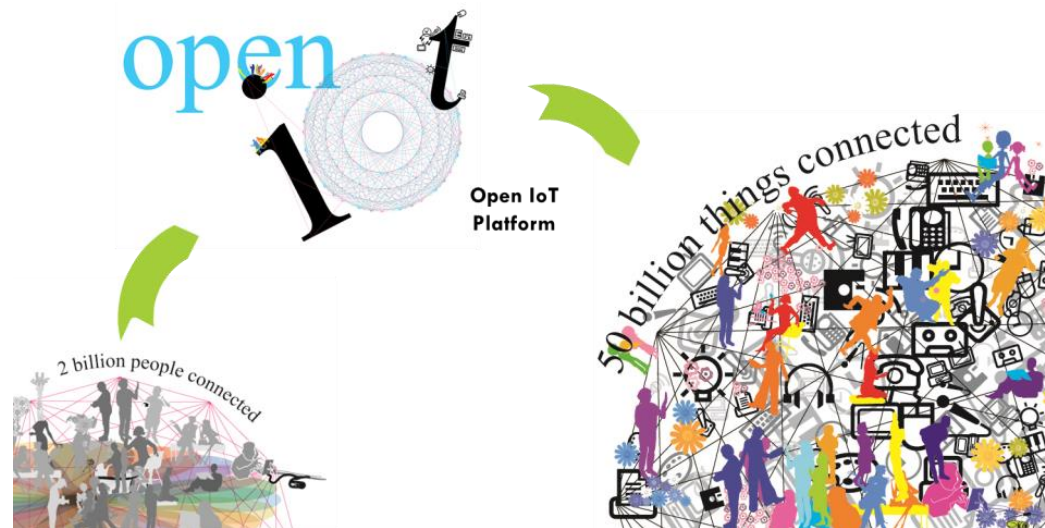


oneM2M Application AndroidSample UserGuide

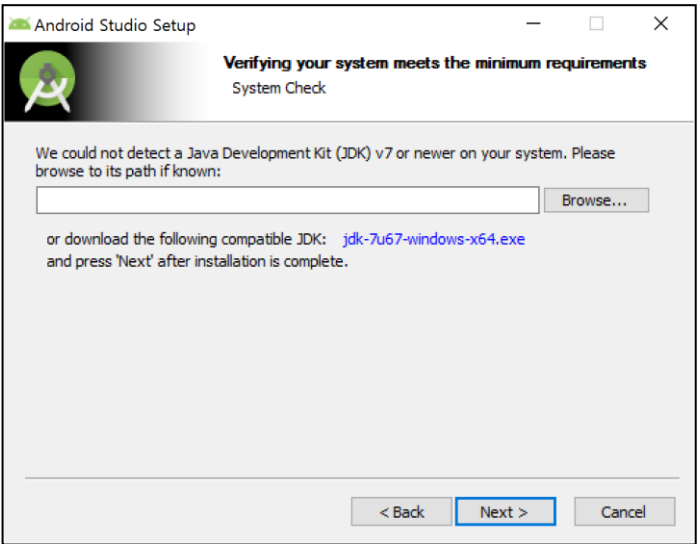


KETI (Korea Electronic Technology Institute)
(araha@keti.re.kr)

Android Studio Installation

■ Android Studio download and installation

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



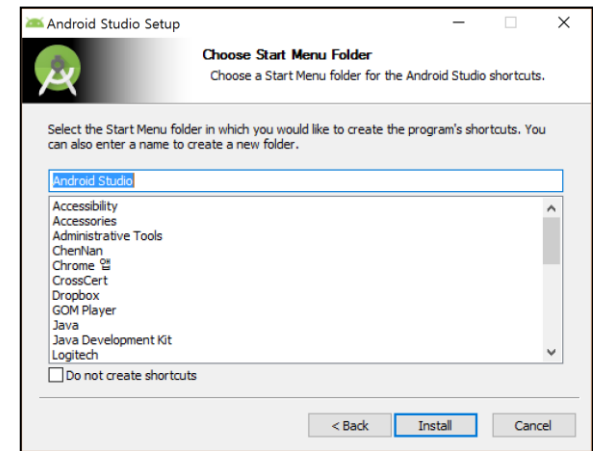
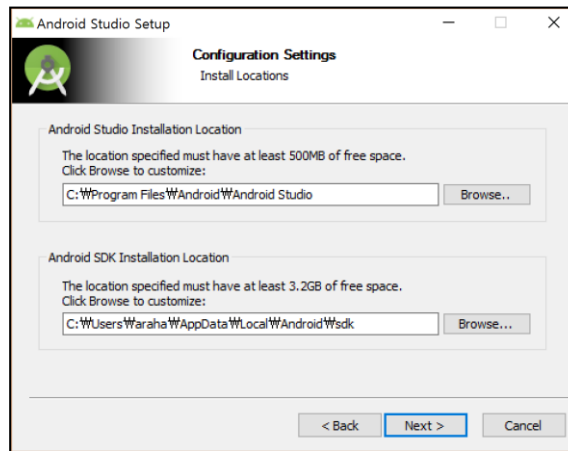
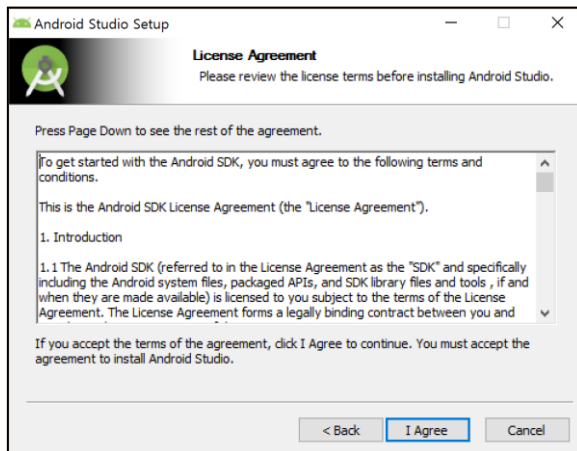
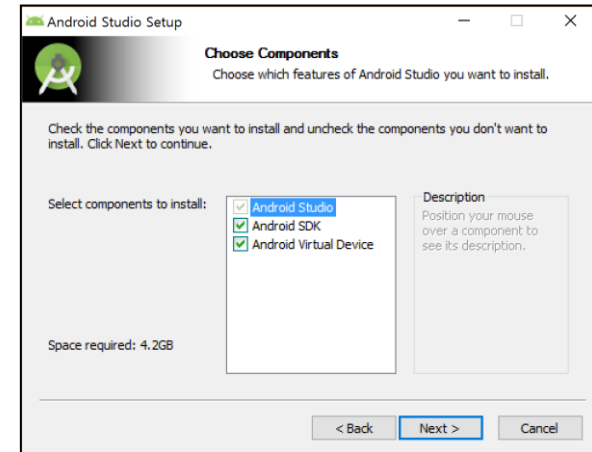
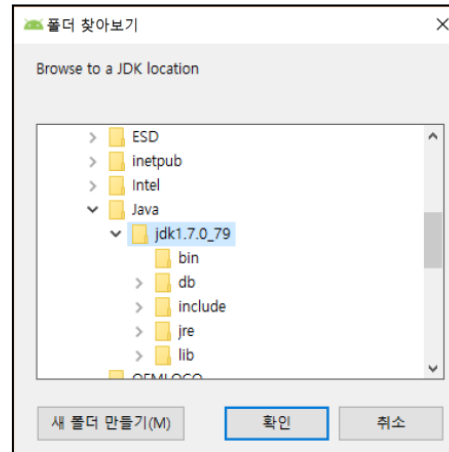
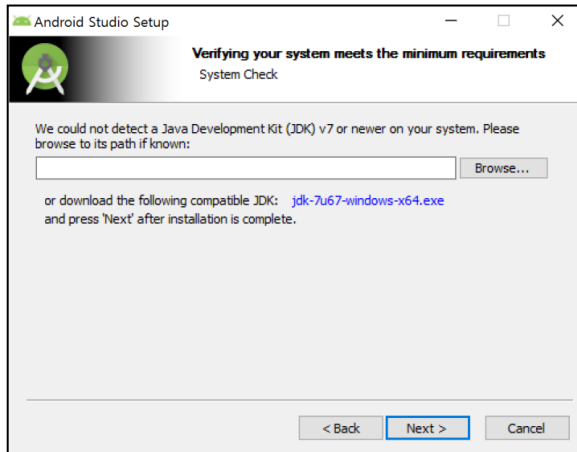
Java SE Development Kit 7u79		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
☐ Accept License Agreement ☒ Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	130.4 MB	jdk-7u79-linux-i586.rpm
Linux x86	147.6 MB	jdk-7u79-linux-i586.tar.gz
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

Android Studio Installation

■ Android Studio download and installation

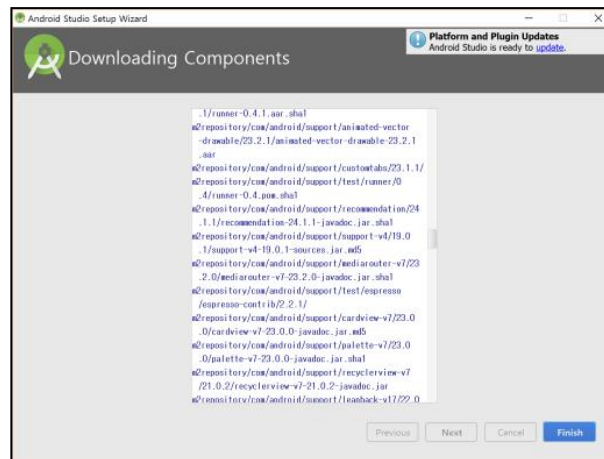
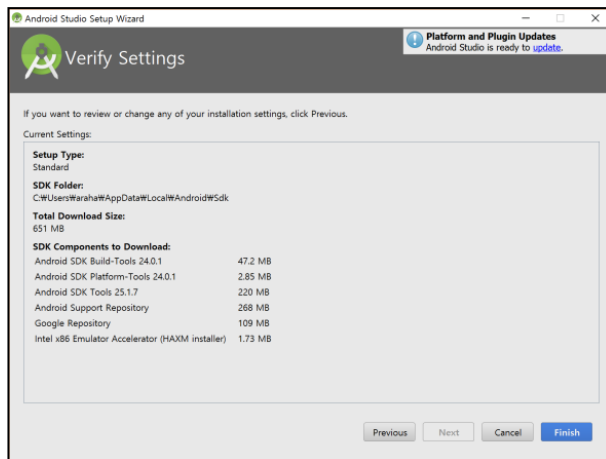
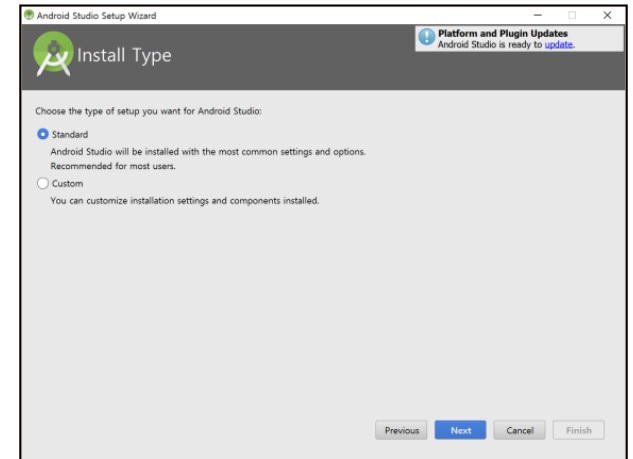
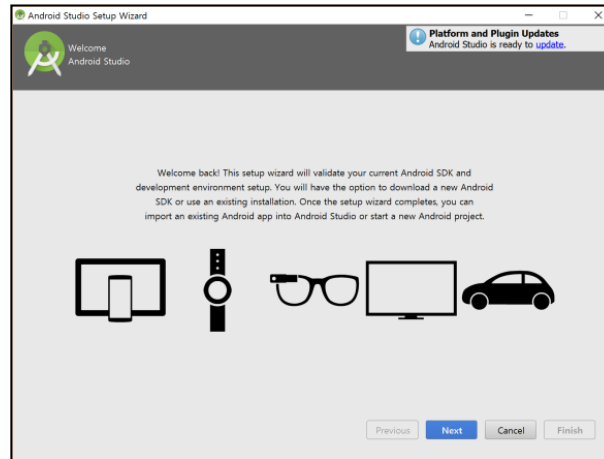
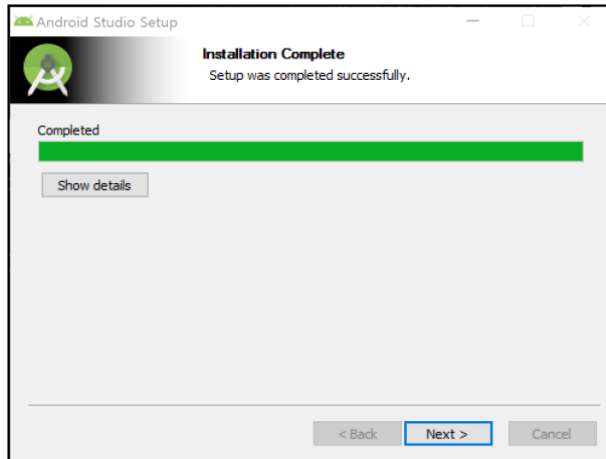
<https://developer.android.com/sdk/index.html> (Android Studio Install)



Android Studio Installation

■ Android Studio download and installation

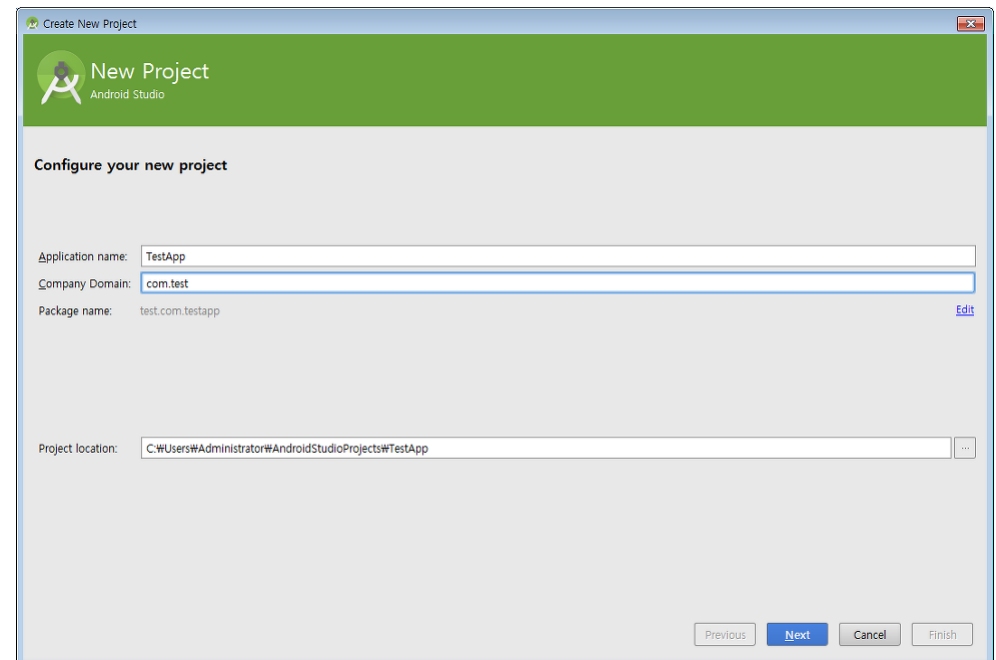
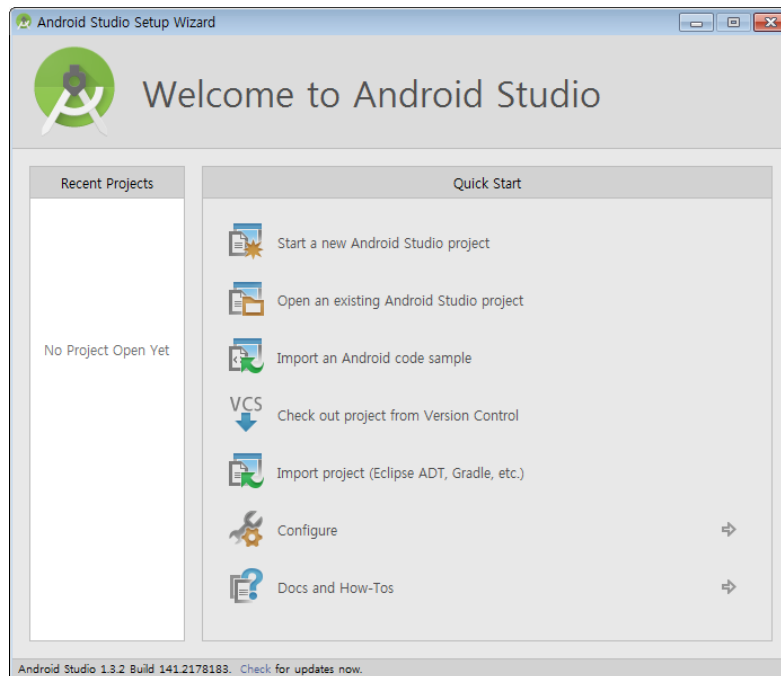
<https://developer.android.com/sdk/index.html> (Android Studio Install)



Android Studio Installation

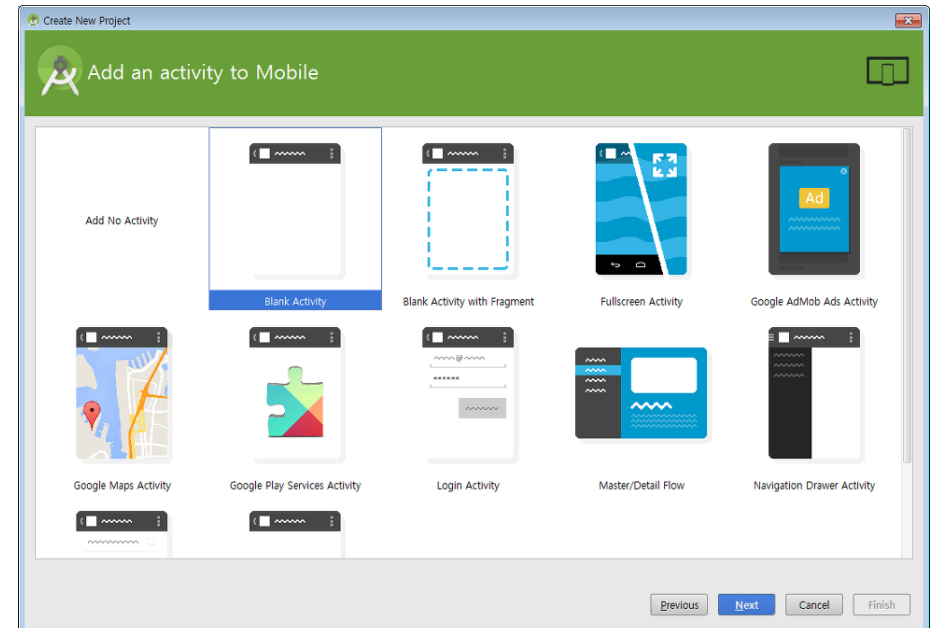
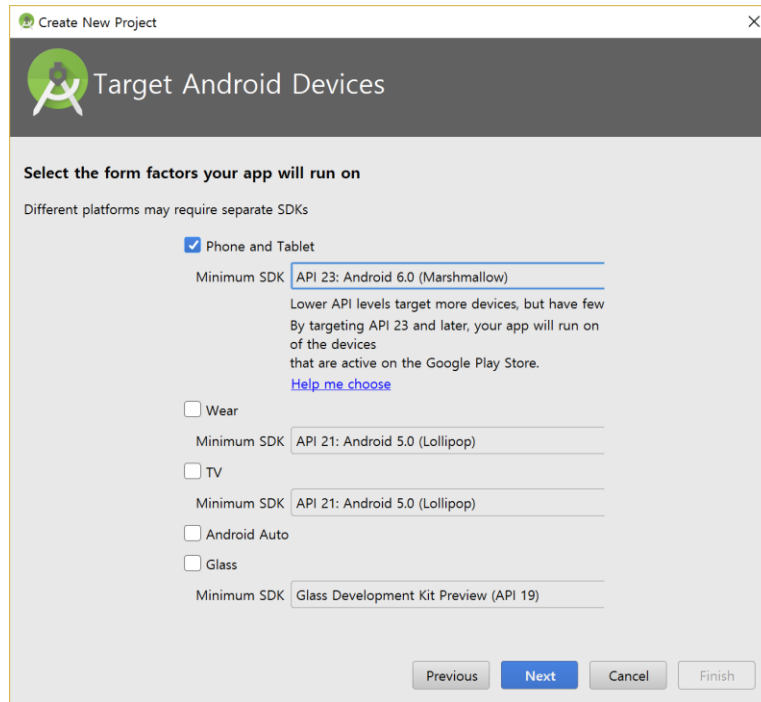
■ Execute Android Studio

- Click “Start a new Android Studio project”.
- Type the application name and company domain



Android Studio Installation

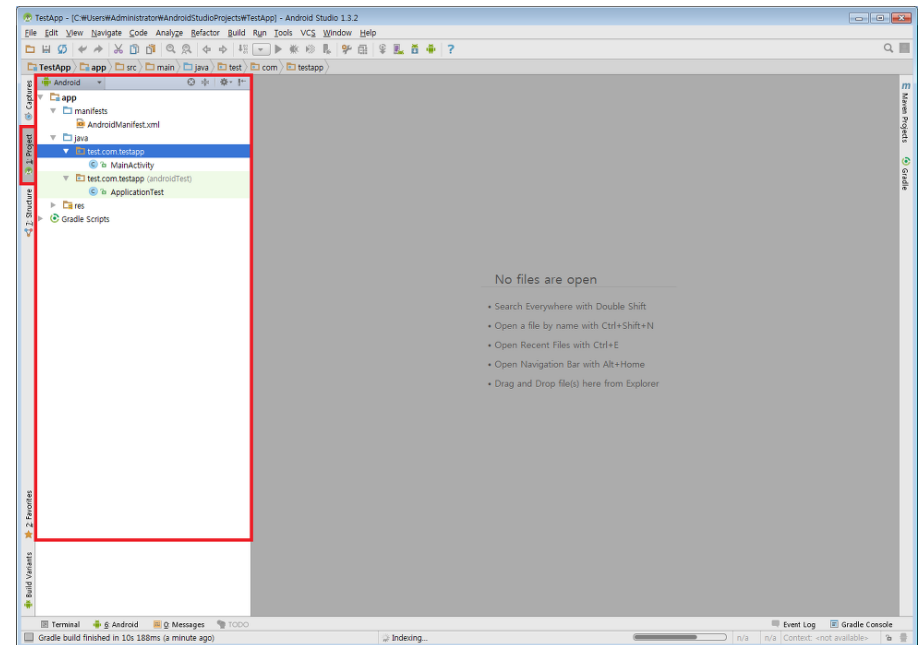
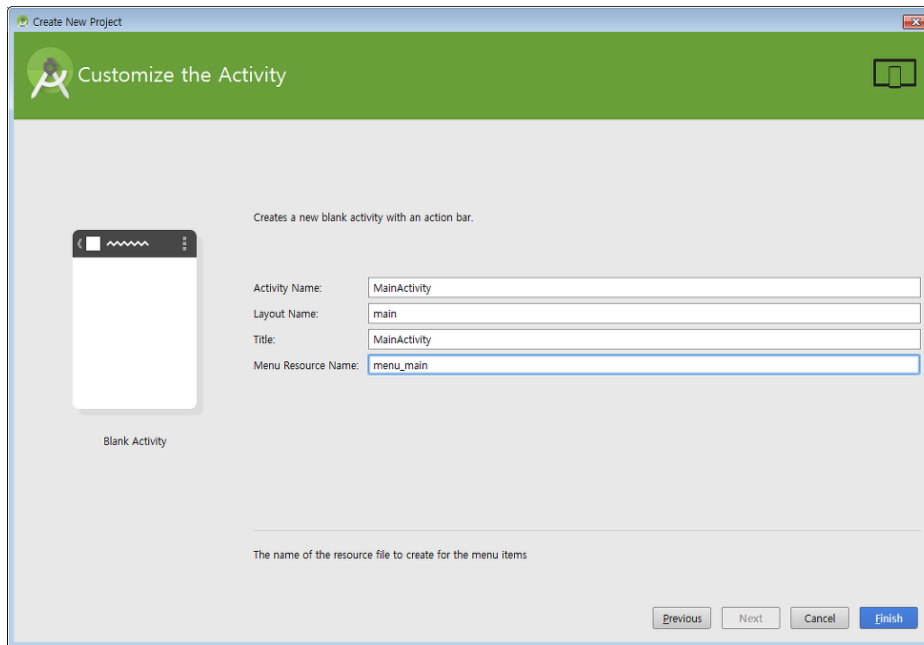
- Configure ADK version and activity for targeting android devices
 - Change Minimum SDK to API 23
 - Select “Blank Activity”



Android Studio Installation

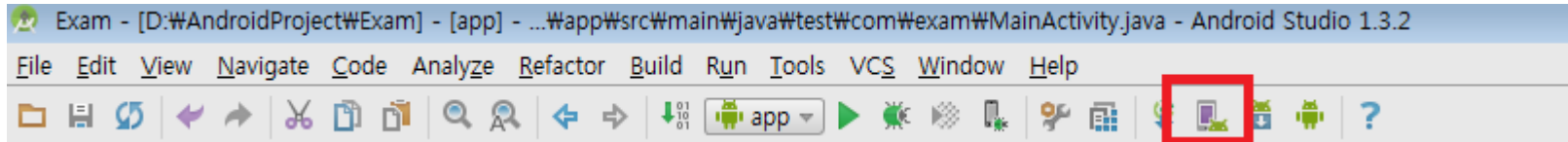
■ Customize the activity

- Input activity name and click “Finish” button.
- Right menu show the project file structure.



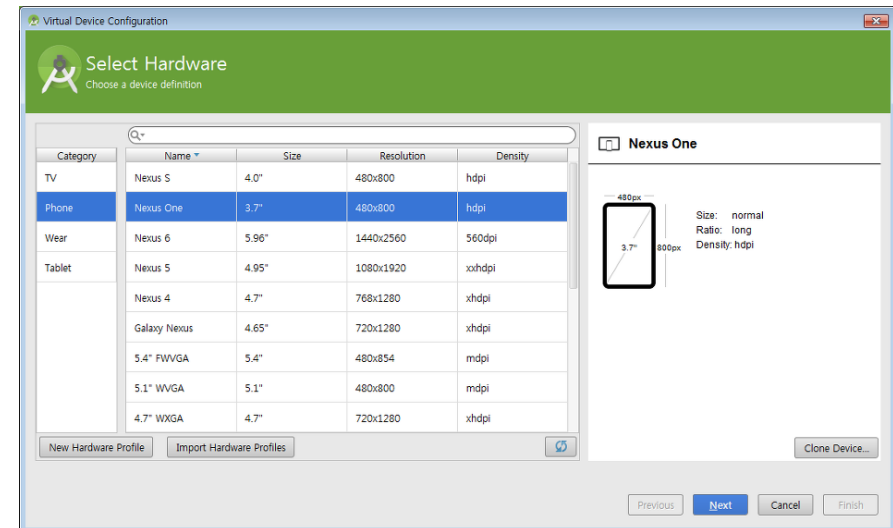
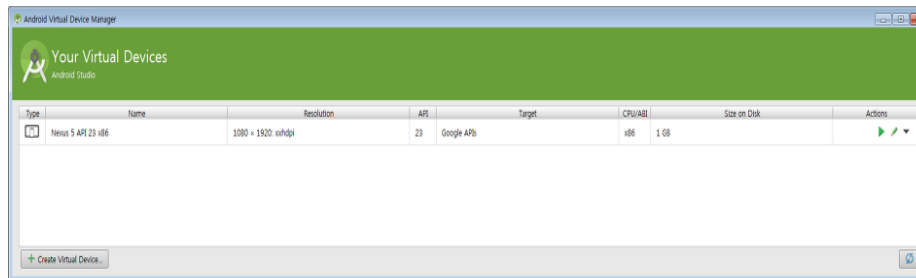
Android Studio Installation

■ Android Studio Virtual Devices



■ Click “AVD Manager” menu

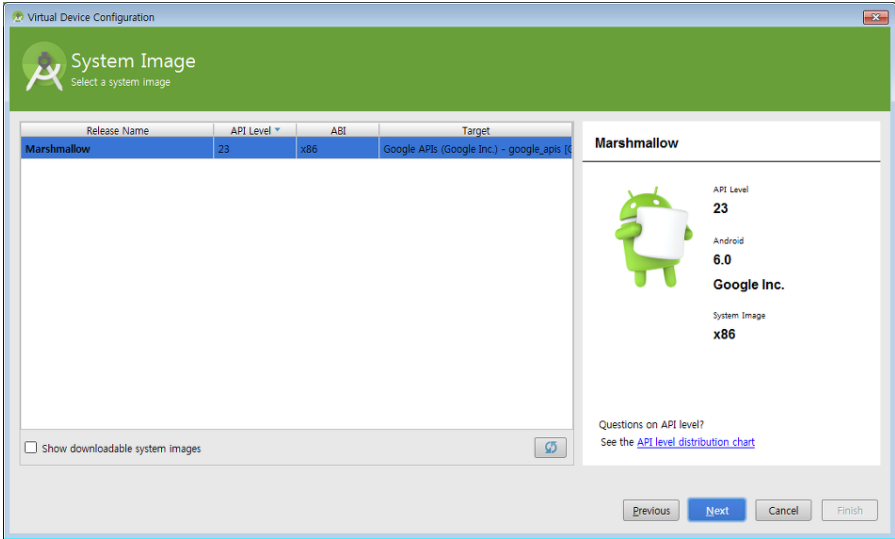
1. The default virtual device is Nexus 5
2. [Add a new virtual device]
3. Click “Create Virtual Device”
4. Select a the phone type as VD



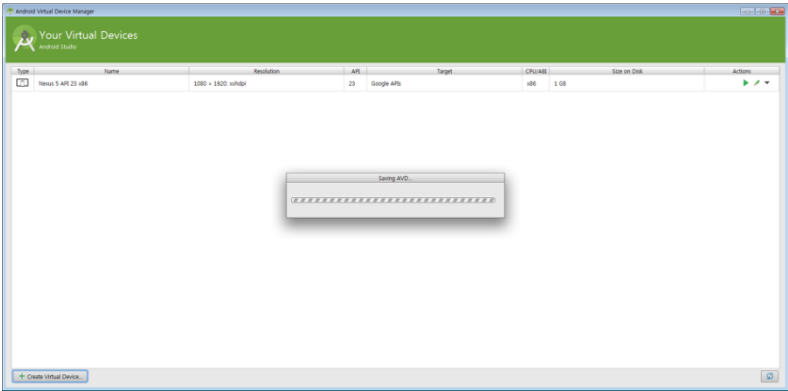
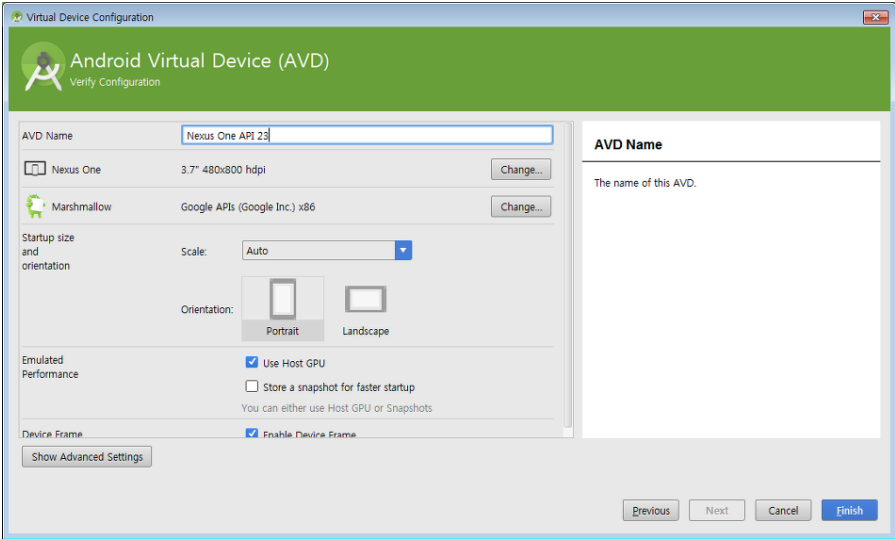
Android Studio Installation

■ Android Studio Virtual Devices

Select OS image



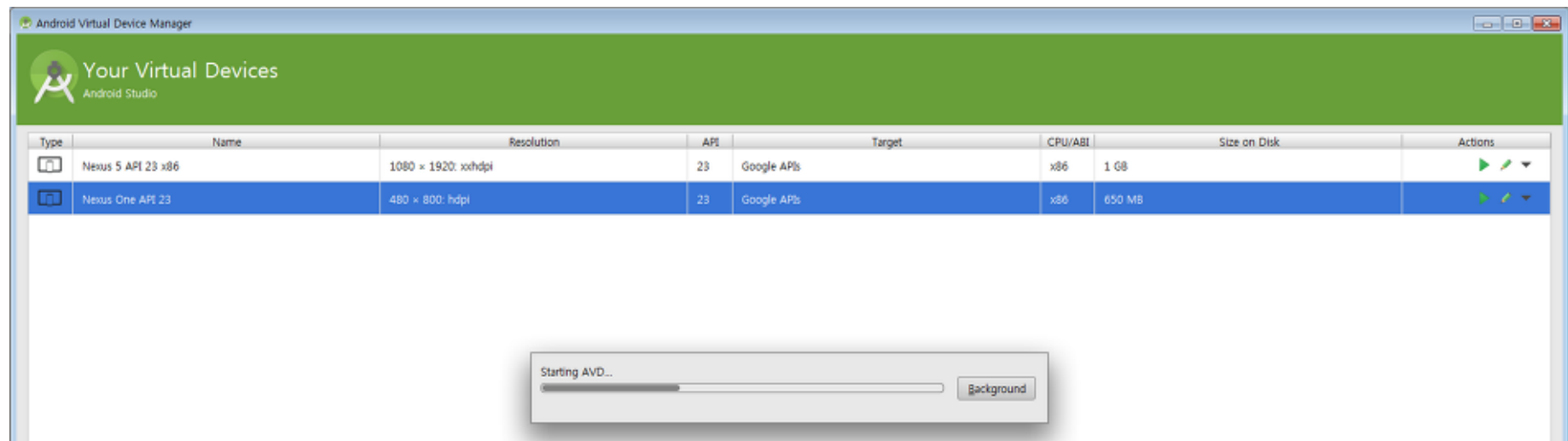
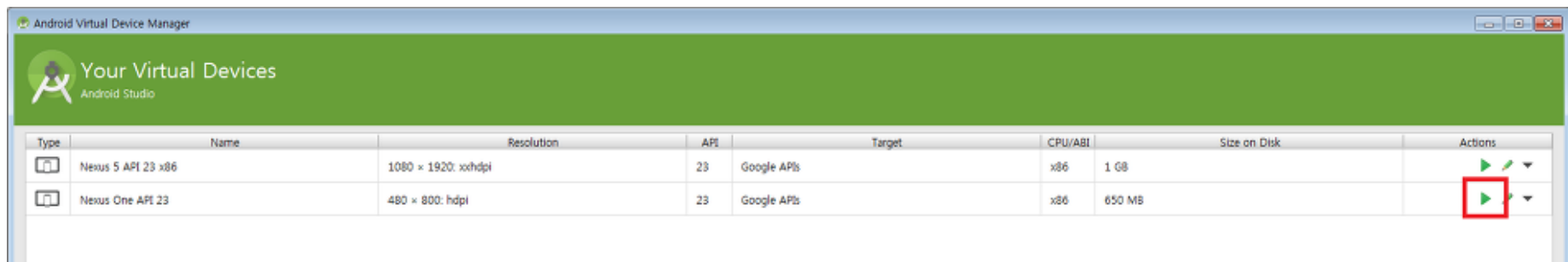
Android Virtual Device(AVD) Configuration



Android Studio Installation

■ Android Studio Virtual Devices

- Click start button to turn on the VD



Android Studio Installation

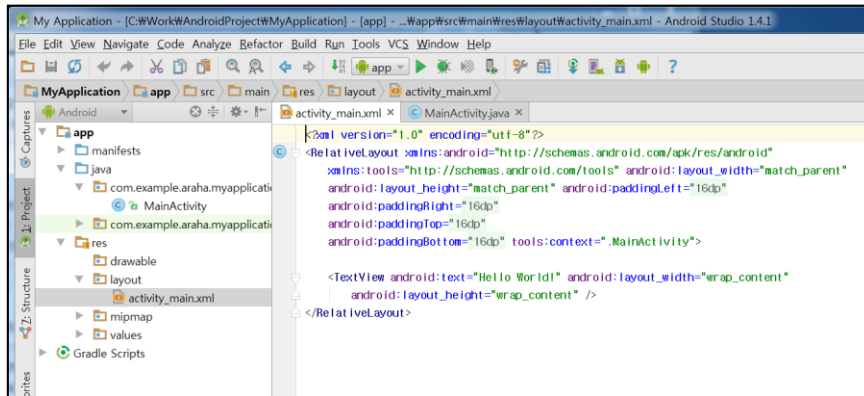
- Android Studio Virtual Devices
 - Wait for the VD system loading



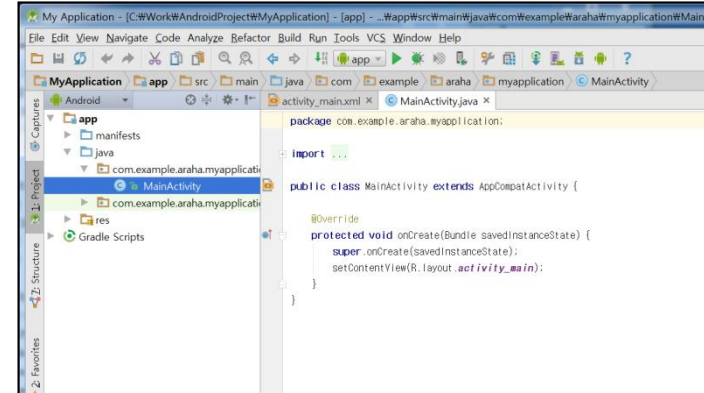
Web View APP

■ Basic Default Code

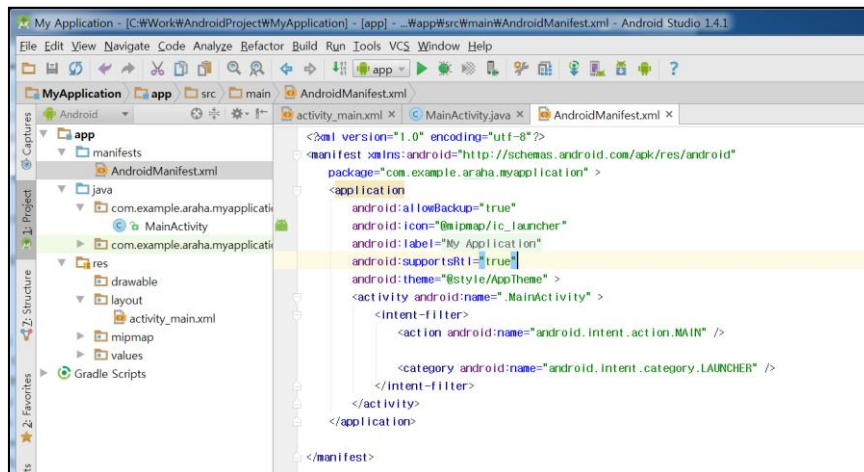
activity_main.xml



MainActivity.Java



AndroidManifest.xml



AndroidManifest?

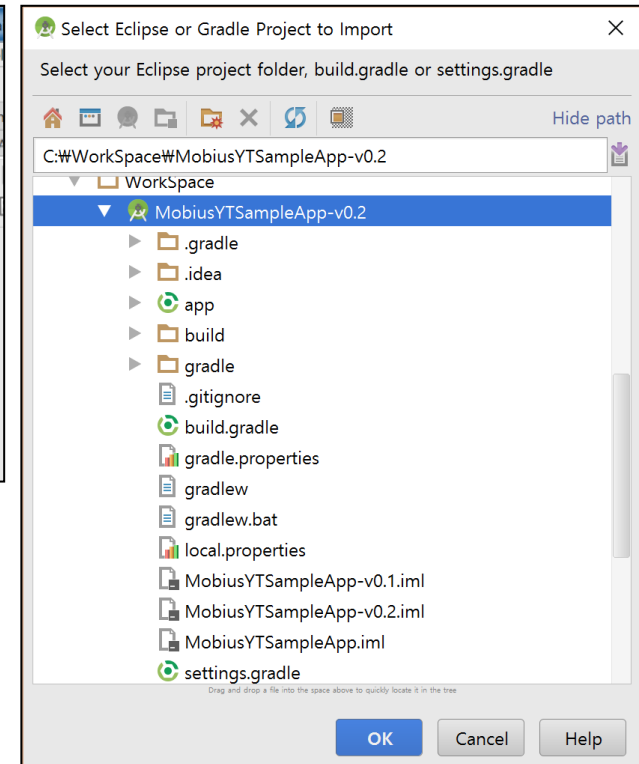
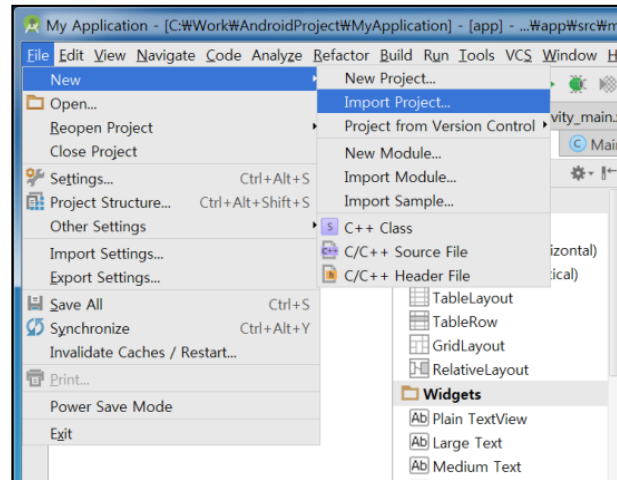
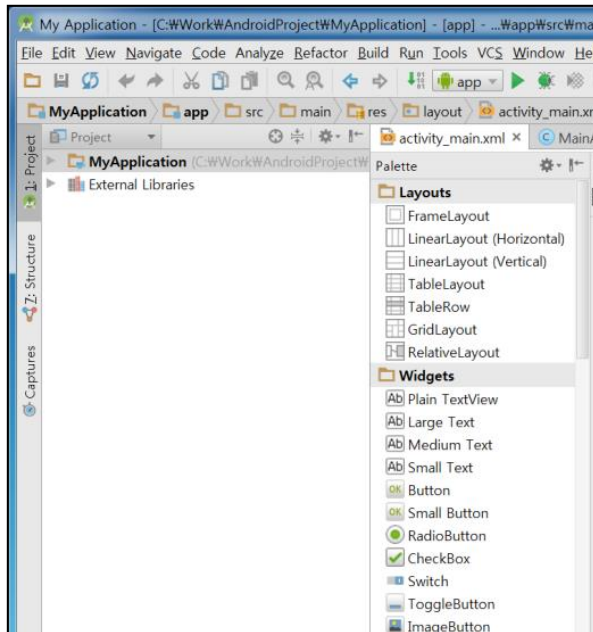
Descript and configure the android project

- App name, ID, icon, theme
- App access right
- Activity, Service, Intent

Mobius and APP connectivity

■ Import sample project

Android Project Import

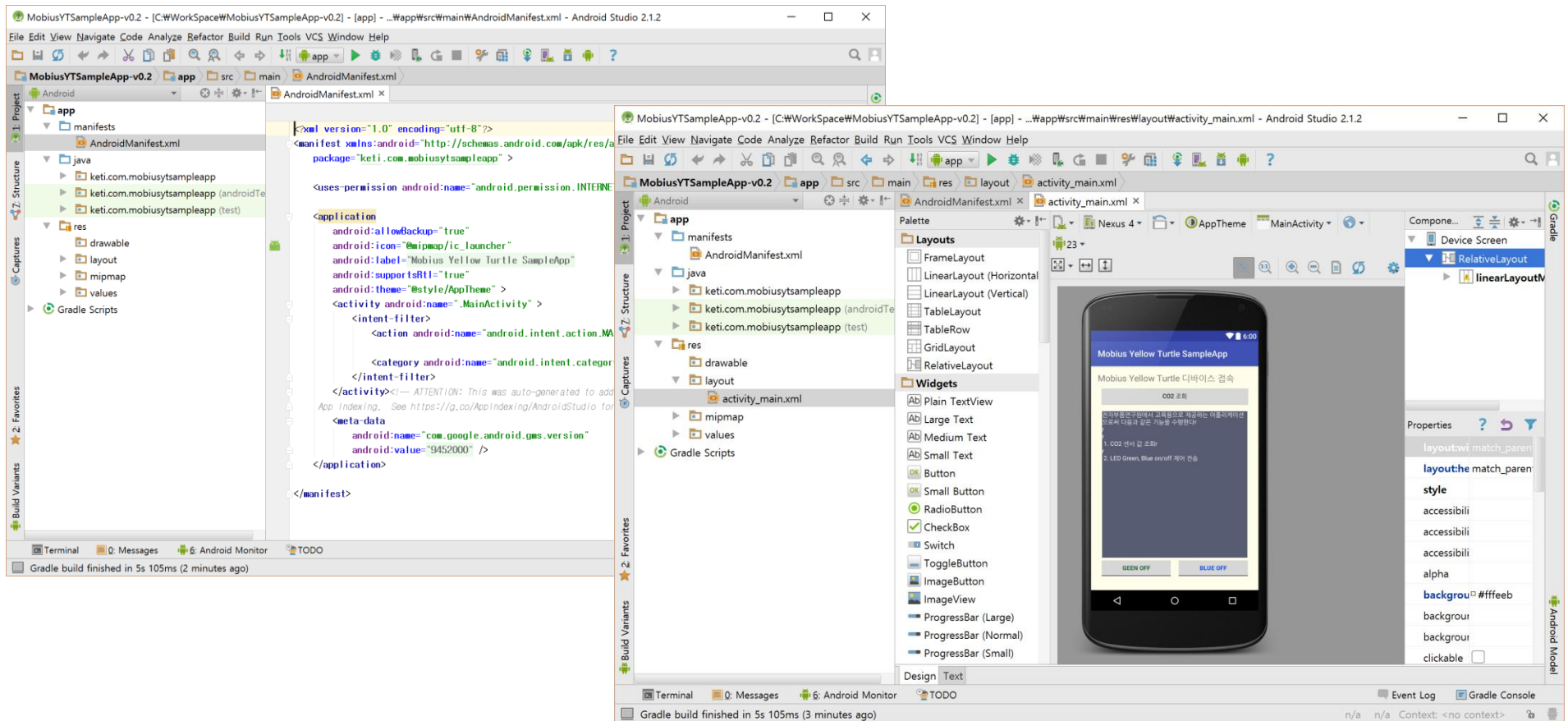


* Download the sample source from IoT OCEAN (www.iotocean.org)

Mobius and APP connectivity

■ Import sample project

Android/app/manifest(AndroidManifest.xml)

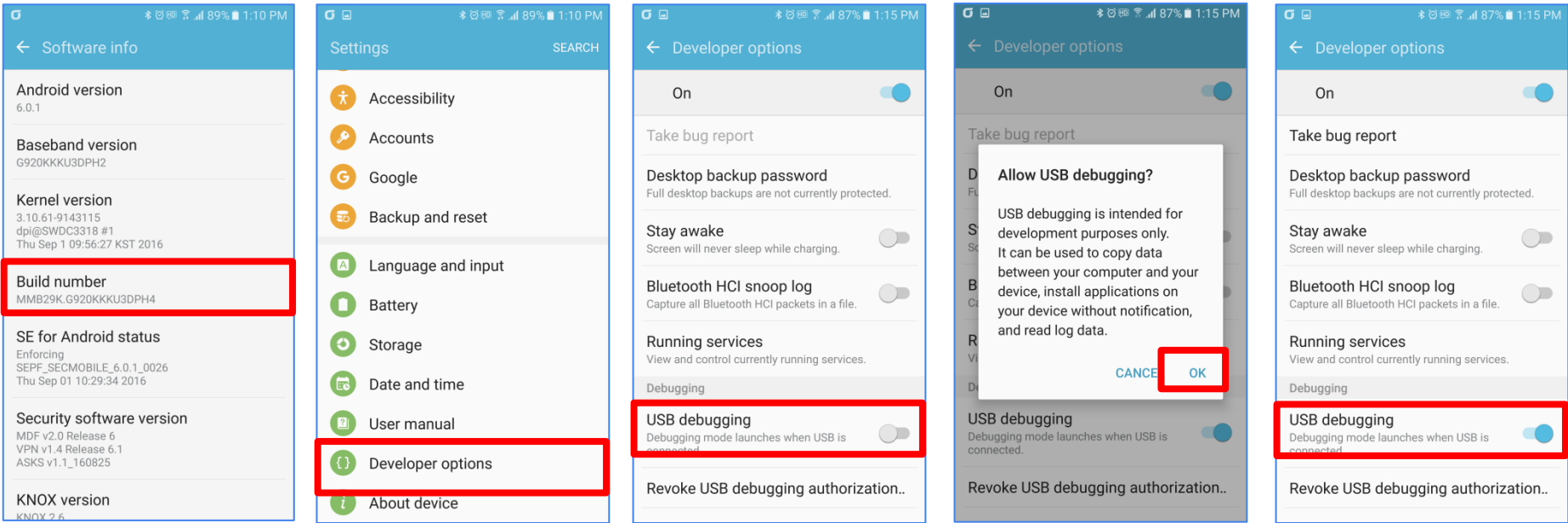


Android/app/res/layout/activity_main.xml(Design)

Mobius and APP connectivity

■ USB Debugging Mode on Android

Change device to debugging mode

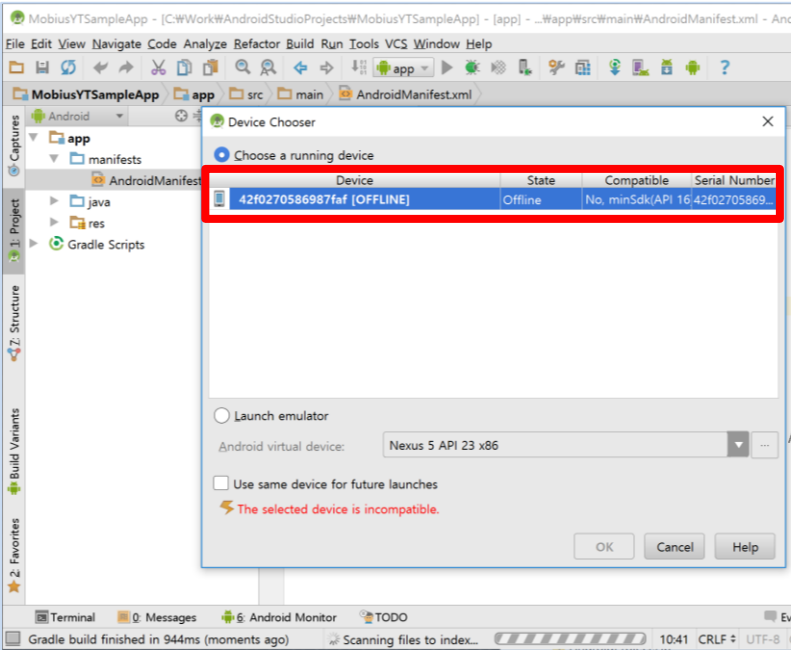


Click “Build number” continue

Mobius and APP connectivity

■ App Running

Android Phone Device Debugging Mode Connect



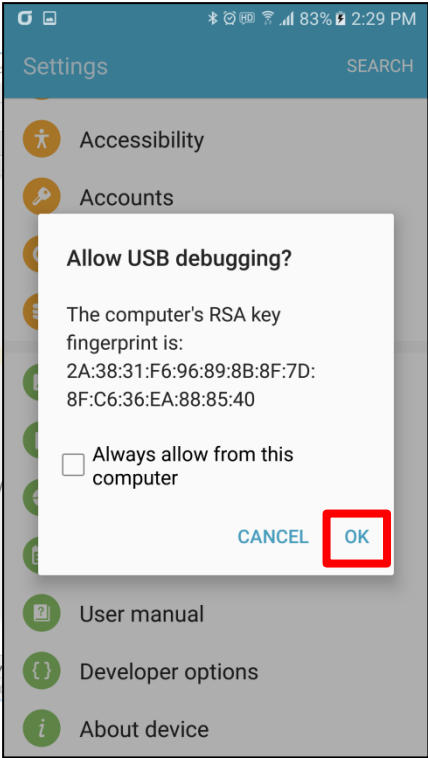
Device	State	Compatible	Serial Number
42f0270586987faf [OFFLINE]	Offline	No, minSdk(API 16)>42f02705869...	

Launch emulator
Android virtual device: Nexus 5 API 23 x86

Use same device for future launches

The selected device is incompatible.

OK Cancel Help



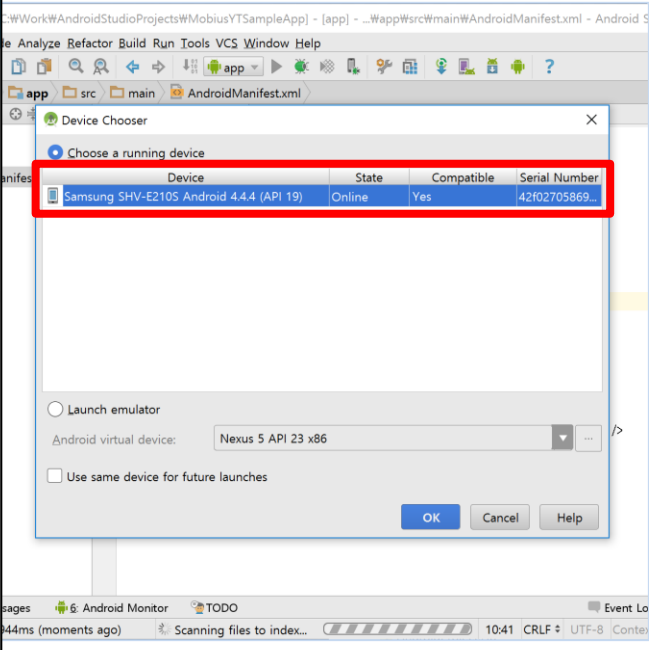
Settings

Allow USB debugging?

The computer's RSA key fingerprint is:
2A:38:31:F6:96:89:8B:8F:7D:
8F:C6:36:EA:88:85:40

☐ Always allow from this computer

CANCEL OK



Device	State	Compatible	Serial Number
Samsung SHV-E210S Android 4.4.4 (API 19)	Online	Yes	42f02705869...

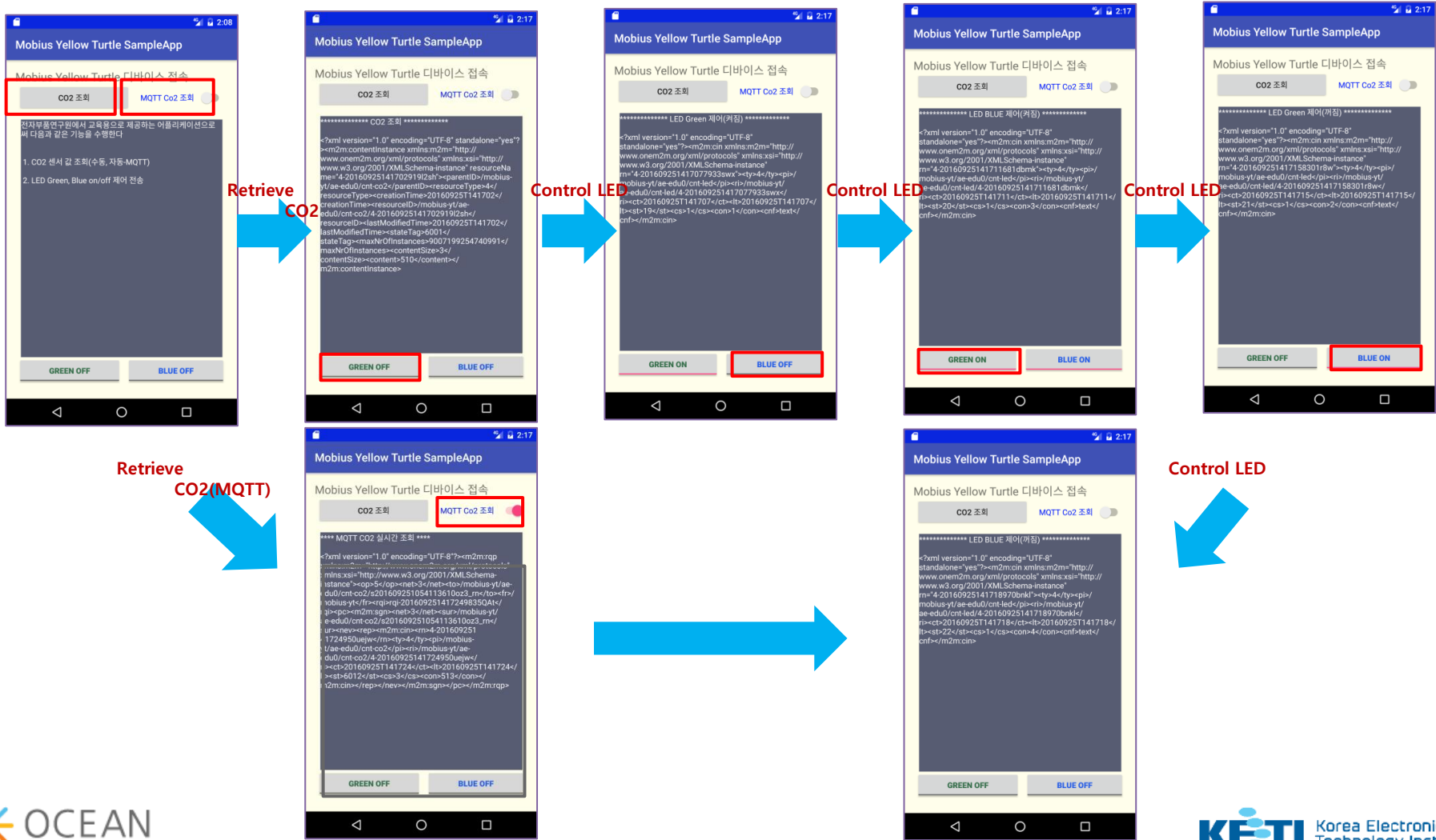
Launch emulator
Android virtual device: Nexus 5 API 23 x86

Use same device for future launches

OK Cancel Help

Mobius and APP connectivity

■ App Running



App Code Review

■ Internet Permission Setting [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>
```

App Code Review

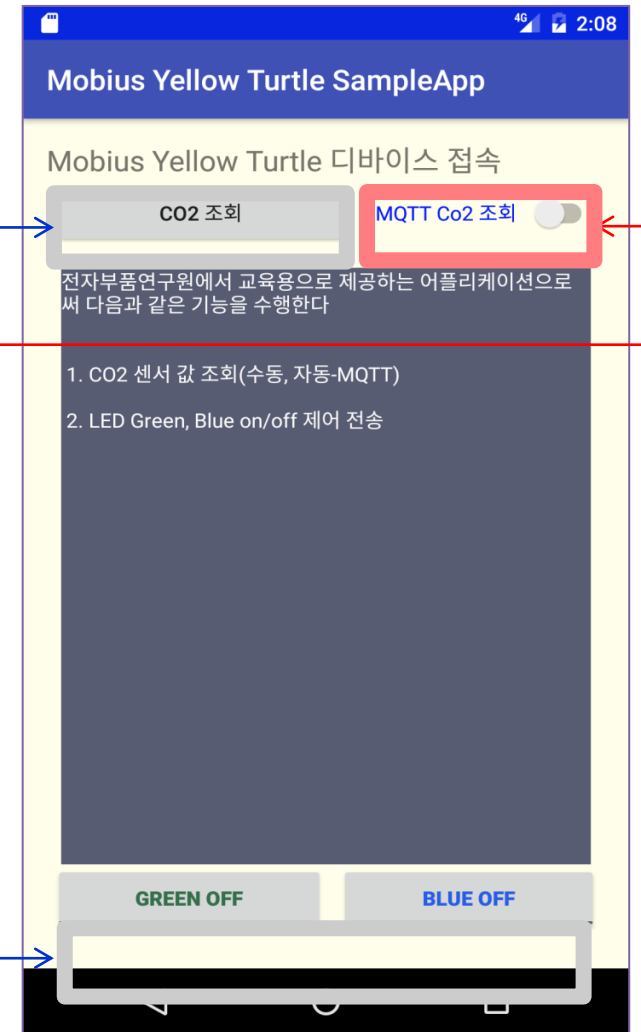
■ Code Structure Review

```
// Main
public MainActivity() { handler = new Handler(); }
/* onCreate */
protected void onCreate(Bundle savedInstanceState) {...}

/* AE Create for Android AE */
public void GetAEInfo() {...}
/* Switch - Get Co2 Data With MQTT */
public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {...}
/* MQTT Subscription */
public void MQTT_Create(boolean mqttStart) {...}
/* 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 {...}
/* Request AE Creation */
class aeCreateRequest extends Thread {...}
/* Retrieve AE-ID */
class aeRetrieveRequest extends Thread {...}
/* Subscribe Co2 Content Resource */
class SubscribeResource extends Thread {...}
```



App Code Review

■ Code Structure Review [AE Create or AE Retrieve ...]

```

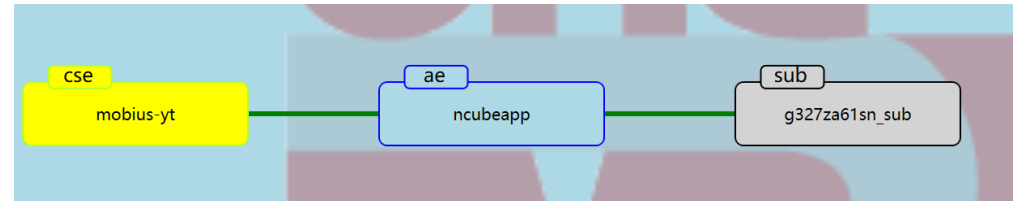
/* Request AE Creation */
class aeCreateRequest extends Thread {
    private final Logger LOG = Logger.getLogger(aeCreateRequest.class.getName());
    String TAG = aeCreateRequest.class.getName();
    private IReceived receiver;
    int responseCode=0;
    public ApplicationEntityObject applicationEntity;
    public void setReceiver(IReceived handler) { this.receiver = handler; }
    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 handler) { this.receiver = handler; }

    @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>

```

Note: In oneM2M standard Android Application is mapping with Application Entity(AE). It should use the AE-ID as request “origin” to do other resource operation(Create, Retrieve, Update, Delete).

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 handler) { this.receiver = handler; }

    @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";

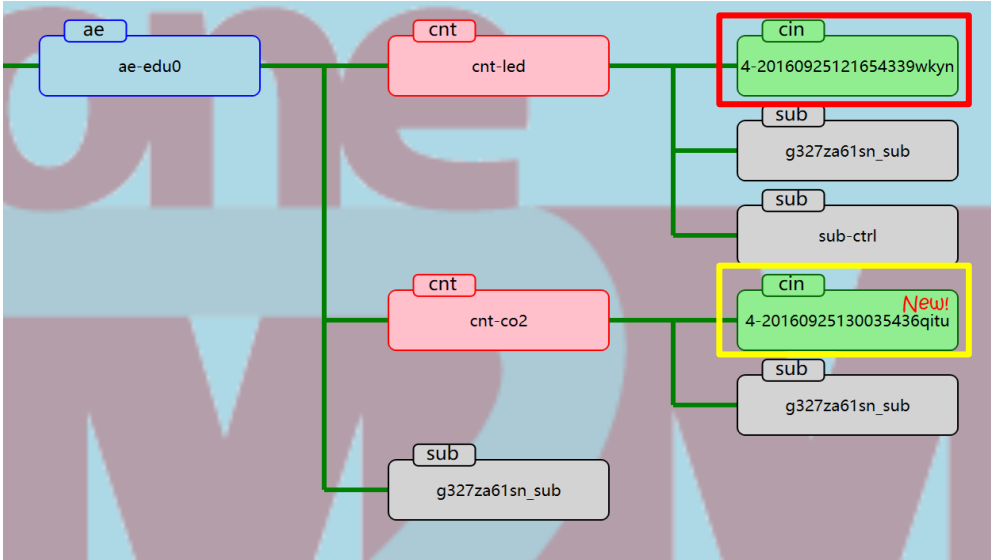
    public ContentInstanceObject contentinstance;

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

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

    @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>

App Code Review

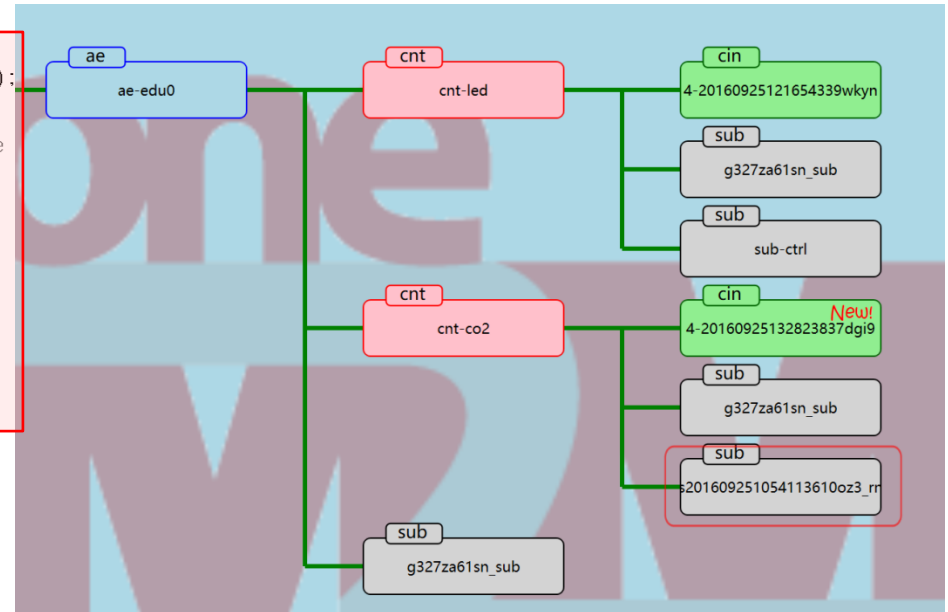
■ Code Structure Review [Subscription create for MQTT]

/* 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 subscribelnstance;
    public SubscribeResource() {...}
    public void setReceiver(IReceived handler) { this.receiver = handler; }

    @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="s201609251054113610oz3_rn"><pi>/mobius-yt/ae-edu0/cnt-
co2</pi><ty>23</ty><ct>20160925T133651</ct><ri>/mobius-yt/ae-edu0/cnt-
co2/s201609251054113610oz3_rn</ri><lt>20160925T133651</lt><st>0</st><enc
><net>3</net></enc><nu>mqtt://192.168.25.47/S201609251054113610oz3_sub</
nu><nct>2</nct><cr>S201609251054113610oz3</cr></m2m:sub>
```

App Code Review

■ Code Structure Review [Button Click Event]

```
public void onClick(View v) {

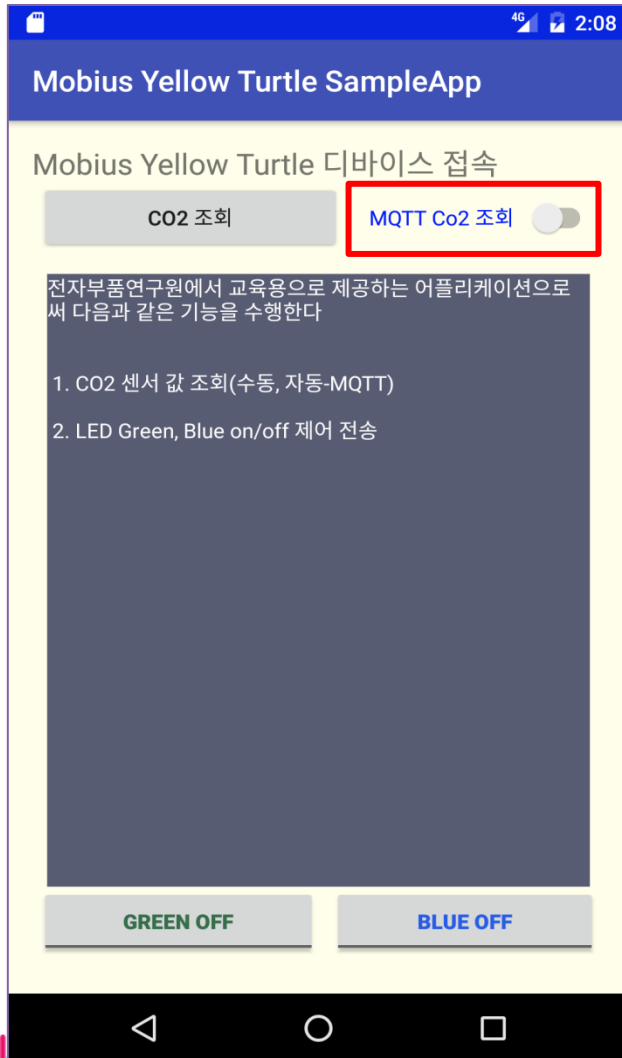
    switch (v.getId()) {

        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;
}
```

App Code Review

■ Code Structure Review [Switch onCheckedChangeListener 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 subscribeResource = new SubscribeResource();
        subscribeResource.setReceiver(new IReceived() {
            public void getResponseBody(final String msg) {
                handler.post(new Runnable() {
                    public void run() { ... }
                });
            }
        });
        subscribeResource.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;
    }
}
}
```


App Code Review

Retrieve by POSTMAN [CO2 Sensing Data]

M-IF77. contentInstance 리소스 조회

GET

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

Params

Send

Save

Authorization

Headers (4)

BodyPre-request ScriptTestsCode

✓ Acceptapplication/xml

✓ X-M2M-R12345

✓ X-M2M-OriginSae_edu1

✓ nmtyperlong

keyvalue

Bulk EditPresets

BodyCookiesHeaders (8)TestsStatus: 200 OKTime: 213 ms

PrettyRawPreviewXML

```
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
3   -instance" resourceName="4-20160928171758407S0Md">
4   <parentID>/mobius-yt/ae-edu1/cnt-co2</parentID>
5   <resourceType>4</resourceType>
6   <creationTime>20160928T171758</creationTime>
7   <resourceID>/mobius-yt/ae-edu1/cnt-co2/4-20160928171758407S0Md</resourceID>
8   <lastModifiedTime>20160928T171758</lastModifiedTime>
9   <stateTag>0</stateTag>
10  <maxNrOfInstances>9007199254740991</maxNrOfInstances>
11  <contentSize>3</contentSize>
12  <content>461</content>
13 </m2m:contentInstance>
```

App Code Review

■ Retrieve by Java Code [CO2 Sensing Data]

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

...

public void GetAEInfo() {

    csebase.setInfo("192.168.25.47", "7579", "mobius-yt", "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 handler) { this.receiver = handler; }
}
```

```
@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="1"
<m2m:contentInstance xmlns:m2m="http://www.onem2m
  -instance" resourceName="4-201609281717584079
  <parentID>/mobius-yt/ae-edu1/cnt-co2</parentID>
  <resourceType>4</resourceType>
  <creationTime>20160928T171758</creationTime>
  <resourceID>/mobius-yt/ae-edu1/cnt-co2/4-2016
  <lastModifiedTime>20160928T171758</lastModifiedTime>
  <stateTag>0</stateTag>
  <maxNrOfInstances>9007199254740991</maxNrOfInstances>
  <contentSize>3</contentSize>
  <content>461</content>
</m2m:contentInstance>
```

App Code Review

■ Control by POSTMAN [LED Control]

M-IF74. contentInstance 리소스 생성

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

AuthorizationHeaders (6)BodyPre-request ScriptTests

Acceptapplication/xml

localeko

X-M2M-R12345

X-M2M-OriginSae-edu1

Content-Typeapplication/vnd.onem2m-res+xml; ty=4

nmtypeshort

keyvalue

M-IF74. contentInstance 리소스 생성

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

AuthorizationHeaders (6)BodyPre-request ScriptTests

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

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

BodyCookiesHeaders (10)Tests

Status: 201 CreatedTime: 240 ms

PrettyRawPreviewXML

```
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="4-20160928183913558u4ys">
3   <ty>4</ty>
4   <pi>/mobius-yt/ae-edu1/cnt-led</pi>
5   <ri>/mobius-yt/ae-edu1/cnt-led/4-20160928183913558u4ys</ri>
6   <ct>20160928T183913</ct>
7   <lt>20160928T183913</lt>
8   <st>0</st>
9   <cs>1</cs>
10  <con>1</con>
11  <cnf>text</cnf>
12 </m2m:cin>
```

1: Green LED ON

2: Green LED OFF

3: Blue LED ON

4: Blue LED OFF

App Code Review

■ Control by Java Code [LED Control]

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

```
...
```

```
public void GetAEInfo() {
```

```
    csebase.setInfo("192.168.25.47", "7579", "mobius-yt", "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 handler) { this.receiver = handler; }
```

```
@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());
```

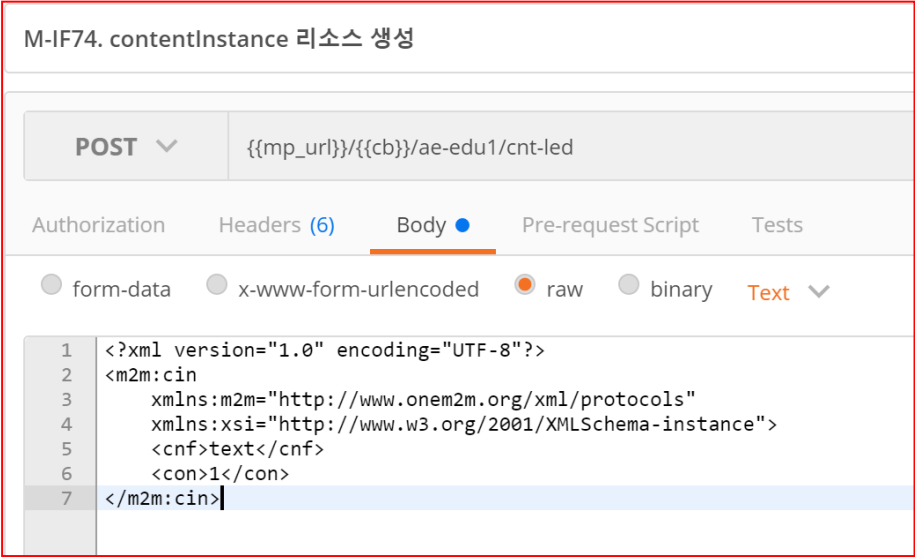
```
    }
```

```
}
```

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

App Code Review

■ Content Instance Creation Java Code



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

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

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

        return xml;
    }
}
```

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

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

    return xml;
}
```



App Code Review

■ AE Create by POSTMAN

M-IF50. AE 리소스 생성

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

AuthorizationHeaders (6)BodyPre-request ScriptTests

Acceptapplication/xml

localeko

X-M2M-R12345

X-M2M-OriginS

Content-Typeapplication/vnd.onem2m-res+xml; ty=2

nmtype

short

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

ParamsSend

AuthorizationHeaders (6)BodyPre-request ScriptTests

form-data

x-www-form-urlencoded

raw

binary

Text

1<?xml version="1.0" encoding="UTF-8"?>

2<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="ncubeapp">

3<api>0.2.481.2.0001.001.0001112</api>

4<rr>true</rr>

5</m2m:ae>

BodyCookiesHeaders (10)Tests

Status: 201 Created

PrettyRawPreviewXML

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<ty>2</ty>

4<pi>/mobius-yt</pi>

5<ri>/mobius-yt/ncubeapp</ri>

6<ct>20160928T193850</ct>

7<lt>20160928T193850</lt>

8<api>0.2.481.2.0001.001.0001112</api>

9<rr>true</rr>

10<aei>ncubeapp</aei>

11</m2m:ae>

BodyCookiesHeaders (9)Tests

Status: 409 Conflict

PrettyRawPreviewXML

1<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

2<m2m:rsp xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

3<cap>resource is already exist</cap>

4</m2m:rsp>

App Code Review

■ AE Create by Java Code

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

```
...
```

```
public void GetAEInfo() {
```

```
    csebase.setInfo("192.168.25.47", "7579", "mobius-yt", "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");
        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");
            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>
```

App Code Review

■ AE Retrieve POSTMAN [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

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

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>

App Code Review

■ AE Retrieve Java Code

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

...

public void GetAEInfo() {

    csebase.setInfo("192.168.25.47","7579","mobius-yt","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 handler) {
        this.receiver = handler;
    }
}
```

```
@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>
```

App Code Review

■ Subscription Creation by POSTMAN

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

POST

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

Authorization

Headers (6)

Body

Pre-request Script

Tests

form-data

x-www-form-urlencoded

raw

binary

Text

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

Body

Cookies

Headers (10)

Tests

Status: 201 CreatedTime: 150 ms

Pretty

Raw

Preview

XML

```
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <m2m:sub xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" rn="ncubeapp_rn">
3   <pi>/mobius-yt/ae-edu1/cnt-co2</pi>
4   <ty>23</ty>
5   <ct>201609281171145</ct>
6   <ri>/mobius-yt/ae-edu1/cnt-co2/ncubeapp_rn</ri>
7   <lt>201609281171145</lt>
8   <st>0</st>
9   <enc>
10    <net>3</net>
11  </enc>
12  <nu>mqtt://192.168.25.47/ncubeapp_rn</nu>
13  <nct>2</nct>
14  <cr>ncubeapp</cr>
15 </m2m:sub>
```

App Code Review

■ Subscription Creation by Java Code

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

public void GetAEInfo() {

    csebase.setInfo("192.168.25.47","7579","mobius-yt","1883");
    ae.setAppName("ncubeapp");
    ...
}

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

/* 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 handler) { this.receiver = handler; }
```

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

public void GetAEInfo() {

    csebase.setInfo("192.168.25.47","7579","mobius-yt","1883");
    ae.setAppName("ncubeapp");
    ...
}

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

/* 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 handler) { this.receiver = handler; }
```



App Code Review

■ MQTT Client Subscription, Listener, Callback, Response Java Code

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

public void GetAEInfo() {

    csebase.setInfo("192.168.25.47","7579","mobius-yt","1883");
    ae.setAppName("ncubeapp");
    ...
    MQTT_Req_Topic = "/oneM2M/req:/mobius-yt/"+ae.getAEid()+"_sub"+"xml";
    MQTT_Resp_Topic = "/oneM2M/resp:/mobius-yt/"+ae.getAEid()+"_sub"+"xml";
}

/* MQTT Subscription */
public void MQTT_Create(boolean mtqStart) {
    if (mtqStart && mqttClient == null) {
        /* Subscription Resource Create to Yellow Turtle */
        SubscribeResource subscribeResource = new SubscribeResource();
        subscribeResource.setReceiver(new IReceived() {
            ...
            subscribeResource.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("\t|\n", ""));
        ArrayList<String> mqttMessage = new ArrayList<String>();
        mqttMessage = MqttClientRequestParser.notificationParse(message.toString());
        String responseMessage = MqttClientRequest.notificationResponse(mqttMessage);

        /* 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");
    }
};
```

App Code Review

■ Android Studio Generate APK

- Build -> Generate Signed APK...
- If have Key store, Select “Choose existing”
- If do not have Key store, Select “Create new”
- Save location and input file name, Password, Alias, Certification information.
- Select key file
- Locate the APK path.

