLayout 두개를 수직배치.
4. LinearLayout 의 Child View를 수평배치.
5. 두번째 LinearLayout 의 Child View 수직배치.

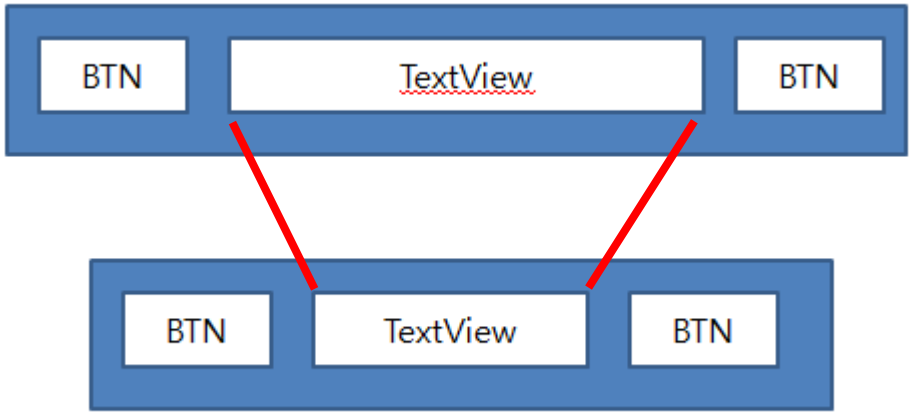



```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:orientation="vertical"
  >
  <LinearLayout
     android:layout_width="match_parent"
     android:layout_height="wrap_content"
     android:orientation="horizontal"
     android:background="#000000"
     android:gravity="center"
     >
     <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button1"
        />

     <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button1"
        />
     <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button1"
        />
  </LinearLayout>
  <LinearLayout
     android:layout_width="match_parent"
     android:layout_height="wrap_content"
     android:orientation="vertical"
     android:background="#d8d8d8"
     android:gravity="center"
     >
     <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button1"
        />

     <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button1"
        />
     <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button1"
        />
  </LinearLayout>
```

Relative Layout(렐러티브 레이아웃)

- ✓ LinearLayout 과 더불어 가장 많이 사용하는 Layout.
- ✓ LinearLayout 으로 구현할 수도 있지만 **Relative Layout을** 이용하면 더욱 쉽게 구현 가능함.
- ✓ 아래와 같이 양쪽에 버튼이 있고 가운데 TextView가 있는 형태에서 양쪽의 버튼은 고정사이즈이면 가운데 Text View는 유동적으로 너비를 줄 경우 필요함.

  (이유? : 다양한 사이즈의 Android 환경이 있기에 이런 식으로 Layout을 배치해야 깔끔한 UI를 볼 수 있기 때문임)



Relative Layout(렐러티브 레이아웃)

[Relative Layout에서 사전 고려 사항]

보통 Layout은 코드상 먼저 작성한 View가 먼저 배치된다. 하지만 **Relative Layout**은 상대적인 배치를 하기 때문에 순서가 중요함.

View의 입장에서 검은색 배경이 부모 레이아웃인 **Relative Layout 임**.

1. 버튼 1을 부모의 왼쪽에 배치

2. 버튼 2를 부모의 오른쪽에 배치

3. Text View를 버튼1의 왼쪽과 버튼2의 오른쪽 부모의 중앙에 배치

이런 순서로 선언해야 하는 이유는?

xml은 순서대로 코드를 파싱되기 때문에

2번과 3번의 순서를 변경하면 Text View를 정렬할 때 버튼2를 찾을 수 없다고 오류가 발생함.

Relative Layout(렐러티브 레이아웃)

버튼 두개에 각각 아이디를 주고 각각 부모의 왼쪽과 오른쪽에 배치.

각 속성은 layout_alignParentLeft와 layout_alignParentRight.

Text View는 버튼1의 오른쪽 버튼2의 왼쪽에 배치.

(layout_toRightOf, layout_toLeftOf)

Text View 경우 width를 match_parent로 하여 최대로 주면

버튼1의 오른쪽과 버튼2의 왼쪽에 꽉 차는 레이아웃이 완성됨.

반대의 방법도 가능함.

바로 위에 버튼 아래에 버튼 그리고 남는 공간은 TextView로 채우는 방법

```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent“
    android:layout_height="wrap_content"
    android:background="#000000">

<Button android:id="@+id/btn1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:text="BUTTON1" />

<Button android:id="@+id/btn2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:text="BUTTON2" />

<TextView android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_toRightOf="@id/btn1"
    android:layout_toLeftOf="@id/btn2"
    android:layout_centerVertical="true"
    android:text="TextView" android:background="#ffffff" />

</RelativeLayout>
```

Relative Layout(렐러티브 레이아웃)

layout_alignParentTop, layout_alignParentBottom을 이용해

버튼을 위아래로 배치.

TextView는 버튼1의 아래에 배치 (layout_below) 됨과 동시에

버튼2의 위쪽에 배치(above)됨.

RelativeLayout과 LinearLayout

두 가지를 사용하시면 웬만한 구성은 가능.

```
<RelativeLayout
        xmlns:android=http://schemas.android.com/apk/res/android
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#000000" >

<Button android:id="@+id/btn1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:text="BUTTON1" />

<Button android:id="@+id/btn2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:text="BUTTON2" />

<TextView android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/btn1"
        android:layout_above="@id/btn2"
        android:layout_centerVertical="true"
        android:text="TextView" android:background="#ffffff" />

</RelativeLayout>
```

Frame Layout(프레임 레이아웃)

Callback Method?

✓ 정의된 특정 상황 및 시점에 자동으로 호출되는 함수.

(전화번호를 알려주고 끊으면 상대편에서 전화를 하는 것과 같다고 하여 붙은 명칭)

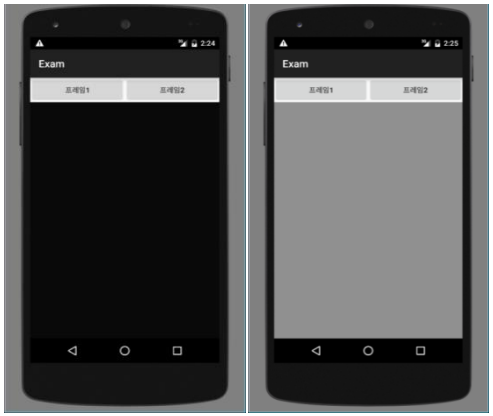
✓ Android 개발 중 Callback은 onClick Event 발생시 가장 많이 사용됨.

※ Listener와 Callback Method의 차이

✓ 대상 뷰에 해당하는 V(View)를 가지는가 아닌가의 차이.

✓ Callback은 **특정 View 클래스 소속이므로 이벤트 발생 대상이 정해져 있음**.

✓ Listener는 여러 위젯에 의해 공유될 수 있으므로 **대상 View가 누구인지 전달 받아야 함.**



FrameLayout

✓ FrameLayout 은 여러 개의 View를 겹칠 수 있는 Layout임.

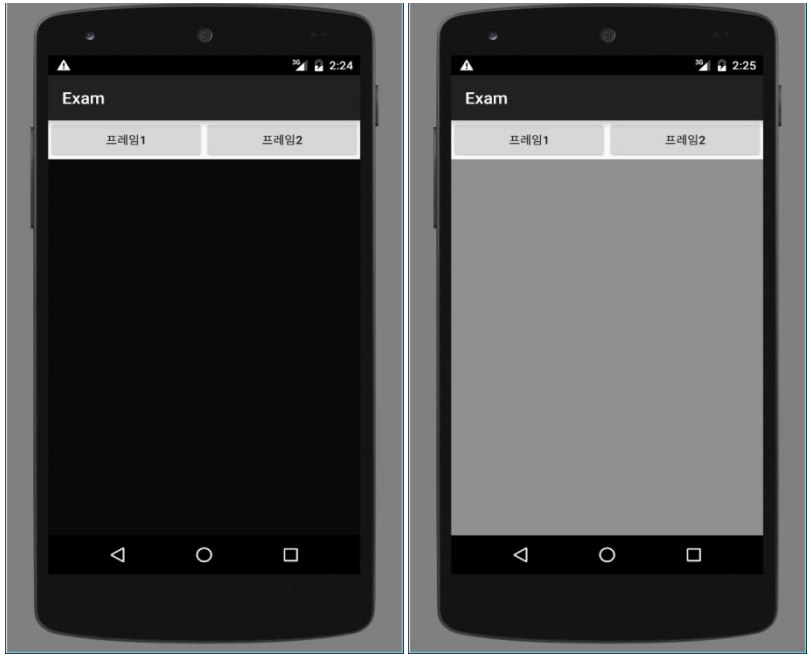
예) 버튼을 두 개 배치한 후 버튼 클릭 시 View들을 교체하는 레이아웃.

1 Android 소개

Frame Layout(프레임 레이아웃)

버튼 클릭 시 View들을 교체하는 예.

✓ FrameLayout과 OnClickListener를 이용.




```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:orientation="vertical"
  >

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <Button
      android:id="@+id/btn1"
      android:layout_width="0dp"
      android:layout_height="wrap_content"
      android:layout_weight="1"
      android:text="프레임1"
      />
    <Button
      android:id="@+id/btn2"
      android:layout_width="0dp"
      android:layout_height="wrap_content"
      android:layout_weight="1"
      android:text="프레임2"
      />

  </LinearLayout>

  <FrameLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <View
      android:id="@+id/view1"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:background="#ffffff"
      />
    <View
      android:id="@+id/view2"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:background="#9c9c9c"
      android:visibility="gone"
      />
  </FrameLayout>
</LinearLayout>
```

Frame Layout(프레임 레이아웃)

[앞 예제 설명]

상단 버튼 너비의 비율을 동일하게 주기 위해 weight를 이용

버튼의 비율을 너비의 50%씩 주기 위해서 layout_weight = 1

즉 weight는 비중이란 뜻으로 1:1의 비율로 나눠가지겠다는 의미임.

weight를 줄 경우 width나 height는 0dp로 지정해 줘야 함.

weight의 비중의 방향은 LinearLayout의 orientation의 영향을 받음.

View의 기본적인 visibility 속성

- ✓ **Visible:** 기본속성으로 보임.
- ✓ **Invisible:** 자리는 차지하고 화면에서 보이지 않음.
- ✓ **gone:** 보이지도 영역도 차지 않고 사라지게 됨.

FrameLayout은 가장 나중에 선언한 view가 위로 오게 됨.

(Red, Blue, **Green**)

Frame Layout(프레임 레이아웃)

onClick Event

✓ View.OnClickListener interface를 implements하게 되면 반드시 onClick를 구현해야 함.

onClick을 구현하기 위한 2가지 과정.

1. Method 구현 (public void onClick(View v)...)

2. Listener 등록 (btnX.setOnClickListener(this);)

Listener를 통해서 onClick Method를 호출함.

클래스 자체에서 인터페이스를 구현했기 때문에

btn1.setOnClickListener(this)로 Listener 등록이 가능.

onClick Method에서 클릭을 했을 때

들어오는 ID에 따라서 view의 visibility를 바꿔주는 코드.

위의 코드와 XML의 조합으로 버튼 클릭 시 View가 바뀌게 됨.

```
package test.com.exam;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

  private Button btn1;
  private Button btn2;

  private View view1;
  private View view2;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btn1 = (Button)findViewById(R.id.btn1);
    btn2 = (Button)findViewById(R.id.btn2);

    view1 = findViewById(R.id.view1);
    view2 = findViewById(R.id.view2);

    btn1.setOnClickListener(this);
    btn2.setOnClickListener(this);
  }

  @Override
  public void onClick(View v) {

    switch (v.getId()){
      case R.id.btn1 :
        view1.setVisibility(View.VISIBLE);
        view2.setVisibility(View.GONE);
        break;
      case R.id.btn2 :
        view1.setVisibility(View.GONE);
        view2.setVisibility(View.VISIBLE);
        break;
    }
  }
}
```

* Android OnClick

OnClickMethod는 OnClickListener interface에 정의되어 있는 Method.

Listener 안에 Method를 구현해 놓으면 특정 이벤트를 계속 듣고 있다가 이벤트가 발생하는 시점에 Method를 호출해 줌

View의 OnClickListener에 있는 OnClickMethod를 구현하는 방법 4가지

1. OnClickListener interface를 implements 하여 Method 구현
2. View의 인자에 바로 Listener을 구현
3. clickListener을 별도의 객체로 생성하여 set하는 방법
4. xml에서 Method를 선언하고 구현하는 방법

위 4가지 방법으로 Android OnClick Method를 구현할 수 있으나 4번은 거의 사용하지 않음

1 Android 소개

* Android OnClick

1) OnClickListener interface를 implements하여 Method 구현

구현한 Listener를 등록.

✓ 주로 Event Lister를 많이 작성해야 하는 상황에서 사용됨.

```
package test.com.exam;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

  private Button btn1;
  private Button btn2;

  private View view1;
  private View view2;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btn1 = (Button)findViewById(R.id.btn1);
    btn2 = (Button)findViewById(R.id.btn2);

    view1 = findViewById(R.id.view1);
    view2 = findViewById(R.id.view2);

    btn1.setOnClickListener(this);
    btn2.setOnClickListener(this);
  }

  @Override
  public void onClick(View v) {

    switch (v.getId()){
      case R.id.btn1 :
        view1.setVisibility(View.VISIBLE);
        view2.setVisibility(View.GONE);
        break;
      case R.id.btn2 :
        view1.setVisibility(View.GONE);
        view2.setVisibility(View.VISIBLE);
        break;
    }
  }
}
```

1 Android 소개

* Android OnClick

2) View의 인자에 바로 Listener구현

1번의 방법과 같이 Android 개방방법에서 가장 많이 사용되는 방식.

- ✓ 구현 이벤트의 수가 적을 경우
- ✓ Listener를 재사용하지 않을 경우

```
package test.com.exam;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity{

  private Button btn1;
  private Button btn2;

  private View view1;
  private View view2;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btn1 = (Button)findViewById(R.id.btn1);
    btn2 = (Button)findViewById(R.id.btn2);

    view1 = findViewById(R.id.view1);
    view2 = findViewById(R.id.view2);

    btn1.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        view1.setVisibility(View.VISIBLE);
        view2.setVisibility(View.GONE);

      }
    });
    btn2.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        view1.setVisibility(View.GONE);
        view2.setVisibility(View.VISIBLE);
      }
    });
  }
}
```

1 Android 소개

* Android OnClick

3) Listener를 객체로 선언해두고 set하는 방법

Listener를 재 사용해야 되는 상황에서 많이 사용하는 기법.

✓ 단 Android에서 Listener를 재 사용하는 경우는 많이 없음.

```
package test.com.exam;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity{

  private Button btn1;
  private Button btn2;

  private View view1;
  private View view2;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btn1 = (Button)findViewById(R.id.btn1);
    btn2 = (Button)findViewById(R.id.btn2);

    view1 = findViewById(R.id.view1);
    view2 = findViewById(R.id.view2);

    btn1.setOnClickListener(btn1Listener);
    btn2.setOnClickListener(btn2Listener);
  }

  View.OnClickListener btn1Listener = new View.OnClickListener() {
    @Override
    public void onClick(View v) {
      view1.setVisibility(View.VISIBLE);
      view2.setVisibility(View.GONE);
    }
  };

  View.OnClickListener btn2Listener = new View.OnClickListener() {
    @Override
    public void onClick(View v) {
      view1.setVisibility(View.GONE);
      view2.setVisibility(View.VISIBLE);
    }
  };
```

1 Android 소개

* Android OnClick

4) xml에서 Method를 선언하고 구현하는 방법

xml에 onClick:"onClick1", onClick:"onClick2" 이라는 코드가 들어감.

여기에 들어가는 onClick1이 바로 java 소스에서 작성할 Method 명임.

```
<Button
    android:id="@+id/btn1"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:onClick="onClick1"
    android:text="프레임1"
    />
<Button
    android:id="@+id/btn2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:onClick="onClick2"
    android:text="프레임2"

    />
```

- ✓ 버튼을 객체로 생성하여 **Listerner를 등록하는 과정 없이** 바로 Method가 호출됨.
- ✓ 코드가 간결.
- ✓ 이 구현방법은 기능을 작성하는 Code와

  View를 담당하는 Layout간의 의존성이 강해지기 때문에 잘 사용하지 않음

```
package test.com.exam;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity{

  private View view1;
  private View view2;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    view1 = findViewById(R.id.view1);
    view2 = findViewById(R.id.view2);

  }

  public void onClick1(View v) {
    view1.setVisibility(View.VISIBLE);
    view2.setVisibility(View.GONE);

  }

  public void onClick2(View v) {
    view1.setVisibility(View.GONE);
    view2.setVisibility(View.VISIBLE);

  }
}
```

Grid Layout(그리드 레이아웃)

- ✓ GridLayout의 Grid는 격자 눈금 등을 뜻함. (바둑판 형식의 레이아웃)
- ✓ GridLayout은 TableLayout과 비슷하나 API14레벨부터 추가된 Layout.
- ✓ LinearLayout, RelativeLayout, TableLayout의 단점을 보완하여 만든 Layout.
- ✓ 메모리 효율이 좋음.
- ✓ 단 계산기 같이 표로 구성되어야 하는 경우 외에는 잘 사용하지 않음.

Grid Layout(그리드 레이아웃)

GridLayout 은 간단함.

GridLayout 속성

- ✓ columnCount = 가로(행)수
- ✓ rowCount = 세로(열)수
- ✓ columnSpan = "2" 컬럼합치기
- ✓ rowSpan = "2" 로우합치기
- ✓ layout_gravity ="fill_horizontal"

```
<GridLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:columnCount="4"
   android:rowCount="4"
    >

   <Button android:text="7"/>
   <Button android:text="8"/>
   <Button android:text="9"/>
   <Button android:text="/"/>

   <Button android:text="4"/>
   <Button android:text="5"/>
   <Button android:text="6"/>
   <Button android:text="*"/>

   <Button android:text="1"/>
   <Button android:text="2"/>
   <Button android:text="3"/>
   <Button android:text="-"/>

   <Button android:text="0"
      android:layout_columnSpan="2"
      android:layout_gravity="fill_horizontal"
      />
   <Button android:text="."/>
   <Button android:text="+"/>


</GridLayout>
```

가로가 4개인데 columnSpan="2"의 속성을 가진 View를 지정하면

가로 2개의 영역을 할당 받고, 나머지 공간은 2개만 남음.

코드를 보면 다른 버튼은 4개씩 배치되어 있지만 columnSpan="2"인 것은 3개의 라인 코드만 있음.

1 Android 소개

Table Layout(테이블 레이아웃)

GridLayout은 TableLayout의 단점을 보완해서 만든 Layout임.

TableRow라는 것을 통해서 한 줄 한 줄 구현 해야 함.

stretchColumns 옵션으로 특정 컬럼을 자동으로 늘려주는 역할을 수행.


```
<TableLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:stretchColumns="*"
  >

  <TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:padding="1dp"
    >
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginRight="1dp"
      android:text="number"
      android:background="#777777"
      android:textSize="12sp"
      android:gravity="center_horizontal"
      />
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginRight="1dp"
      android:text="이름"
      android:background="#777777"
      android:textSize="12sp"
      android:gravity="center_horizontal"
      />
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="나이"
      android:background="#777777"
      android:textSize="12sp"
      android:gravity="center_horizontal"
      />
  </TableRow>
```

```
  <TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:padding="1dp"
    >
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginRight="1dp"
      android:text="1"
      android:background="#ffffff"
      android:textSize="12sp"
      android:gravity="center_horizontal"
      />
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginRight="1dp"
      android:text="number"
      android:background="#ffffff"
      android:textSize="12sp"
      android:gravity="center_horizontal"
      />
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="number"
      android:background="#ffffff"
      android:textSize="12sp"
      android:gravity="center_horizontal"
      />
  </TableRow>

</TableLayout>
```

Table를 사용할 일이 있으면 GridLayout를 사용하는 것이 더욱 간단하게 구현 가능함.

Absolute Layout(앱솔루트 레이아웃)

초기 기기가 적었던 시절은 이 Layout을 사용했지만

다양한 기기들이 나오면서 절대적인 좌표로 구성되는 **Absolute Layout은 사용하지 않음.**

Absolute Layout은 deprecated 됨(더 이상 중요도가 떨어져 사용하지 않을 것이며 앞으로 사라지게 될 거라는 뜻)

버튼을 생성하여 특정 좌표로 위치를 지정하는 방식.


```
<AbsoluteLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="match_parent"
  android:layout_height="match_parent">

  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button"
    android:layout_x="100dp"
    android:layout_y="100dp"
    />

</AbsoluteLayout>
```

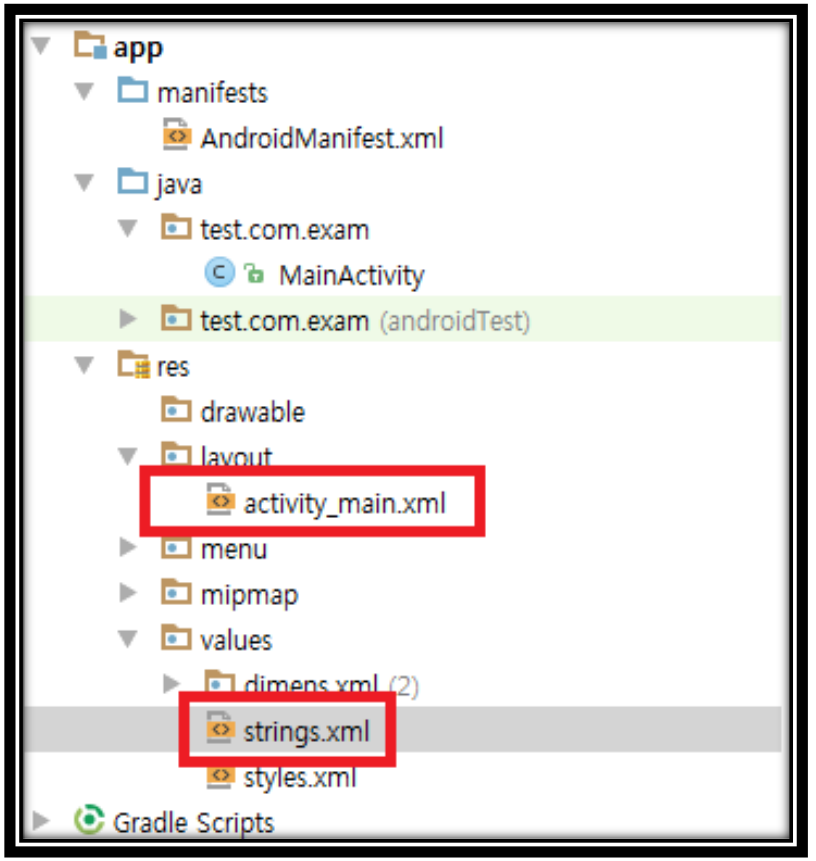
Android TextView and string.xml

View중에서 가장 기본이 되는 View인 TextView

TextView는 Button과 EditText의 부모 클래스로

Android에서 Text를 표현해주는 기본 View임.

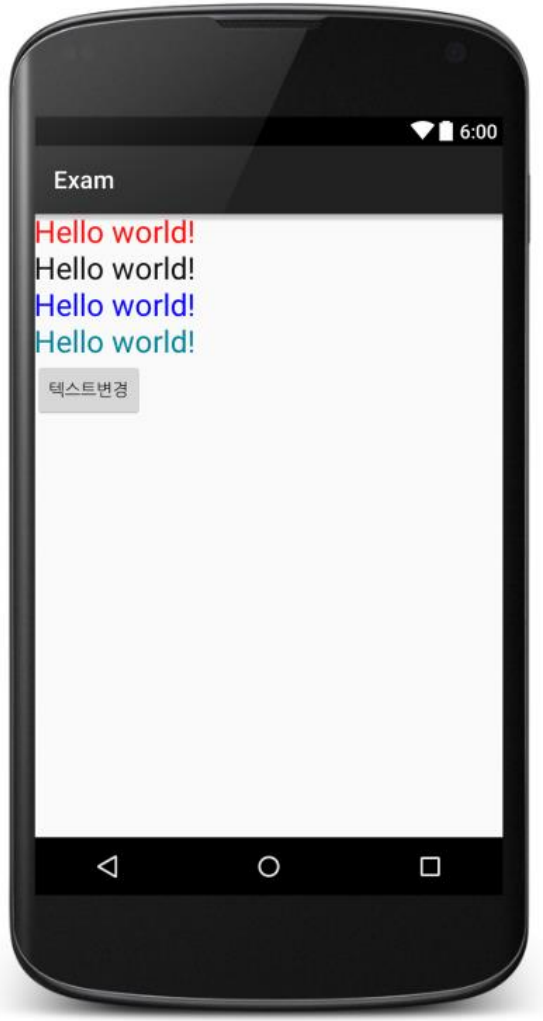
TextView의 Text는 보통 res/strings.xml에 선언하여 사용함.



Android TextView and strings.xml

일반적으로 개발을 할 때는 View영역과 Business Logic영역으로 구분하여 개발함.

- ✓ View 영역은 큰 틀인 Layout와 Layout에 사용될 Resources 로 세분화됨.
- ✓ strings.xml은 문자열 resource를 나열해 놓은 xml임.



문자열을 resource를 따로 처리하는 이유 ?

- ✓ 관리의 용이성.
- ✓ 재 사용성.
- ✓ 언어별 문자열 표기 기능

부가 설명) xml에 문자열이 고정되어 있다면

언어가 다른 두 개의 국가를 지원해야 할 경우 layout을 별도로 생성해 줘야 하지만

문자열을 strings.xml로 관리를 하면 해당 언어별 strings.xml만 하나 더 만들면 됨.

Android TextView and strings.xml

TextView를 만들고 버튼 클릭 시 해당 TextView의 텍스트 속성 변경.

[사용법]

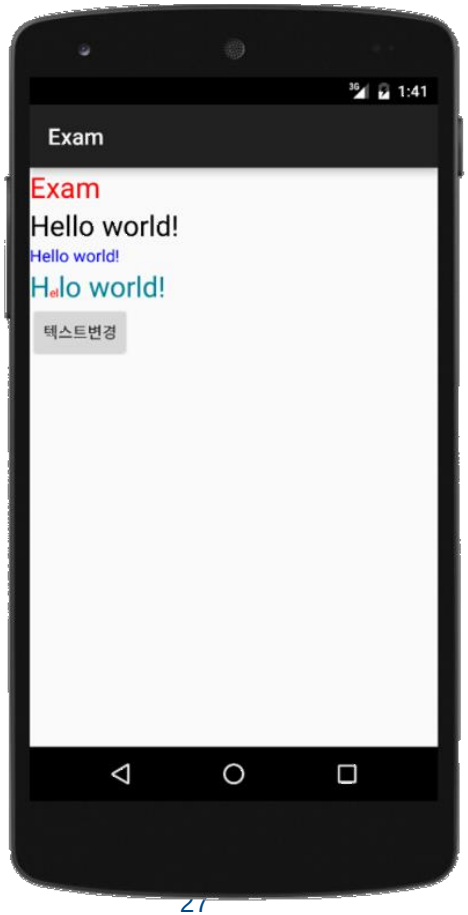
strings.xml에 있는 resource를 사용하고자 하면

text부분을 @string/xxxx 로 표현.

dimens.xml는 textSize

color.xml는 textColor

```
<resources>
    <string name="app_name">Exam</string>
    <string name="hello_world">Hello world!</string>
    <string name="action_settings">Settings</string>
    <string name="text_change_button">텍스트변경</string>
</resources>
```




```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/textViewText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world"
        android:textSize="25sp"
        android:textColor="#FF0000"
        />

    <TextView
        android:id="@+id/textViewText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world"
        android:textSize="25sp"
        android:textColor="#090909"
        />

    <TextView
        android:id="@+id/textViewText3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world"
        android:textSize="25sp"
        android:textColor="#0100FF"
        />

    <TextView
        android:id="@+id/textViewText4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world"
        android:textSize="25sp"
        android:textColor="#008299"
        />
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/text_change_button"
        />

</LinearLayout>
```

Android TextView and strings.xml

- setText : textView의 텍스트를 변경

  resource에 접근 시 Context를 통해서 접근

- setTextColor : textView의 색상을 변경

  Color.BLACK (Android 기본적인 resource)

- setTextSize : 텍스트 크기를 변경

```
button1 = (Button) findViewById(R.id.button1);

button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        textViewText1.setText(getBaseContext().getText(R.string.app_name));
        textViewText2.setTextColor(Color.BLACK);
        textViewText3.setTextSize(TypedValue.COMPLEX_UNIT_SP, 15);


        SpannableStringBuilder builder = new SpannableStringBuilder(textViewText3.getText());
        builder.setSpan(new ForegroundColorSpan(Color.RED), 1, 3, Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
        builder.setSpan(new AbsoluteSizeSpan(30), 1, 3, Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);

        textViewText4.setText(builder);
    }
});
```

일부 문자열 속성 변경 (setSpan Method 사용) 를 통해서 Color을 변경하고, 텍스트 Size를 변경함

- SpannableStringBuilder 생성
- setSpan Method를 이용하여 문자열의 시작과 끝 인덱스를 이용해 범위를 지정하고, 속성(Color, Size)를 지정함
- textView에 setText를 해주면 텍스트뷰 4번째와 같이 특정 영역만 바뀌게 됨.

많이 사용되는 예) 검색결과에 검색어 색상만 다르게 보여줄 경우.

1	Android 소개
2	**Android Studio 개발 환경 구성**
3	Web View App 실습
4	개방형 IoT Platform 연동 App Review
5	App Code Review
6	Generate APK

2 Android Studio 개발 환경 구성

JDK 설치 및 환경 설정 (이전 방식)

1. Download

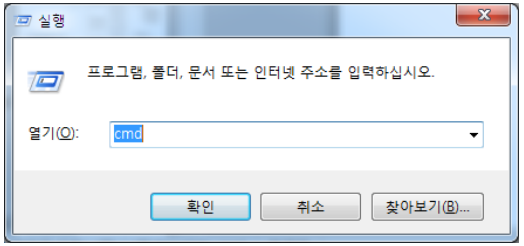
http://www.oracle.com/index.html로 접속하여 Download

2. Java 환경 구성

[내 컴퓨터]-[속성]-[고급]-[환경변수]를 클릭

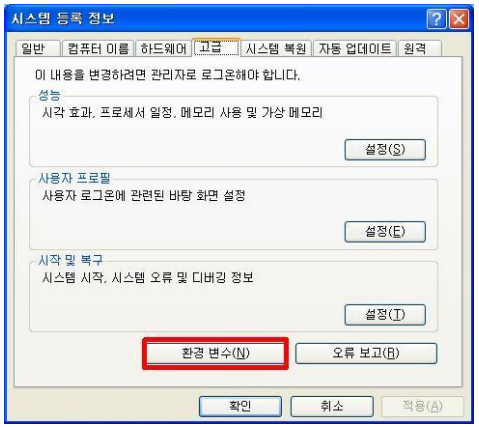
- 환경 변수: 변수 이름/변수 값 : JAVA_HOME / Java 설치 경로

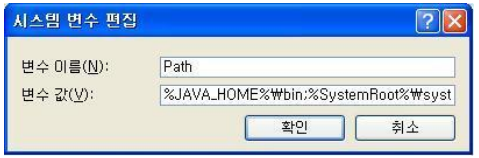
- Java 경로: 변수 이름/변수 값 : Path / Java bin 경로 추가

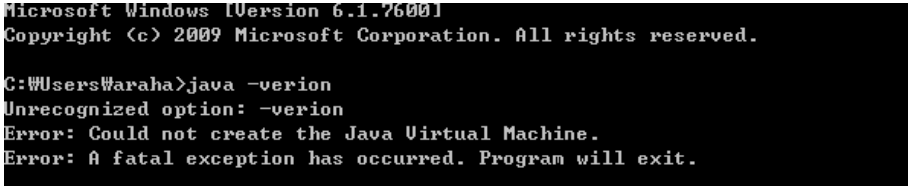


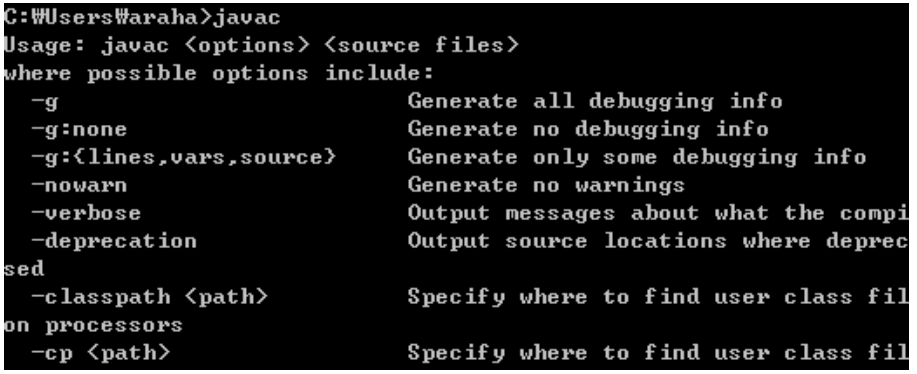
3. 실행 환경 확인

- WindowsKey+R, cmd 입력 후 확인
- Javac –version
- Javac





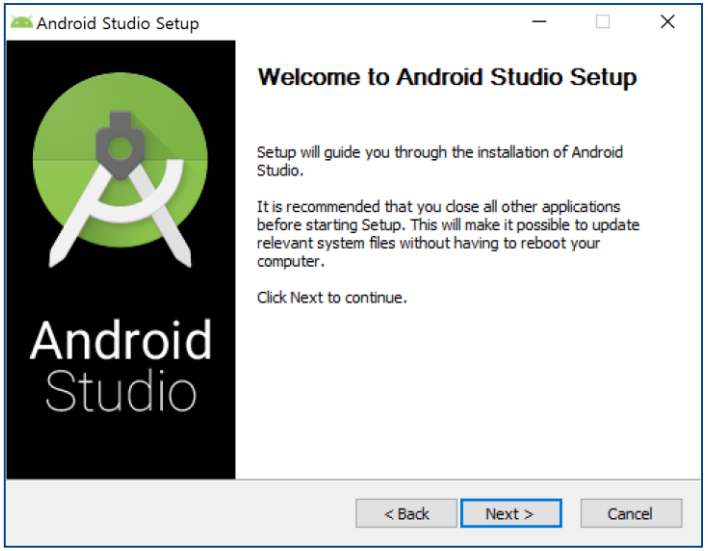


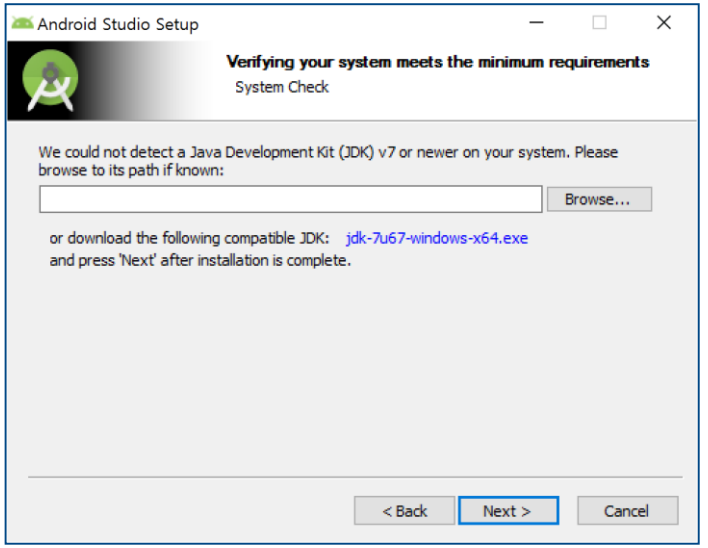


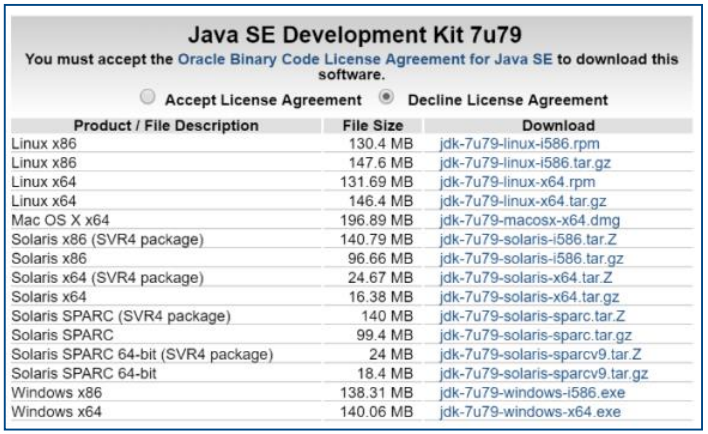
2 Android Studio 개발 환경 구성

Android Studio Install

https://developer.android.com/sdk/index.html (Android Studio Install ..., Java Development Kit 7u79 Install)





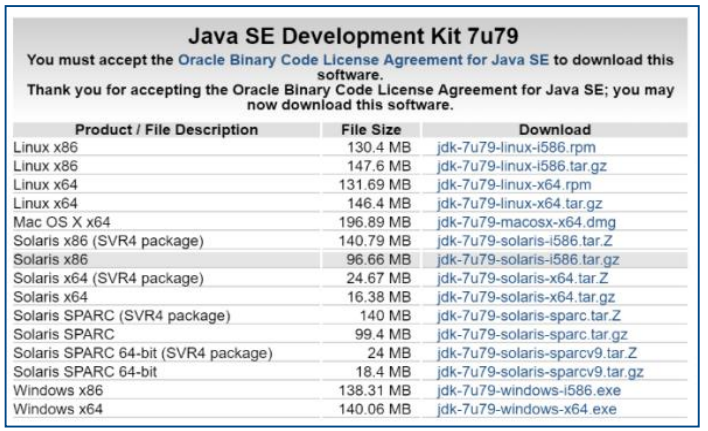


Java SE Development Kit 7u79

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	130.4 MB	jdk-7u79-linux-i586.rpm
Linux x86	147.6 MB	jdk-7u79-linux-i586.tar.gz
Linux x64	131.69 MB	jdk-7u79-linux-x64.rpm
Linux x64	146.4 MB	jdk-7u79-linux-x64.tar.gz
Mac OS X x64	196.89 MB	jdk-7u79-macosx-x64.dmg
Solaris x86 (SVR4 package)	140.79 MB	jdk-7u79-solaris-i586.tar.Z
Solaris x86	96.66 MB	jdk-7u79-solaris-i586.tar.gz
Solaris x64 (SVR4 package)	24.67 MB	jdk-7u79-solaris-x64.tar.Z
Solaris x64	16.38 MB	jdk-7u79-solaris-x64.tar.gz
Solaris SPARC (SVR4 package)	140 MB	jdk-7u79-solaris-sparc.tar.Z
Solaris SPARC	99.4 MB	jdk-7u79-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	24 MB	jdk-7u79-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	18.4 MB	jdk-7u79-solaris-sparcv9.tar.gz
Windows x86	138.31 MB	jdk-7u79-windows-i586.exe
Windows x64	140.06 MB	jdk-7u79-windows-x64.exe



Java SE Development Kit 7u79

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

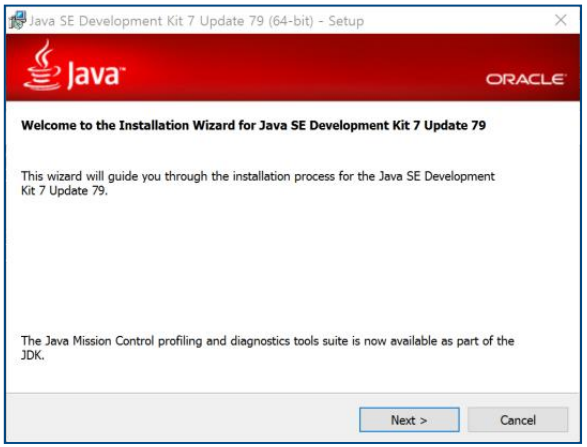
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.

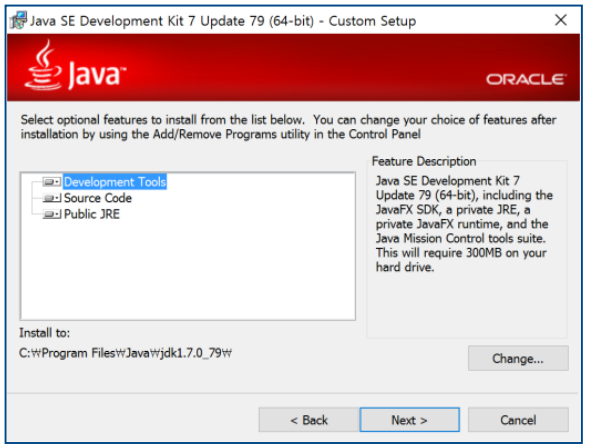
Product / File Description	File Size	Download
Linux x86	130.4 MB	jdk-7u79-linux-i586.rpm
Linux x86	147.6 MB	jdk-7u79-linux-i586.tar.gz
Linux x64	131.69 MB	jdk-7u79-linux-x64.rpm
Linux x64	146.4 MB	jdk-7u79-linux-x64.tar.gz
Mac OS X x64	196.89 MB	jdk-7u79-macosx-x64.dmg
Solaris x86 (SVR4 package)	140.79 MB	jdk-7u79-solaris-i586.tar.Z
Solaris x86	96.66 MB	jdk-7u79-solaris-i586.tar.gz
Solaris x64 (SVR4 package)	24.67 MB	jdk-7u79-solaris-x64.tar.Z
Solaris x64	16.38 MB	jdk-7u79-solaris-x64.tar.gz
Solaris SPARC (SVR4 package)	140 MB	jdk-7u79-solaris-sparc.tar.Z
Solaris SPARC	99.4 MB	jdk-7u79-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	24 MB	jdk-7u79-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	18.4 MB	jdk-7u79-solaris-sparcv9.tar.gz
Windows x86	138.31 MB	jdk-7u79-windows-i586.exe
Windows x64	140.06 MB	jdk-7u79-windows-x64.exe

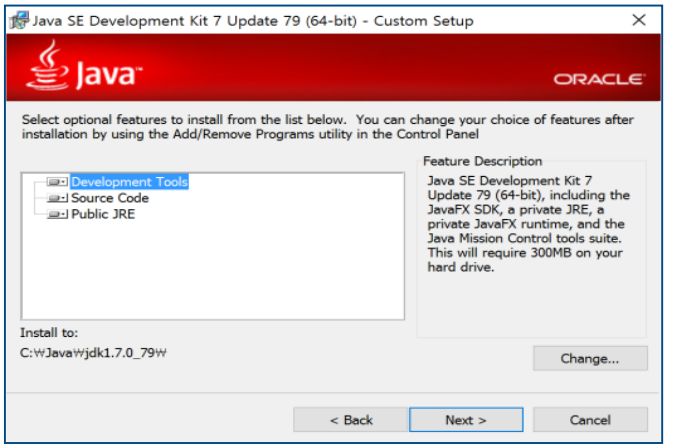
2 Android Studio 개발 환경 구성

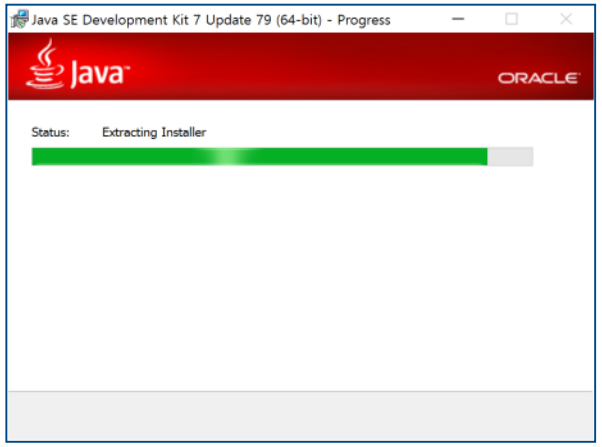
Android Studio Install

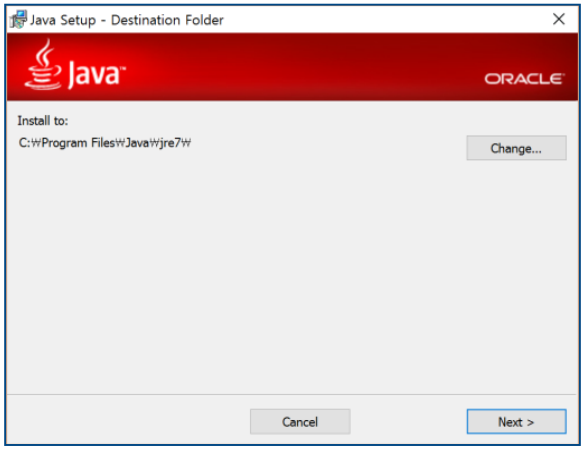
https://developer.android.com/sdk/index.html (Android Studio Install ..., Java Development Kit 7u79 Install)







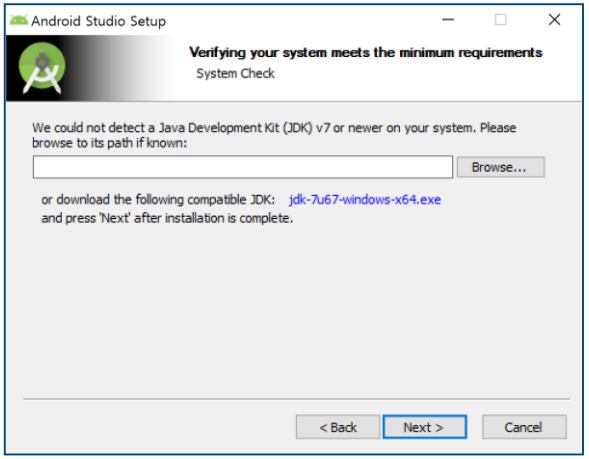


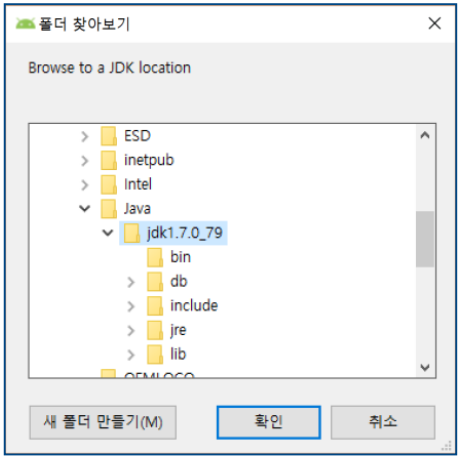


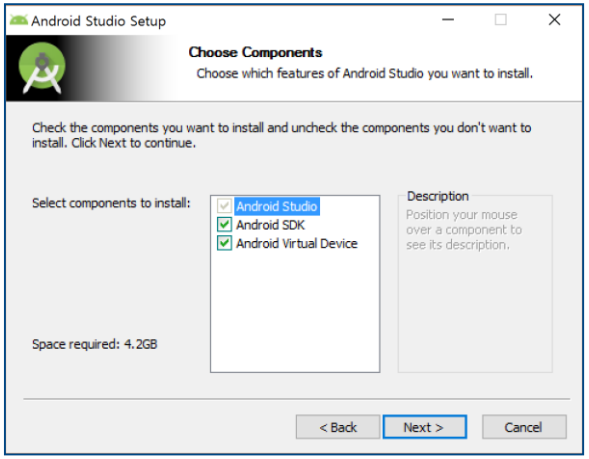
2 Android Studio 개발 환경 구성

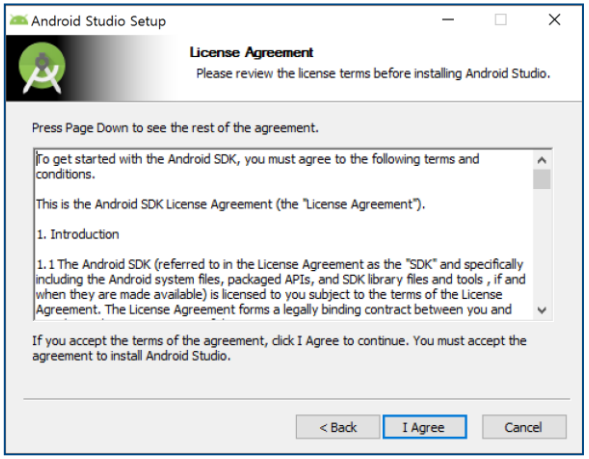
Android Studio Install

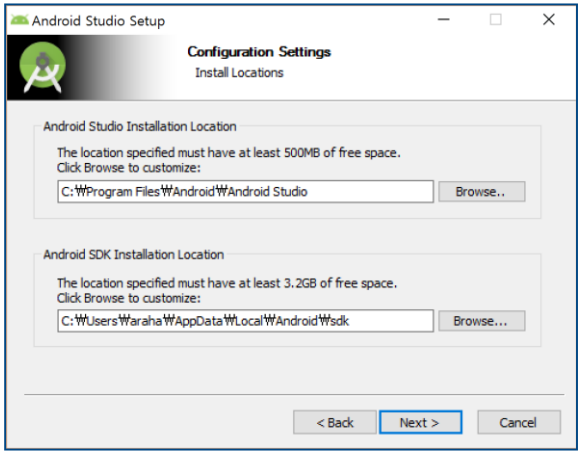
https://developer.android.com/sdk/index.html (Android Studio Install)

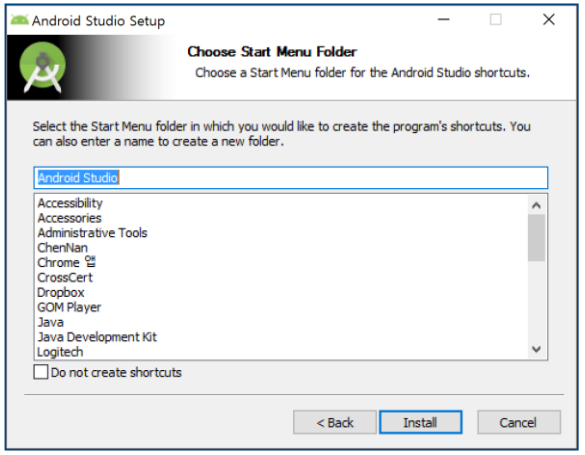








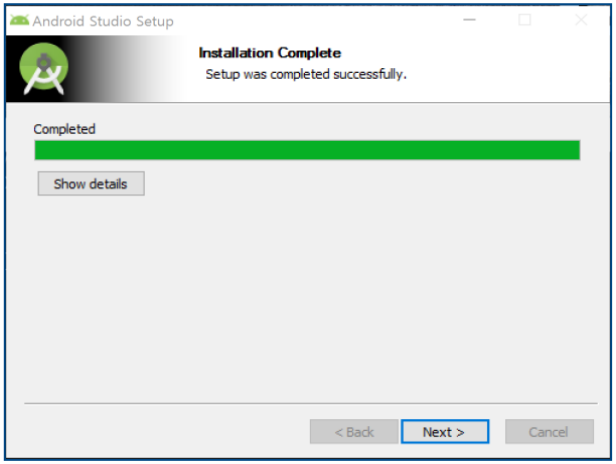


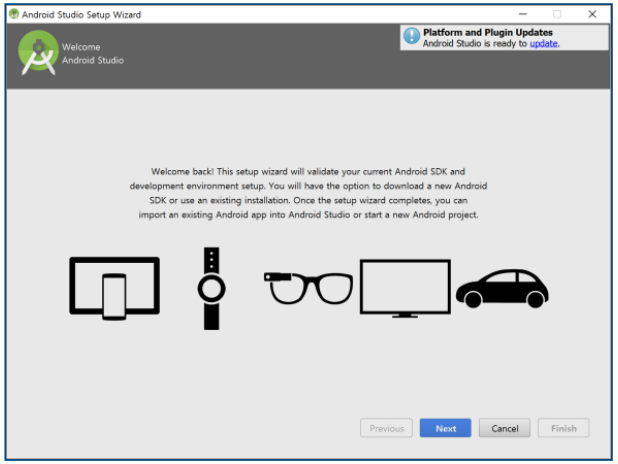


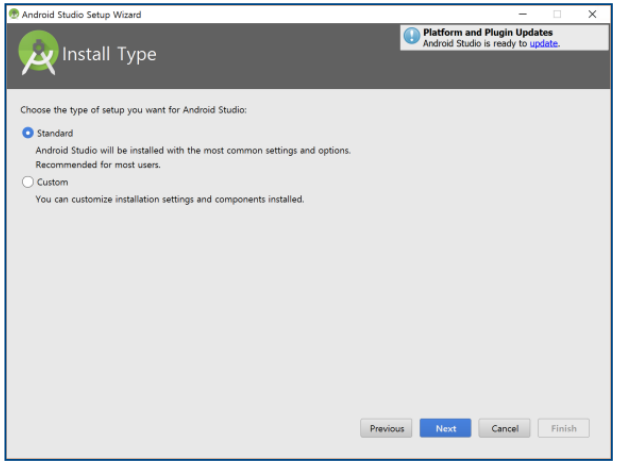
2 Android Studio 개발 환경 구성

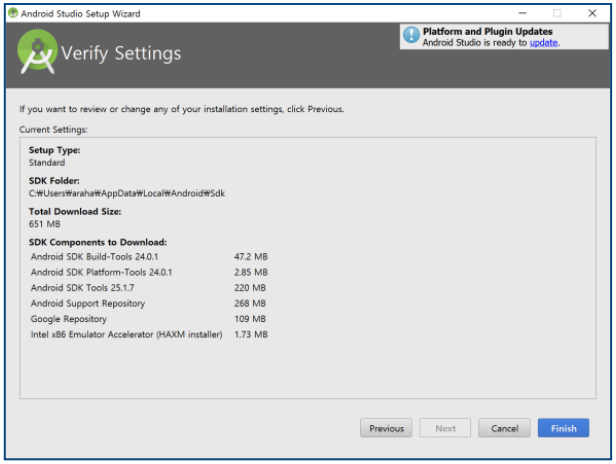
Android Studio Install

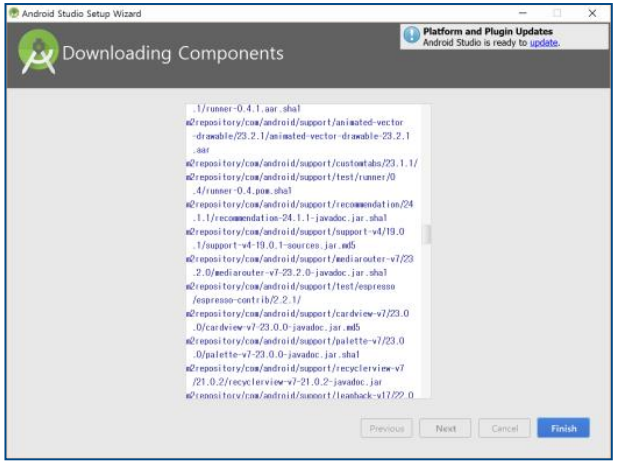
https://developer.android.com/sdk/index.html (Android Studio Install)







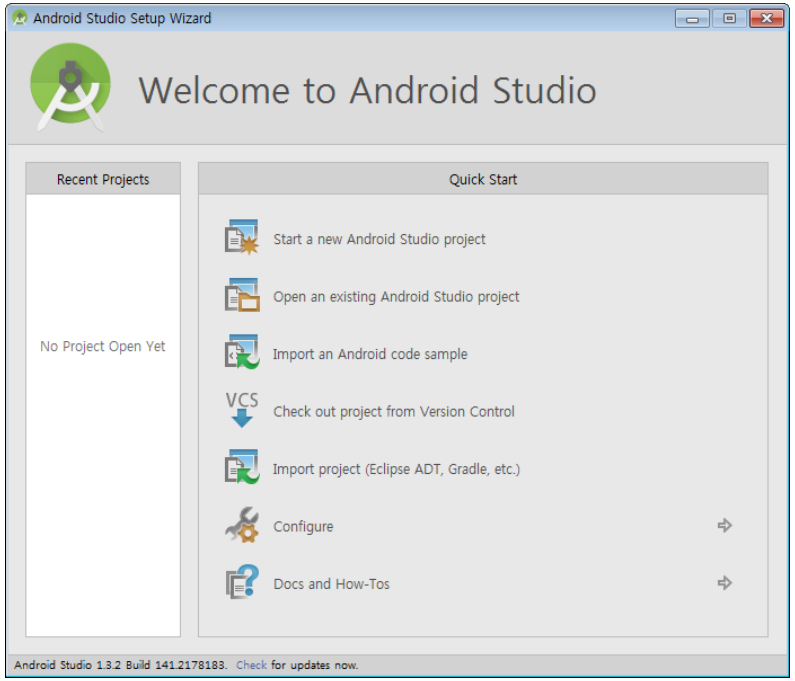


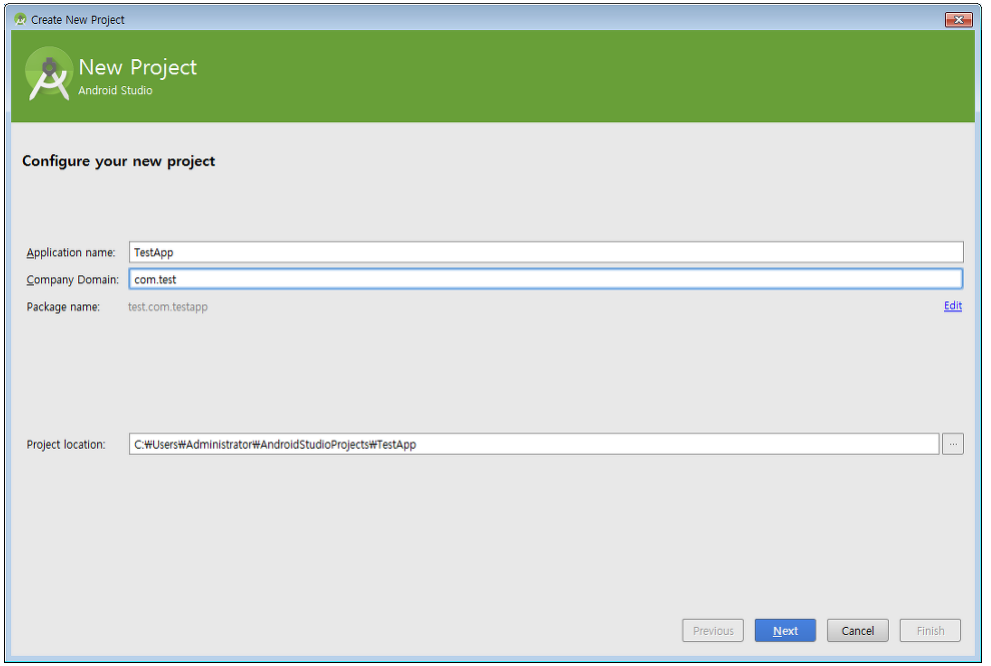


Android Studio Start

Android Studio 실행

- ✓ Start a new Android Studio project를 클릭.
- ✓ Application name과 Company Domain을 입력하면 packge name는 자동으로 생성됨.





2 Android Studio 개발 환경 구성

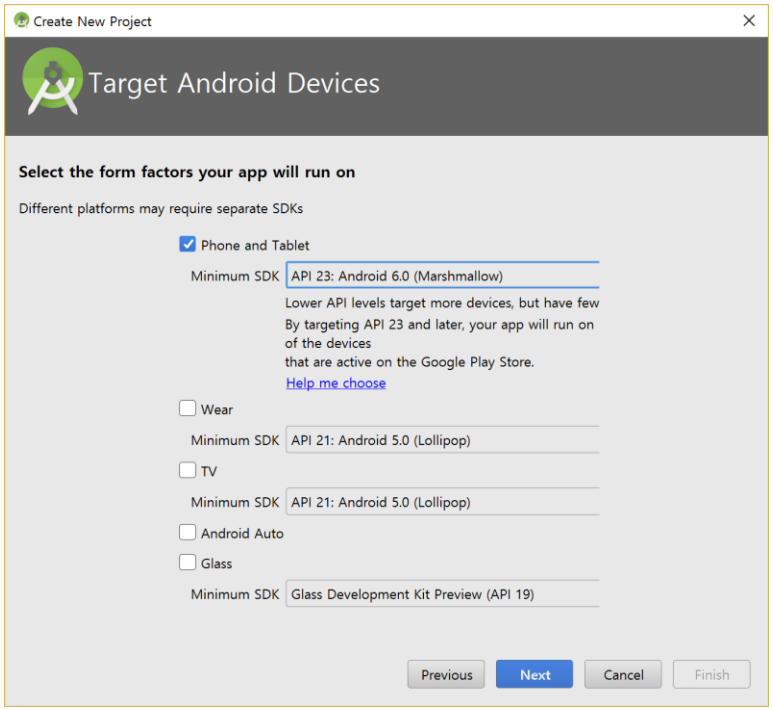
Android Studio Start

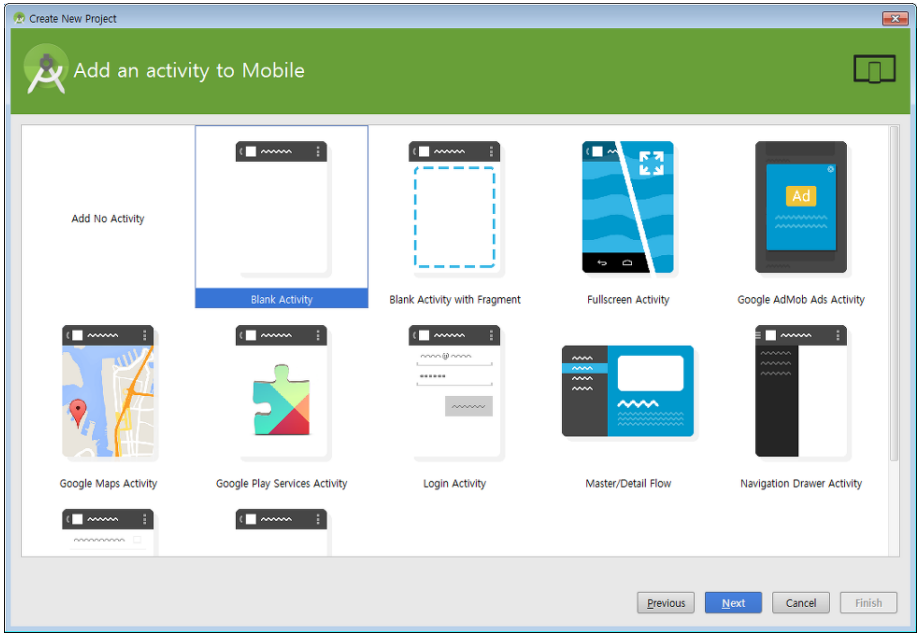
Target Android Devices

✓ Phone and Tablet를 선택하고 **Minimun SDK를 Mashmallow인 API 23로 설정**

(해당 어플리케이션은 API 23이상의 모바일만 지원하겠다는 것임)

✓ 기본 Activity는 지정. (예: Blank Activity를 선택)

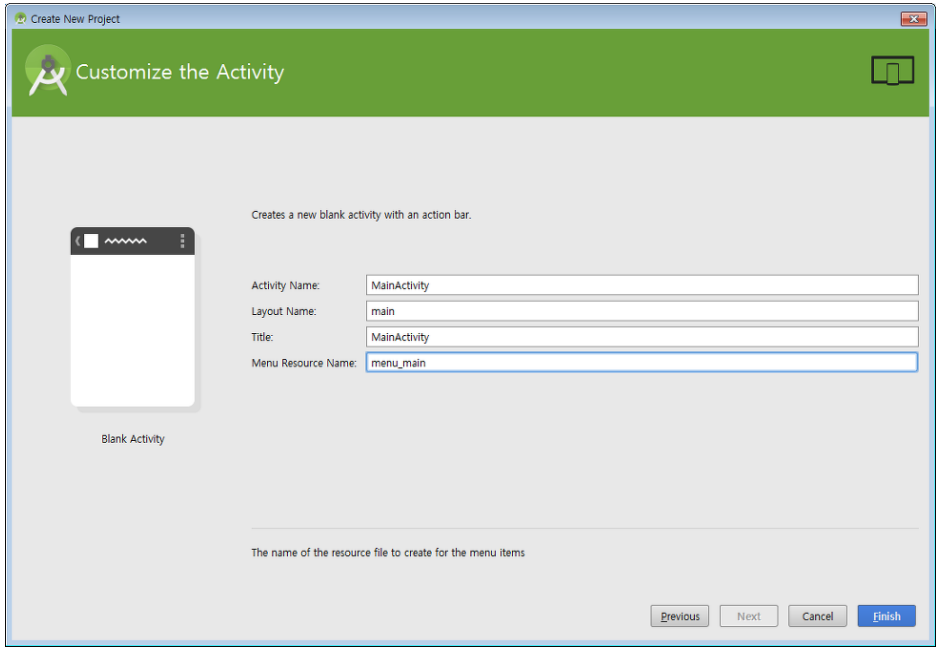


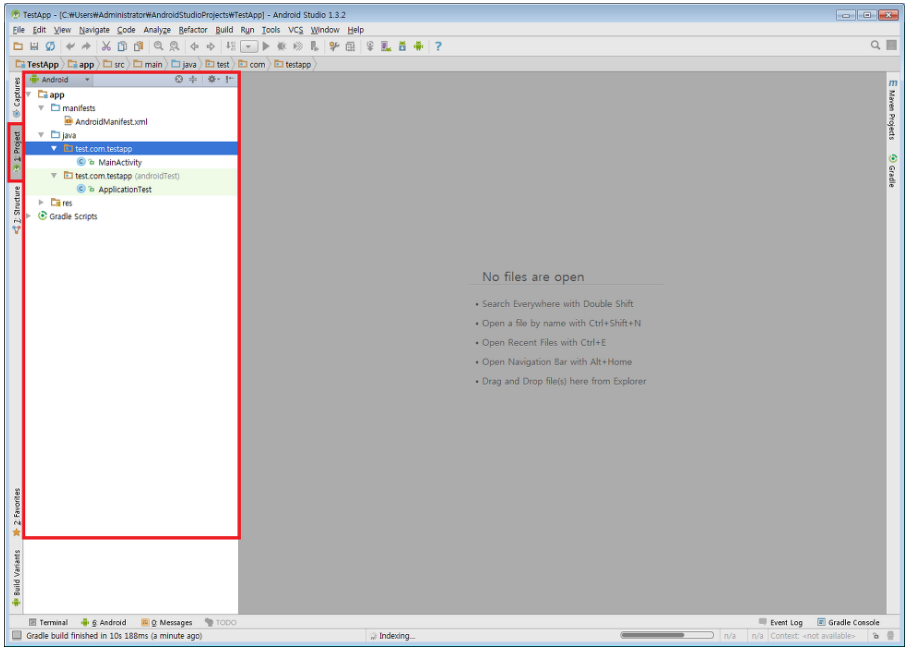


Android Studio Start

Customize the Activity

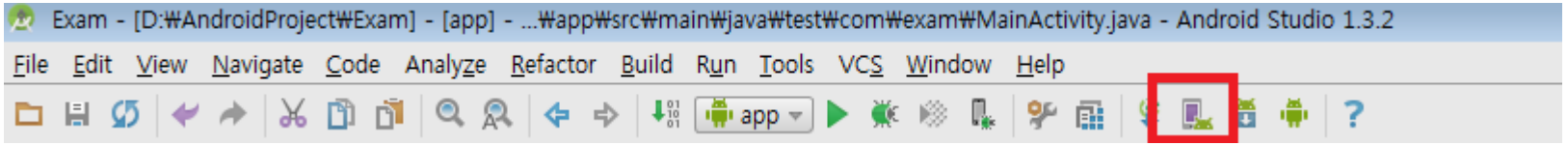
- ✓ Activity명 기타 관련 정보를 입력 후 Finish를 클릭.
- ✓ 옆의 project를 클릭하면 프로젝트 네비게이션이 나타남.
- ✓ Project에 보면 Gradle가 있는데 Android Studio는 Gradle BuildTool을 사용함.





2 Android Studio 개발 환경 구성

Android Studio Emulator



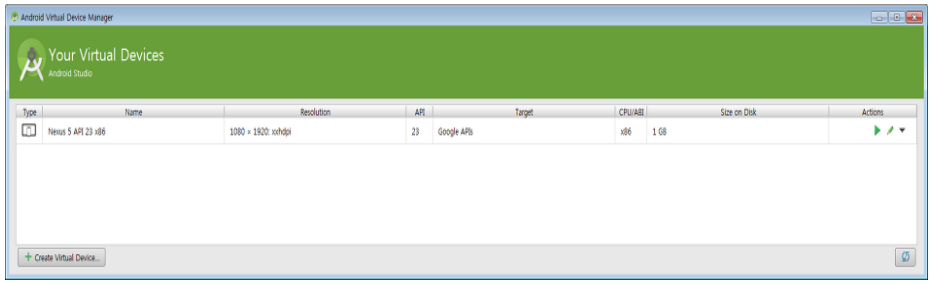
- 상단 메뉴에서 AVD Manager 아이콘 클릭

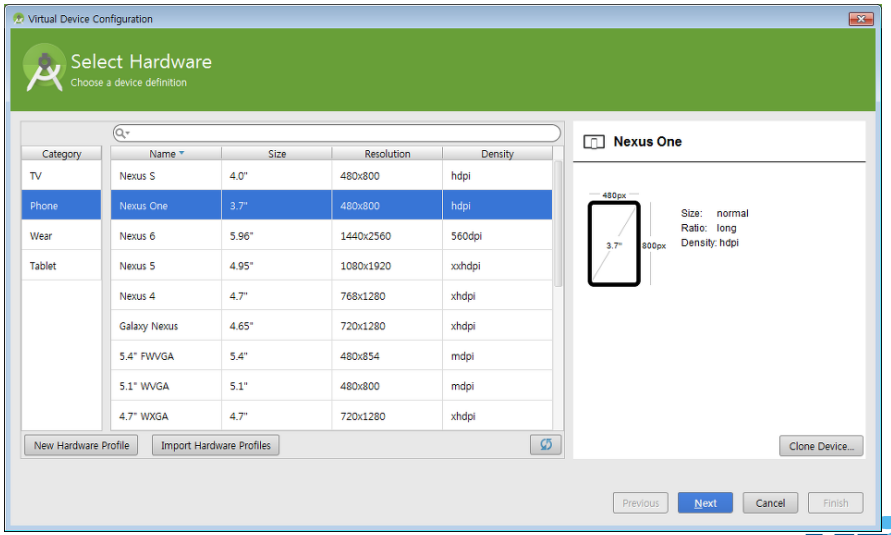
- 기본적으로 Nexus 5 Virtual Device가 생성되어 있음

[신규 Virtual Device 생성 방법]

- Create Virtual Device를 클릭.

- Category Phone선택하여 특정 해상도를 지정 및 선택.

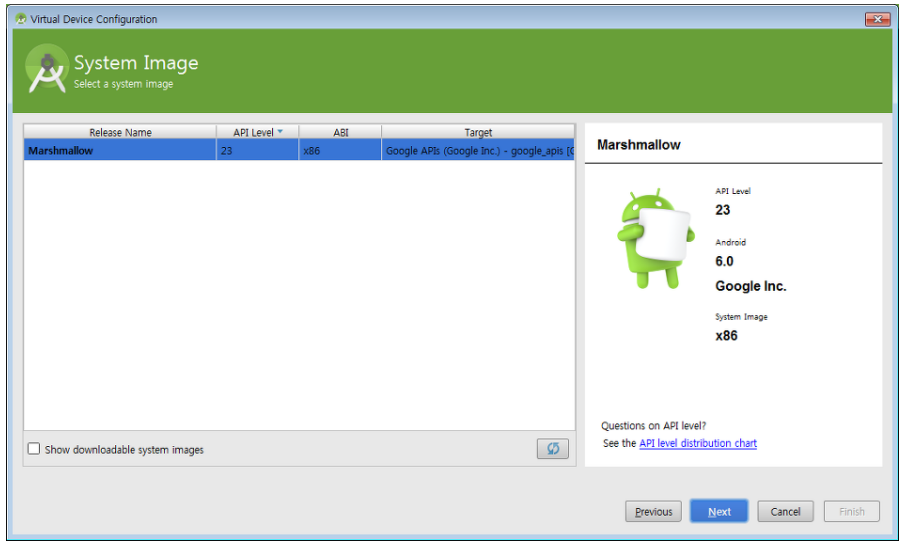




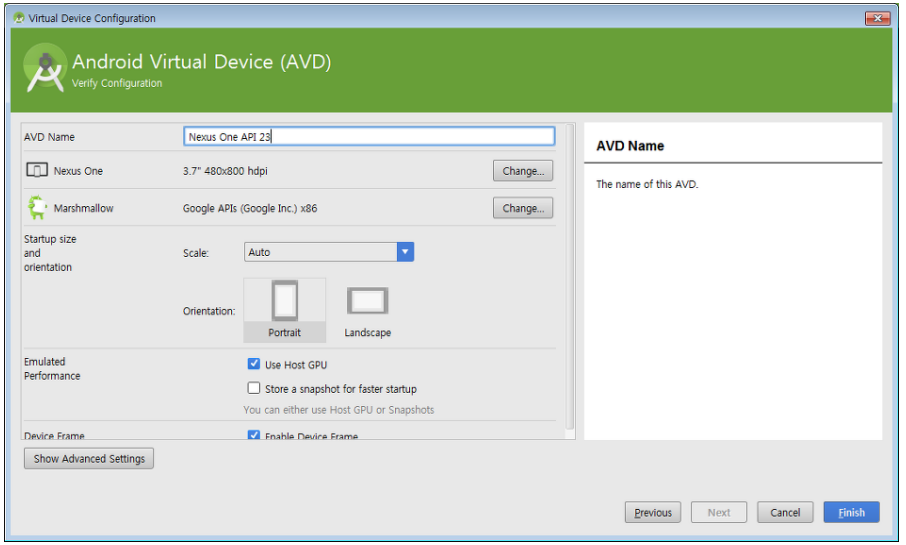
2 Android Studio 개발 환경 구성

Android Studio Emulator

시스템 이미지를 선택.

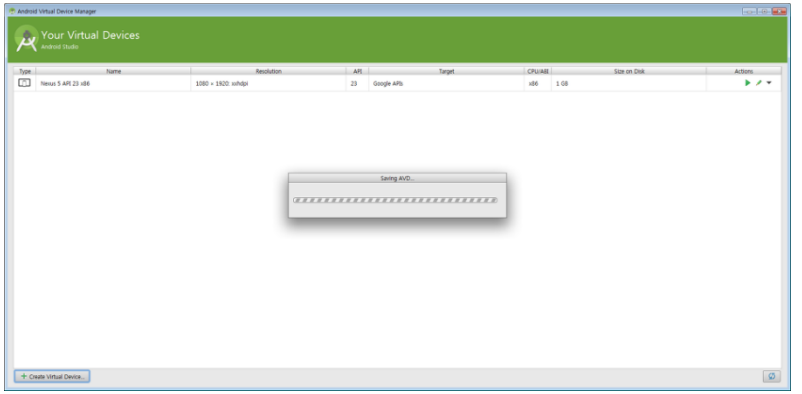


Android Virtual Device(AVD) Configuration



AVD 이름과 기타 옵션을 선택함.

AVD 생성 중...



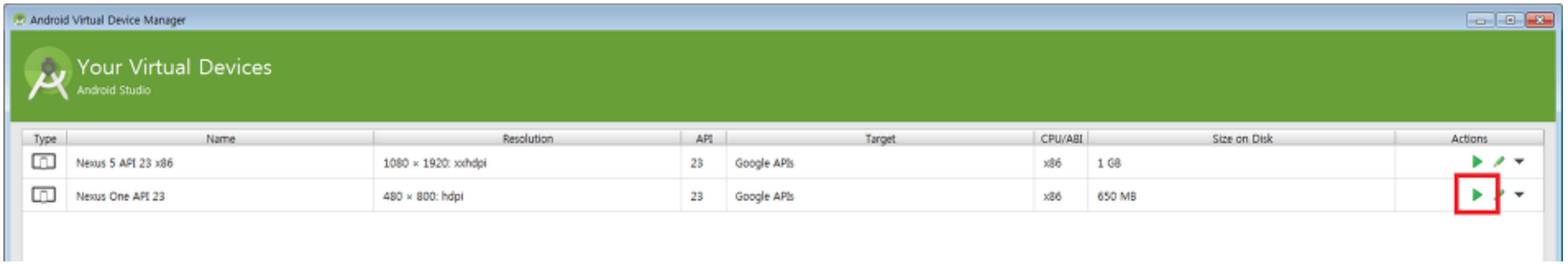
2 Android Studio 개발 환경 구성

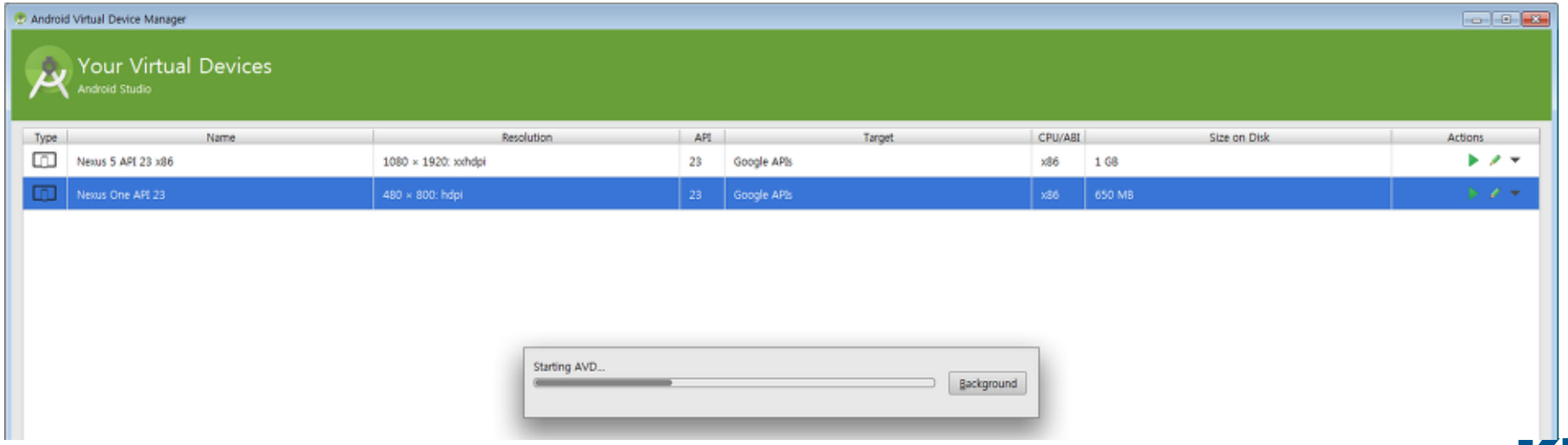
Android Studio Emulator

AVD 생성 후 우측의 실행 버튼을 클릭

진행 상태 바가 나타나며 Emulator가 실행됨

Wait... Wait... Wait





Android Studio Emulator

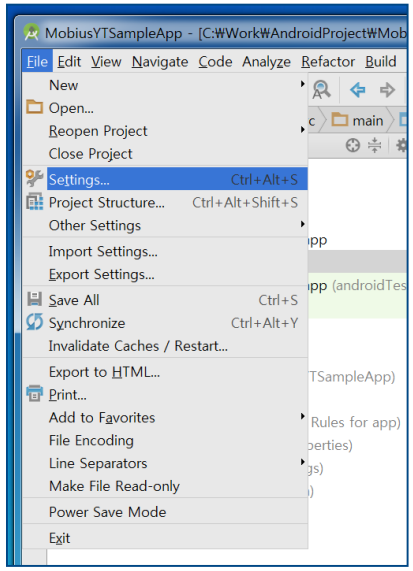
Android Emulator가 뜨고 Android 가 로딩됨.

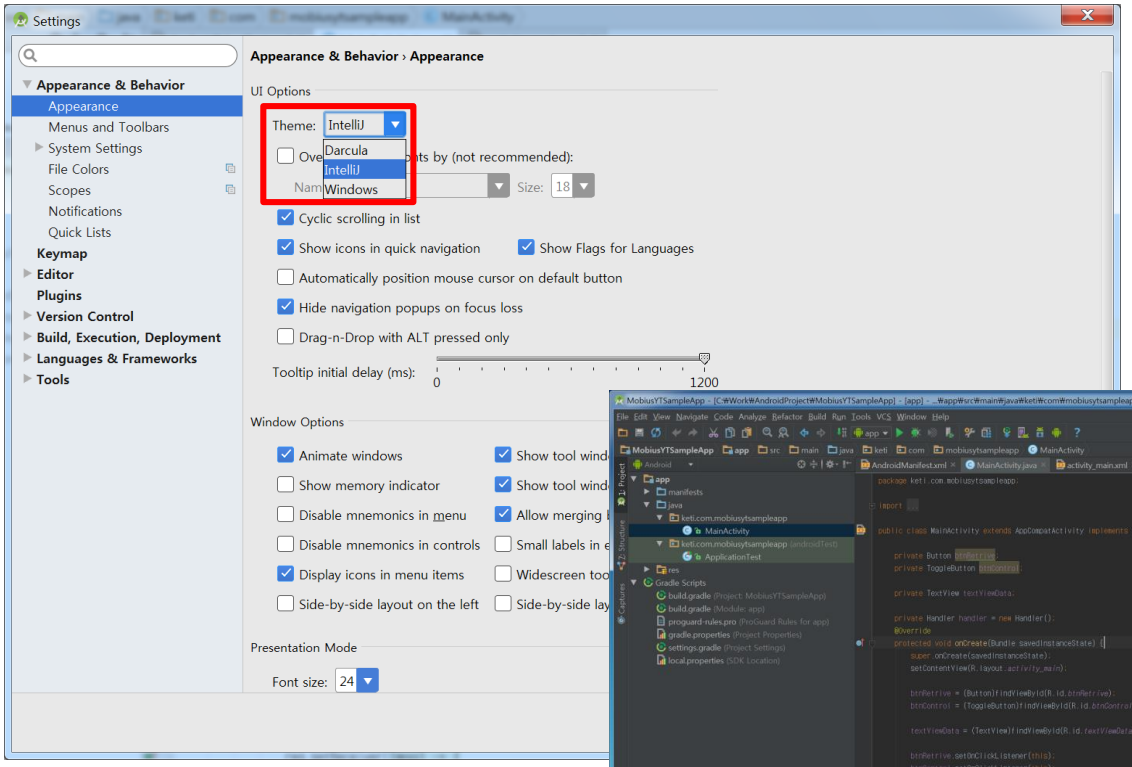
Wait... Wait... Wait

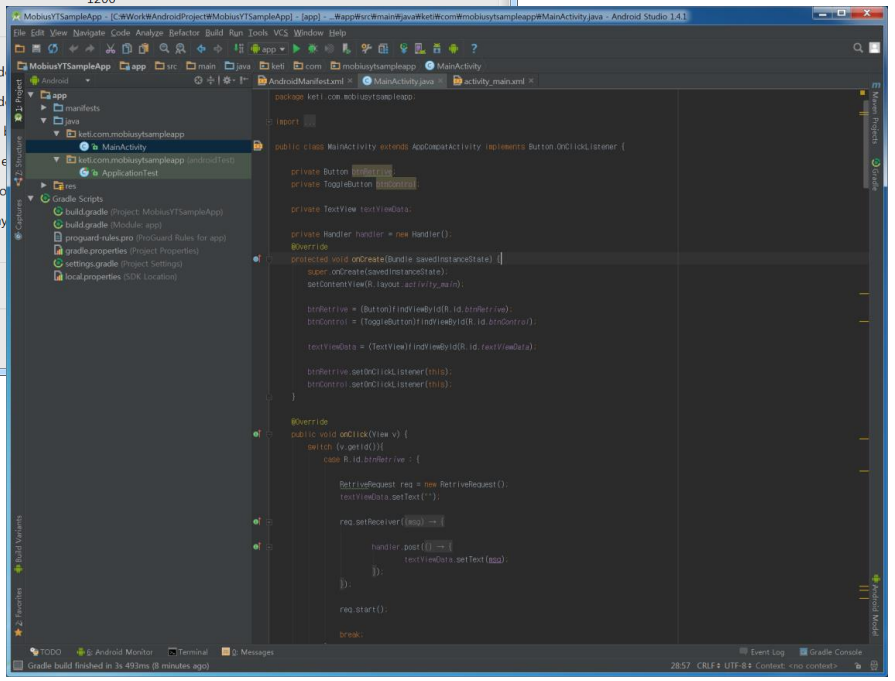
로딩 완료...

2 Android Studio 개발 환경 구성

Tip: Android Studio Theme Change





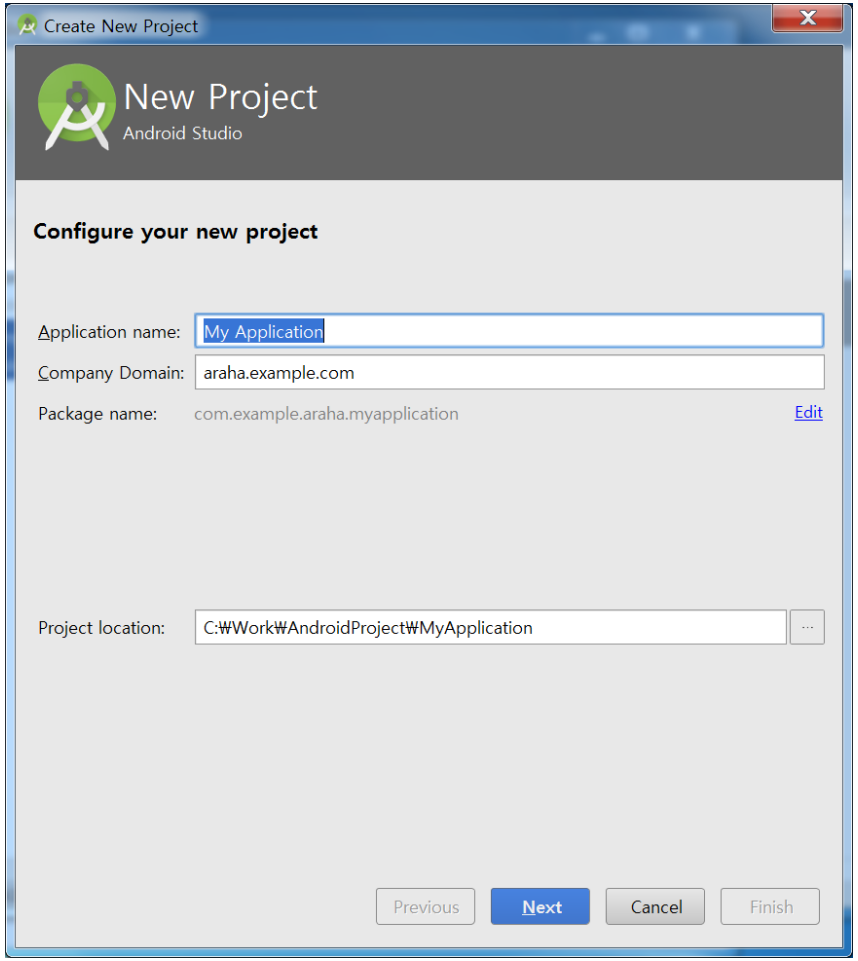


1	Android 소개
2	Android Studio 개발 환경 구성
3	**Web View App 실습**
4	개방형 IoT Platform 연동 App Review
5	App Code Review
6	Generate APK

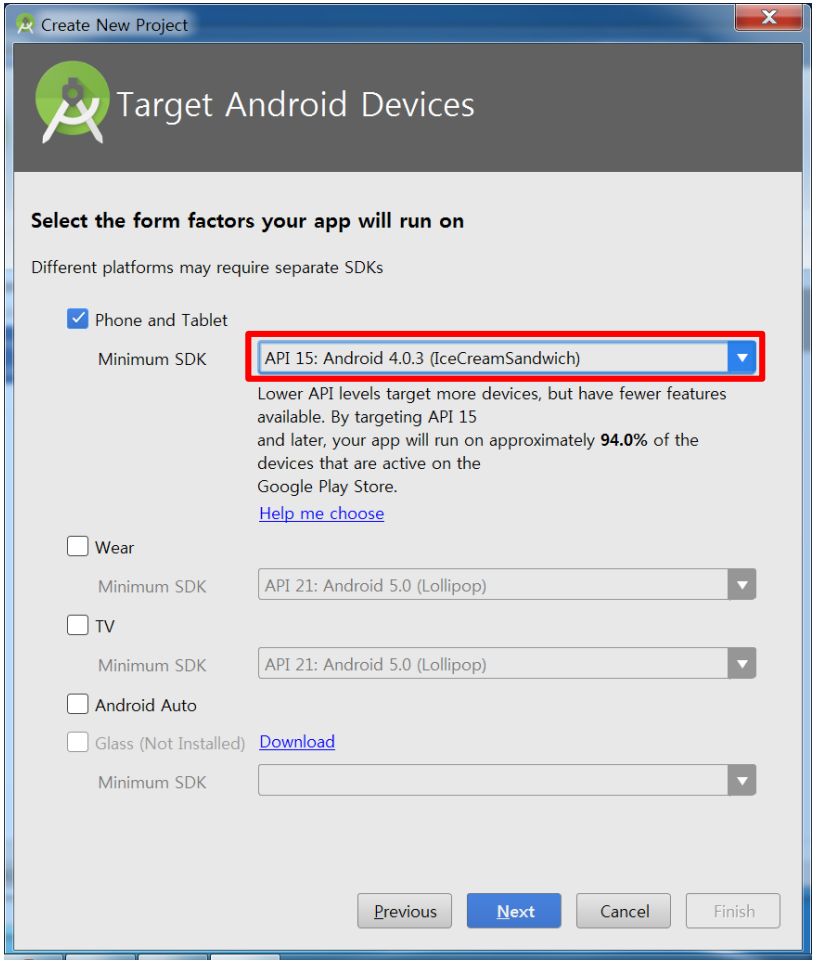
3 Web View App 실습

New Project

Application Name -> Company Domain -> Project location



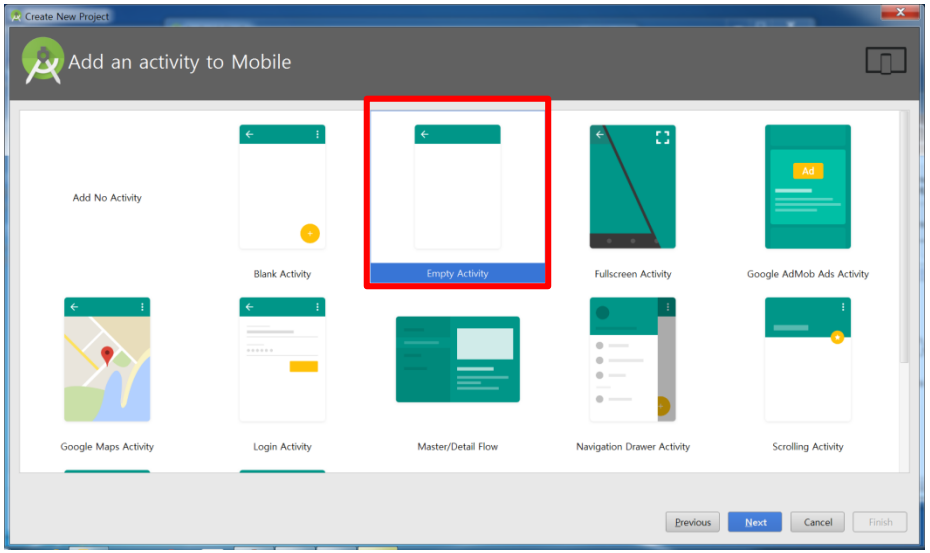
Minimum SDK (Android 구동 Version)



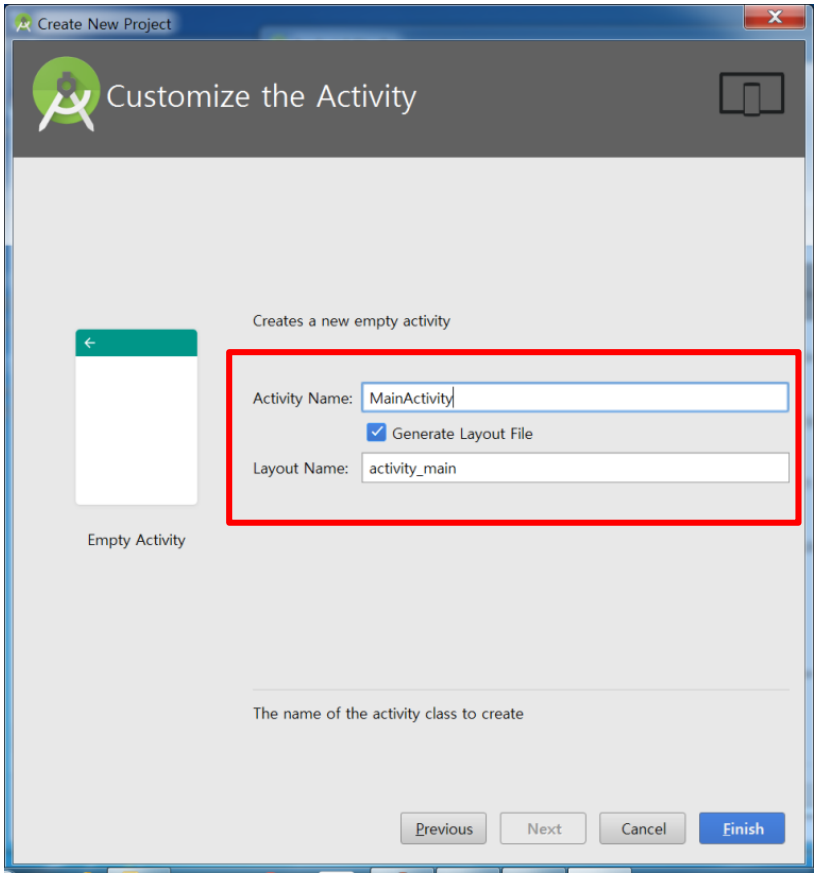
3 Web View App 실습

Activity 화면 선택 및 이름 명명

Empty Activity



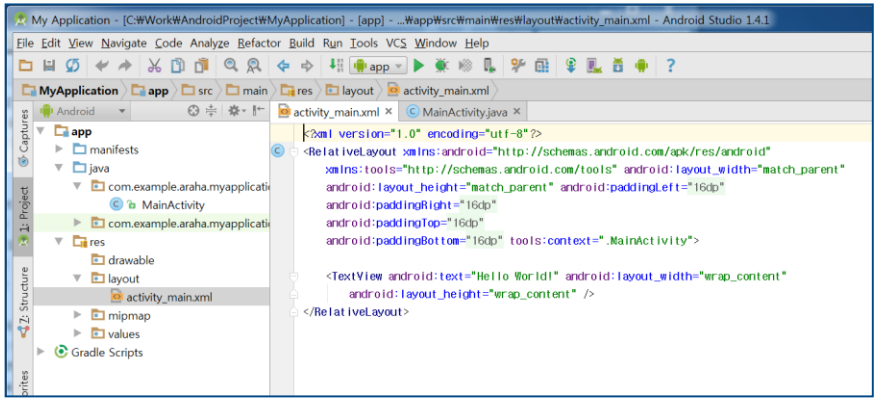
Activity Name, Layout Name



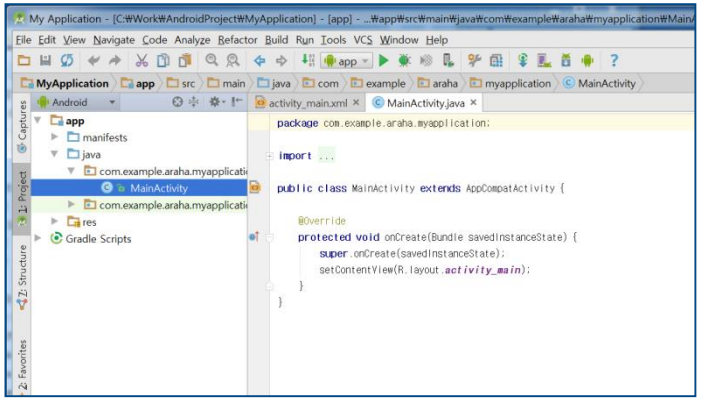
3 Web View App 실습

Basic Default Code

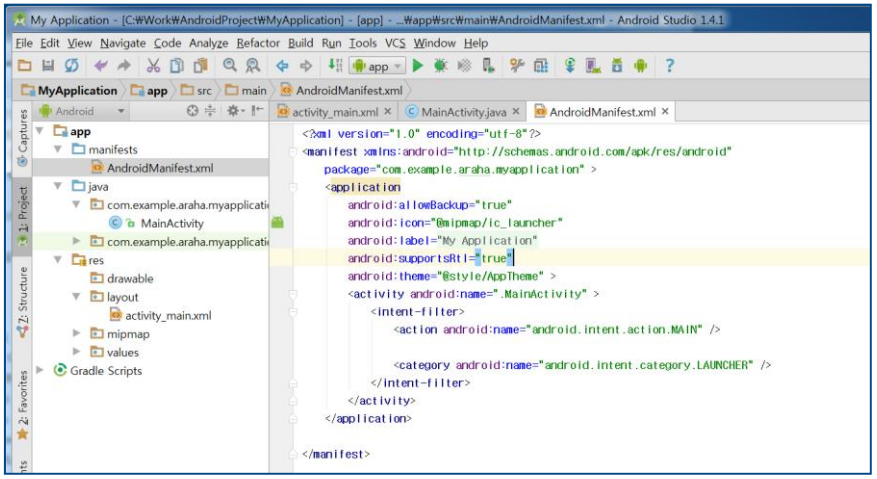
Activity_main.xml



MainActivity.java



AndroidManifest.xml



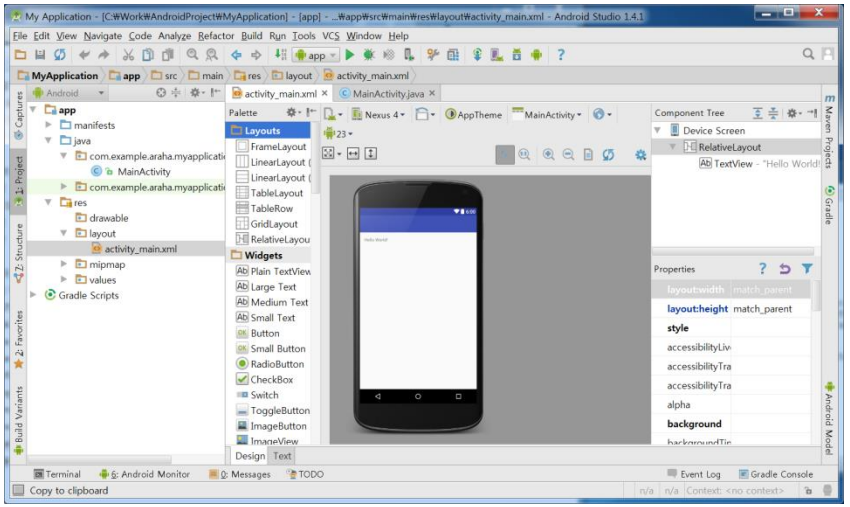
AndroidManifest ?

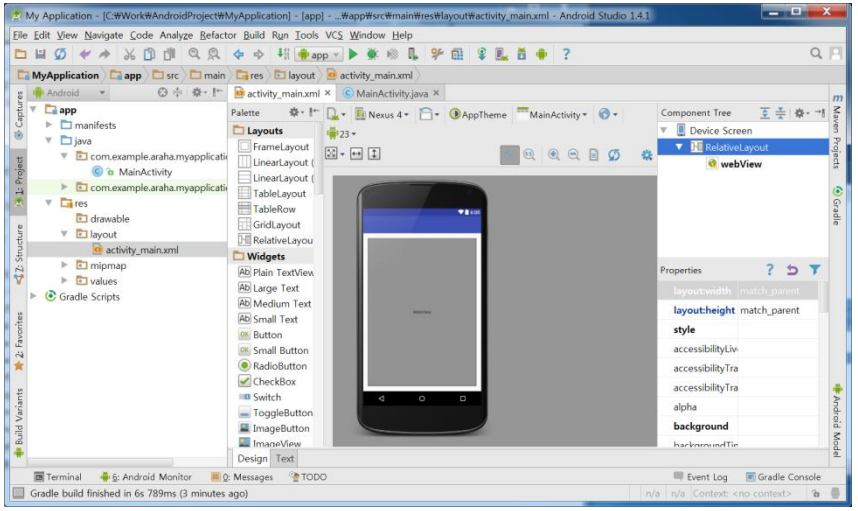
응용 프로그램에 대한 여러 가지 속성정보와
응용 프로그램에 포함 되어있는 각종 컴포넌트를 정의한
명세서 파일

- 주로 권한 관련된 지정 가능
- 액티비티, 서비스, 리시버 등을 지정함

3 Web View App 실습

WebView Layout





Activity_main.xml

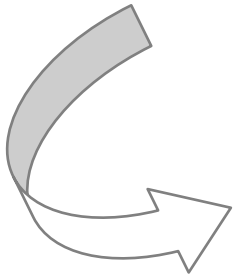
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin" tools:context=".MainActivity">
    <WebView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/webView"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"/>
</RelativeLayout>
```

3 Web View App 실습

WebView 구성 설정

MainActivity.java

```
package com.example.araha.myapplication;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```



```
package com.example.araha.myapplication;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Window;
import android.webkit.WebChromeClient;
import android.webkit.WebView;
import android.webkit.WebViewClient;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        WebView webView = new WebView(this);
        webView.getSettings().setJavaScriptEnabled(true);
        webView.setWebChromeClient(new WebChromeClient());
        webView.getSettings().setBuiltInZoomControls(true);
        webView.setWebViewClient(new WebViewClient());
        webView.loadUrl("http://www.naver.com");
        setContentView(webView);
    }
}
```

WebView 설정

- Javascript 사용
- Zoom 기능 사용
- 기본 URL 지정
- ContentView에 WebView 활성화

Internet Permission 설정

AndroidManifest.xml

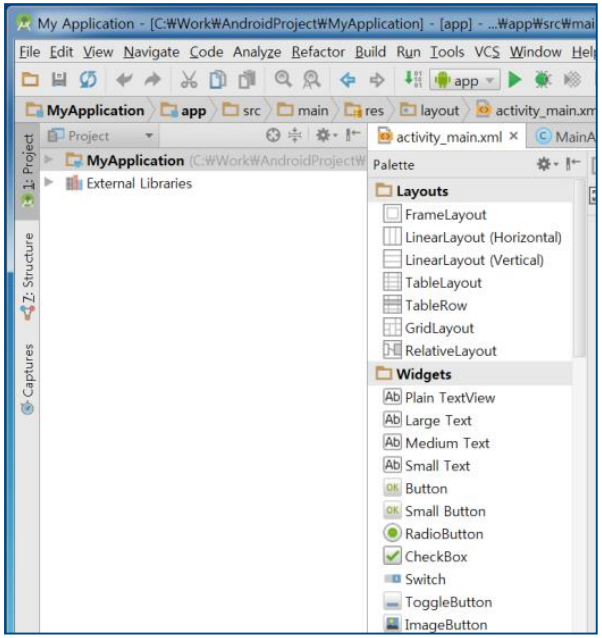
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.araha.myapplication" >
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme" >
        <activity android:name=".MainActivity" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

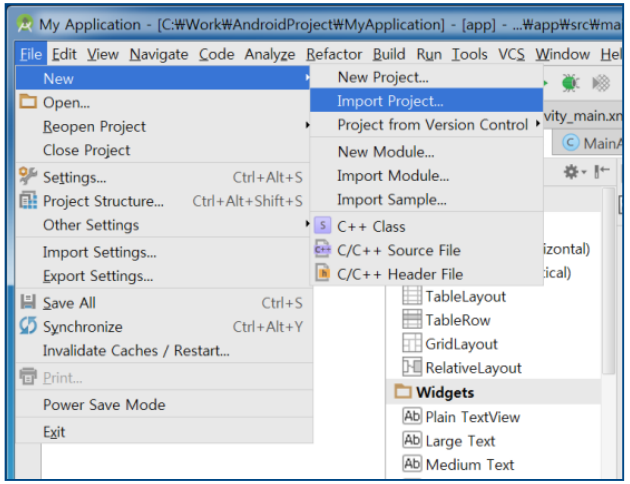
1	Android 소개
2	Android Studio 개발 환경 구성
3	Web View App 실습
4	**개방형 IoT Platform 연동 App Review**
5	App Code Review
6	Generate APK

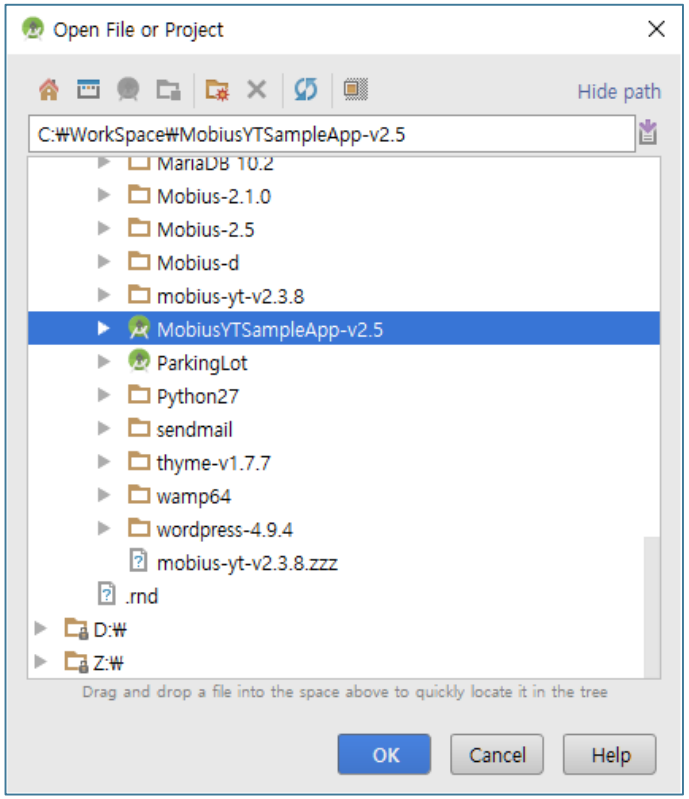
4 개방형 IoT Platform 연동 App Review

Project Import

Android Project Import





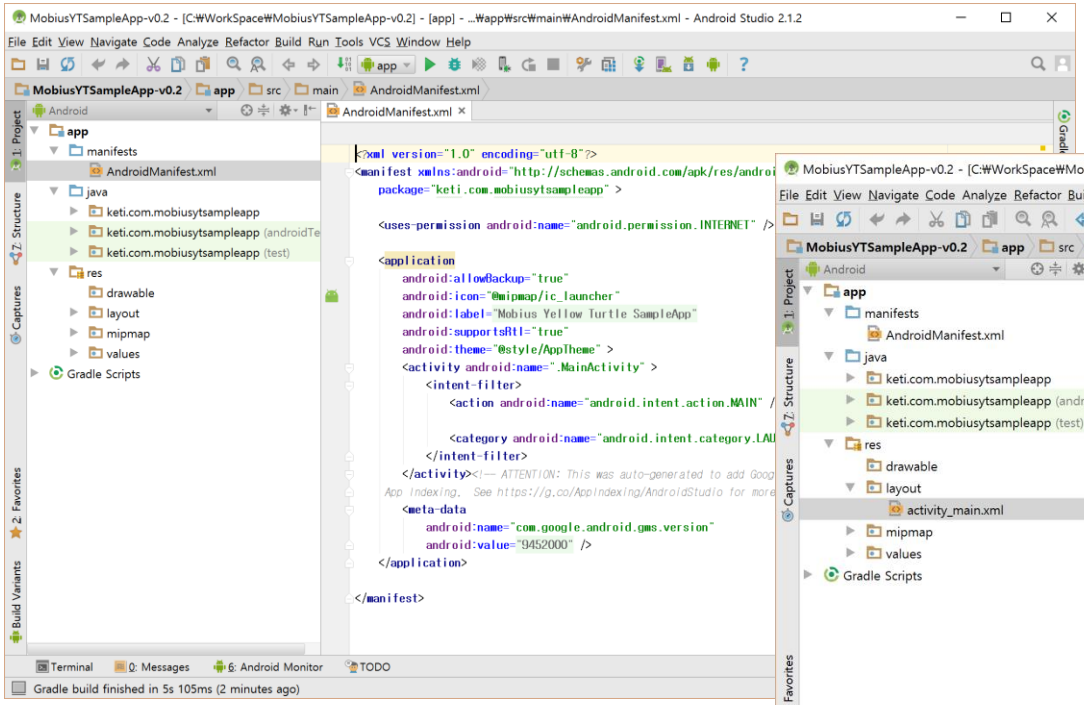


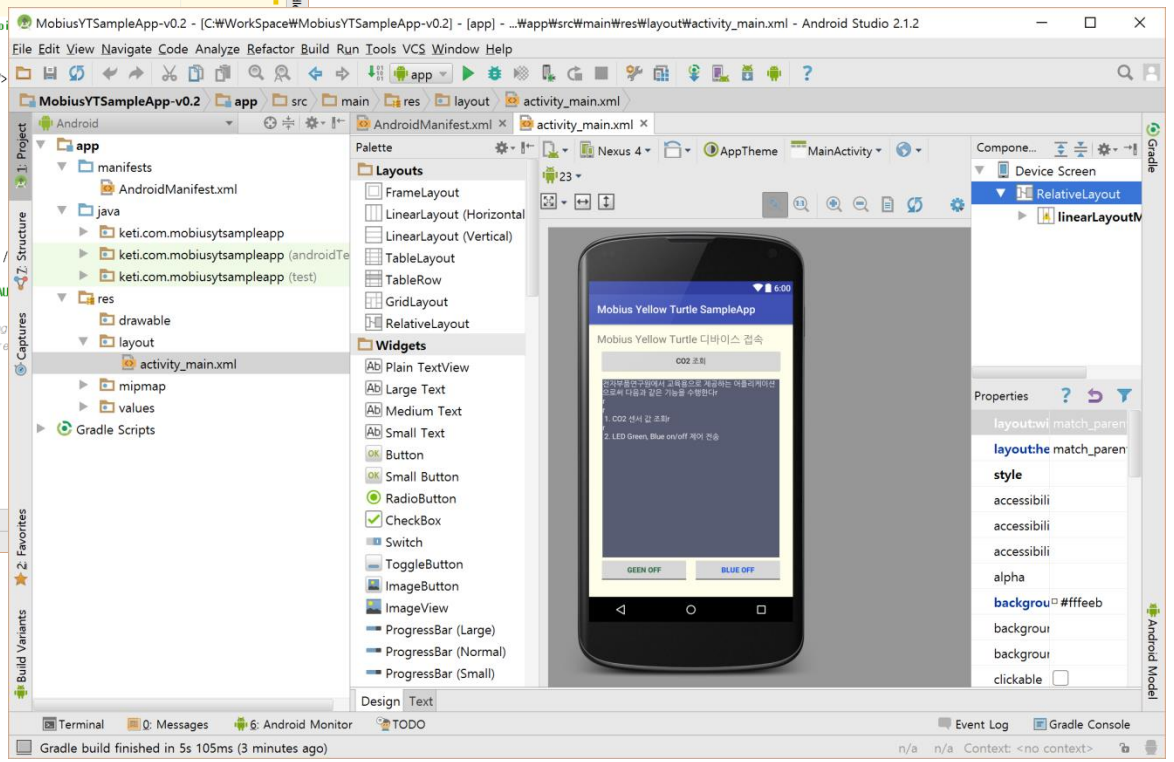
*** OCEAN Developers 에 배포 (http://developers.iotocean.org)**

[MobiusTYSampleApp-v2.5 , 2018.5.11]

Project Import

Android/ app/ manifest (AndroidManifest.xml)



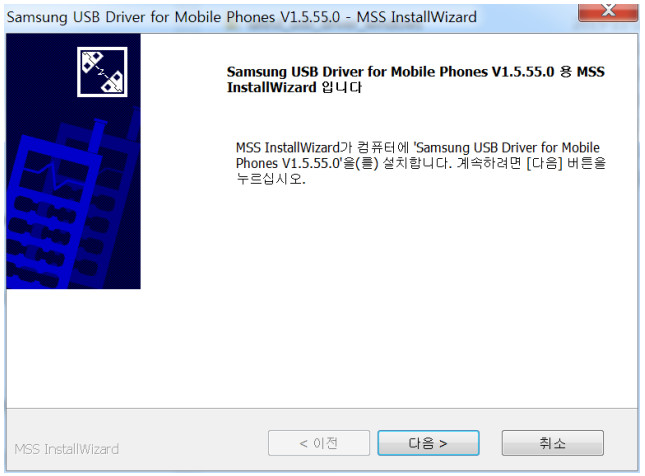


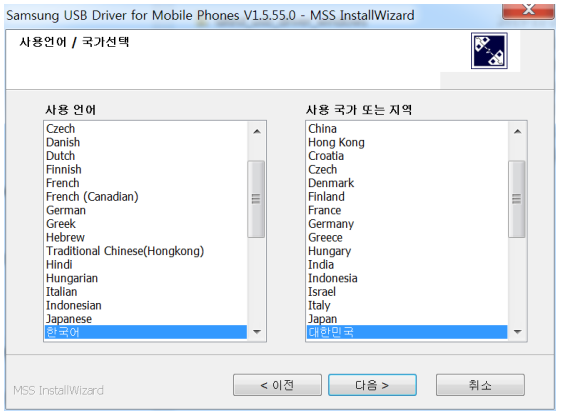
Android/ app/ res/ layout/ activity_main.xml (Design)

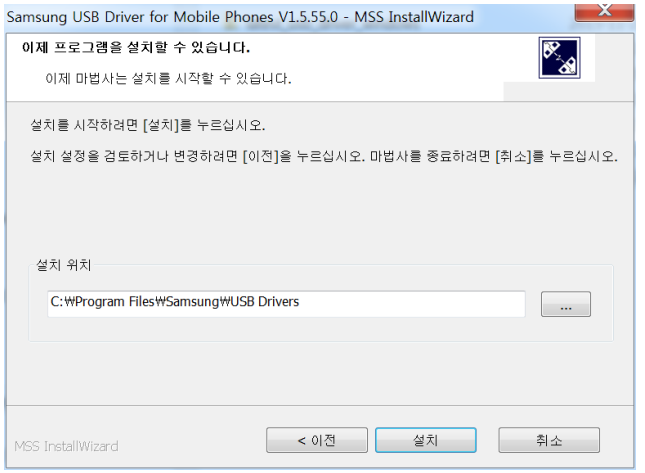
4 개방형 IoT Platform 연동 App Review

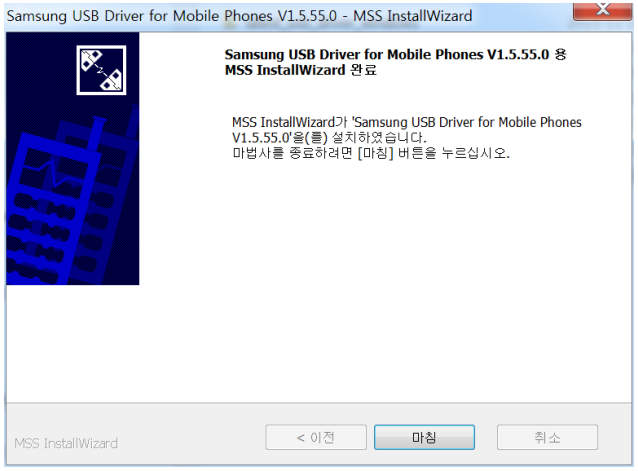
Android USB Driver Install

Samsung USB 통합 Driver Install (http://local.sec.samsung.com/comLocal/support/down/kies_main.do?kind=usb)





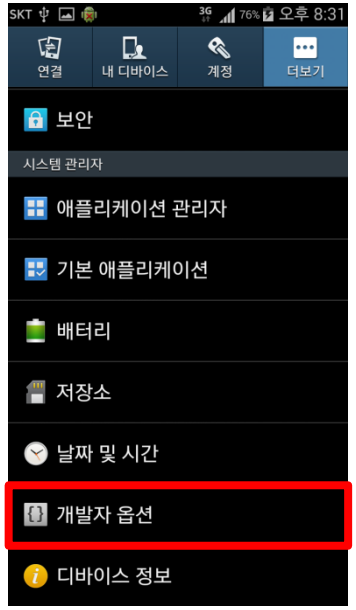


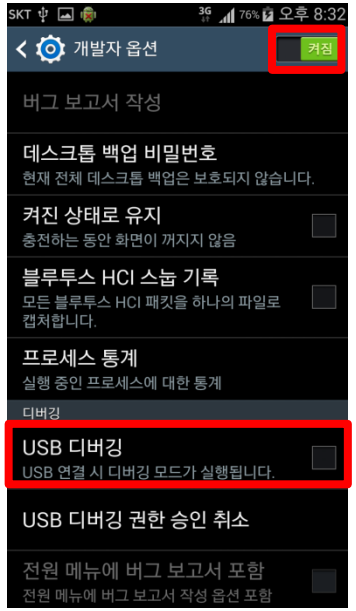


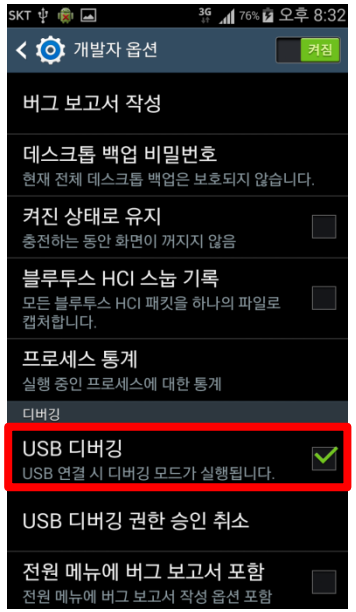
4 개방형 IoT Platform 연동 App Review

USB Debugging Mode on Android

Common Debugging Mode



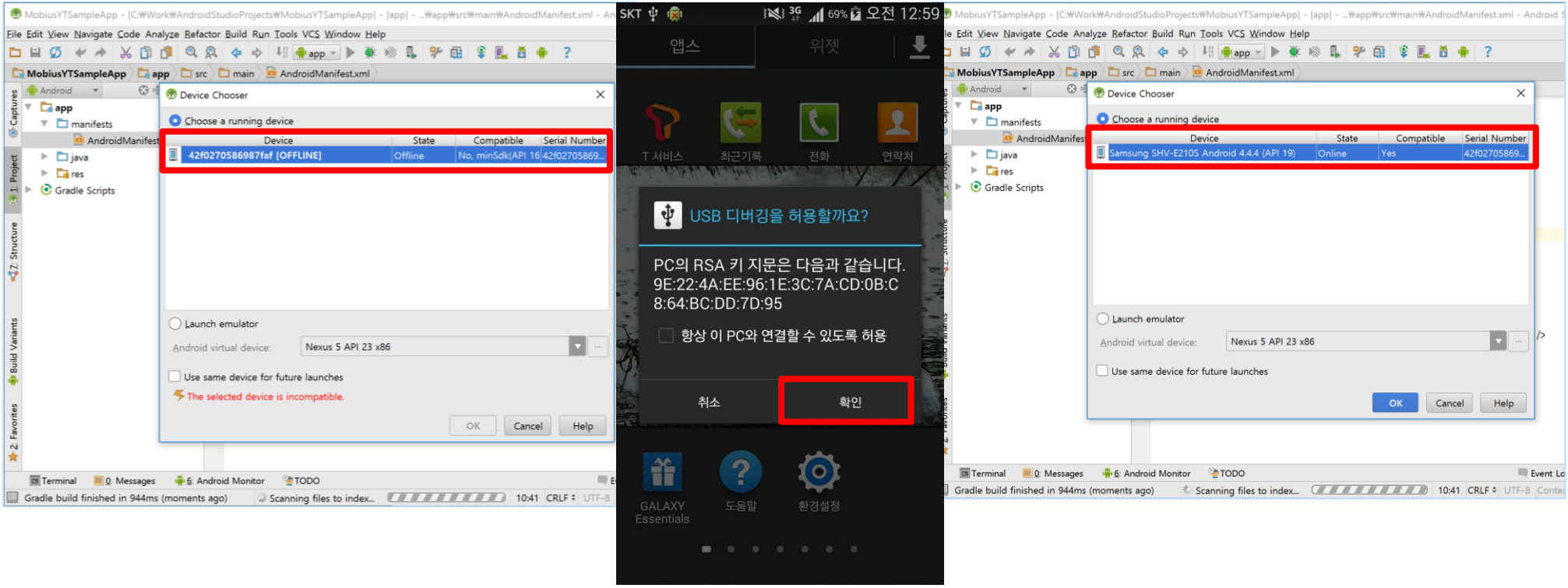




4 개방형 IoT Platform 연동 App Review

App Running

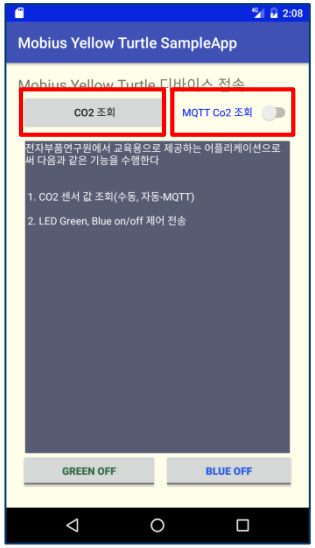
Android Phone Device Debugging Mode Connect



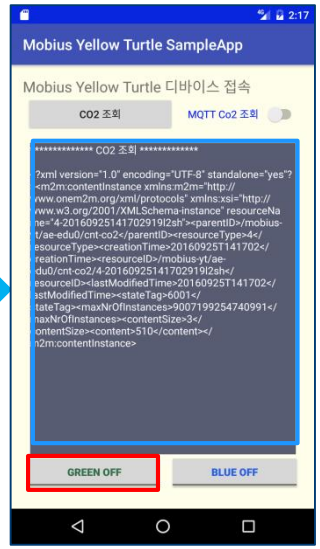
4 개방형 IoT Platform 연동 App Review

App Running

초기 화면



CO2 조회



Gree LED On

Blue LED On

Green LED Off

MQTT Co2 조회

Blue LED Off

1	Android 소개
2	Android Studio 개발 환경 구성
3	Web View App 실습
4	개방형 IoT Platform 연동 App Review
5	App Code Review
6	Generate APK

Internet Permission 설정 [MQTT]

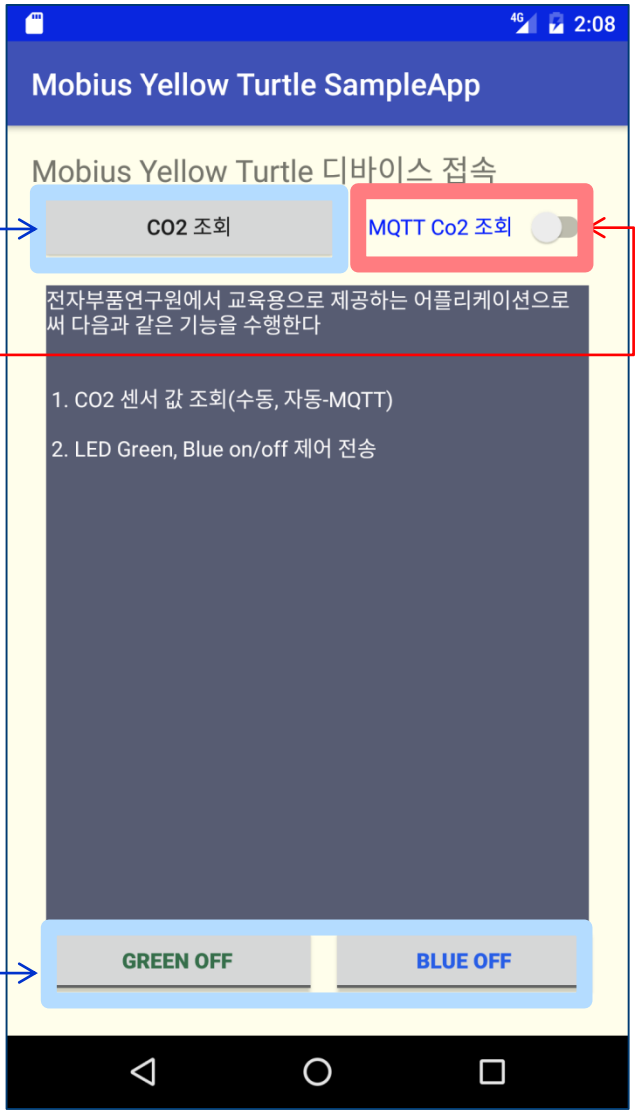
AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="keti.com.mobiusytsampleapp" >

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme" >
        <activity android:name=".MainActivity" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <service android:name="org.eclipse.paho.android.service.MqttService" />
    </application>

</manifest>
```

5 App Code Review

Code Structure Review

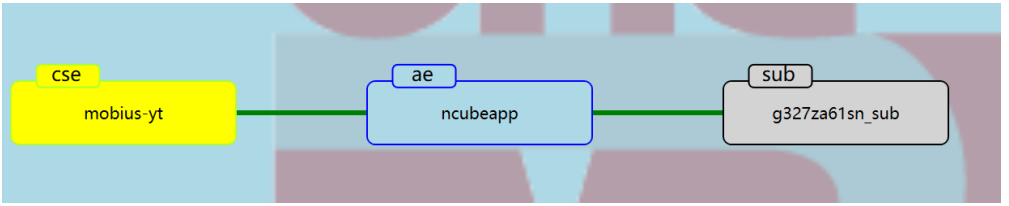
```
// Main
public MainActivity() { handler = new Handler(); }
/* onCreate */
protected void onCreate(Bundle savedInstanceState) {...}
/* AE Create for Androdi AE */
public void GetAEInfo() {...}
/* Switch - Get Co2 Data With MQTT */
public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {...}
/* MQTT Subscription */
public void MQTT_Create(boolean mtqqStart) {...}
/* MQTT Listener */
private IMqttActionListener mainIMqttActionListener = new IMqttActionListener() {...};
/* MQTT Broker Message Received */
private MqttCallback mainMqttCallback = new MqttCallback() {...};
@Override
public void onClick(View v) {...}
@Override
public void onStart() {...}
@Override
public void onStop() {...}
/* Response callback Interface */
public interface IReceived {...}

/* Retrieve Co2 Sensor */
class RetrieveRequest extends Thread {...}
/* Request Control LED */
class ControlRequest extends Thread {...}
/* Regeust AE Creation */
class aeCreateRequest extends Thread {...}
/* Retrieve AE-ID */
class aeRetrieveRequest extends Thread {...}
/* Subscribe Co2 Content Resource */
class SubscribeResource extends Thread {...}
```



5 App Code Review

Code Structure Review [AE Create or AE Retrieve … Get AE-ID]

```
/* Request AE Creation */
class aeCreateRequest extends Thread {
    private final Logger LOG = Logger.getLogger(aeCreateRequest.class.getName());
    String TAG = aeCreateRequest.class.getName();
    private IReceived receiver;
    int responseCode=0;
    public ApplicationEntityObject applicationEntity;
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }
    public aeCreateRequest(){
        applicationEntity = new ApplicationEntityObject();
        applicationEntity.setResourceName(ae.getappName());
    }
    @Override
    public void run() {...}
}
/* Retrieve AE-ID */
class aeRetrieveRequest extends Thread {
    private final Logger LOG = Logger.getLogger(aeCreateRequest.class.getName());
    private IReceived receiver;
    int responseCode=0;

    public aeRetrieveRequest() {...}
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }

    @Override
    public void run() {...}
}
```




```
<?xml version="1.0" encoding="utf-16" standalone="yes"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
rn="ncubeapp">
  <pi>/mobius-yt</pi>
  <ty>2</ty>
  <ct>20160928T161212</ct>
  <ri>/mobius-yt/ncubeapp</ri>
  <lt>20160928T161212</lt>
  <api>0.2.481.2.0001.001.0001114</api>
  <aei>xxxxxx</aei>
  <rr>true</rr>
</m2m:ae>
```

※ oneM2M 표준에서는 Android Application 자체도 Application Entity(AE)로 간주하기에 신규로 AE를 생성하거나 생성된 AE를 통해 얻어진 AE-ID를 이용하여 Orginator로 이용해야 함.

실제 AE-ID(aei) 는 서버에 AE 생성 요청시 임의로 생성되어 정의됨.

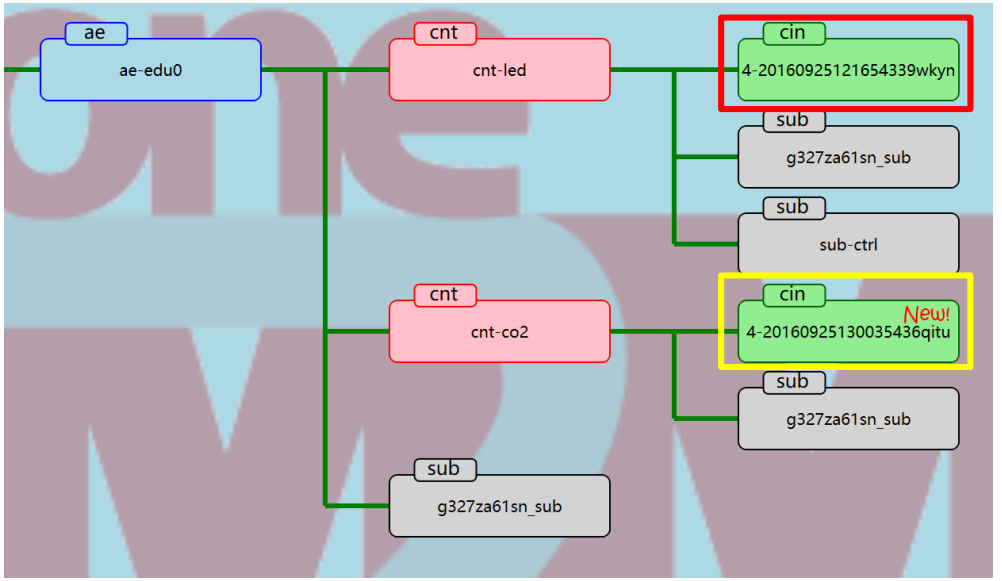
5 App Code Review

Code Structure Review [Co2 Retrieve, LED Control]

```
/* Retrieve Co2 Sensor */
class RetrieveRequest extends Thread {
    private final Logger LOG = Logger.getLogger(RetrieveRequest.class.getName());
    private IReceived receiver;
    private String ContainerName = "cnt-co2";

    public RetrieveRequest(String containerName) {
        this.ContainerName = containerName;
    }
    public RetrieveRequest() {}

    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }

    @Override
    public void run() {...}
}
```

```
/* Request Control LED */
class ControlRequest extends Thread {
    private final Logger LOG = Logger.getLogger(ControlRequest.class.getName());
    private IReceived receiver;
    private String container_name = "cnt-led";

    public ContentInstanceObject contentinstance;

    public ControlRequest(String comm) {
        contentinstance = new ContentInstanceObject();
        contentinstance.setContent(comm);
    }

    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }

    @Override
    public void run() {...}
}
```

<?xml version="1.0" encoding="utf-16" standalone="yes"?><m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="4-20160925121654339wkyn"><pi>/mobius-yt/ae-edu0/cnt-led</pi><ty>4</ty><ct>20160925T121654</ct><ri>/mobius-yt/ae-edu0/cnt-led/4-20160925121654339wkyn</ri><lt>20160925T121654</lt><st>18</st><mni>9007199254740991</mni><cs>1</cs><cnf>text</cnf><con>4</con></m2m:cin>

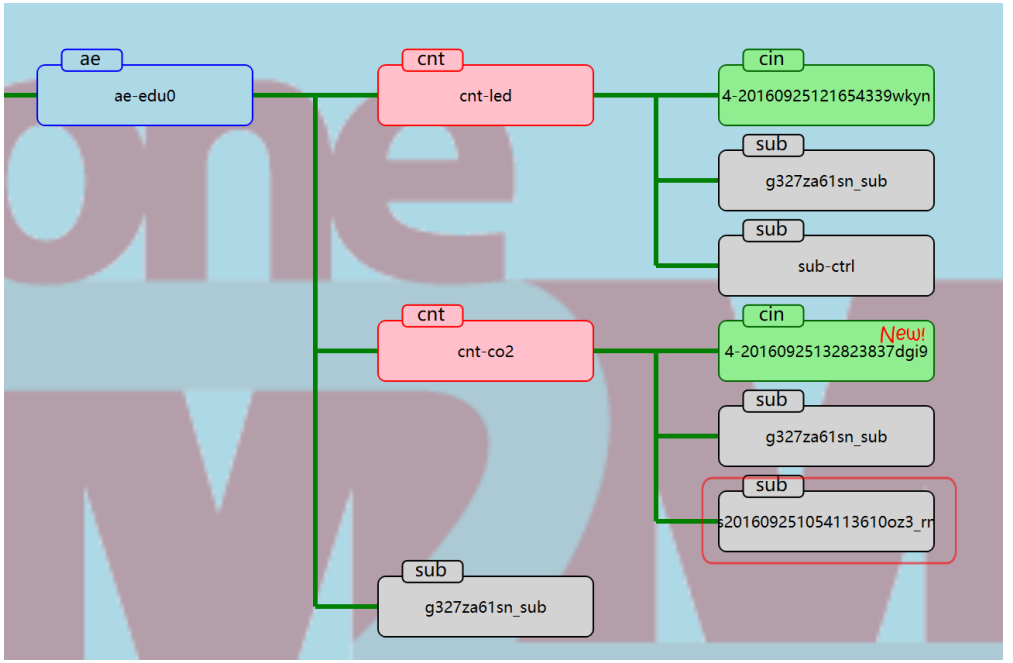


<?xml version="1.0" encoding="utf-16" standalone="yes"?><m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="4-20160925130205542urps"><pi>/mobius-yt/ae-edu0/cnt-co2</pi><ty>4</ty><ct>20160925T130205</ct><ri>/mobius-yt/ae-edu0/cnt-co2/4-20160925130205542urps</ri><lt>20160925T130205</lt><st>3756</st><mni>9007199254740991</mni><cs>3</cs><con>719</con></m2m:cin>

5 App Code Review

Code Structure Review [Subscription Content(cnt-co2) Resource Create for MQTT]

```
/* Subscribe Co2 Content Resource */
class SubscribeResource extends Thread {
    private final Logger LOG = Logger.getLogger(SubscribeResource.class.getName());
    private IReceived receiver;
    private String container_name = "cnt-co2"; //change to control container name

    public ContentSubscribeObject subscribeInstance;
    public SubscribeResource() {...}
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }

    @Override
    public void run() {...}
}
```




```
<?xml version="1.0" encoding="utf-16" standalone="yes"?><m2m:sub
xmlns:m2m="http://www.onem2m.org/xml/protocols"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
rn="s20160925105411361oz3_rn"><pi>/mobius-yt/ae-edu0/cnt-
co2</pi><ty>23</ty><ct>20160925T133651</ct><ri>/mobius-yt/ae-edu0/cnt-
co2/s20160925105411361oz3_rn</ri><lt>20160925T133651</lt><st>0</st><enc
><net>3</net></enc><nu>mqtt://192.168.25.47/S20160925105411361oz3_sub</
nu><nct>2</nct><cr>S20160925105411361oz3</cr></m2m:sub>
```

5 App Code Review

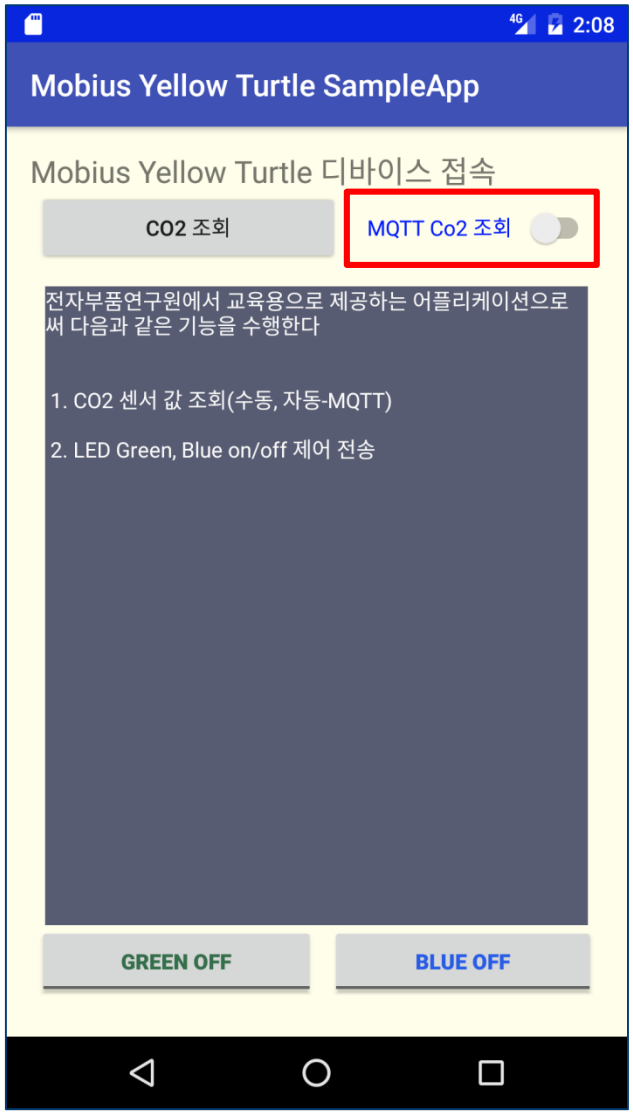
Code Structure Review [Button Click Event]

```
public void onClick(View v) {


        switch (v.getId()) {


          case R.id.btnRetrieve: {
             RetrieveRequest req = new RetrieveRequest();
             textViewData.setText("");
             req.setReceiver(new IReceived() {
                @Override
                public void getResponseBody(final String msg) {
                   handler.post(new Runnable() {
                      @Override
                      public void run() {
                         textViewData.setText("********** CO2 조회 ***;
                      }
                   });
                }
             });
             req.start();
            break;
          }
         …

         ..

         .
```

```
case R.id.btnControl_Green: {
        if (((ToggleButton) v).isChecked()) {
           ControlRequest req = new ControlRequest("1");
           req.setReceiver(new IReceived() {
              @Override
              public void getResponseBody(final String msg) {
                 handler.post(new Runnable() {
                    @Override
                    public void run() {
                       textViewData.setText("********** LED Green 제어(켜짐)..");
                    }
                 });
              }
           });
           req.start();
        } else {
           ControlRequest req = new ControlRequest("2");
           req.setReceiver(new IReceived() {
              @Override
              public void getResponseBody(final String msg) {
                 handler.post(new Runnable() {
                    @Override
                    public void run() {
                       textViewData.setText("********** LED Green 제어(꺼짐)..");
                    }
                 });
              }
           });
           req.start();
        }
        break;
     }
```

5 App Code Review

Code Structure Review [Switch onCheckChanged Event]




```
public void onCheckedChanged(CompoundButton buttonView, boolean isChecked)
if (isChecked) {
      Log.d(TAG, "MQTT Create");
      MQTT_Create(true);
   } else {
      Log.d(TAG, "MQTT Close");
      MQTT_Create(false);
   }
}

/* MQTT Subscription */
public void MQTT_Create(boolean mtqqStart) {
   if (mtqqStart && mqttClient == null) {
      /* Subscription Resource Create to Yellow Turtle */
      SubscribeResource subcribeResource = new SubscribeResource();
      subcribeResource.setReceiver(new IReceived() {
         public void getResponseBody(final String msg) {
            handler.post(new Runnable() {
               public void run() {   ...   }
            });
         }
      });
      subcribeResource.start();

      /* MQTT Subscribe */
      mqttClient = new MqttAndroidClient(this.getApplicationContext(), "tcp://" + csebase.getHost()
+ ":" + csebase.getMQTTPort(), MqttClient.generateClientId());
      mqttClient.setCallback(mainMqttCallback);
      try {
         IMqttToken token = mqttClient.connect();
         token.setActionCallback(mainIMqttActionListener);
      } catch (MqttException e) {
         e.printStackTrace();
      }
   } else {
      /* MQTT unSubscribe or Client Close */
      mqttClient.setCallback(null);
      mqttClient.close();
      mqttClient = null;
   }
}
```

5 App Code Review

RetrieveRequest POSTMAN Script (CO2 Sensing Data)

M-IF77. contentInstance 리소스 조회

GET {{mp_url}}/{{cb}}/ae-edu1/cnt-co2/latest Params Send Save

Authorization Headers (4) Body Pre-request Script Tests Code

Accept	application/xml	
X-M2M-RI	12345	
X-M2M-Origin	Sae_edu1	
nmtype	long	
key	value	

Bulk Edit Presets

Body Cookies Headers (8) Tests Status: 200 OK Time: 213 ms

Pretty Raw Preview XML Save Response

```
1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <m2m:contentInstance xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema
       -instance" resourceName="4-20160928171758407S0Md">
3      <parentID>/mobius-yt/ae-edu1/cnt-co2</parentID>
4      <resourceType>4</resourceType>
5      <creationTime>20160928T171758</creationTime>
6      <resourceID>/mobius-yt/ae-edu1/cnt-co2/4-20160928171758407S0Md</resourceID>
7      <lastModifiedTime>20160928T171758</lastModifiedTime>
8      <stateTag>0</stateTag>
9      <maxNrOfInstances>9007199254740991</maxNrOfInstances>
10     <contentSize>3</contentSize>
11     <content>461</content>
12 </m2m:contentInstance>
```

5 App Code Review

RetrieveRequest Java Code Review

```
private String ServiceAEName = "ae-edu1";

...

public void GetAEInfo() {

    csebase.setInfo("192.168.0.135","7579","Mobius","1883"); //Sample IP
    ...

    ...
}
```

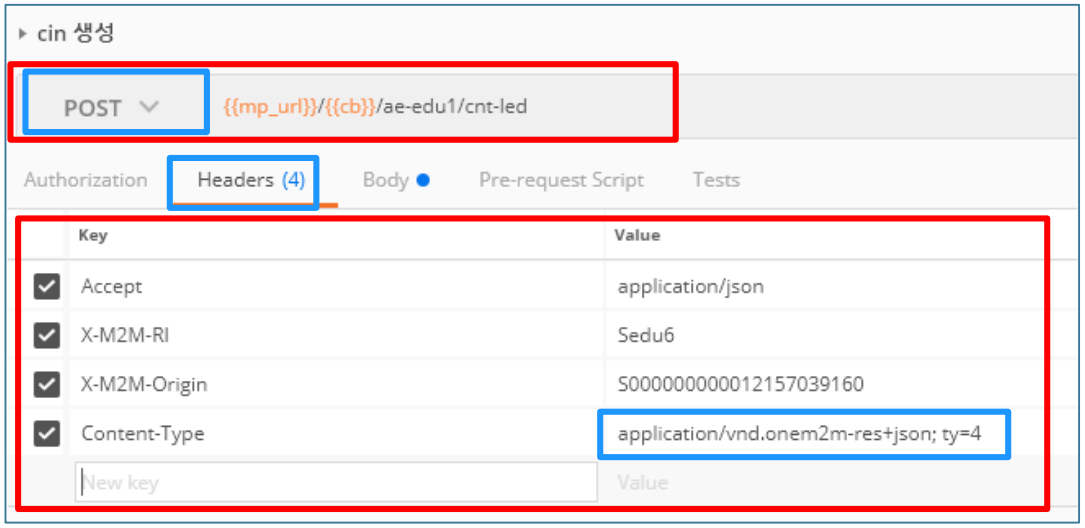
```
/* Retrieve Co2 Sensor */
class RetrieveRequest extends Thread {
    private final Logger LOG = Logger.getLogger(RetrieveRequest.class.getName());
    private IReceived receiver;
    private String ContainerName = "cnt-co2";

    public RetrieveRequest(String containerName) {
        this.ContainerName = containerName;
    }
    public RetrieveRequest() {}
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }
```

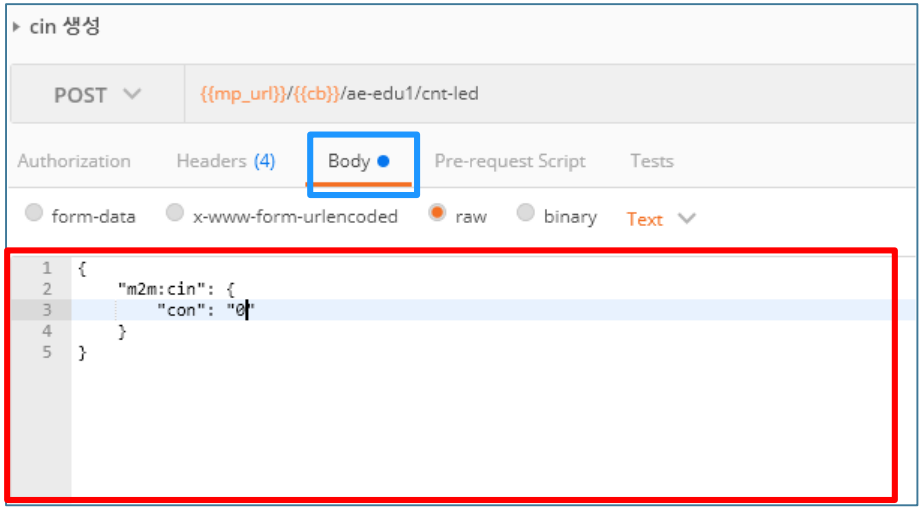
```
@Override
public void run() {
    try {
        String sb = csebase.getServiceUrl() + "/" + ServiceAEName + "/" + ContainerName + "/" + "latest";

        URL mUrl = new URL(sb);

        HttpURLConnection conn = (HttpURLConnection) mUrl.openConnection();
        conn.setRequestMethod("GET");
        conn.setDoInput(true);
        conn.setDoOutput(false);

        conn.setRequestProperty("Accept", "application/xml");
        conn.setRequestProperty("X-M2M-RI", "12345");
        conn.setRequestProperty("X-M2M-Origin", ae.getAEid() );
        conn.setRequestProperty("nmtype", "long");
        conn.connect();

        String strResp = "";
        BufferedReader in = new BufferedReader(new InputStreamReader(conn.getInputStream()));

        String strLine= "";
        while ((strLine = in.readLine()) != null) {
            strResp += strLine;
        }

        if ( strResp != "" ) {
            receiver.getResponseBody(strResp);
        }
        conn.disconnect();

    } catch (Exception exp) {
        LOG.log(Level.WARNING, exp.getMessage());
    }
  }
}
```

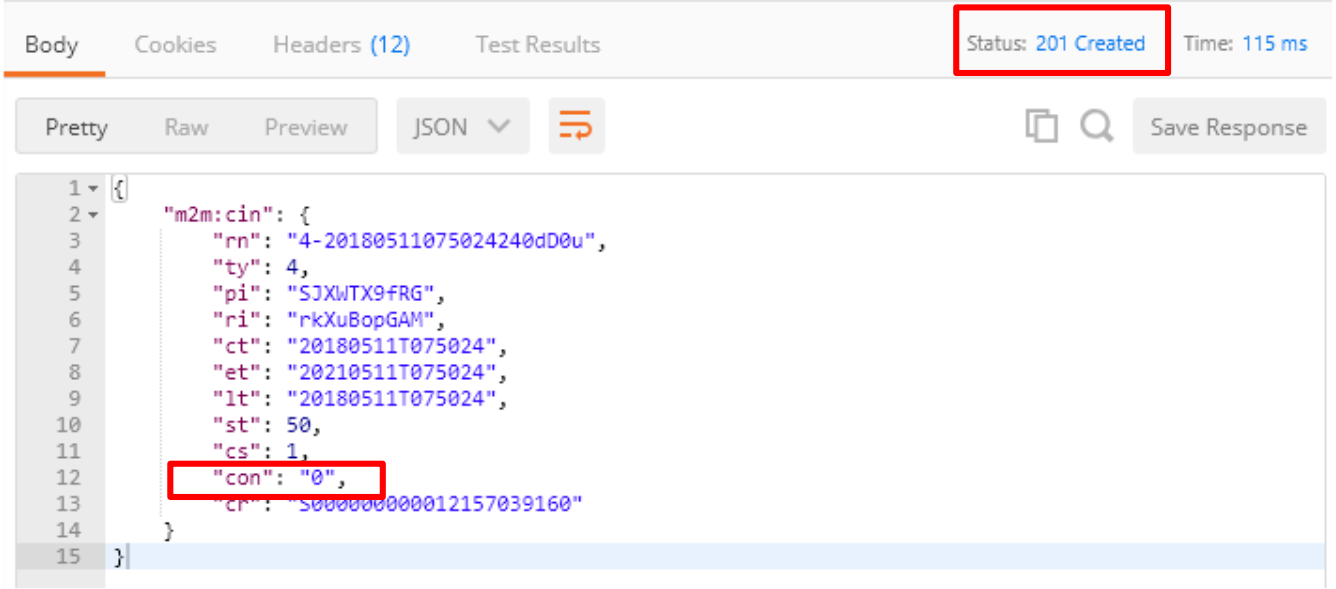
```
<?xml version="1.0" encoding="UTF-8" standalone="
<m2m:contentInstance xmlns:m2m="http://www.onem2m
    -instance" resourceName="4-20160928171758407
    <parentID>/mobius-yt/ae-edu1/cnt-co2</parentI
    <resourceType>4</resourceType>
    <creationTime>20160928T171758</creationTime>
    <resourceID>/mobius-yt/ae-edu1/cnt-co2/4-2016
    <lastModifiedTime>20160928T171758</lastModifi
    <stateTag>0</stateTag>
    <maxNrOfInstances>9007199254740991</maxNrOfIn
    <contentSize>3</contentSize>
    <content>461</content>
</m2m:contentInstance>
```

5 App Code Review

ControlRequest POSTMAN Script (LED Control)







1: Green LED ON

2: Green LED OFF

3: Blue LED ON

4: Blue LED OFF

5 App Code Review

ControlRequest Java Code Review

```
private String ServiceAEName = "ae-edu1";

...



public void GetAEInfo() {

    csebase.setInfo("192.168.0.135","7579","Mobius","1883");
    ...

    ...
}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<m2m:cin
    xmlns:m2m="http://www.onem2m.org/xml/protocols"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <cnf>text</cnf>
    <con>1</con>
</m2m:cin>
```

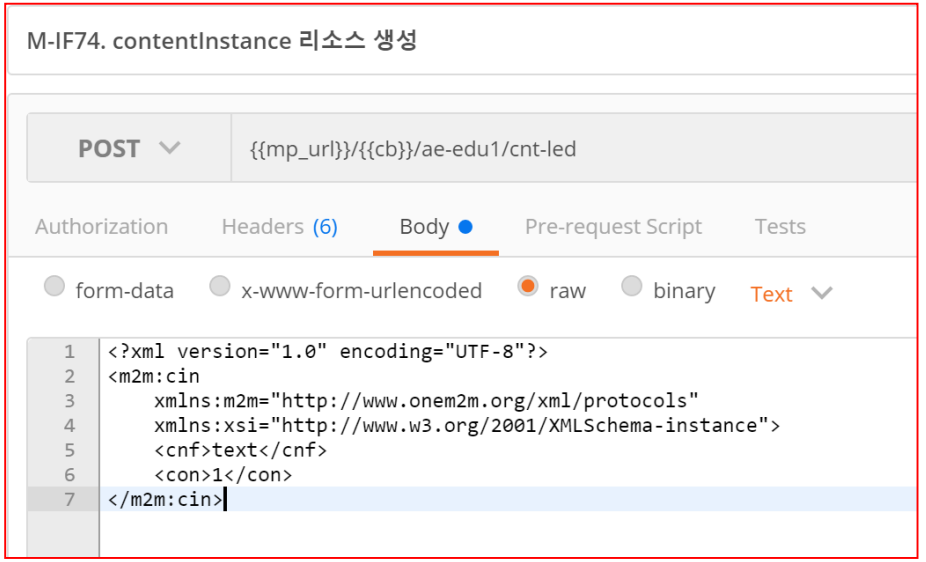
```
class ControlRequest extends Thread {
    private final Logger LOG = Logger.getLogger(ControlRequest.class.getName());
    private IReceived receiver;
    private String container_name = "cnt-led";

    public ContentInstanceObject contentinstance;
    public ControlRequest(String comm) {
        contentinstance = new ContentInstanceObject();
        contentinstance.setContent(comm);
    }
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }
```

```
    @Override
    public void run() {
        try {
            String sb = csebase.getServiceUrl() +"/" + ServiceAEName + "/" + container_name;

            URL mUrl = new URL(sb);

            HttpURLConnection conn = (HttpURLConnection) mUrl.openConnection();
            conn.setRequestMethod("POST");
            conn.setDoInput(true);
            conn.setDoOutput(true);
            conn.setUseCaches(false);
            conn.setInstanceFollowRedirects(false);

            conn.setRequestProperty("Accept", "application/xml");
            conn.setRequestProperty("Content-Type", "application/vnd.onem2m-res+xml;ty=4");
            conn.setRequestProperty("locale", "ko");
            conn.setRequestProperty("X-M2M-RI", "12345");
            conn.setRequestProperty("X-M2M-Origin", ae.getAEid() );

            String reqContent = contentinstance.makeXML();
            conn.setRequestProperty("Content-Length", String.valueOf(reqContent.length()));

            DataOutputStream dos = new DataOutputStream(conn.getOutputStream());
            dos.write(reqContent.getBytes());
            dos.flush();
            dos.close();

            BufferedReader in = new BufferedReader(new InputStreamReader(conn.getInputStream()));

            String resp = "";
            String strLine="";
            while ((strLine = in.readLine()) != null) {
                resp += strLine;
            }
            if (resp != "") {
                receiver.getResponseBody(resp);
            }
            conn.disconnect();

        } catch (Exception exp) {
            LOG.log(Level.SEVERE, exp.getMessage());
        }
    }
}
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes
<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protoc
    -20160928183913558u4ys">
    <ty>4</ty>
    <pi>/mobius-yt/ae-edu1/cnt-led</pi>
    <ri>/mobius-yt/ae-edu1/cnt-led/4-201609281839135
    <ct>20160928T183913</ct>
    <lt>20160928T183913</lt>
    <st>0</st>
    <cs>1</cs>
    <con>1</con>
    <cnf>text</cnf>
</m2m:cin>
```

5 App Code Review

ContentInstanceObject Java Code Review (nmtype - shot name, long name)




```
public class ContentInstanceObject {
    private String content = "";
    public void setContent(String contentValue) {
        this.content = contentValue; }

    public String makeXML() {
        String xml = "";

        xml += "<?xml version=₩"1.0₩" encoding=₩"UTF-8₩"?>";
        xml += "<m2m:cin ";
        xml += "xmlns:m2m=₩"http://www.onem2m.org/xml/protocols₩" ";
        xml += "xmlns:xsi=₩"http://www.w3.org/2001/XMLSchema-instance₩">";
        xml += "<cnf>text</cnf>";
        xml += "<con>" + content + "</con>";
        xml += "</m2m:cin>";

        return xml;
    }
}
```

Short Name Type Recommaned

5 App Code Review

AE Create POSTMAN Script [AE-ID]

M-IF50. AE 리소스 생성

POST {{mp_url}}/{{cb}}

Authorization | Headers (6) | Body ● | Pre-request Script | Tests

Accept	application/xml
locale	ko
X-M2M-RI	12345
X-M2M-Origin	S
Content-Type	application/vnd.onem2m-res+xml; ty=2
nmtype	short

POST {{mp_url}}/{{cb}} Params Send

Authorization | Headers (6) | Body ● | Pre-request Script | Tests

form-data | x-www-form-urlencoded | raw | binary | Text

```
<?xml version="1.0" encoding="UTF-8"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http
    /XMLSchema-instance" rn="ncubeapp">
  <api>0.2.481.2.0001.001.0001112</api>
  <rr>true</rr>
</m2m:ae>
```

Body | Cookies | Headers (10) | Tests — Status: 201 Created

Pretty | Raw | Preview | XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/
    /XMLSchema-instance" rn="ncubeapp">
    <ty>2</ty>
    <pi>/mobius-yt</pi>
    <ri>/mobius-yt/ncubeapp</ri>
    <ct>20160928T193850</ct>
    <lt>20160928T193850</lt>
    <api>0.2.481.2.0001.001.0001112</api>
    <rr>true</rr>
    <aei>ncubeapp</aei>
</m2m:ae>
```

Body | Cookies | Headers (9) | Tests — Status: 409 Conflict

Pretty | Raw | Preview | XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:rsp xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3
    /XMLSchema-instance">
    <cap>resource is already exist</cap>
</m2m:rsp>
```

5 App Code Review

AE Create Java Code Review

```
private String ServiceAEName = "ae-edu1";

...

public void GetAEInfo() {

    csebase.setInfo("192.168.0.135","7579","Mobius","1883");
    ae.setAppName("ncubeapp");

    ...
    ...
}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<m2m:ae
xmlns:m2m="http://www.onem2m.org/xml/protocols"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
rn="ncubeapp">
<api>0.2.481.2.0001.001.0001114</api>
<rr>true</rr>
</m2m:ae>
```

```
class aeCreateRequest extends Thread {
    private final Logger LOG = Logger.getLogger(aeCreateRequest.class.getName());
    String TAG = aeCreateRequest.class.getName();
    private IReceived receiver;
    int responseCode=0;
    public ApplicationEntityObject applicationEntity;
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }
    public aeCreateRequest(){
        applicationEntity = new ApplicationEntityObject();
        applicationEntity.setResourceName(ae.getappName());
    }
```

```
public void run() {
    try {
        String sb = csebase.getServiceUrl();
        URL mUrl = new URL(sb);
        HttpURLConnection conn = (HttpURLConnection) mUrl.openConnection();
        conn.setRequestMethod("POST");
        conn.setDoInput(true);
        conn.setDoOutput(true);
        conn.setUseCaches(false);
        conn.setInstanceFollowRedirects(false);
        conn.setRequestProperty("Content-Type", "application/vnd.onem2m-res+xml;ty=2");
        conn.setRequestProperty("Accept", "application/xml");
        conn.setRequestProperty("locale", "ko");
        conn.setRequestProperty("X-M2M-Origin", "S"+ ae.getappName());
        conn.setRequestProperty("X-M2M-RI", "12345");
        conn.setRequestProperty("X-M2M-NM", ae.getappName() );
        String reqXml = applicationEntity.makeXML();
        conn.setRequestProperty("Content-Length", String.valueOf(reqXml.length()));
        DataOutputStream dos = new DataOutputStream(conn.getOutputStream());
        dos.write(reqXml.getBytes());
        dos.flush();
        dos.close();
        responseCode = conn.getResponseCode();
        BufferedReader in = null;
        String aei = "";
        if (responseCode == 201) {
            // Get AEID from Response Data
            in = new BufferedReader(new InputStreamReader(conn.getInputStream()));
            String resp = "";
            String strLine;
            while ((strLine = in.readLine()) != null) {
                resp += strLine;
            }
            ParseElementXml pxml = new ParseElementXml();
            aei = pxml.GetElementXml(resp, "aei");
            ae.setAEid( aei );
            Log.d(TAG, "Create Get AEID[" + aei + "]");
            in.close();
        }
        if (responseCode != 0) {
            receiver.getResponseBody( Integer.toString(responseCode) );        }
        conn.disconnect();
    } catch (Exception exp) {
        LOG.log(Level.SEVERE, exp.getMessage());
    }
```

```
<?xml version="1.0" encoding="UTF-8" s
<m2m:ae xmlns:m2m="http://www.onem2m.o
    -instance" rn="ncubeapp">
    <pi>/mobius-yt</pi>
    <ty>2</ty>
    <ct>20160928T193850</ct>
    <ri>/mobius-yt/ncubeapp</ri>
    <lt>20160928T193850</lt>
    <api>0.2.481.2.0001.001.0001112</a
    <aei>ncubeapp</aei>
    <rr>true</rr>
</m2m:ae>
```

5 App Code Review

AE Retrieve POSTMAN Script [AE-ID]

M-IF56. AE 리소스 조회

GET {{mp_url}}/{{cb}}/ncubeapp Params Send Save

Authorization Headers (4) Body Pre-request Script Tests Code

Accept	application/xml
X-M2M-RI	12345
X-M2M-Origin	Sandroid
nmtype	short
key	value

Bulk Edit Presets

Body Cookies Headers (8) Tests Status: 200 OK Time: 121 ms

Pretty Raw Preview XML Save Response

```
1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001
       /XMLSchema-instance" rn="ncubeapp">
3      <pi>/mobius-yt</pi>
4      <ty>2</ty>
5      <ct>20160928T193850</ct>
6      <ri>/mobius-yt/ncubeapp</ri>
7      <lt>20160928T193850</lt>
8      <api>0.2.481.2.0001.001.0001112</api>
9      <aei>ncubeapp</aei>
10     <rr>true</rr>
11 </m2m:ae>
```

5 App Code Review

AE Retrieve Java Code Review

```
private String ServiceAEName = "ae-edu1";

...

public void GetAEInfo() {

    csebase.setInfo("192.168.0.135","7579","Mobius","1883");
    ae.setAppName("ncubeapp");

    ...

    ...
}
```

```
/* Retrieve AE-ID */
class aeRetrieveRequest extends Thread {
    private final Logger LOG = Logger.getLogger(aeCreateRequest.class.getName());
    private IReceived receiver;
    int responseCode=0;

    public aeRetrieveRequest() {
    }
    public void setReceiver(IReceived hanlder) {
        this.receiver = hanlder;
    }
```

```
    @Override
public void run() {
    try {
        String sb = csebase.getServiceUrl()+"/"+ ae.getappName();
        URL mUrl = new URL(sb);

        HttpURLConnection conn = (HttpURLConnection) mUrl.openConnection();
        conn.setRequestMethod("GET");
        conn.setDoInput(true);
        conn.setDoOutput(false);

        conn.setRequestProperty("Accept", "application/xml");
        conn.setRequestProperty("X-M2M-RI", "12345");
        conn.setRequestProperty("X-M2M-Origin", "Sandoroid");
        conn.setRequestProperty("nmtype", "short");
        conn.connect();

        responseCode = conn.getResponseCode();

        BufferedReader in = null;
        String aei = "";
        if (responseCode == 200) {
            // Get AEID from Response Data
            in = new BufferedReader(new InputStreamReader(conn.getInputStream()));

            String resp = "";
            String strLine;
            while ((strLine = in.readLine()) != null) {
                resp += strLine;
            }

            ParseElementXml pxml = new ParseElementXml();
            aei = pxml.GetElementXml(resp, "aei");
            ae.setAEid( aei );
            Log.d(TAG, "Retrieve Get AEID[" + aei + "]");
            in.close();
        }
        if (responseCode != 0) {
            receiver.getResponseBody( Integer.toString(responseCode) );
        }
        conn.disconnect();
    } catch (Exception exp) {
        LOG.log(Level.SEVERE, exp.getMessage());
    }
}
```

```
<?xml version="1.0" encoding="UTF-8" s
<m2m:ae xmlns:m2m="http://www.onem2m.o
    -instance" rn="ncubeapp">
    <pi>/mobius-yt</pi>
    <ty>2</ty>
    <ct>20160928T193850</ct>
    <ri>/mobius-yt/ncubeapp</ri>
    <lt>20160928T193850</lt>
    <api>0.2.481.2.0001.001.0001112</a
    <aei>ncubeapp</aei>
    <rr>true</rr>
</m2m:ae>
```

5 App Code Review

Subscription Resource Create POSTMAN Script

M-IF58. subscription 리소스 생성

POST {{mp_url}}/{{cb}}/ae-edu1/cnt-co2

Authorization Headers (6) Body Pre-request Script Tests

Accept	application/xml
X-M2M-RI	12345
X-M2M-Origin	Sxxxx
Content-Type	application/vnd.onem2m-res+xml; ty=23
locale	ko
nmtype	short
key	value

M-IF58. subscription 리소스 생성

POST {{mp_url}}/{{cb}}/ae-edu1/cnt-co2 Params

Authorization Headers (6) Body Pre-request Script Tests

form-data x-www-form-urlencoded raw binary Text

```
<?xml version="1.0" encoding="UTF-8"?>
<m2m:sub
    xmlns:m2m="http://www.onem2m.org/xml/protocols"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="ncubeapp_rn">
    <enc>
        <net>3</net>
    </enc>
    <nu>mqtt://192.168.25.47/ncubeapp_rn</nu>
</m2m:sub>
```

Body Cookies Headers (10) Tests Status: 201 Created Time: 150 ms

Pretty Raw Preview XML Save Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:sub xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn
    ="ncubeapp_rn">
    <pi>/mobius-yt/ae-edu1/cnt-co2</pi>
    <ty>23</ty>
    <ct>20160928T171145</ct>
    <ri>/mobius-yt/ae-edu1/cnt-co2/ncubeapp_rn</ri>
    <lt>20160928T171145</lt>
    <st>0</st>
    <enc>
        <net>3</net>
    </enc>
    <nu>mqtt://192.168.25.47/ncubeapp_rn</nu>
    <nct>2</nct>
    <cr>ncubeapp</cr>
</m2m:sub>
```

5 App Code Review

Subscription Resource Create Java Code Review

```
private String ServiceAEName = "ae-edu1";

public void GetAEInfo() {

    csebase.setInfo("192.168.0.135","7579","Mobius","1883");
    ae.setAppName("ncubeapp");
    ...
}

/* MQTT Subscription */
public void MQTT_Create(boolean mtqqStart) {
    if (mtqqStart && mqttClient == null) {
        /* Subscription Resource Create to Yellow Turtle */
        SubscribeResource subcribeResource = new SubscribeResource();
        subcribeResource.setReceiver(new IReceived() {
            public void getResponseBody(final String msg) {
                handler.post(new Runnable() {
                    public void run() {
                        ...
                    }
                });
            }
        });
        subcribeResource.
        ...
}

 /* Subscribe Co2 Content Resource */
class SubscribeResource extends Thread {
    private IReceived receiver;
    private String container_name = "cnt-co2"; //change to control container name

    public ContentSubscribeObject subscribeInstance;
    public SubscribeResource() {
        subscribeInstance = new ContentSubscribeObject();
        subscribeInstance.setUrl(csebase.getHost());
        subscribeInstance.setResourceName(ae.getAEid()+"_rn");
        subscribeInstance.setPath(ae.getAEid()+"_sub");
        subscribeInstance.setOrigin_id(ae.getAEid());
    }
    public void setReceiver(IReceived hanlder) { this.receiver = hanlder; }
```

```
<?xml version="1.0" encoding="UTF-8"?>
<m2m:sub
    xmlns:m2m="http://www.onem2m.org/xml/protocols"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="ncubeapp_rn">
    <enc>
        <net>3</net>
    </enc>
    <nu>mqtt://192.168.25.47/ncubeapp_rn</nu>
</m2m:sub>
```

```
public void run() {
    try {
        String sb = csebase.getServiceUrl() + "/" + ServiceAEName + "/" + container_name;
        URL mUrl = new URL(sb);

        HttpURLConnection conn = (HttpURLConnection) mUrl.openConnection();
        conn.setRequestMethod("POST");
        conn.setDoInput(true);
        conn.setDoOutput(true);
        conn.setUseCaches(false);
        conn.setInstanceFollowRedirects(false);

        conn.setRequestProperty("Accept", "application/xml");
        conn.setRequestProperty("Content-Type", "application/vnd.onem2m-res+xml; ty=23");
        conn.setRequestProperty("locale", "ko");
        conn.setRequestProperty("X-M2M-RI", "12345");
        conn.setRequestProperty("X-M2M-Origin", ae.getAEid());

        String reqmqttContent = subscribeInstance.makeXML();
        conn.setRequestProperty("Content-Length", String.valueOf(reqmqttContent.length()));

        DataOutputStream dos = new DataOutputStream(conn.getOutputStream());
        dos.write(reqmqttContent.getBytes());
        dos.flush();
        dos.close();

        BufferedReader in = new BufferedReader(new InputStreamReader(conn.getInputStream()));

        String resp = "";
        String strLine="";
        while ((strLine = in.readLine()) != null) {
            resp += strLine;
        }

        if (resp != "") {
            receiver.getResponseBody(resp);
        }
        conn.disconnect();

    } catch (Exception exp) {
        LOG.log(Level.SEVERE, exp.getMessage());
    }
  }
}
```

5 App Code Review

MQTT Client Subscribe, Listener, Callback , Response Java Code Review

```
private String ServiceAEName = "ae-edu1";

public void GetAEInfo() {

    csebase.setInfo("192.168.0.135","7579","Mobius","1883");
    ae.setAppName("ncubeapp");
    ...
    MQTT_Req_Topic = "/oneM2M/req/mobius-yt/"+ae.getAEid()+"_sub"+"/#";
    MQTT_Resp_Topic = "/oneM2M/resp/mobius-yt/"+ae.getAEid()+"_sub"+"/json";
}

/* MQTT Subscription */
public void MQTT_Create(boolean mtqqStart) {
    if (mtqqStart && mqttClient == null) {
        /* Subscription Resource Create to Yellow Turtle */
        SubscribeResource subcribeResource = new SubscribeResource();
        subcribeResource.setReceiver(new IReceived() {
        ...
        subcribeResource.start();

        /* MQTT Subscribe */
        mqttClient = new MqttAndroidClient(this.getApplicationContext(), "tcp://" +
csebase.getHost() + ":" + csebase.getMQTTPort(), MqttClient.generateClientId());
        mqttClient.setCallback(mainMqttCallback);
        try {
            IMqttToken token = mqttClient.connect();
            token.setActionCallback(mainIMqttActionListener);
        } catch (MqttException e) {
            e.printStackTrace();
        }
    } else {
        /* MQTT unSubscribe or Client Close */
        mqttClient.setCallback(null);
        mqttClient.close();
        mqttClient = null;
    }
}
```

```
/* MQTT Listener */
private IMqttActionListener mainIMqttActionListener = new IMqttActionListener() {
    @Override
    public void onSuccess(IMqttToken asyncActionToken) {
        String payload = "";
        int mqttQos = 1; /* 0: NO QoS, 1: No Check , 2: Each Check */

        MqttMessage message = new MqttMessage(payload.getBytes());
        try {
            mqttClient.subscribe(MQTT_Req_Topic, mqttQos);
        } catch (MqttException e) {
            e.printStackTrace();
        }
    }
    ...
};
/* MQTT Broker Message Received */
private MqttCallback mainMqttCallback = new MqttCallback() {
    @Override
    public void connectionLost(Throwable cause) {  Log.d(TAG, "connectionLost"); }
    @Override
    public void messageArrived(String topic, MqttMessage message) throws Exception {
        Log.d(TAG, "messageArrived");
        textViewData.setText("");
        textViewData.setText("**** MQTT CO2 실시간 조회 ****₩r₩n₩r₩n" +
message.toString().replaceAll("","", "₩n"));
        /* Json Type Response Parsing */
        String retrqi = MqttClientRequestParser.notificationJsonParse(message.toString());
        String responseMessage = MqttClientRequest.notificationResponse(retrqi);

        /* Make xml for MQTT Response Message */
        MqttMessage resmessage = new MqttMessage(responseMessage.getBytes());

        try {
            mqttClient.publish(MQTT_Resp_Topic, resmessage);
        } catch (MqttException e) { e.printStackTrace(); }
    }
    @Override
    public void deliveryComplete(IMqttDeliveryToken token) {
        Log.d(TAG, "deliveryComplete");
    }
};
```

1	Android 소개
2	Android Studio 개발 환경 구성
3	Web View App 실습
4	개방형 IoT Platform 연동 App Review
5	App Code Review
6	Generate APK

6 Generate APK

Android Studio Generate APK

1. 메뉴 Build -> Generate Signed APK...

2.1. Key store가 있다면 Choose existing 선택

2.2. Key store가 없다면 Create new 선택

2.2.1. 저장위치 및 파일명 입력,
비밀번호 입력, Alias, Password, Confirm 입력

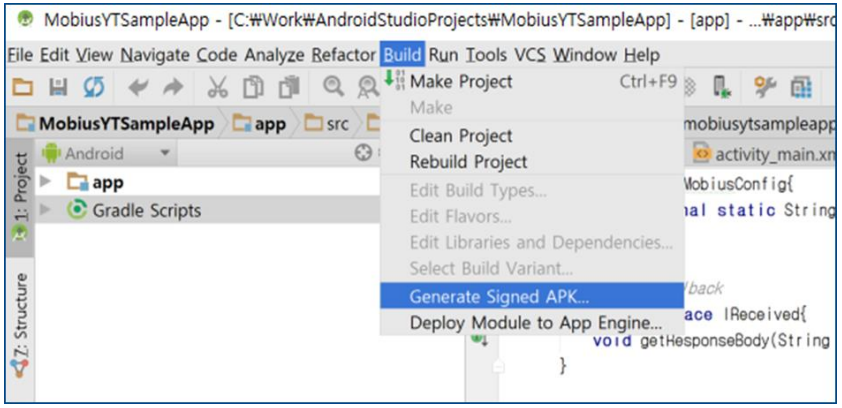
2.2.2. 필요하면 Certificate 에 인증 정보를 입력

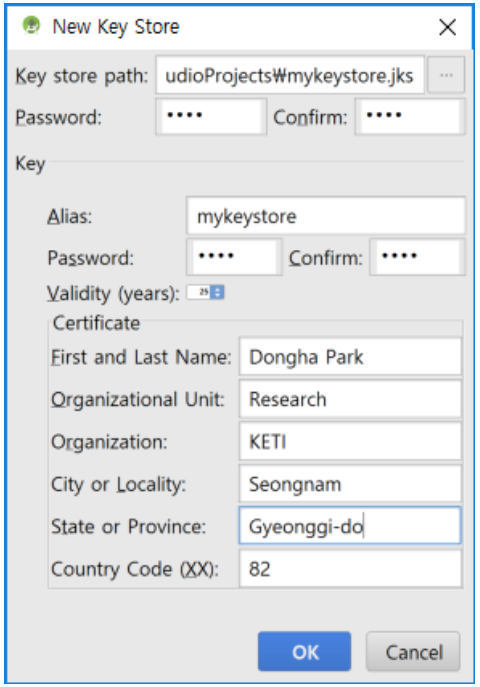
3. Key store password, alias, key password 입력.

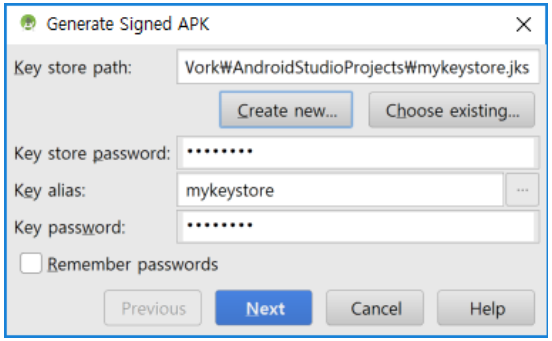
(new로 만들었다면 필요한 정보를 채워줌)

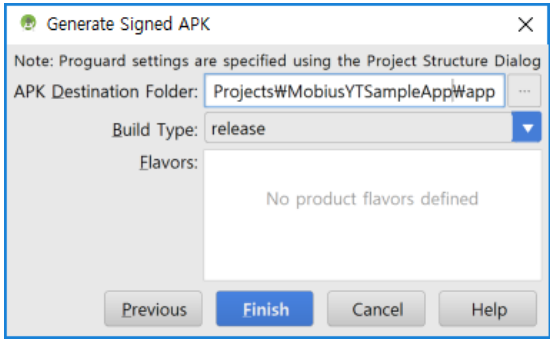
4. 디컴파일 방지가 필요하면 ProGuard 체크 아니면 Finish.

5. Show in Explorer 로 생성된 파일을 확인







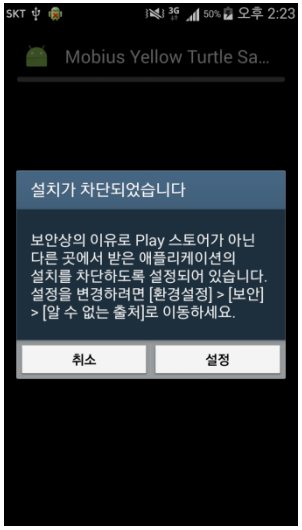


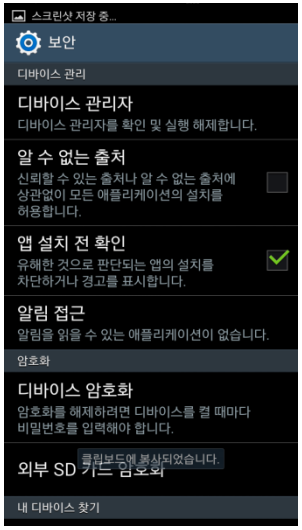
6 Generate APK

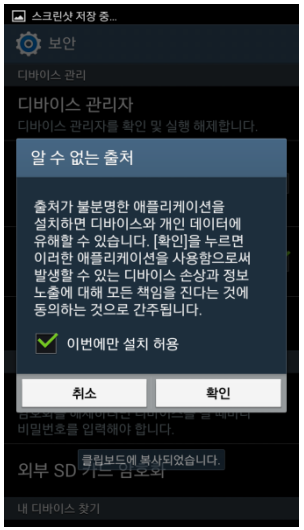
Android Studio Generate APK Install

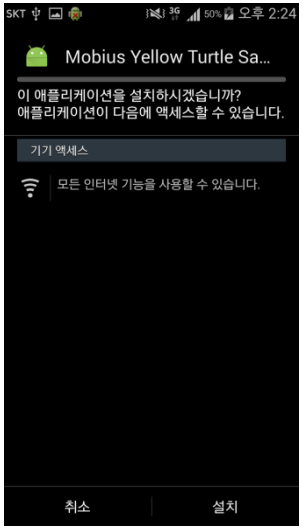
탐색기를 이용하여 생성된 APK 파일을 스마트폰에 옮김.

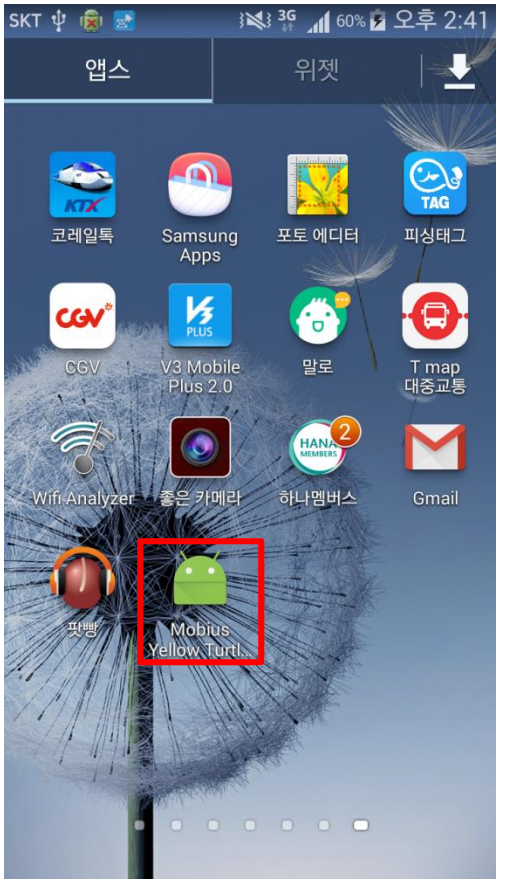
스마트폰에서 내 파일 App을 실행하여 옮겨진 APK 파일을 실행











Thank you

문 의 : 박동하 책임 araha@keti.re.kr

031-789-7580