

การพัฒนา Mobile application สำหรับ Android OS

อ. พิชัย จอดพิมาย

Email: pichai.j@cpc.ac.th

Room: 970

โครงร่างเนื้อหา

- 1 แนะนำ Android Studio
- 2 การสร้างหน้า Login
- 3 การเพิ่ม ลบ แก้ไข และแสดงรายการ
- 4 การแสดงกราฟแท่ง กราฟวงกลม กราฟเส้น และเกจชาร์ต
- 5 การจัดการกับกล้องและวิดีโอ (Camera and Video)
- 6 การสร้างและอ่าน QR Code
- 7 การแสดงตำแหน่งบน Google Map
- 8 ภาคผนวก ก : ทบทวนภาษา Java
- 9 ภาคผนวก ข : การพัฒนา RESTful API

1. แนะนำ Android Studio

ก่อนการพัฒนาต้องมีการเตรียมความพร้อมโดยการติดตั้ง Android Studio เพื่อใช้สำหรับการพัฒนา Android OS-based Project



1.1 ตรวจสอบความต้องการก่อนติดตั้ง Android Studio

Windows

- Microsoft® Windows® 7/8/10 (32- or 64-bit)
The Android Emulator only supports 64-bit Windows ([learn more](#)).
- 4 GB RAM minimum, 8 GB RAM recommended
- 2 GB of available disk space minimum,
4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

Mac

- Mac® OS X® 10.10 (Yosemite) or higher, up to 10.14 (macOS Mojave)
- 4 GB RAM minimum, 8 GB RAM recommended
- 2 GB of available disk space minimum,
4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

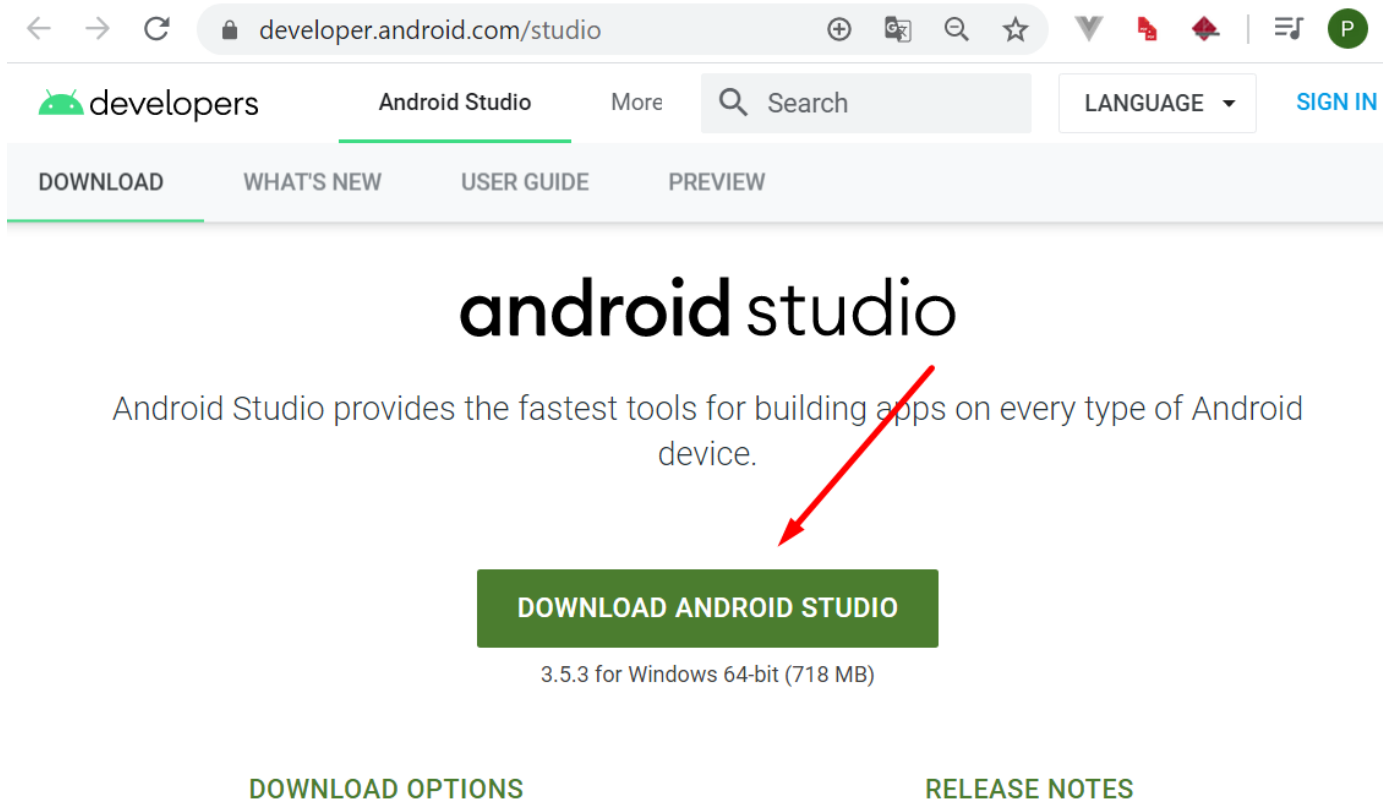
Linux

- GNOME or KDE desktop
Tested on gLinux based on Debian (4.19.67-2rodete2).
- 64-bit distribution capable of running 32-bit applications
- GNU C Library (glibc) 2.19 or later
- 4 GB RAM minimum, 8 GB RAM recommended
- 2 GB of available disk space minimum,
4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

1.2 ติดตั้ง Android Studio

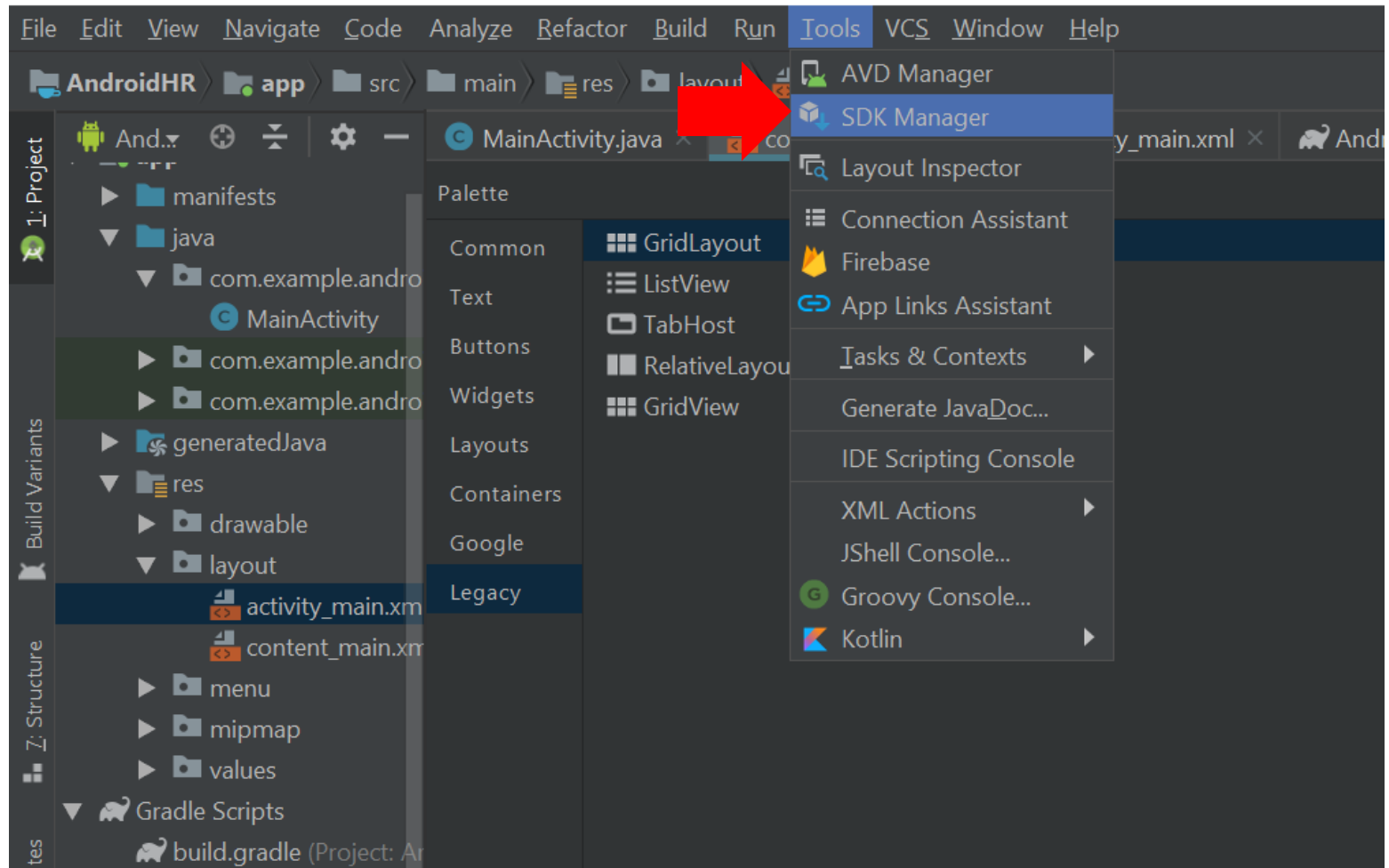
ดาวน์โหลดตัวติดตั้ง ที่:

<https://developer.android.com/studio>



ดำเนินการติดตั้งจนกว่าจะเสร็จ

ติดตั้ง SDK



เลือก SDK Version ที่ต้องการ แล้วคลิกปุ่ม "OK"

Appearance & Behavior > System Settings > Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: [Edit](#)

SDK Platforms SDK Tools SDK Update Sites

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

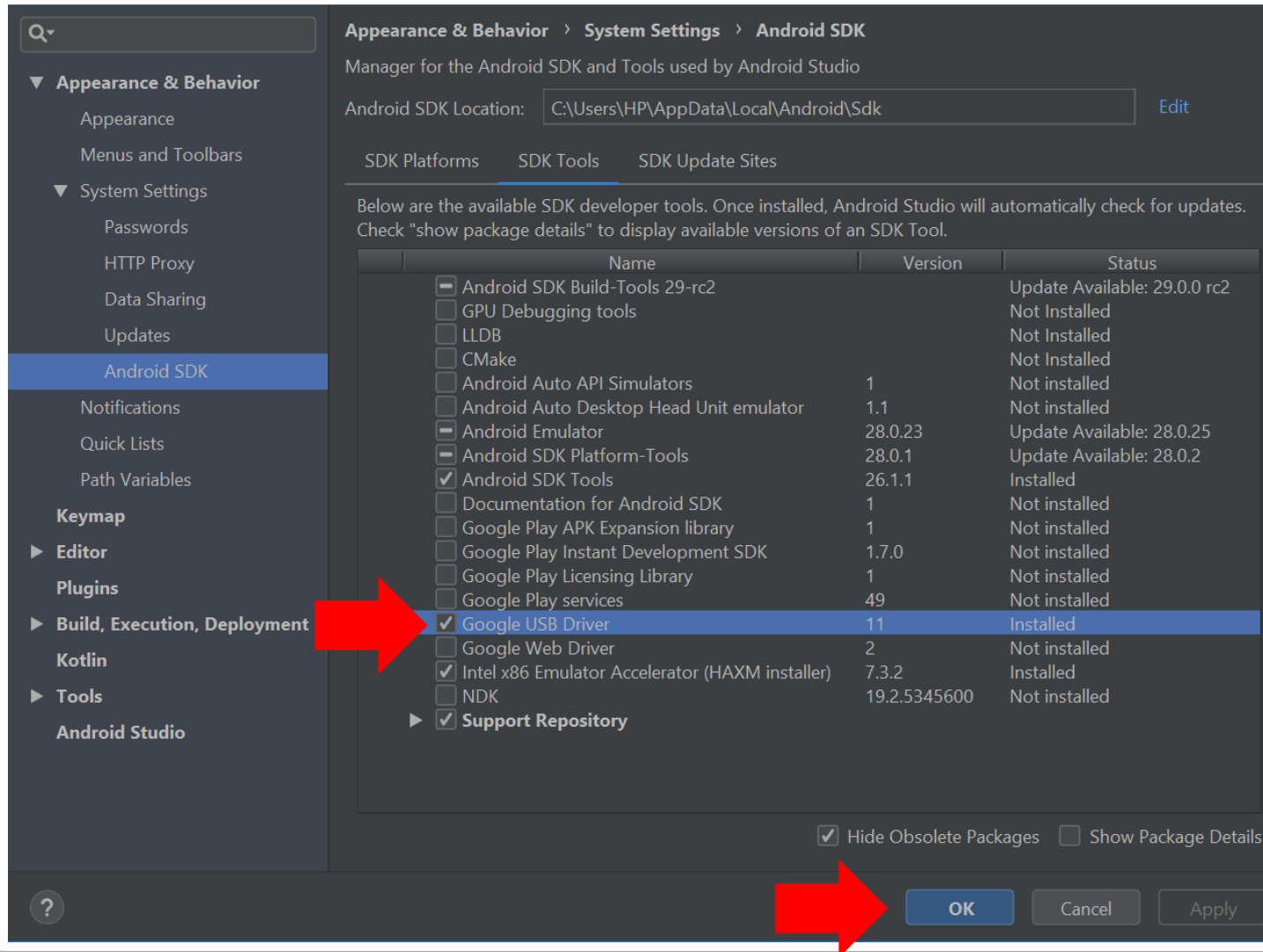
Name	API Level	Revision	Status
<input type="checkbox"/> Android Q Preview	Q	2	Not installed
<input checked="" type="checkbox"/> Android 9.0 (Pie)	28	6	Installed
<input checked="" type="checkbox"/> Android 8.1 (Oreo)	27	3	Installed
<input type="checkbox"/> Android 8.0 (Oreo)	26	2	Partially installed
<input type="checkbox"/> Android 7.1.1 (Nougat)	25	3	Not installed
<input type="checkbox"/> Android 7.0 (Nougat)	24	2	Not installed
<input type="checkbox"/> Android 6.0 (Marshmallow)	23	3	Partially installed
<input type="checkbox"/> Android 5.1 (Lollipop)	22	2	Not installed
<input type="checkbox"/> Android 5.0 (Lollipop)	21	2	Not installed
<input type="checkbox"/> Android 4.4W (KitKat Wear)	20	2	Not installed
<input type="checkbox"/> Android 4.4 (KitKat)	19	4	Not installed
<input type="checkbox"/> Android 4.3 (Jelly Bean)	18	3	Not installed
<input type="checkbox"/> Android 4.2 (Jelly Bean)	17	3	Not installed
<input type="checkbox"/> Android 4.1 (Jelly Bean)	16	5	Not installed
<input type="checkbox"/> Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
<input type="checkbox"/> Android 4.0 (IceCreamSandwich)	14	4	Not installed
<input type="checkbox"/> Android 3.2 (Honeycomb)	13	1	Not installed
<input type="checkbox"/> Android 3.1 (Honeycomb)	12	3	Not installed
<input type="checkbox"/> Android 3.0 (Honeycomb)	11	2	Not installed
<input type="checkbox"/> Android 2.3.3 (Gingerbread)	10	2	Not installed
<input type="checkbox"/> Android 2.3 (Gingerbread)	9	2	Not installed

☒ Hide Obsolete Packages ☐ Show Package Details

OK Cancel Apply

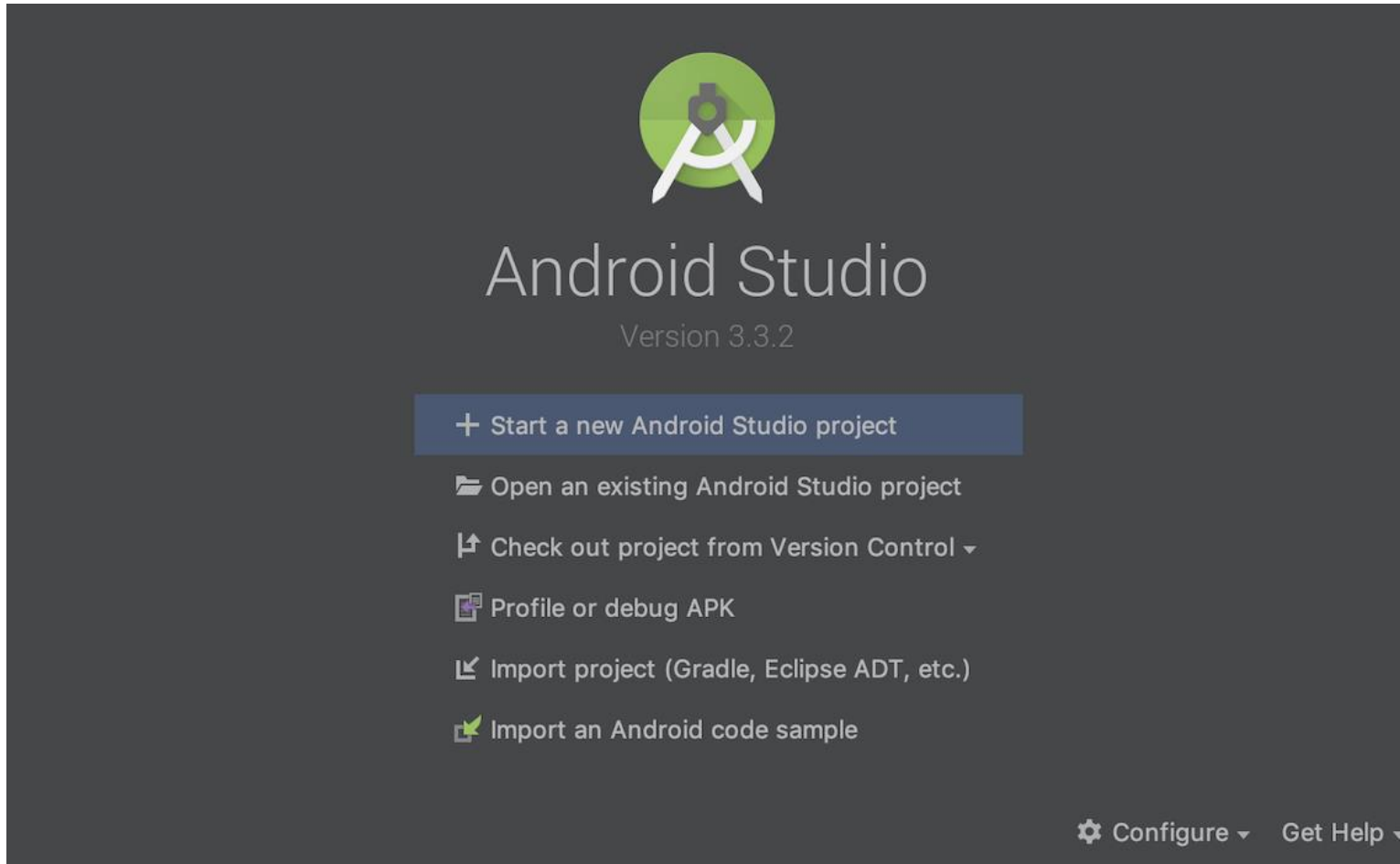
ติดตั้ง Google USB Driver

เลือก Google USB Driver แล้วคลิกปุ่ม "OK"

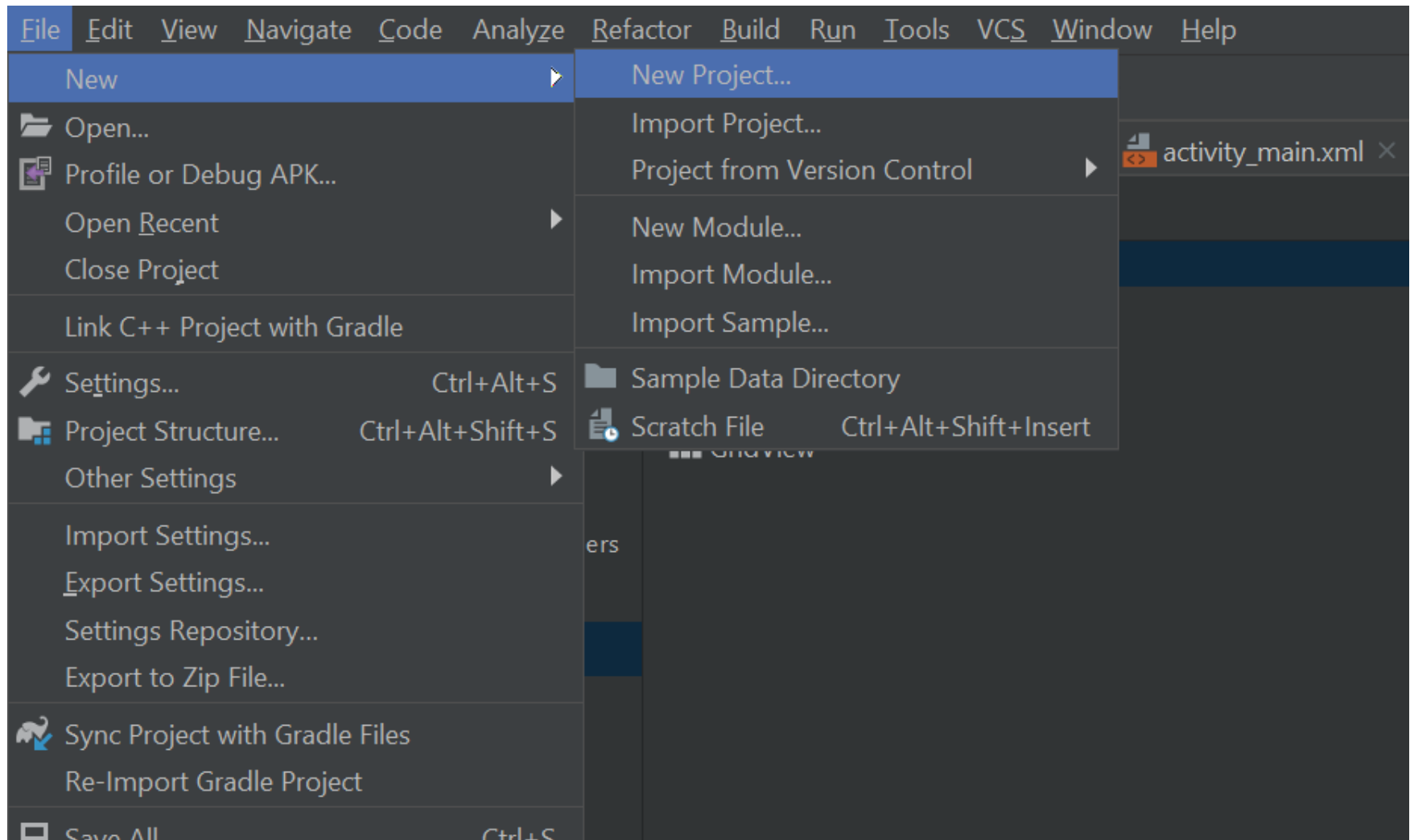


1.3 การสร้าง Project

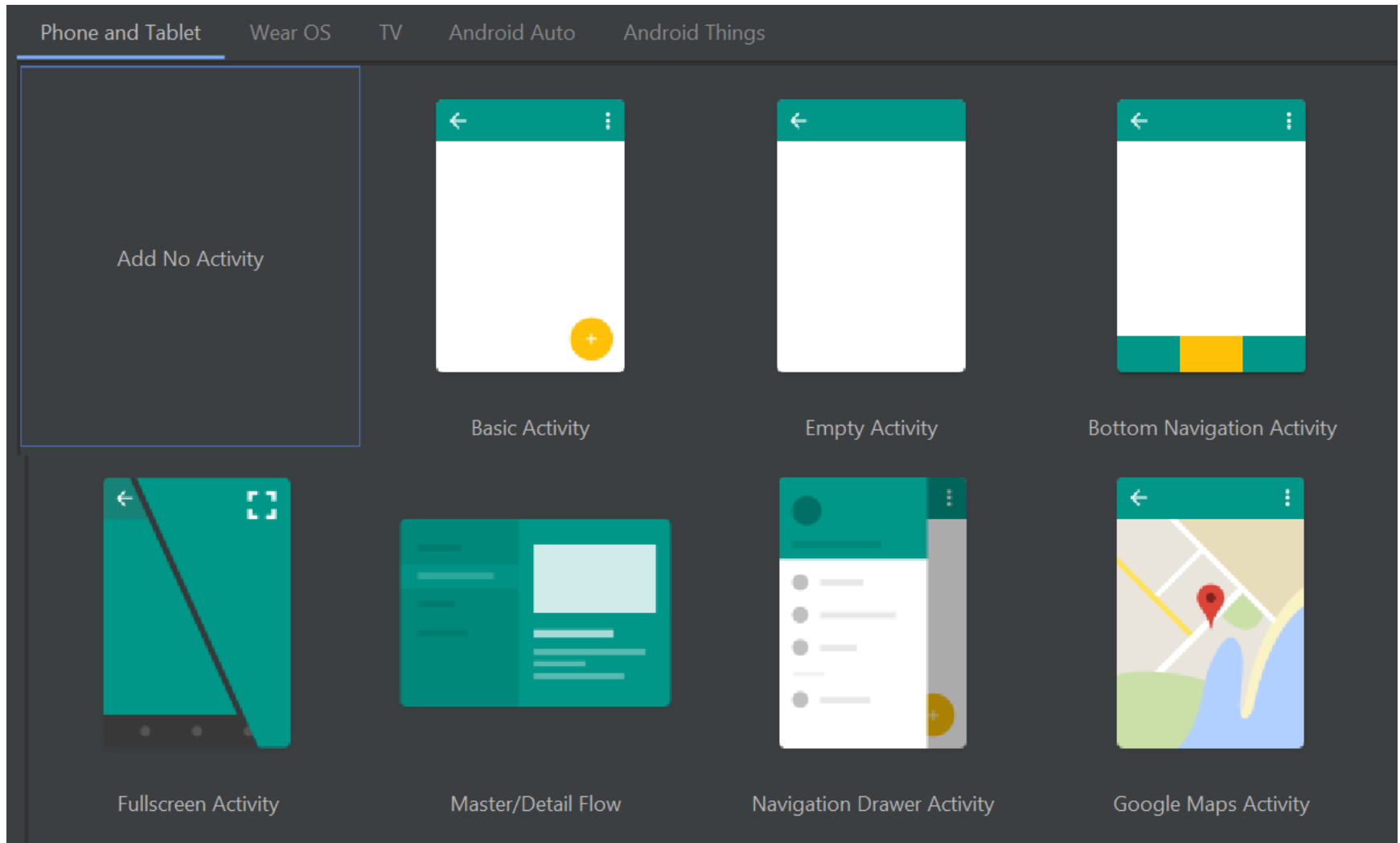
กรณีที่ยังไม่มีการสร้าง Project เลยหรือไม่มีการเปิด Project ค้างไว้ จะพบหน้านี้ ให้ทำการคลิกที่ "Start a new android..."



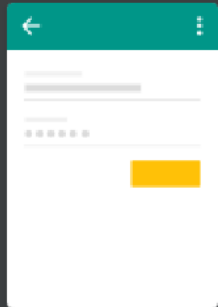
กรณีที่มีการเปิด Project ค้างไว้ จะพบหน้านี้ ให้ทำการคลิกที่ File->New->New Project



จากนั้นทำการเลือก Activity ที่ต้องการ จากทั้งหมด 16 รูปแบบ

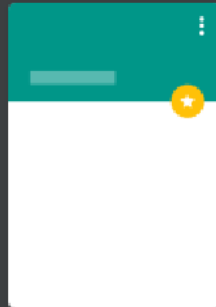


Fullscreen Activity



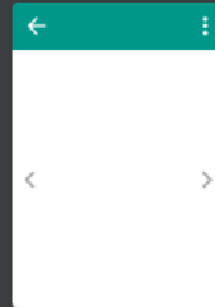
Login Activity

Master/Detail Flow



Scrolling Activity

Navigation Drawer Activity



Tabbed Activity

Google Maps Activity



Native C++

Empty Activity

Creates a new empty activity

Previous

Next

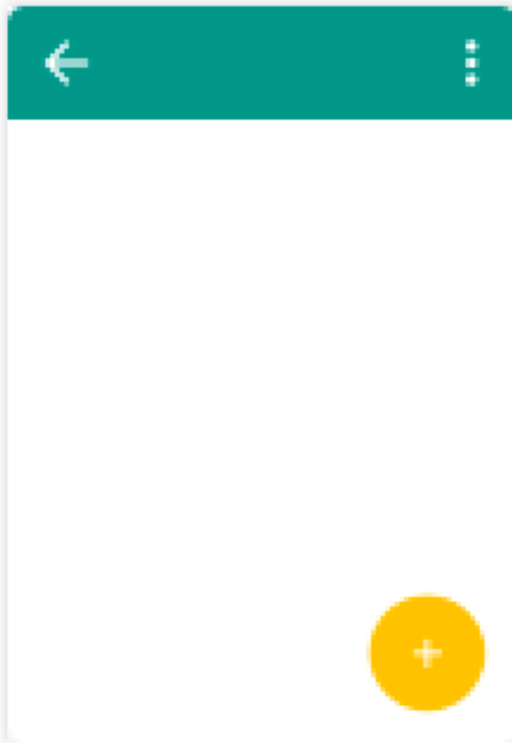
Cancel

Finish

No Activity

ไม่มีการสร้าง Activity หรือ Template เลย
ดังนั้นผู้พัฒนาจะต้องดำเนินการสร้าง Activity ขึ้นมาใช้งานเอง

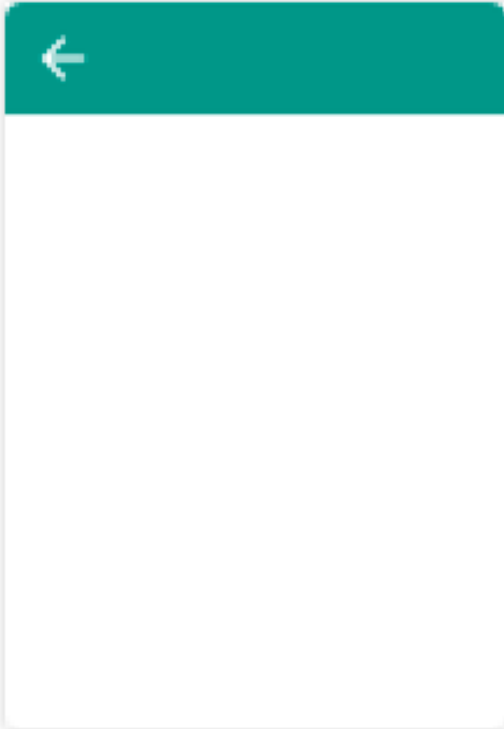
Basic Activity



Activity นี้จะสร้าง UI พื้นฐานประกอบด้วย

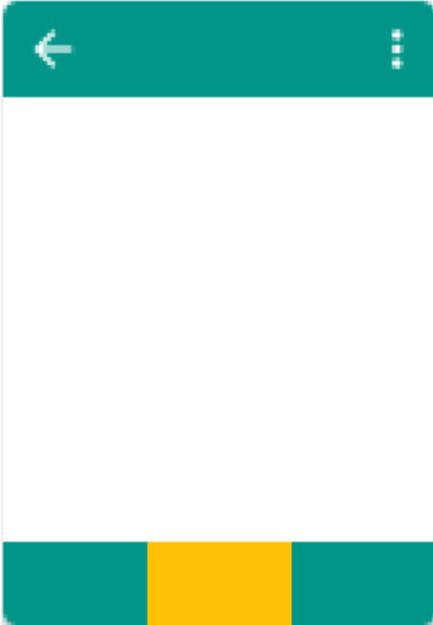
- AppBar นั่นคือ Application Bar
- FloatingActionButton นั่นคือ ปุ่มลอย โดยปกติจะมีลักษณะวงกลม ซึ่งสามารถลอยทับ UI อื่น ๆ

Empty Activity



Activity นี้จะเป็น Activity เปล่า ๆ ผู้พัฒนา
ต้องระบุ UI ที่ต้องการใช้เองทั้งหมด

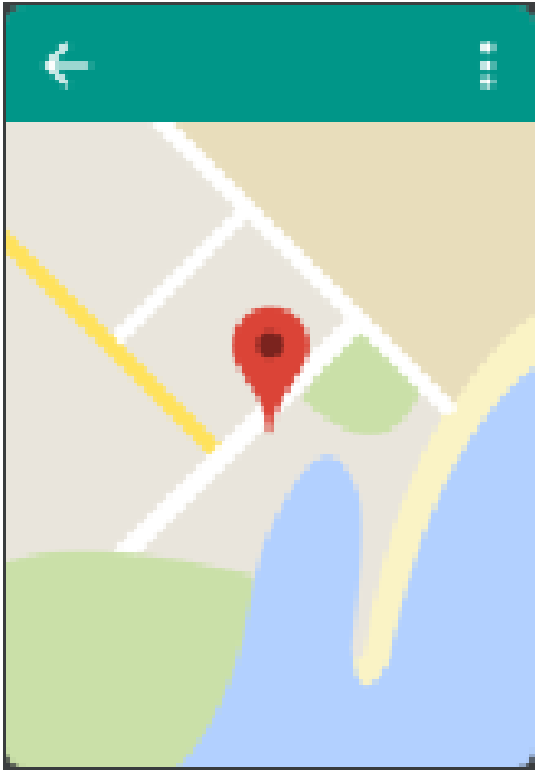
Bottom Navigation Activity



Activity นี้จะสร้าง UI
ประกอบด้วย

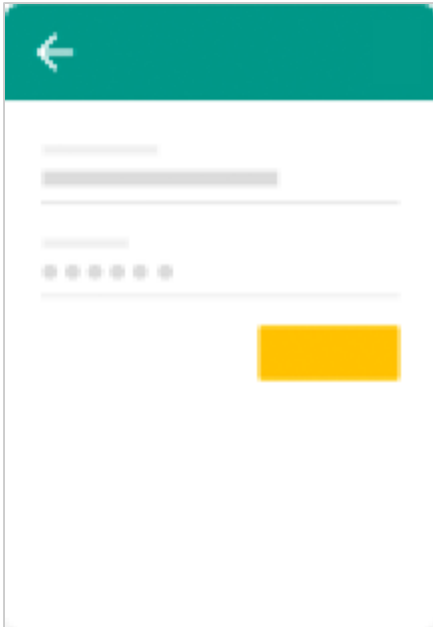
- AppBar นั่นคือ Application Bar
- Bottom Navigation นั่นคือ ปุ่มที่อยู่ด้านล่างของจอ

Google Map Activity



Activity นี้จะสร้าง Fragment สำหรับแสดงแผนที่ของ Google Map

Login Activity



Activity นี้จะสร้าง UI ประกอบด้วย

- Email และ password input fields
- Sing-in button

นอกจาก UI ยังมีการสร้าง Progress bar เพื่อแสดงความก้าวหน้าของการ Load ข้อมูลผ่านทาง Network

ระบบ

- ชื่อ Application
- ชื่อ Package
- ตำแหน่งที่จัดเก็บ Code
- ภาษา (Java/Kotlin)
- API level ที่ App. นี้

รองรับ

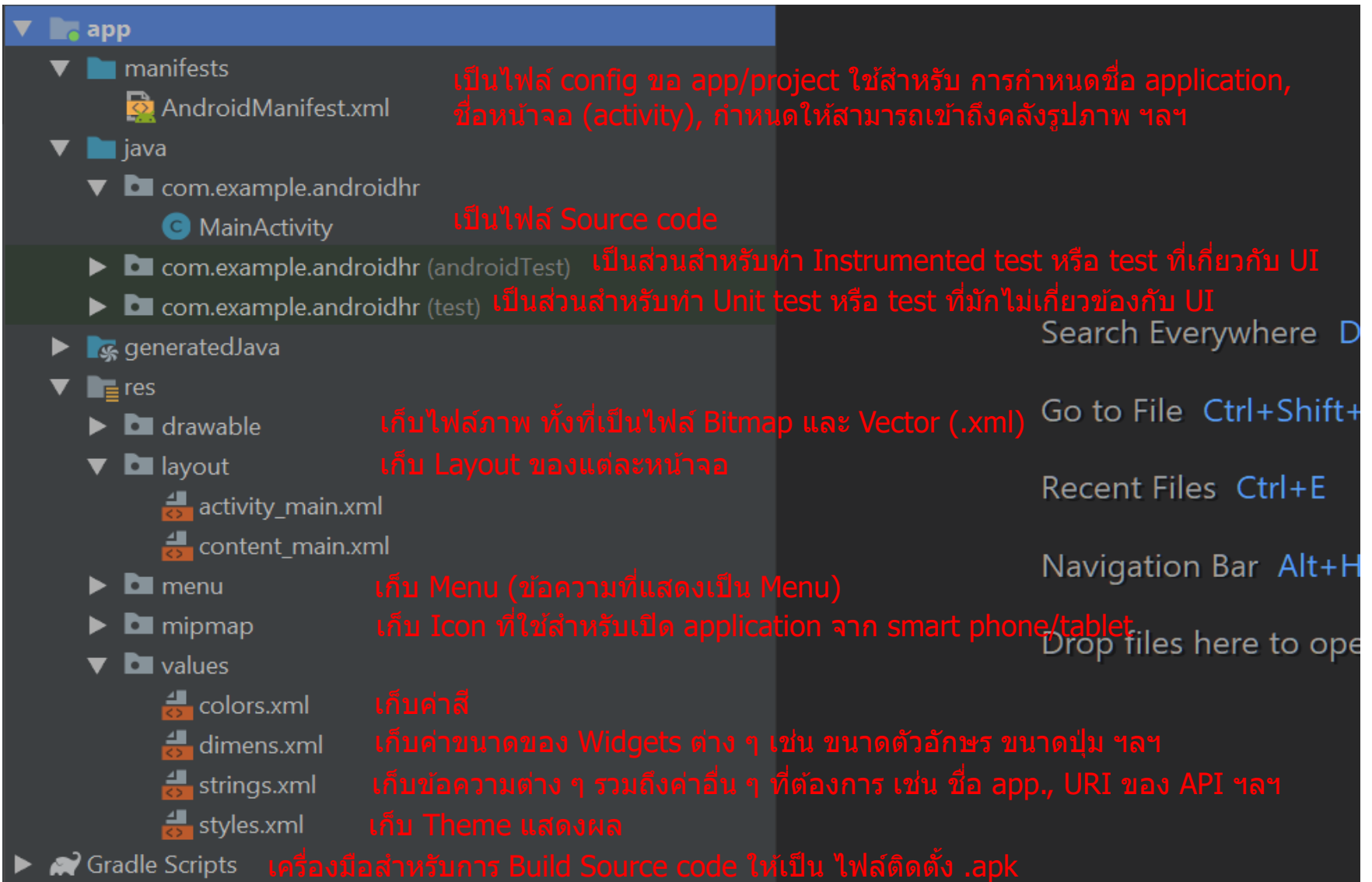
(* API 18 จะรองรับ Smart phone/Tablet บนระบบปฏิบัติการ Android ได้ประมาณ 95.9% ของเครื่องที่มีขายทั้งหมด)

The screenshot shows the 'New Project' dialog in Android Studio. The fields are filled with the following information:

- Name:** My Application
- Package name:** com.example.androidhr
- Save location:** C:\pichai\Thesis\MyApplication
- Language:** Java
- Minimum API level:** API 18: Android 4.3 (Jelly Bean)

Below the fields, there is an information icon and a message: "Your app will run on approximately 95.9% of devices." with a link "Help me choose". There are also two unchecked checkboxes: "This project will support instant apps" and "Use AndroidX artifacts". At the bottom, there are four buttons: "Previous", "Next", "Cancel", and "Finish".

1.4 โครงสร้างของ Project

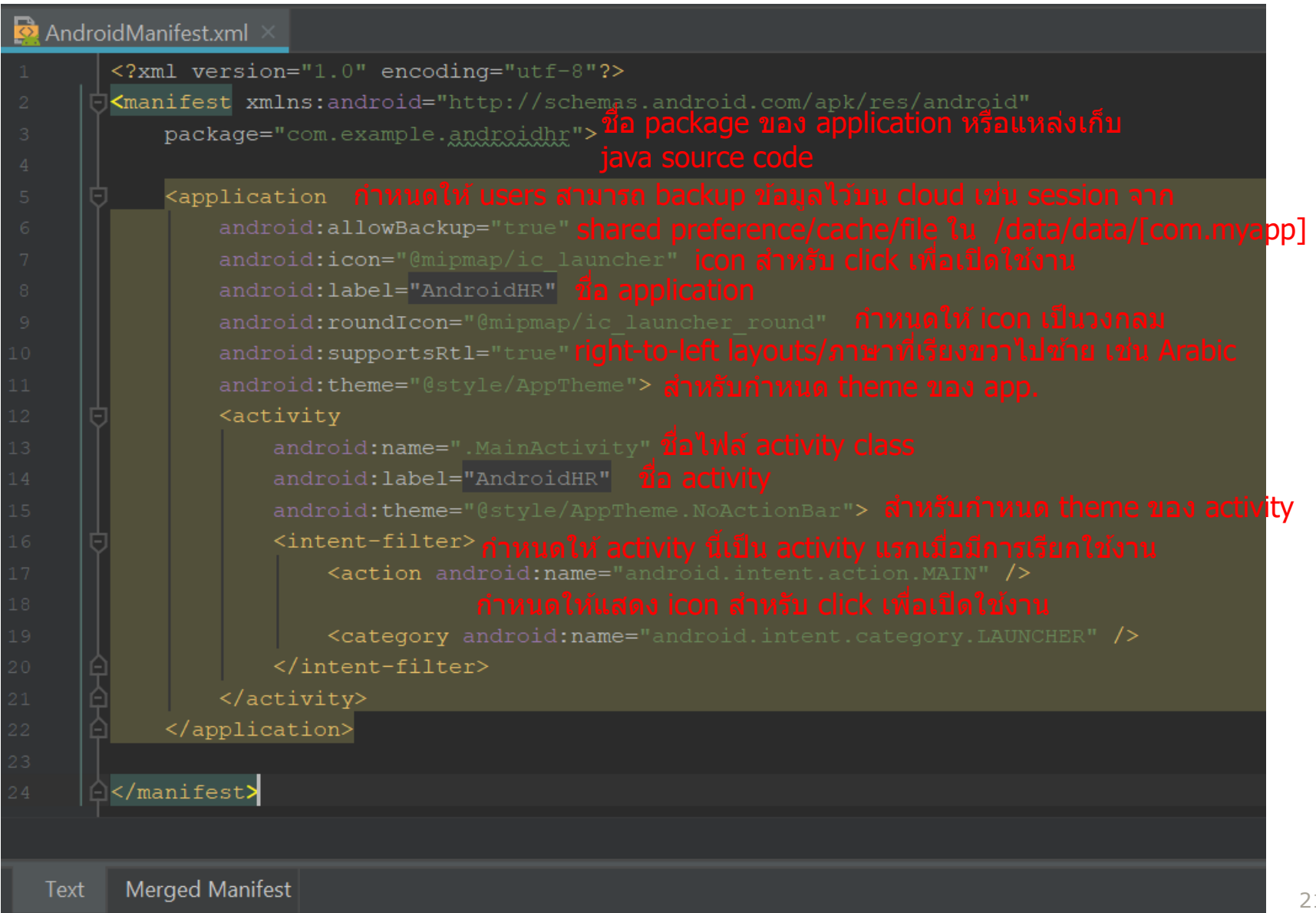


The image shows a screenshot of an Android Studio project structure with various folders and files. Red Thai text annotations explain the purpose of each component. On the right side, there are faint, semi-transparent text elements from the IDE interface.

Folder/File	Annotation (Thai)
app	
manifests	
AndroidManifest.xml	เป็นไฟล์ config ของ app/project ใช้สำหรับ การกำหนดชื่อ application, ชื่อหน้าจอ (activity), กำหนดให้สามารถเข้าถึงคลังรูปภาพ ฯลฯ
java	
com.example.androidhr	
MainActivity	เป็นไฟล์ Source code
com.example.androidhr (androidTest)	เป็นส่วนสำหรับทำ Instrumented test หรือ test ที่เกี่ยวข้องกับ UI
com.example.androidhr (test)	เป็นส่วนสำหรับทำ Unit test หรือ test ที่มักไม่เกี่ยวข้องกับ UI
generatedJava	
res	
drawable	เก็บไฟล์ภาพ ทั้งที่เป็นไฟล์ Bitmap และ Vector (.xml)
layout	เก็บ Layout ของแต่ละหน้าจอ
activity_main.xml	
content_main.xml	
menu	เก็บ Menu (ข้อความที่แสดงเป็น Menu)
mipmap	เก็บ Icon ที่ใช้สำหรับเปิด application จาก smart phone/tablet
values	
colors.xml	เก็บค่าสี
dimens.xml	เก็บค่าขนาดของ Widgets ต่าง ๆ เช่น ขนาดตัวอักษร ขนาดปุ่ม ฯลฯ
strings.xml	เก็บข้อความต่าง ๆ รวมถึงค่าอื่น ๆ ที่ต้องการ เช่น ชื่อ app., URI ของ API ฯลฯ
styles.xml	เก็บ Theme แสดงผล
Gradle Scripts	เครื่องมือสำหรับการ Build Source code ให้เป็น ไฟล์ติดตั้ง .apk

Search Everywhere D
Go to File Ctrl+Shift+
Recent Files Ctrl+E
Navigation Bar Alt+H
Drop files here to open

องค์ประกอบของไฟล์ Androidmanifest.xml



The image shows a screenshot of an AndroidManifest.xml file in an IDE. The file is titled "AndroidManifest.xml" and contains the following XML code:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.androidhr">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="AndroidHR"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="AndroidHR"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Annotations in Thai explain the elements:

- <manifest>**: ชื่อ package ของ application หรือแหล่งเก็บ java source code
- <application>**: กำหนดให้ users สามารถ backup ข้อมูลไว้บน cloud เช่น session จาก shared preference/cache/file ใน /data/data/[com.myapp]
- android:icon**: icon สำหรับ click เพื่อเปิดใช้งาน
- android:label**: ชื่อ application
- android:roundIcon**: กำหนดให้ icon เป็นวงกลม
- android:supportsRtl**: right-to-left layouts/ภาษาที่เรียงขวาไปซ้าย เช่น Arabic
- android:theme**: สำหรับกำหนด theme ของ app.
- <activity>**: ชื่อไฟล์ activity class
- android:label**: ชื่อ activity
- android:theme**: สำหรับกำหนด theme ของ activity
- <intent-filter>**: กำหนดให้ activity นี้เป็น activity แรกเมื่อมีการเรียกใช้งาน
- <action android:name="android.intent.action.MAIN" />**: กำหนดให้แสดง icon สำหรับ click เพื่อเปิดใช้งาน
- <category android:name="android.intent.category.LAUNCHER" />**: กำหนดให้แสดง icon สำหรับ click เพื่อเปิดใช้งาน

The IDE interface shows a "Text" tab and a "Merged Manifest" tab at the bottom.

องค์ประกอบของไฟล์ MainActivity.java

```
1 package com.example.androidhr;
2
3 import ...
4
11
12 public class MainActivity extends AppCompatActivity {
13
14     @Override
15     protected void onCreate(Bundle savedInstanceState) {
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_main);
18         Toolbar toolbar = findViewById(R.id.toolbar);
19         setSupportActionBar(toolbar);
20
21         FloatingActionButton fab = findViewById(R.id.fab);
22         fab.setOnClickListener(new View.OnClickListener() {
23             @Override
24             public void onClick(View view) {
25                 Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH_LONG)
26                     .setAction("Action", null).show();
27             }
28         });
29     }
30
31     @Override
32     public boolean onCreateOptionsMenu(Menu menu) {...}
33
34
35
36
37     @Override
38     public boolean onOptionsItemSelected(MenuItem item) {...}
39
40
41
42
43
44
45
46
47
48
49
50
51 }
```

MainActivity Class ซึ่ง Inherit มาจาก AppCompatActivity เพื่อให้สามารถเข้าถึง UI ใน Activity Layout ได้

Method onCreate ซึ่งเป็น Method แรกที่จะทำงานเมื่อมีการเข้าสู่หน้าจอหรือ Activity

ชื่อไฟล์ Activity ที่อยู่ในไฟล์เดอร์ Layout

องค์ประกอบของโฟลเดอร์ drawable

เก็บไฟล์ภาพ ทั้งที่เป็นไฟล์ Bitmap และ Vector (.xml)

ไฟล์ ic_launcher_background.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <vector xmlns:android="http://schemas.android.com/apk/res/android"
3      android:width="108dp"
4      android:height="108dp"
5      android:viewportWidth="108"
6      android:viewportHeight="108">
7      <path
8          android:fillColor="#008577"
9          android:pathData="M0,0h108v108h-108z" />
10     <path
11         android:fillColor="#00000000"
12         android:pathData="M9,0L9,108"
13         android:strokeWidth="0.8"
14         android:strokeColor="#33FFFFFF" />
15     <path
16         android:fillColor="#00000000"
17         android:pathData="M19,0L19,108"
18         android:strokeWidth="0.8"
19         android:strokeColor="#33FFFFFF" />
```

องค์ประกอบของโฟลเดอร์ layout

ใช้สำหรับเก็บไฟล์ Layout หรือ Activity

ไฟล์ activity_main.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <android.support.design.widget.CoordinatorLayout
3      xmlns:android="http://schemas.android.com
4      xmlns:app="http://schemas.android.com/apk
5      xmlns:tools="http://schemas.android.com/t
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9
10     <android.support.design.widget.AppBarLayout
11
12
13
14
15
16
17
18
19
20
21
22
23
24     <include layout="@layout/content_main" />
25
26     <android.support.design.widget.FloatingActionButton...>
27
28
29
30
31
32
33
34 </android.support.design.widget.CoordinatorLayout>
```



ไฟล์ content_main.xml

```
content_main.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <android.support.constraint.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      app:layout_behavior="android.support.design.widget.AppBarLayout$ScrollingVie..."
9      tools:context=".MainActivity"
10     tools:showIn="@layout/activity_main">
11
12
13 </android.support.constraint.ConstraintLayout>
```

องค์ประกอบของฟลเดอร์ menu

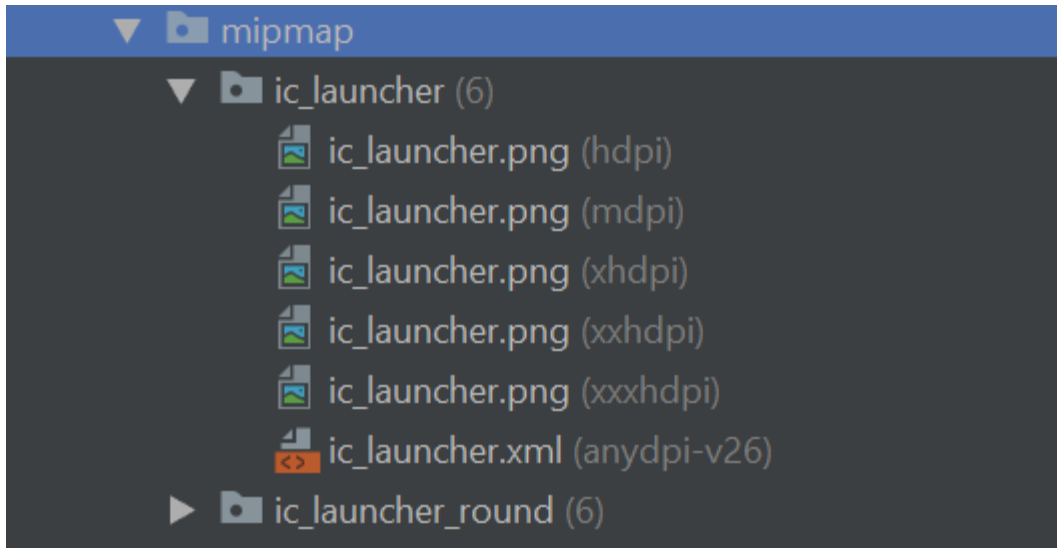
ใช้สำหรับเก็บไฟล์ Menu

ไฟล์ menu_main.xml

```
1 <menu xmlns:android="http://schemas.android.com/apk/res/android"
2       xmlns:app="http://schemas.android.com/apk/res-auto"
3       xmlns:tools="http://schemas.android.com/tools"
4       tools:context="com.example.myapplication.MainActivity">
5     <item
6         android:id="@+id/action_settings"
7         android:orderInCategory="100"
8         android:title="Settings"
9         app:showAsAction="never" />
10  </menu>
```

องค์ประกอบของโฟลเดอร์ minmap

ใช้สำหรับเก็บรูป icon ของ Application (แสดงบนหน้าจอเพื่อให้คลิกเข้าสู่ระบบ)



สนับสนุน ความละเอียดของหน้าจอที่แตกต่างกัน

mdpi = medium dot per inch

hdpi = high dot per inch

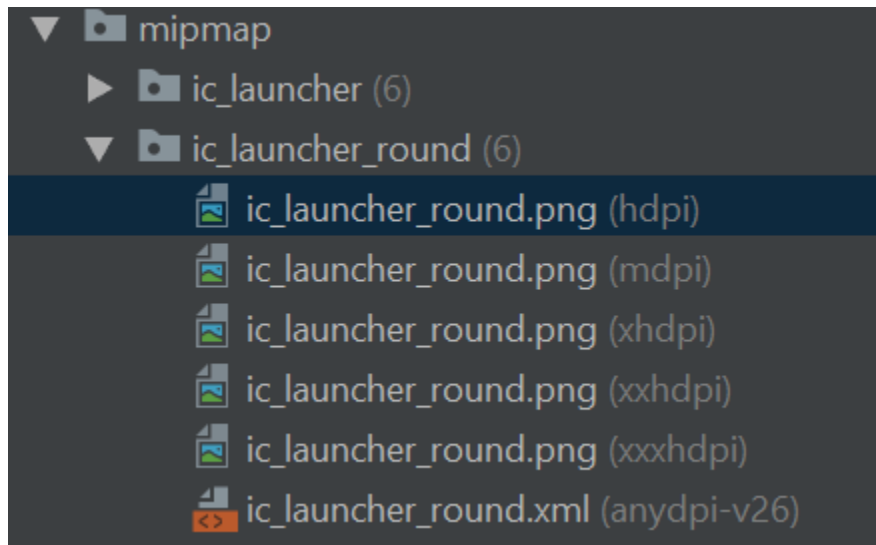
xhdpi = extra-high dot per inch

xxhdpi = extra-extra-high dot per inch

xxxhdpi = extra-extra-extra-high dot per inch



ไฟล์ Icon รูปสี่เหลี่ยม
ในขนาดต่าง ๆ



สนับสนุน ความละเอียดของหน้าจอที่แตกต่างกัน

mdpi = medium dot per inch

hdpi = high dot per inch

xhdpi = extra-high dot per inch

xxhdpi = extra-extra-high dot per inch

xxxhdpi = extra-extra-extra-high dot per inch



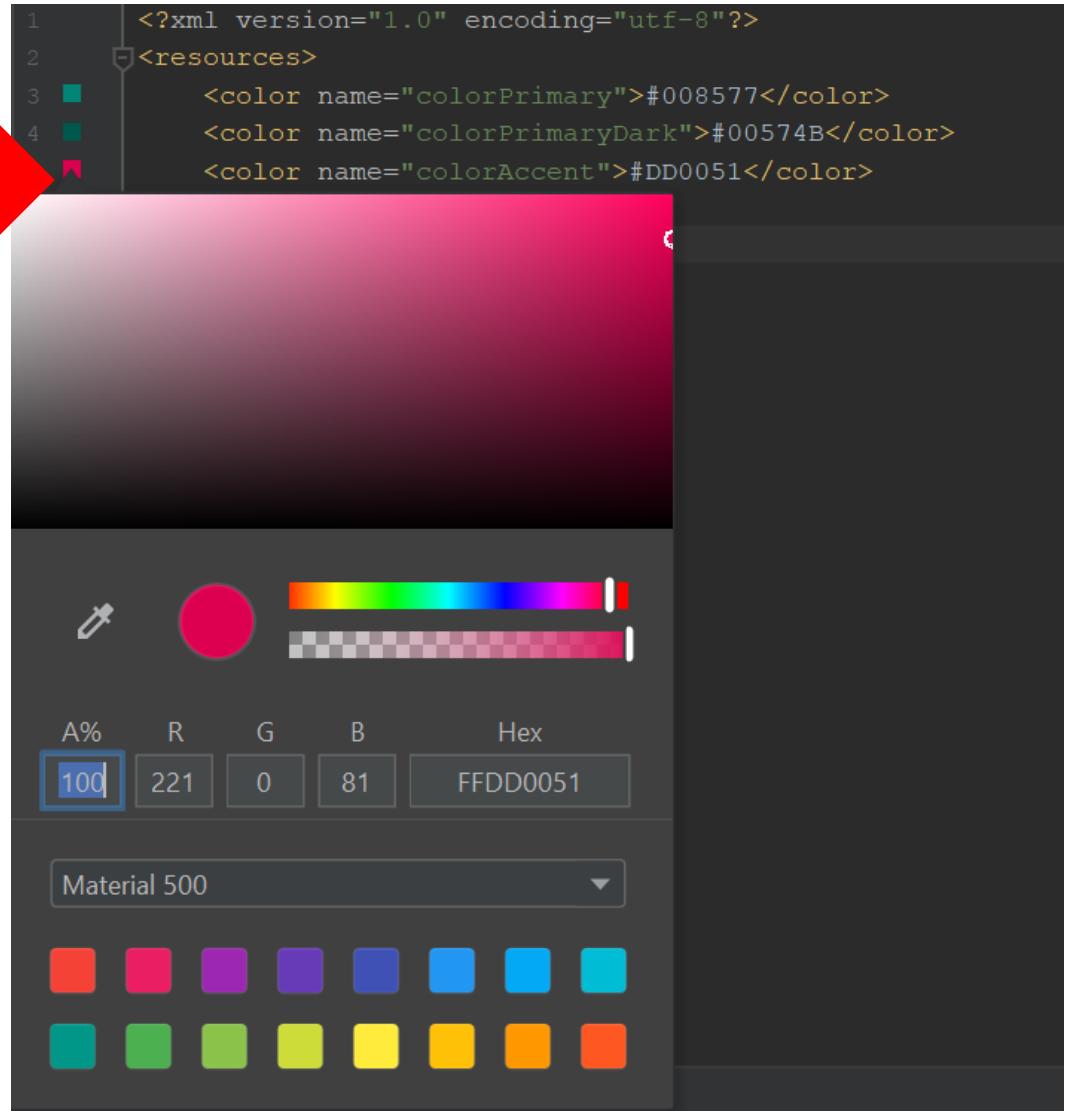
**ไฟล์ Icon รูปวงกลม
ในขนาดต่าง ๆ**

องค์ประกอบของไฟล์ color.xml

ใช้สำหรับกำหนดค่าสีของช่อกรอกข้อมูล ขนาดปุ่ม ขนาดตัวอักษร และขนาด Widgets อื่น ๆ

กำหนดค่าสีในระบบ
ARGB

- A=Alpha ใช้สำหรับกำหนด Transparency
- R=Red
- G=Green
- B=Blue



องค์ประกอบของไฟล์ **dimens.xml**

ใช้สำหรับเก็บขนาดช่องกรอกข้อมูล ขนาดปุ่ม ขนาดตัวอักษร และขนาด Widgets อื่น ๆ

```
1 <resources>
2     <dimen name="text_input">60dp</dimen>
3     <dimen name="button">60dp</dimen>
4     <dimen name="h1">32sp</dimen>
5     <dimen name="h2">24sp</dimen>
6     <dimen name="h3">16sp</dimen>
7     <dimen name="h4">14sp</dimen>
8     <dimen name="h5">12sp</dimen>
9 </resources>
```

สรุปเรื่องหน่วยวัดในการแสดงผล

sp = Scale-independent Pixels

dp = Density-independent Pixels

px = Pixels (พิกเซล)

mm = Millimeters (มิลลิเมตร)

in = inches (นิ้ว)

pt = Points (จุด) มีขนาดเท่ากับ $1/72$ inches

sp และ dp จะปรับขนาดให้เหมาะสมกับความละเอียดจอภาพในแต่ละเครื่อง โดยที่ sp มักใช้กับขนาดตัวอักษร ขณะที่ dp มักใช้กับ Widgets อื่น ๆ

px : ความละเอียดของจอภาพที่ต่างกันทำให้จำนวน pixels ต่อ 1 inches แตกต่างกัน ซึ่งส่งผลให้ขนาดของ Widgets เปลี่ยนไปด้วย (แต่การเปลี่ยนไปไม่ได้ถูกปรับให้เหมาะสมกับความละเอียดของจอภาพ)

mm, in, และ pt : ขนาดของ Widgets คงที่ไม่เปลี่ยนแปลงตามความละเอียดของจอภาพ

องค์ประกอบของไฟล์ strings.xml

ใช้สำหรับเก็บข้อความที่แสดงใน Label ช่องกรอกข้อมูล ปุ่ม Widgets อื่น ๆ หรือแม้แต่ตำแหน่งของ API สำหรับรับ-ส่งข้อมูล

```
1 <resources>
2     <string name="app_name">Human Resource Management System</string>
3     <string name="api_uri">http://172.20.10.8/employee/</string>
4     <string name="employee_first_name">ชื่อ</string>
5     <string name="employee_last_name">นามสกุล</string>
6     <string name="employee_address">ที่อยู่</string>
7 </resources>
```


องค์ประกอบของไฟล์ style.xml

ใช้สำหรับกำหนดรูปแบบของ Theme แสดงผลทั้งในระดับ App.
ระดับ Activity หรือระดับ Widget

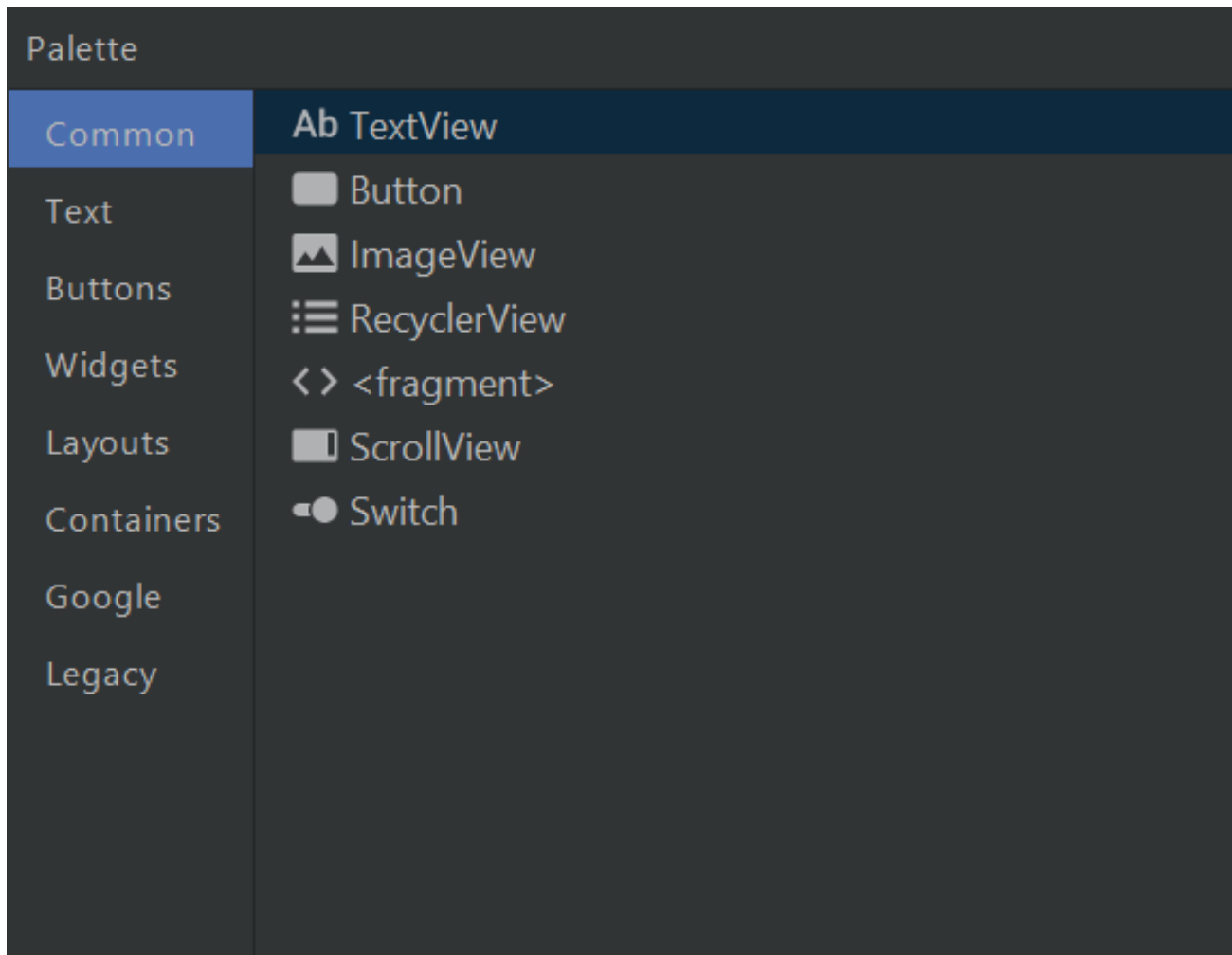
```
1 <resources>
2
3     <!-- Base application theme. -->
4     <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
5         <!-- Customize your theme here. --> การกำหนดสีของ Action Bar
6         <item name="colorPrimary">@color/colorPrimary</item>
7         <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
8         <item name="colorAccent">@color/colorAccent</item>
9     </style>
10
11     <style name="AppTheme.NoActionBar" กำหนดให้ไม่แสดง Action Bar
12         <item name="windowActionBar">false</item>
13         <item name="windowNoTitle">true</item>
14     </style>
15
16     <style name="AppTheme.AppBarOverlay" parent="ThemeOverlay.AppCompat.Dark.ActionBar" />
17
18     <style name="AppTheme.PopupOverlay" parent="ThemeOverlay.AppCompat.Light" />
19
20 </resources>
21
```

1.5 Palette [Component/Widget/Layout]

- ☐ Common
- ☐ Text
- ☐ Buttons
- ☐ Widgets
- ☐ Layouts
- ☐ Containers
- ☐ Legacy

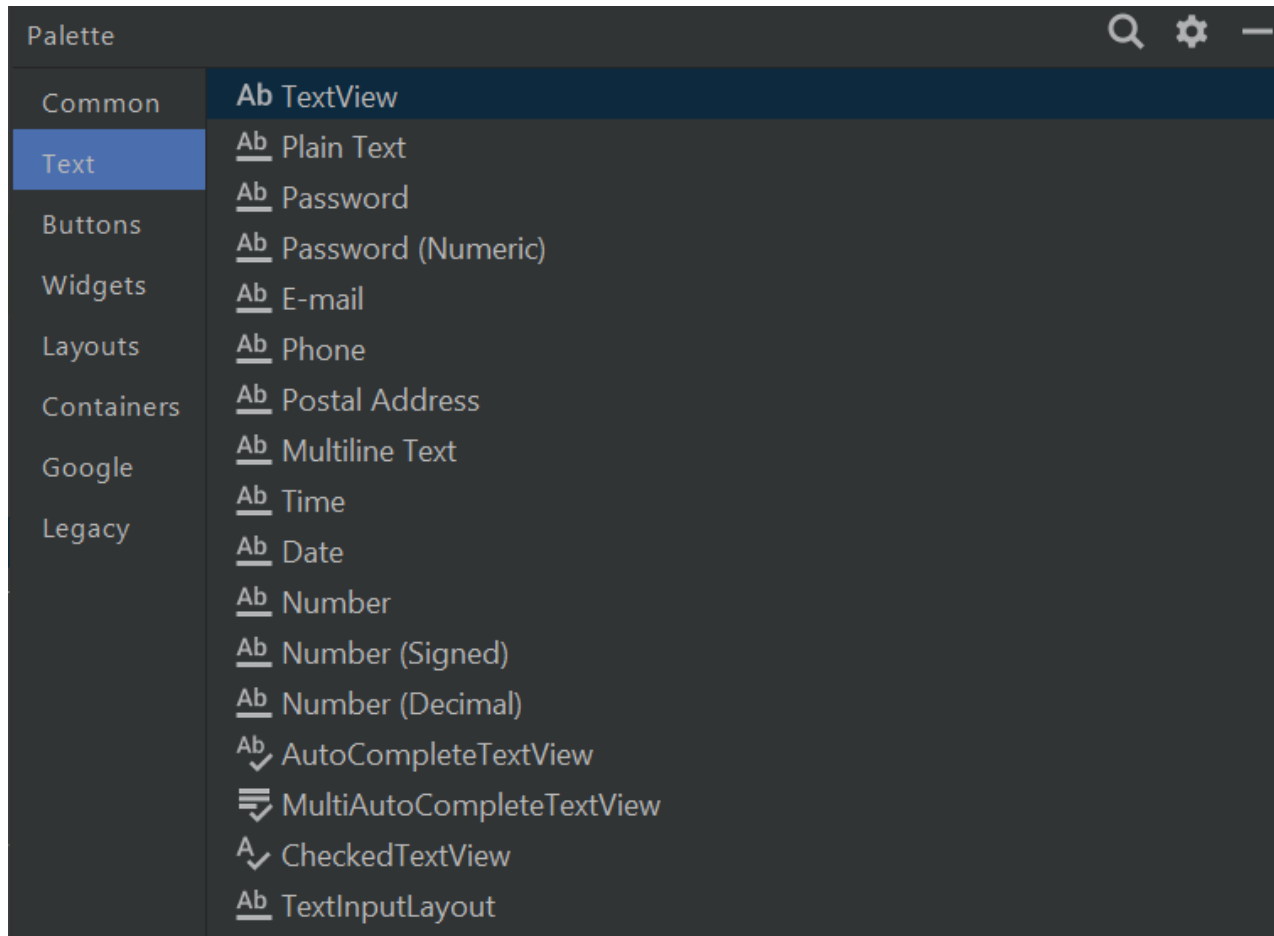
Common

กลุ่มของ Widgets ที่มักถูกใช้บ่อย ๆ เช่น การแสดงข้อความ (TextView), ปุ่ม (Button), แสดงรูปภาพ (ImageView)



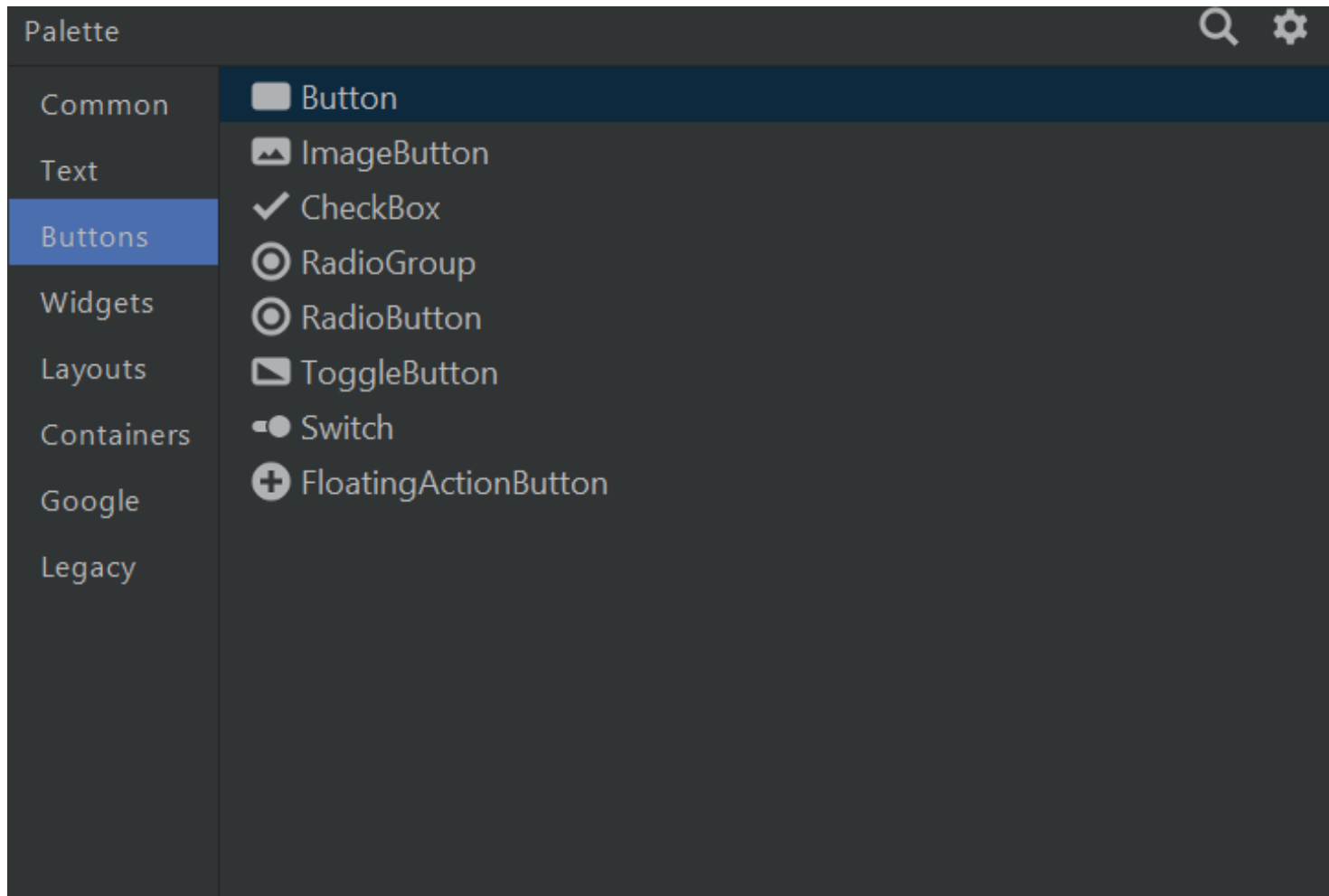
Text

กลุ่มของ Widgets ที่เกี่ยวข้องกับข้อความ เช่น การแสดงข้อความ (TextView), ช่องกรอกข้อมูล (Plain Text), ช่องกรอกรหัสผ่าน (Password) ฯลฯ



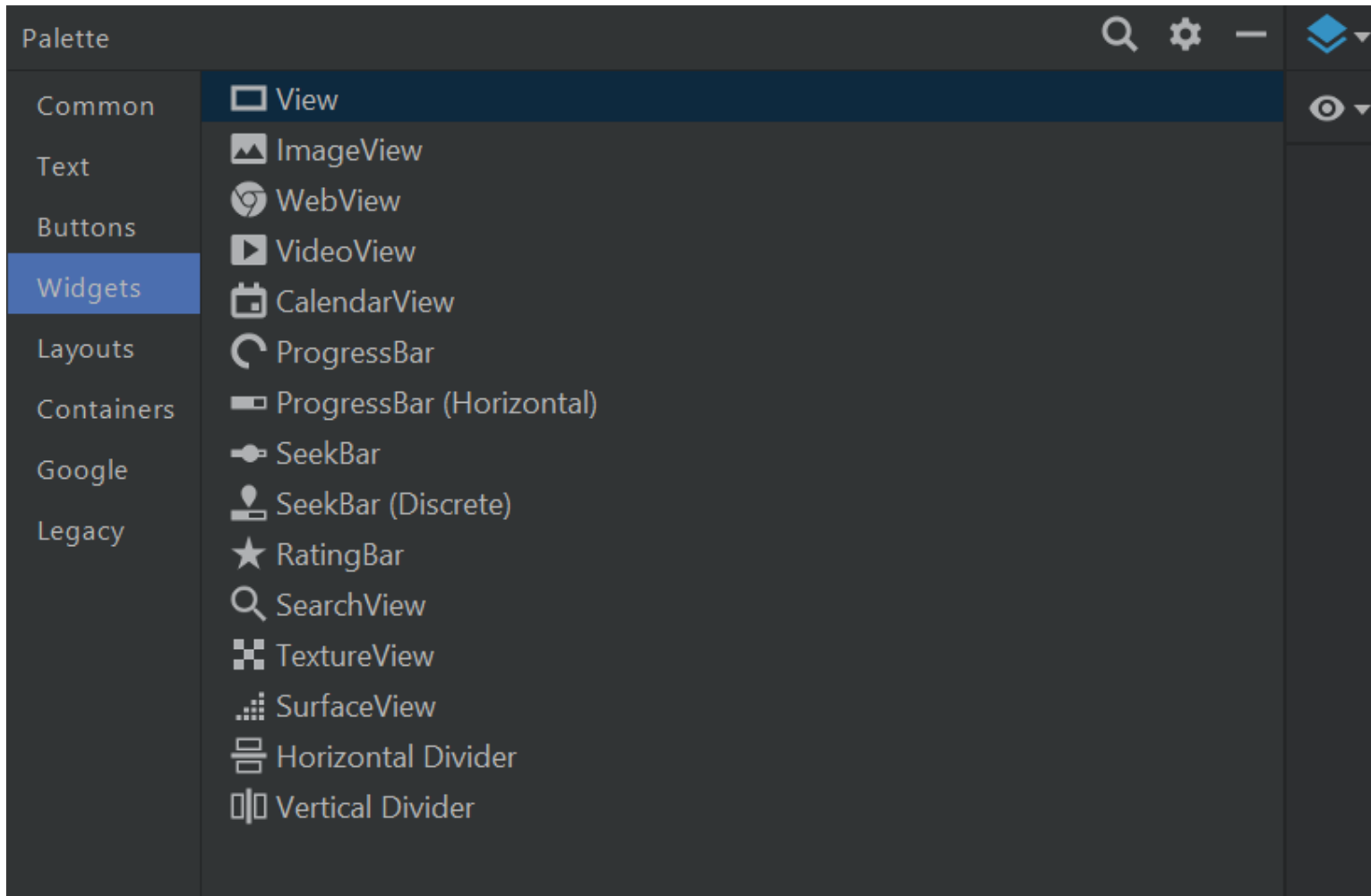
Buttons

กลุ่มของ Widgets ที่เกี่ยวข้องกับปุ่ม เช่น ปุ่มปกติ (Button), ปุ่มรูปภาพ (ImageButton), เช็คบ็อก (CheckBox) ฯลฯ



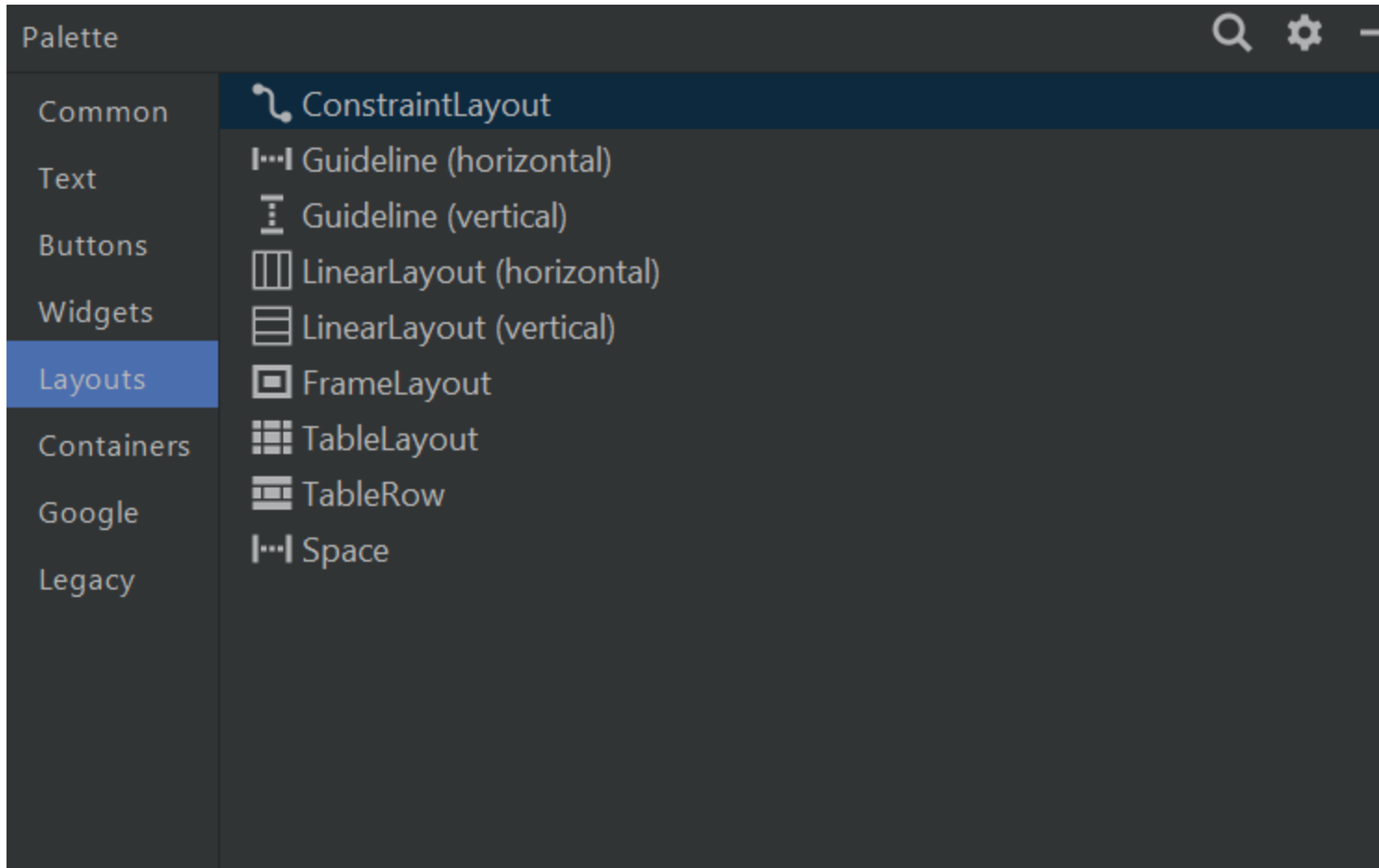
Widgets

กลุ่มของ Widgets อื่น ๆ เช่น การแสดงภาพ (ImageView), แสดงเว็บ (WebView), แสดงวิดีโอ (VideoView) ฯลฯ

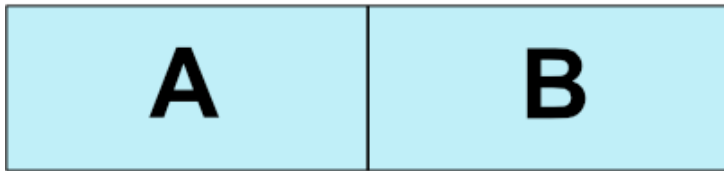
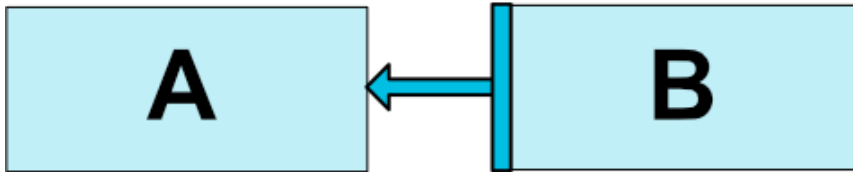


Layout

กลุ่มของ Layouts เช่น ConstraintLayout, LinearLayout, Table Layout ฯลฯ



Constraint Layout



Widgets จะถูกจัดวางโดยการอ้างอิงถึงตำแหน่งซึ่งกันและกัน เช่นเดียวกับ Relative Layout แต่ดำเนินการผ่านทาง GUI โดยตรง

ConstraintLayout allows you to create large and complex layouts with a flat view hierarchy (no nested view groups). It's similar to RelativeLayout in that all views are laid out according to relationships between sibling views and the parent layout, but it's more flexible than RelativeLayout and easier to use with Android Studio's Layout Editor.

Examples

First name

Last name

Address

District

Province

Zip code

SUBMIT



Linear Layout

เป็น Layout ที่จะทำให้ Widgets ที่อยู่ภายในหน้าจอจะเรียงในแนวเดียวกัน เช่น จากซ้ายไปขวา (horizontally) หรือ จากบนลงล่าง (vertically) โดยการกำหนดทิศทางของ Layout ด้วย `android:orientation` attribute.

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

ประกาศ Name space

```
    android:layout_width="match_parent" กว้างเต็มหน้าจอ
```

```
    android:layout_height="match_parent" สูงเต็มหน้าจอ
```

```
    android:paddingLeft="16dp" ขอบซ้าย 16 dp
```

```
    android:paddingRight="16dp" ขอบขวา 16 dp
```

```
    android:orientation="vertical" > แสดงผลแนวตั้ง
```

```
    <EditText
```

```
        android:layout_width="match_parent" กว้างเต็มหน้าจอ
```

```
        android:layout_height="wrap_content" สูงเท่าตัวหนังสือ
```

```
        android:hint="@string/to" /> คำใบ้
```

```
    <EditText
```

```
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
```

```
        android:hint="@string/subject" />
```

```
    <EditText
```

```
        android:layout_width="match_parent"
```

```
        android:layout_height="0dp" ไม่มีความสูง
```

```
        android:layout_weight="1" สูงเท่าส่วนที่เหลือของจอ
```

```
        android:gravity="top" ตัวหนังสืออยู่บนสุด
```

```
        android:hint="@string/message" />
```

```
    <Button
```

```
        android:layout_width="100dp"
```

```
        android:layout_height="wrap_content"
```

```
        android:layout_gravity="right"
```

```
        android:text="@string/send" />
```

```
</LinearLayout>
```

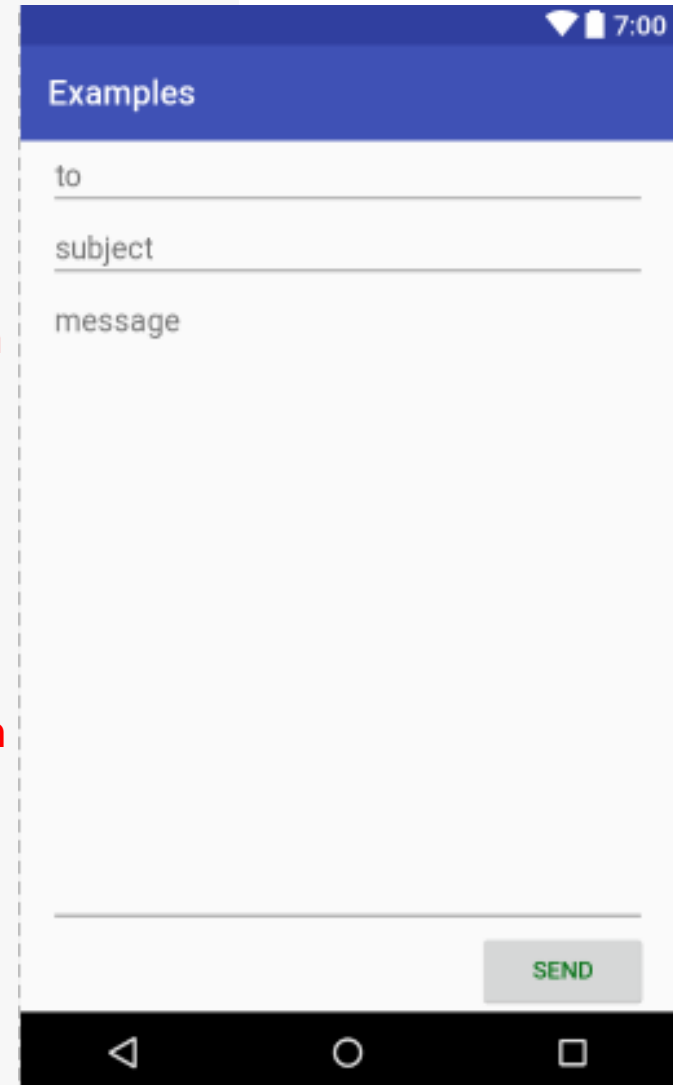
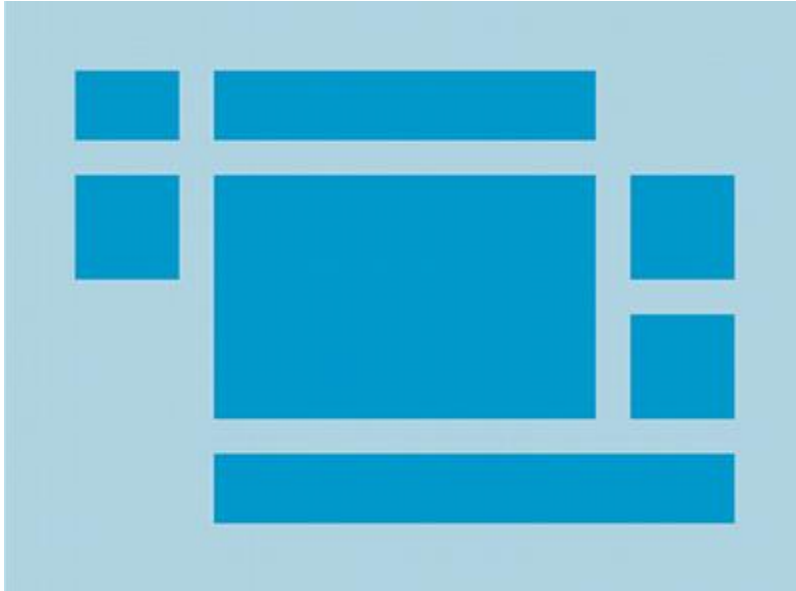


Table Layout



เป็น Layout ที่จะจัดเรียง Widgets ตาม rows and columns หรือ Cell ที่กำหนด

โดยใช้ `<TableRow>` element ในการสร้าง Row ให้กับ Table

```

<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="2">

```

```

<TableRow

```

```

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

```

```

    <ImageView

```

```

        android:id="@+id/imageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_column="1"
        app:srcCompat="@drawable/login" />

```

```

    <TextClock

```

```

        android:id="@+id/textClock"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_column="2"
        android:layout_gravity="bottom" />

```

```

</TableRow>

```

```

<TableRow>

```

```

    <TextView android:text="First Name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_column="1" />

```

```

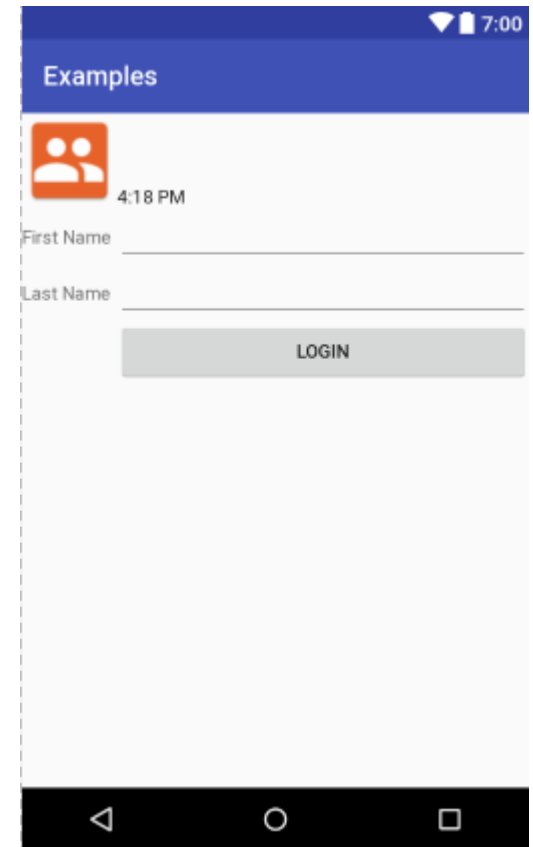
    <EditText android:width="200px"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_column="2"/>

```

```

</TableRow>

```



```
<TableRow
```

```
    android:layout_width="match_parent"  
    android:layout_height="match_parent">
```

```
    <TextView
```

```
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_column="1"  
        android:text="Last Name" />
```

```
    <EditText
```

```
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_column="2"  
        android:width="100px" />
```

```
</TableRow>
```

```
<TableRow
```

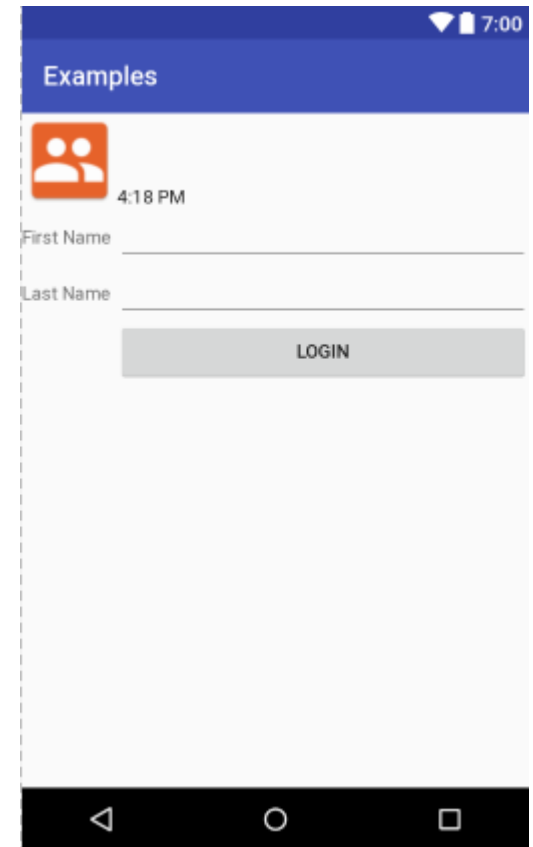
```
    android:layout_width="match_parent"  
    android:layout_height="match_parent">
```

```
    <Button
```

```
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:text="Login"  
        android:id="@+id/button"  
        android:layout_column="2" />
```

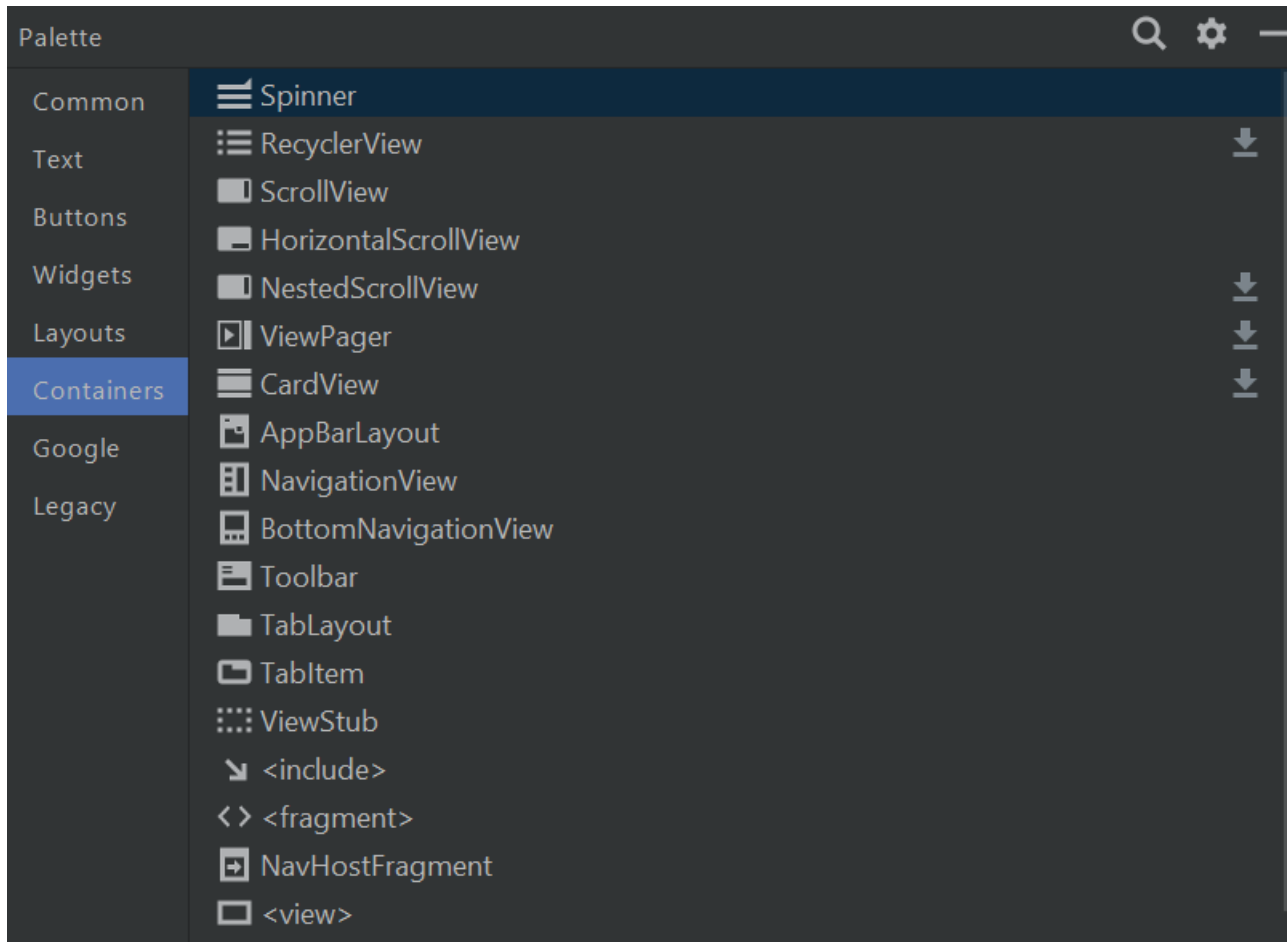
```
</TableRow>
```

```
</TableLayout>
```



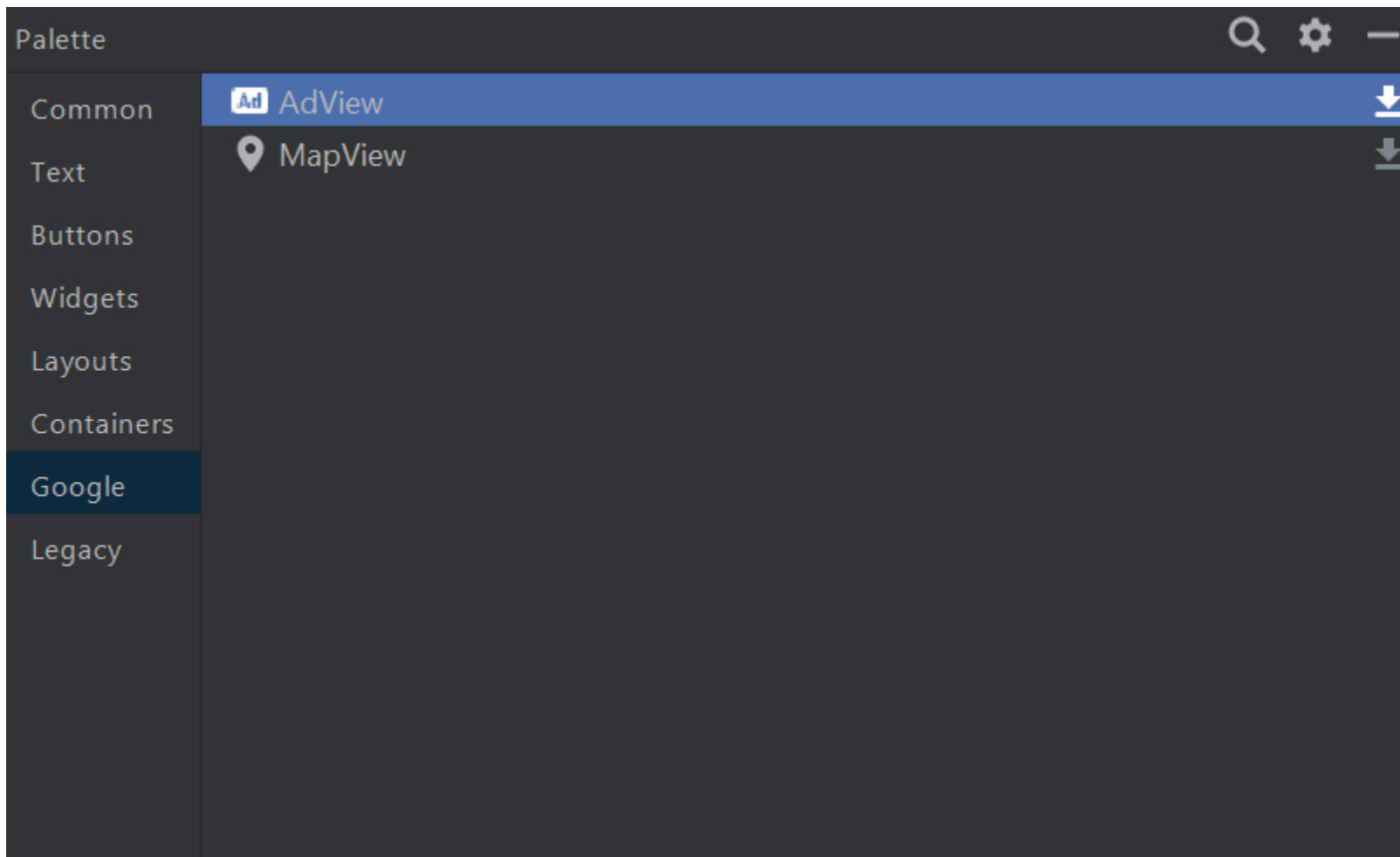
Containers

กลุ่มของ Widgets ที่เป็น Containers เช่น แสดงรายการข้อมูลแบบ Dropdown (Spinner), แสดงรายการข้อมูล (RecyclerView), แสดง Toolbar (Toolbar) ฯลฯ



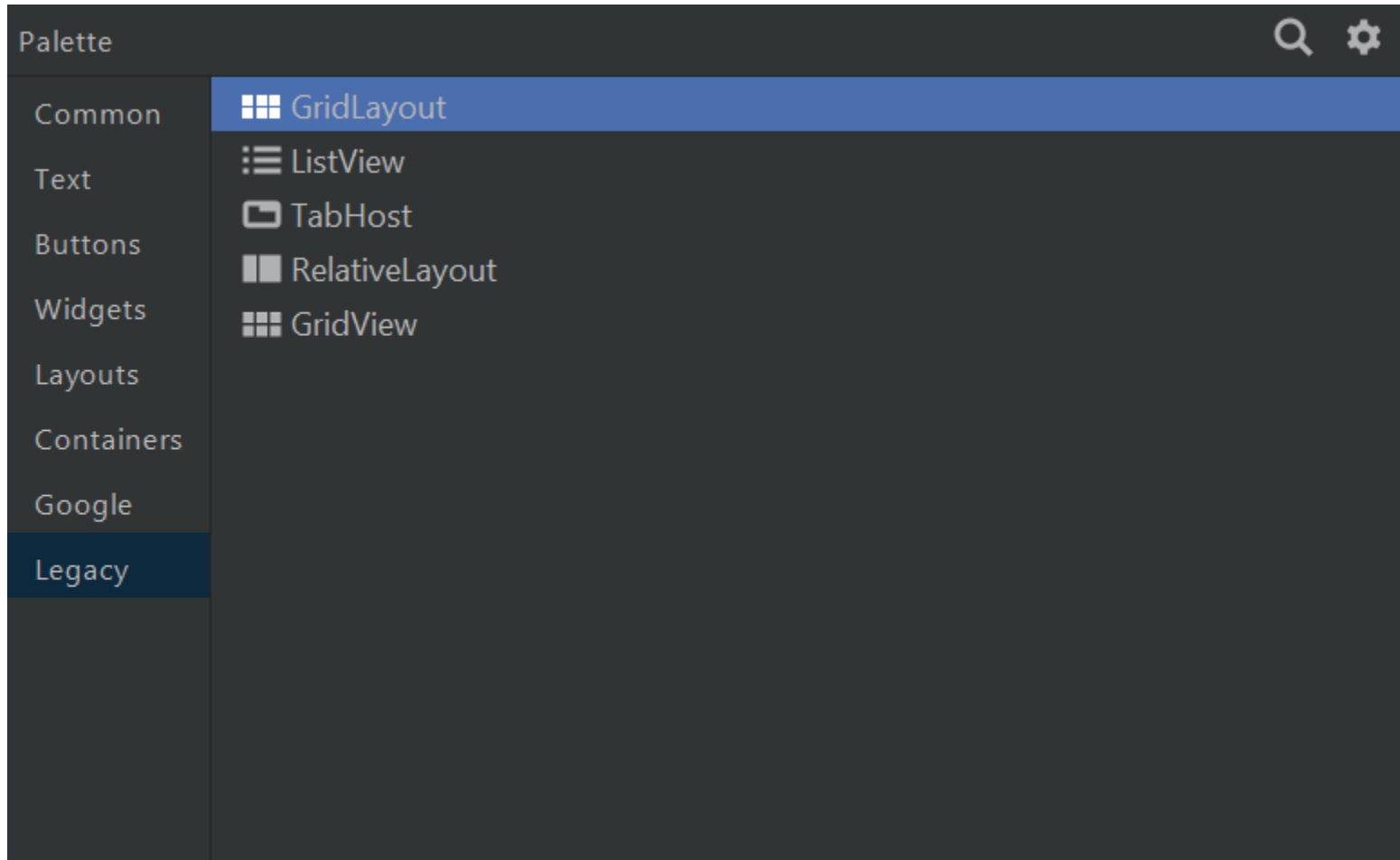
Google

กลุ่มของ Google Widgets ประกอบด้วย แสดงโฆษณา (AdView) และแสดง Google Map (MapView)



Legacy

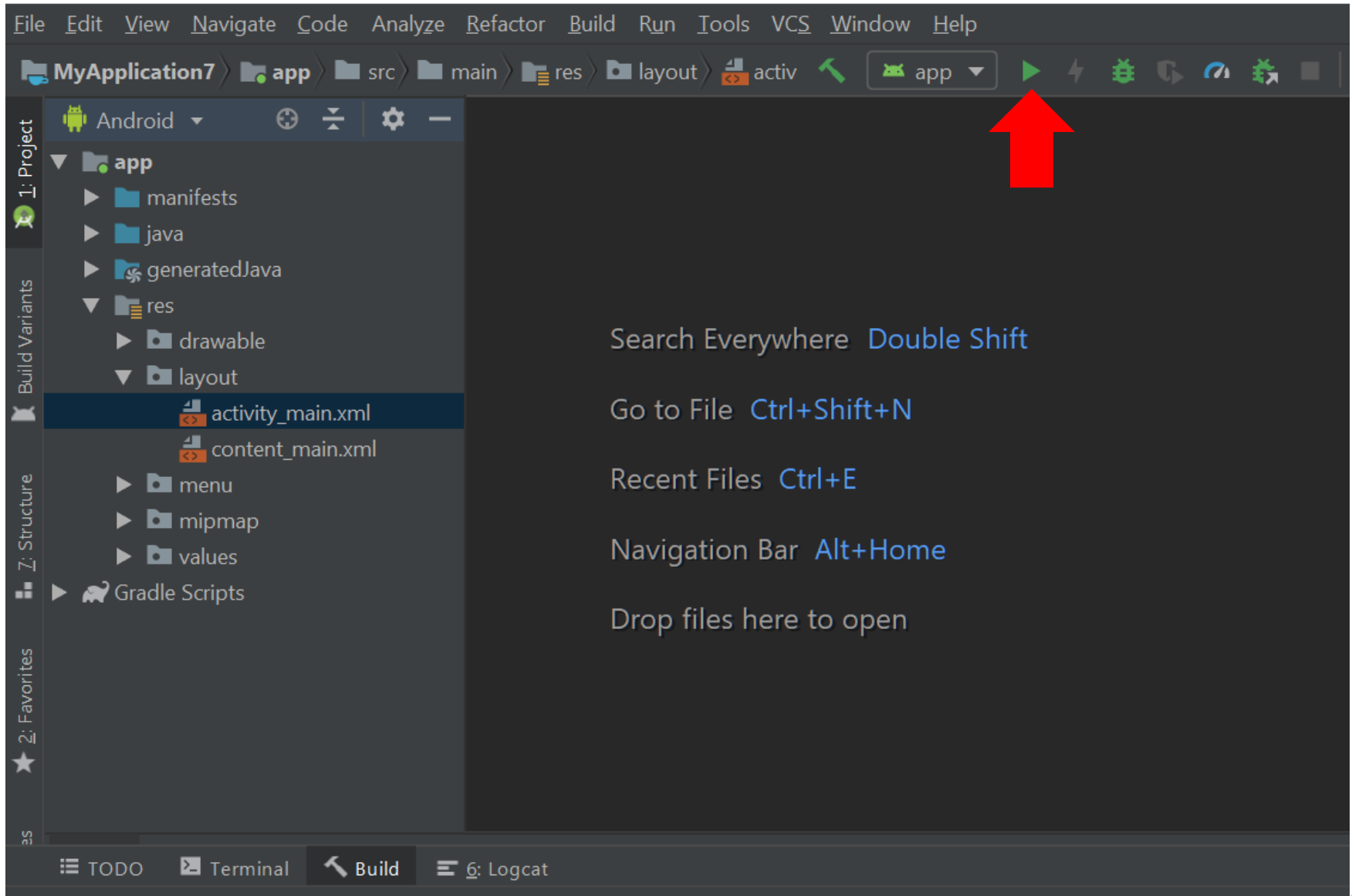
กลุ่มของ Widgets เก่า เช่น GridLayout, ListView, RelativeLayout, GridView ฯลฯ

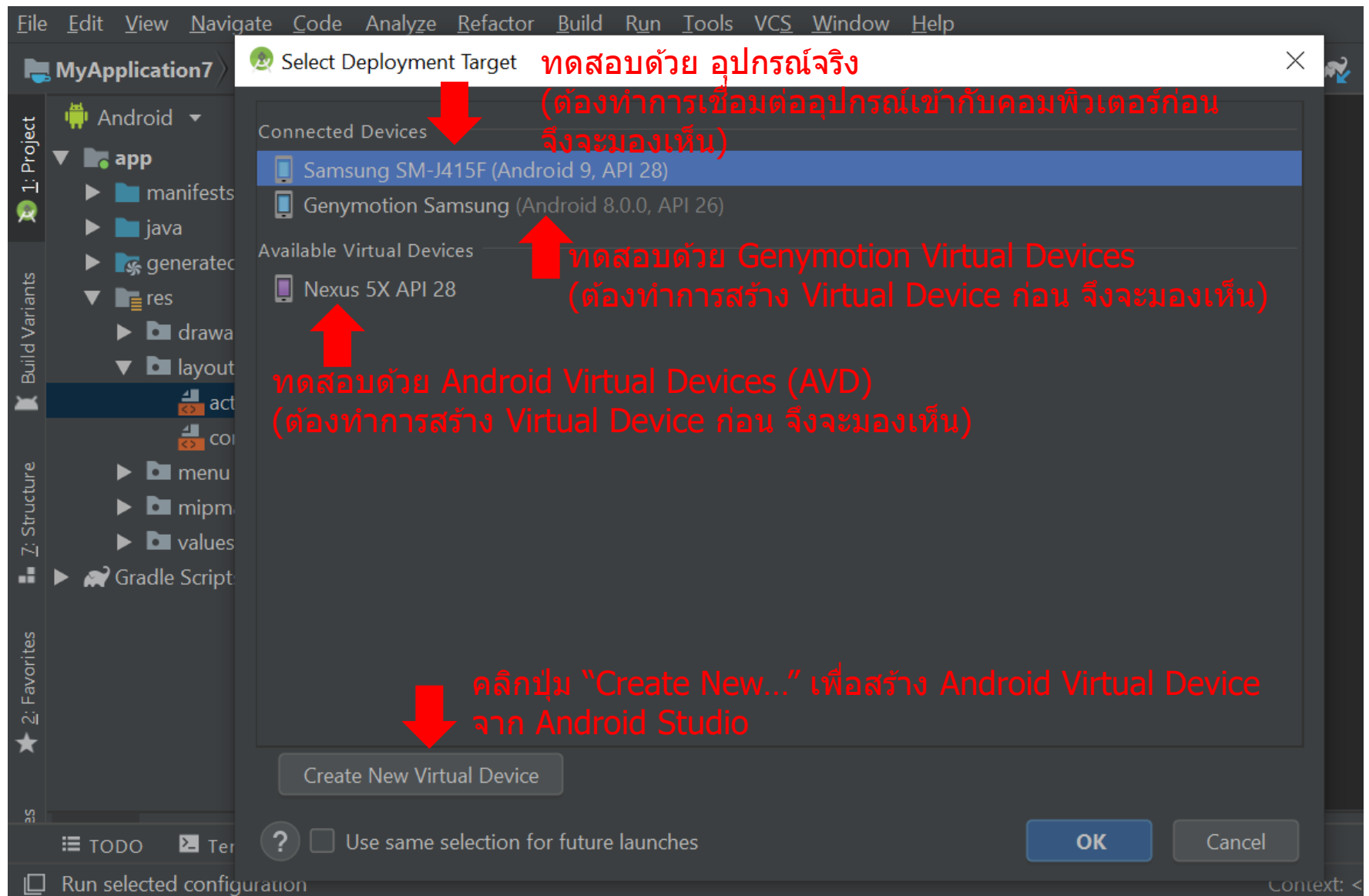


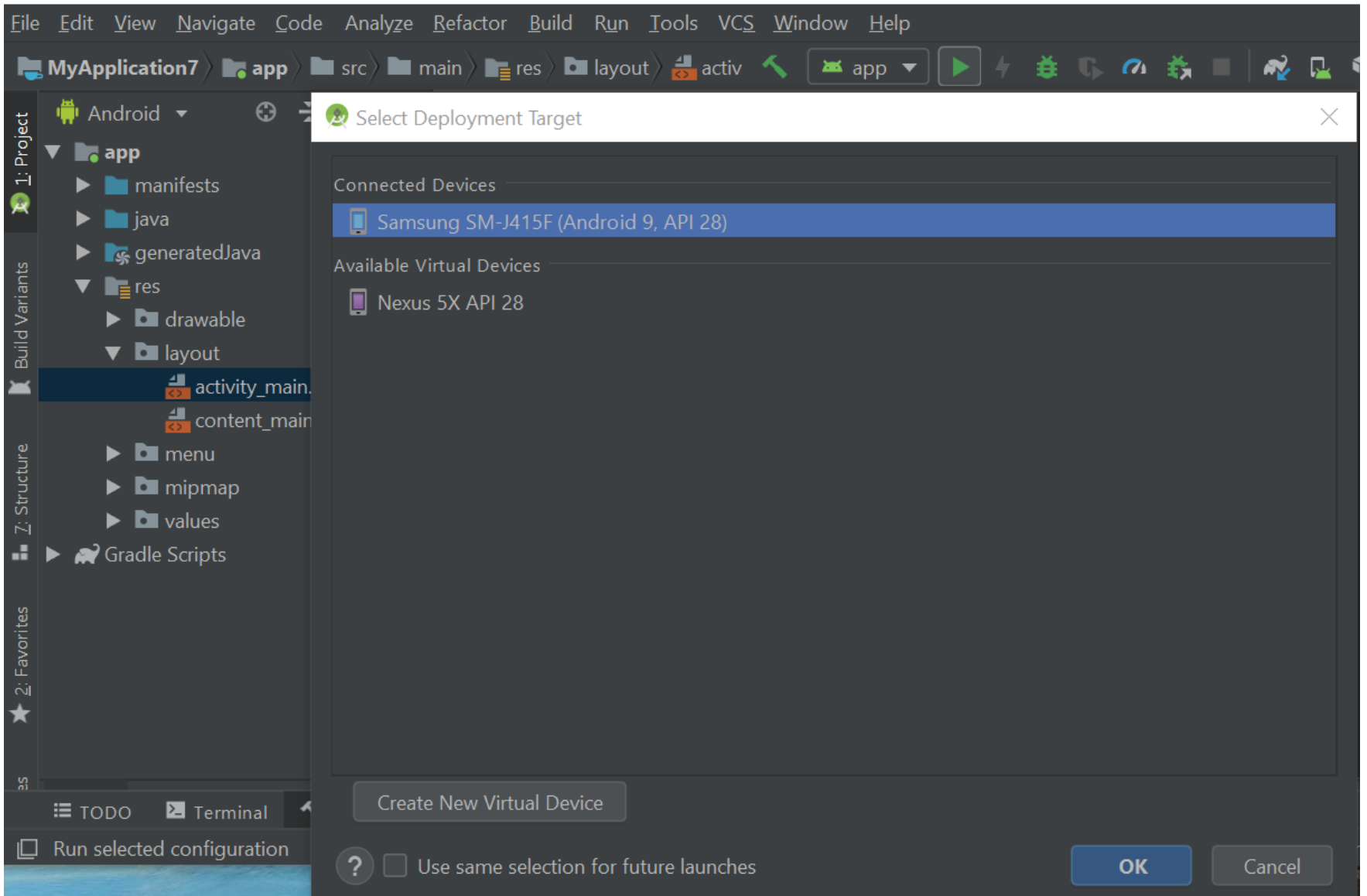
1.6 การทดสอบบนอุปกรณ์

1. ตรวจสอบ Android Version ในอุปกรณ์
โดยไปที่ **Settings -> About phone -> Software information**
2. เปิดใช้งาน USB debugging
โดยไปที่ **Settings -> About phone -> Software information**
แล้วกดที่ **Build number** 7 ครั้ง เพื่อให้แสดง **Developer options**
จากนั้นไปที่ **Settings -> Developer options** แล้วคลิก "On" และ
กำหนด USB debugging เป็น "On" เช่นกัน

1.7 การทดสอบ Application

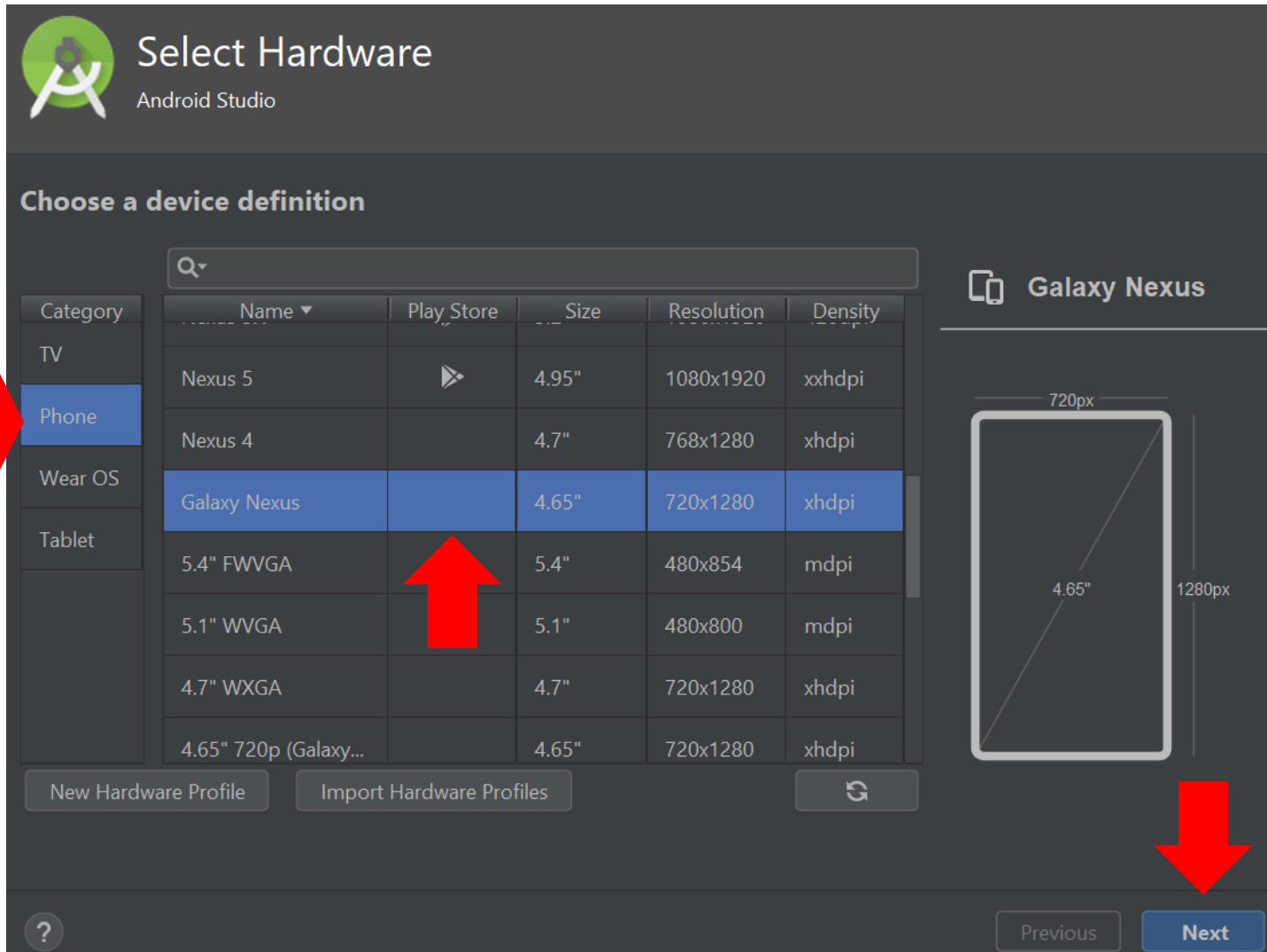






การสร้าง Android Virtual Device (AVD)

หลังจากคลิกปุ่ม "Create New..." แล้วจะพบหน้าให้เลือกอุปกรณ์



1.8 การ publish application ไปยัง google play

โดยเข้าไปที่: <https://play.google.com/apps/publish/signup/>



Sign-in with your Google account



Accept Developer Agreement



Pay Registration Fee



Complete your Account details

You are signed in as...



This is the Google account that will be associated with your Developer Console.

If you would like to use a different account, you can choose from the following options below. If you are an organization, consider registering a new Google account rather than using a personal account.

[SIGN IN WITH A DIFFERENT ACCOUNT](#)

[CREATE A NEW GOOGLE ACCOUNT](#)

Accept developer agreement

Read and agree to the [Google Play Developer distribution agreement](#).



I agree and I am willing to associate my account registration with the Google Play Developer distribution agreement.

Review distribution countries

Review the distribution countries where you can distribute and sell applications.

If you are planning to sell apps or in-app products, check if you can have a merchant account in your country.

Credit card

Make sure you have your credit card handy to pay the \$25 registration fee in the next step.

CONTINUE TO PAYMENT

2. การสร้างหน้า Login

การสร้างหน้า Login ประกอบด้วย

- ☐ Config Session และ API
- ☐ สร้าง Layout ของหน้า Login
- ☐ เขียน Code สำหรับหน้า Login

2.1 Config ค่าต่าง ๆ ที่จำเป็น

1. ไฟล์ AndroidManifest.xml

ประกาศ LoginActivity (ชื่อต้องเหมือนกันกับ ชื่อ Class) ให้สามารถใช้งานได้ และกำหนดให้ทำงานเป็นหน้าจอแรก

```
<activity
    android:name=".LoginActivity"
    android:label="Login">

    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

2. ไฟล์ strings.xml

กำหนดที่อยู่ของ API และข้อความที่เกี่ยวข้องกับการ Login

```
<resources>
    <!-- RESTful API-->
    <string name="root_url">http://172.20.10.8/ci_sale/</string>
    <string name="login_url">index.php/api/user/</string>

    <!-- Login Activity -->
    <string name="login_tex_description">
        Please enter your username and password.</string>
    <string name="login_button_description">Click here to login</string>
    <string name="username"></string>
    <string name="password"></string>
    <string name="logout">Logout</string>
    <string name="exit">Exit without logout</string>
```

3. ไฟล์ strings.xml

กำหนดที่อยู่ของ API และข้อความที่เกี่ยวข้องกับการ Login

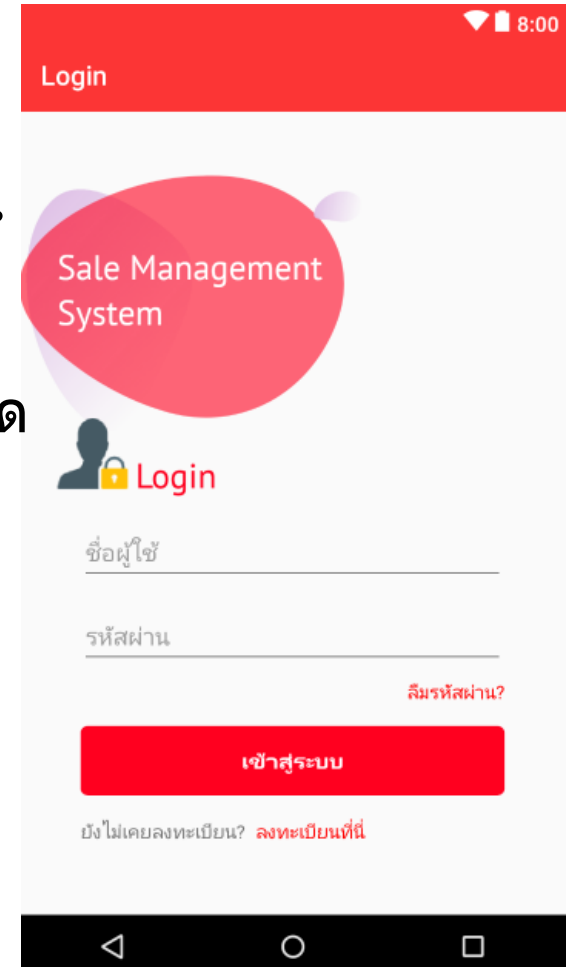
```
<resources>
    <!-- RESTful API-->
    <string name="root_url">http://172.20.10.8/ci_sale/</string>
    <string name="login_url">index.php/api/user/</string>

    <!-- Login Activity -->
    <string name="login_tex_description">
        Please enter your username and password.</string>
    <string name="login_button_description">Click here to login</string>
    <string name="username"></string>
    <string name="password"></string>
    <string name="logout">Logout</string>
    <string name="exit">Exit without logout</string>
```

2.2 สร้าง Layout

สร้าง Layout โดยเก็บไว้ที่ rs/layout/activity_login.xml ภายในไฟล์ประกอบไปด้วย 6 widgets ที่สำคัญ นั่นคือ

1. ConstraintLayout สำหรับวาง Layout
2. TextView สำหรับแสดงข้อความต่าง ๆ
3. EditText สำหรับช่องกรอกชื่อผู้ใช้และรหัสผ่าน
4. Button สำหรับกดเข้าสู่ระบบ
5. ImageView สำหรับแสดงรูปภาพ
6. ProgressBar สำหรับแสดง Progress การโหลด



2.3 เขียน Code สำหรับหน้า Login

โค้ดเก็บไว้ที่ java/[your SW package]/LoginActivity.java ภายในไฟล์ประกอบไปด้วย 4 methods และ 1 subclass นั่นคือ

1. onCreate method

สำหรับ Render หน้าจอ และอ่านค่าจาก Activity Layout

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_login);
    mLoginFormView = findViewById(R.id.login_form);
    mProgressView = findViewById(R.id.login_progress);
    txtUsername = findViewById(R.id.txtRobotName1);
    txtPassword = findViewById(R.id.txtPassword);
}
```

2. onResume method

สำหรับตรวจสอบว่ามีการ Login ค้างไว้อยู่หรือไม่ ถ้าใช่จะ Redirect ไปยังหน้า Main โดยไม่ต้องทำการ Login ซ้ำ

```
//Go to the main page if a user has logged-in
@Override
protected void onResume() {
    sharedPreferences=getSharedPreferences(APP_PREFER,
Context.MODE_PRIVATE);
    if ((sharedPreferences.contains(USERNAME_PREFER)) &&
sharedPreferences.contains(PASSWORD_PREFER)) {
        Intent i = new Intent(this, MainActivity.class);
        startActivity(i);
        finish();
    }
    super.onResume();
}
```

3. login method

สำหรับตรวจสอบว่ามีการกรอก username และ password หรือไม่
ถ้าใช่จะแสดง Progress แล้วทำการ Login ด้วย UserLoginTask

```
public void login(View v) {  
  
    if (mAuthTask != null) { return; }  
    txtUsername.setError(null);  
    txtPassword.setError(null);  
  
    String username = txtUsername.getText().toString();  
    String password = txtPassword.getText().toString();  
  
    if (TextUtils.isEmpty(username)) {  
        txtUsername.setError("กรุณาระบุชื่อผู้ใช้");  
        txtUsername.requestFocus();  
    } else if (TextUtils.isEmpty(password)) {  
        txtPassword.setError("กรุณาระบุรหัสผ่าน");  
        txtPassword.requestFocus();  
    } else {  
        showProgress(true);  
        mAuthTask = new UserLoginTask(username, password);  
        mAuthTask.execute((Void) null);  
    }  
}
```

4. showProgress method

สำหรับซ่อน Login form แล้วแสดง Progress

```
private void showProgress(final boolean show) {

    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.HONEYCOMB_MR2) {
        int shortAnimTime =
getResources().getInteger(android.R.integer.config_shortAnimTime);

        mLoginFormView.setVisibility(show ? View.GONE : View.VISIBLE);
        mLoginFormView.animate().setDuration(shortAnimTime).alpha(
            show ? 0 : 1).setListener(new AnimatorListenerAdapter() {
            @Override
            public void onAnimationEnd(Animator animation) {
                mLoginFormView.setVisibility(show ? View.GONE : View.VISIBLE);
            }
        });
        mProgressView.setVisibility(show ? View.VISIBLE : View.GONE);
        mProgressView.animate().setDuration(shortAnimTime).alpha(
            show ? 1 : 0).setListener(new AnimatorListenerAdapter() {
            @Override
            public void onAnimationEnd(Animator animation) {
                mProgressView.setVisibility(show ? View.VISIBLE : View.GONE);
            }
        });
    } else {
        // The ViewPropertyAnimator APIs are not available, so simply show
        // and hide the relevant UI components.
        mProgressView.setVisibility(show ? View.VISIBLE : View.GONE);
        mLoginFormView.setVisibility(show ? View.GONE : View.VISIBLE);
    }
}
```


5. UserLoginTask class

สำหรับตรวจสอบ username แล้ว Password ในฐานข้อมูล ถ้าถูกต้อง จะทำการสร้าง Session แล้ว Redirect ไปยังหน้า Main

```
public class UserLoginTask extends AsyncTask<Void, Void, Boolean> {

    private final String mUsername;
    private final String mPassword;
    Boolean isSuccess = false;

    UserLoginTask(String username, String password) {
        mUsername = username;
        mPassword = password;
    }

    @Override
    protected Boolean doInBackground(Void... params) {
        try {
            // Simulate network access.
            Thread.sleep(2000);
        } catch (InterruptedException e) {
            return false;
        }
        //login
        String url = getString(R.string.root_url) + getString(R.string.login_url);
        RequestParams rp = new RequestParams();
        rp.add("username", txtUsername.getText().toString());
        rp.add("password", txtPassword.getText().toString());
    }
}
```

```

@Override
public void onSuccess(int statusCode, Header[] headers, JSONObject response) {
    try {
        JSONObject obj = new JSONObject(response.toString());
        String status = (String) obj.get("status");
        //Log.e("Message", status);
        if(status.equals("true")){
            JSONArray data = obj.getJSONArray("data");
            JSONObject item = data.getJSONObject(0);

            //Create shared preference to store user data
            SharedPreferences.Editor editor = sharedPrefer.edit();
            editor.putString(USERNAME_PREFER,
txtUsername.getText().toString());
            editor.putString(PASSWORD_PREFER,
txtPassword.getText().toString());
            editor.putString(EMP_TYPE_PREFER, item.getString("empType"));
            editor.commit();

            //Go to main page
            Intent intent = new Intent(getApplicationContext(),
MainActivity.class);

            startActivity(intent);
            isSuccess = true;
        }else{}
    } catch (JSONException e) {
        e.printStackTrace();
    }
}

});

return isSuccess;
}

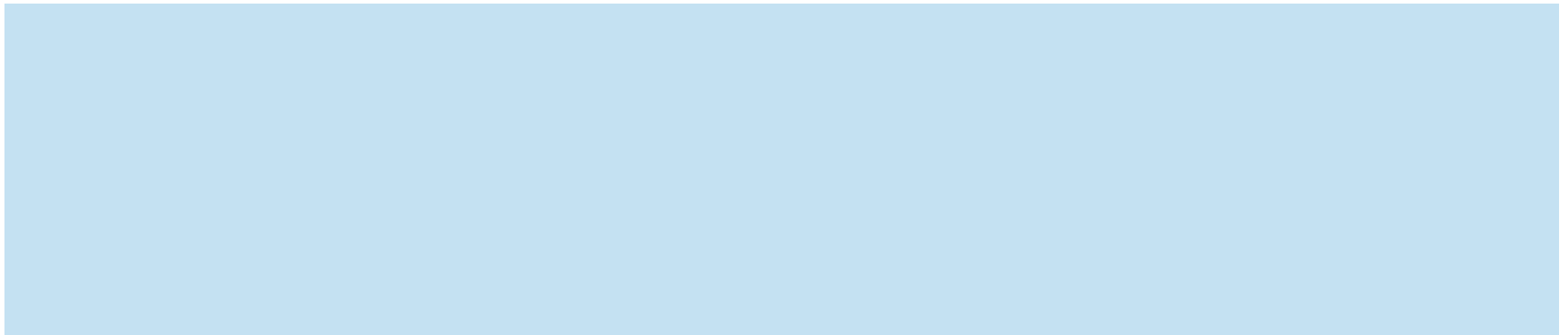
```

```
@Override
protected void onPostExecute(final Boolean success) {
    mAuthTask = null;
    showProgress(false);

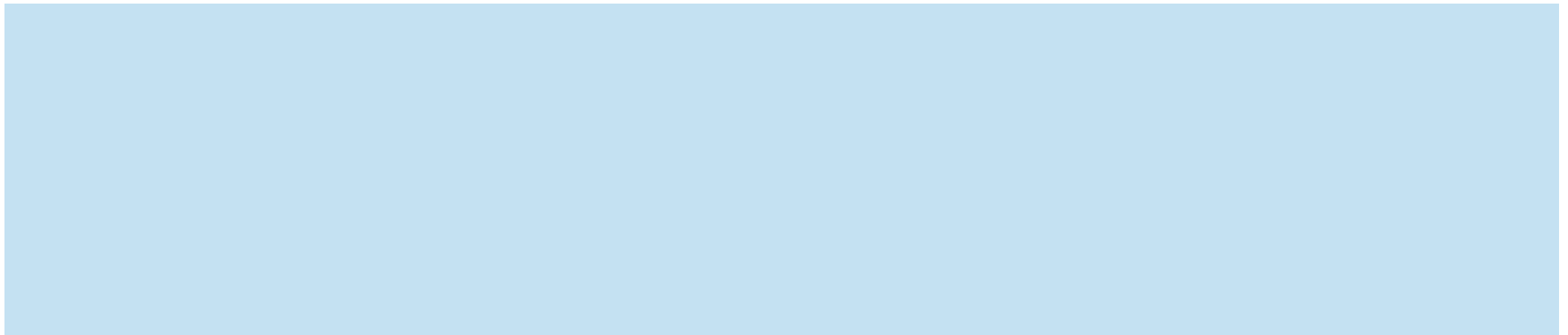
    if (success) {
        finish();
    } else {
        txtUsername.setError("ชื่อผู้ใช้ หรือ รหัสผ่านไม่ถูกต้อง");
        txtUsername.requestFocus();
    }
}

@Override
protected void onCancelled() {
    mAuthTask = null;
    showProgress(false);
}
}
```

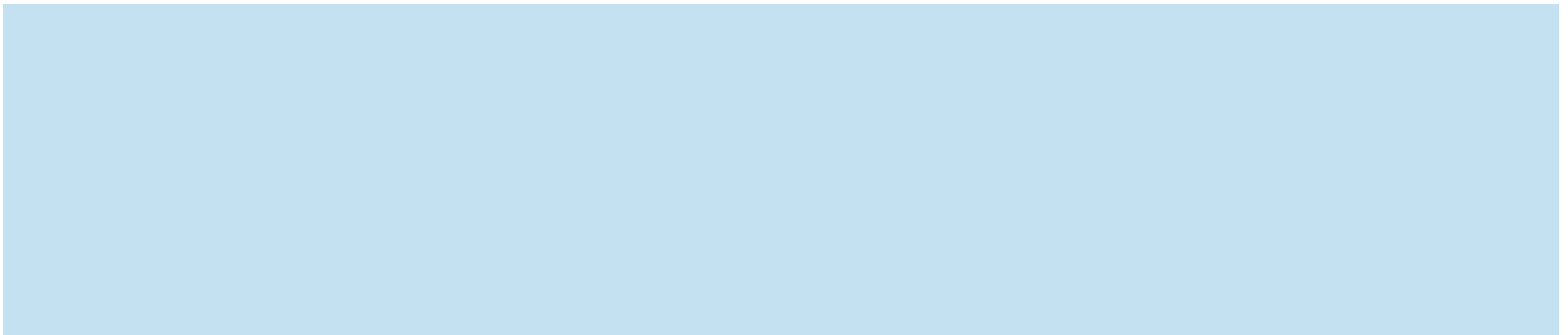
3. การเพิ่ม ลบ แก้ไข และแสดงรายการ



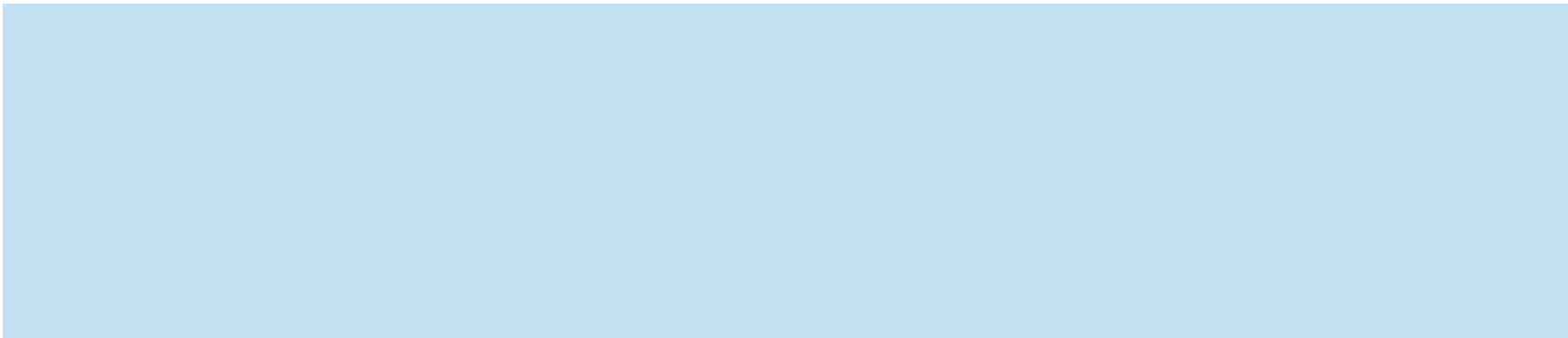
4.การแสดงกราฟแท่ง กราฟวงกลม กราฟเส้น และเกจชาร์ต



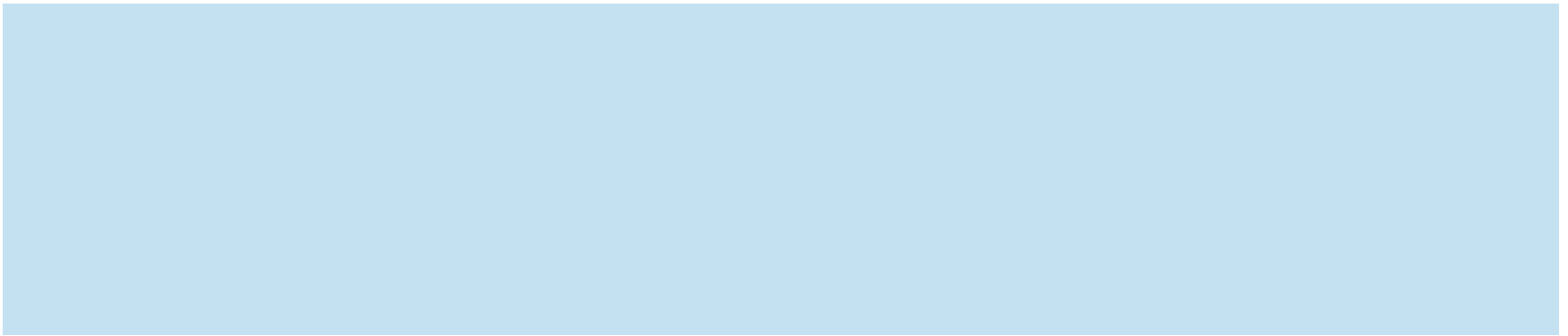
5. การจัดการกับกล้องและวิดีโอ (Camera and Video)



6. การสร้างและอ่าน QR Code



7. การแสดงตำแหน่งบน Google Map



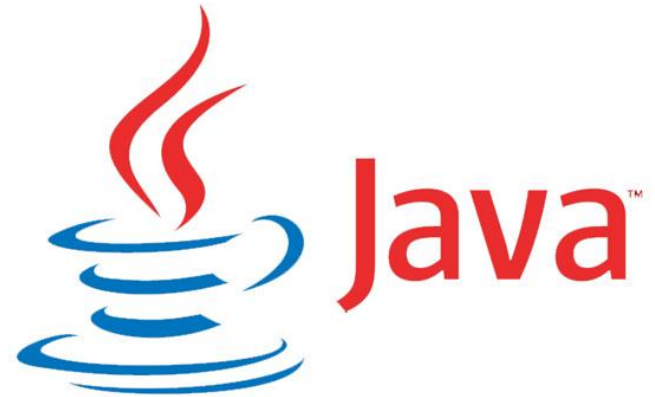
ภาคผนวก ก : ทบทวนภาษา Java

เนื้อหาเบื้องต้นในการเขียนโปรแกรมด้วยภาษา Python ประกอบด้วย

- ☐ แนะนำภาษา Java
- ☐ การตั้งชื่อตัวแปร (Variable Naming)
- ☐ ชนิดข้อมูล (Data Type)
- ☐ อาร์เรย์ (Array)
- ☐ การแสดงผลทางจอภาพและการ Comment
- ☐ การแปลงชนิดข้อมูล (Data Type Conversion)
- ☐ ตัวดำเนินการ (Operator)
- ☐ คำสั่งควบคุม (Control Flow Statement)
- ☐ หลักการ Object-Oriented Programming

แนะนำภาษา Java

- Object-Oriented Programming (OOP)
- Cross-platform programming language
- Open-source
- A huge community support
- 2th popular programming ranking (IEEE Spectrum 2019)



<https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2019>

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/index.html>

<https://www.tutorialspoint.com/java/>

การตั้งชื่อตัวแปร (Variable Naming)

- ห้ามขึ้นต้นชื่อด้วยตัวเลข
- ชื่อตัวแปรประกอบด้วยตัวอักษร ตัวเลข หรือเครื่องหมายขีดเส้นใต้ "_" หรือเครื่องหมาย "\$"
- ตัวหนังสือภาษาอังกฤษตัวใหญ่ต่างจากตัวเล็ก (Case Sensitive)

การแสดงผลทางจอภาพและการ Comment

ตัวอย่างการแสดงผลทางจอภาพ

```
System.out.println("Hello World!");
```

ตัวอย่างการ Comment

```
// Comment บรรทัดเดียว
```

```
/*  
    Comment  
หลายบรรทัดเดียว  
*/
```

ชนิดข้อมูล (Data Type)

Group	Type	Size (bit)	Range
Boolean	boolean	1	true/false
Integer	byte	8	-128 to 127
	short	16	-32,768 to 32,767
	int	32	-2,147,483,648 to 2,147,483,647
	long	64	-2^{63} to $(2^{63} - 1)$
Floating-Point	float	32	$\pm 1.5 \times 10^{-45}$ to $\pm 3.4 \times 10^{38}$
	double	64	$\pm 5.0 \times 10^{-324}$ to $\pm 1.7 \times 10^{-308}$
Character	char	16	any Unicode character (16 bit)

*String type: Java programming language also provides special support for character strings via the [java.lang.String](#) class

ตัวอย่างการประกาศตัวแปร

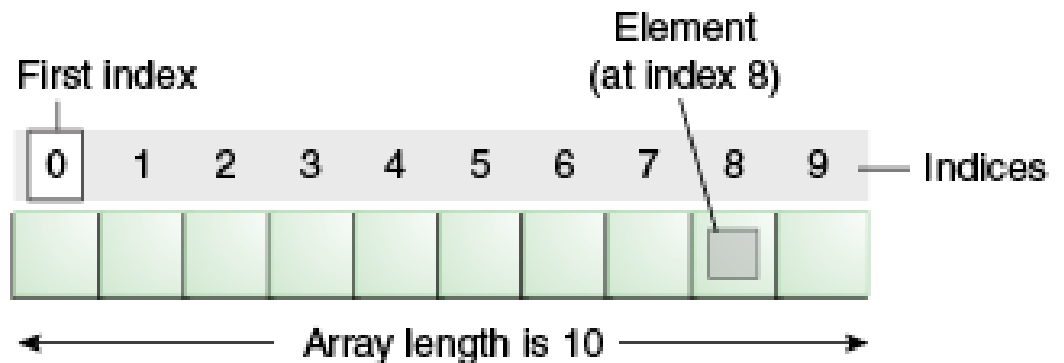
```
class ArrayDemo {  
    public static void main(String[] args) {  
        int a, b, c;           // Declares three ints, a, b, and c.  
        int a = 10, b = 10;    // Example of initialization  
        byte B = 22;          // initializes a byte type variable B.  
        double pi = 3.14159;  // declares and assigns a value of PI.  
        char a = 'a';         // the char variable a is initialized with value 'a'  
  
        System.out.println(a+b);  
  
    }  
}
```

อาร์เรย์ (Array)

```
// declares an array of integers
int[] anArray;
anArray = new int[10]; // allocates memory for 10 integers
anArray[0] = 100;
anArray[1] = 200;
// and so forth
anArray[10] = 1000;
```

Alternatively, you can use the shortcut syntax to create and initialize an array:

```
int[] anArray = {
    100, 200, 300,
    400, 500, 600,
    700, 800, 900, 1000
};
```



```
System.out.println("Element 1 at index 0: " + anArray[0]);  
System.out.println("Element 2 at index 1: " + anArray[1]);  
System.out.println("Element 3 at index 2: " + anArray[2]);
```


ตัวดำเนินการ (Operator)

Type	Sign	Description	Example
Arithmetic (เครื่องหมายทางคณิตศาสตร์)	+	Addition (also used for String concatenation)	3+2
	-	Subtraction	3-2
	*	Multiplication	3*2
	/	Division	3/2
	%	Remainder or mod	3%2
Assignment (กำหนดค่า)	=	Simple assignment	a=3
	+=	Addition, then assignment	a+=3
	-=	Subtraction, then assignment	a-=3
	=	Multiplication , then assignment	a=3
	/=	Division, then assignment	a/=3
	%=	Remainder, then assignment	a%=3

Type	Sign	Description	Example
Unary (Arithmetic และ Assignment ด้วย operand ตัวเดียว)	+	Unary plus operator; indicates positive value	a+
	-	Unary minus operator; negates an expression	a-
	++	Increments a value by 1	a++
	--	Decrements a value by 1	a--
	!	Inverts the value of a boolean	!a
Comparison (การเปรียบเทียบ)	==	Equal to	if(a==b)
	!=	Not equal to	if(a!=b)
	>	Greater than	if(a>b)
	>=	Greater than or equal to	if(a>=b)
	<	Less than	if(a<b)
	<=	Less than or equal to	if(a<=b)

Type	Sign	Description	Example
Conditional (เงื่อนไข)	&&	Conditional-AND	if(a>b && a>c)
		Conditional-OR	if(a>b a>c)
	?:	Ternary (shorthand for if-then-else statement)	(a>b) ? "yes" : "no"
Bitwise (การดำเนินการ ในเลขฐานสอง)	~	Unary bitwise complement	~a
	&	Bitwise AND	a&b
	^	Bitwise exclusive OR	a^b
		Bitwise inclusive OR	a b
Bit Shift (การ shift bit)	<<	Signed left shift (คูณ 2 คูณ 2)	a<<
	>>	Signed right shift (หาร 2 หาร 2)	a>>

การแปลงชนิดข้อมูล (Data Type Conversion)

การแปลง	ตัวอย่าง
แปลงข้อความไปเป็นตัวเลข	<pre>float a = (Float.valueOf("5.12")).floatValue(); float b = (Float.valueOf("7.13")).floatValue(); System.out.println("a + b = " + (a + b)); float a = Float.parseFloat(args[0]); float b = Float.parseFloat(args[1]); System.out.println("a + b = " + (a + b));</pre>
แปลงตัวเลขไปเป็นข้อความ	<pre>int i; String s1 = "" + i; String s2 = String.valueOf(i); int i; double d; String s3 = Integer.toString(i); String s4 = Double.toString(d);</pre>

ตัวอย่างการแปลงข้อมูล

```
public class ToStringDemo {  
  
    public static void main(String[] args) {  
        double d = 858.48;  
        String s = Double.toString(d);  
  
        int dot = s.indexOf('.');  
  
        System.out.println(dot + " digits " +  
            "before decimal point.");  
        System.out.println( (s.length() - dot - 1) +  
            " digits after decimal point.");  
    }  
}
```

คำสั่งควบคุม (Control Flow Statement)

- ☐ IF-Then Statements
- ☐ Switch Statements
- ☐ For Statement
- ☐ While and Do-While Statements
- ☐ Branching Statements

The if-then Statements

```
if (condition) {  
    ...  
}
```

```
if (condition) {  
    ...  
}  
else {  
    ...  
}
```

```
int a = 10;  
int b = 20;  
if (a < b) // if block  
{  
    System.out.println("Yes");  
}  
else  
{  
    System.out.println("No");  
}
```

The Switch Statement

```
switch (expression)  
{  
  case value1 :  
    statement1;  
    break;  
  case value2 :  
    statement2;  
    break;  
  case valueN :  
    statementN;  
    break;  
  default :  
    statement;  
    break;  
}
```

```
string inp;  
string day;  
System.out.print("Please select number:");  
inp = Console.ReadLine();  
switch (inp)  
{  
    case "1":  
        day = "Sunday";  
        break;  
    case "2":  
        day = "Monday";  
        break;  
    case "3":  
        day = "Tuesday";  
        break;  
    default:  
        day = "none";  
        break;  
}  
System.out.println("This day is " + day);  
Console.Read();
```


The for Statement

```
for (counter = first value; counter  
    condition; adjust counter value) {  
    ...  
}
```

```
for(int i=1; i<10; i++){  
    System.out.println("Count is: " + i);  
}
```

The While Statements

```
while (condition) {  
    ...  
}
```

```
int count = 1;  
while (count < 10) {  
    System.out.println(count);  
    count++;  
}
```

The Do-while Statements

```
do {  
  
    ...  
} while (condition);
```

```
int count = 1;  
do  
{  
    System.out.println(count);  
    count++;  
} while (count < 10);
```

Branching Statements

- ❑ The break statement
- ❑ The continue Statement
- ❑ Return Statement

The break statement

An breaks statement terminates the innermost switch, for, while, or do-while statement.

```
for(int i=1; i<10; i++){  
    if (i == 5)  
    {  
        break;  
    }  
    System.out.println("Count is: " + i);  
}
```

The continue Statement

An continue statement skips current iteration of end of the innermost for, while, or do-while statement.

The return Statement

The return statement has two forms: one that returns a value, and one that doesn't.

1. The return a value Statement
2. The doesn't return value Statement

```
public string showMessage()  
{  
    return "Hello!";  
}
```

หลักการ Object-Oriented Programming

☐ Encapsulation

- ☐ wrapping the data (**variables**) and functions (**methods**) together as a single unit.
- ☐ hiding data and functions with public, private, and protect.

☐ Abstraction

- ☐ Abstract classes may or may not contain abstract methods, i.e., methods without body (public void get();)
- ☐ Interface classes contain only abstract methods.

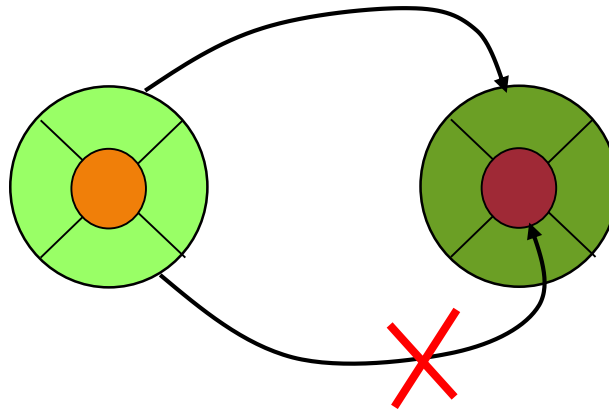
☐ Inheritance

- ☐ Each subclass inherits all variables of its superclass.

☐ Polymorphism

- ☐ Interface same despite different data types .

Encapsulation



- ❑ Method และ Variable ที่มีความสัมพันธ์กันถูกจัดอยู่ใน Class เดียวกัน
- ❑ Class Method และ Variable ถูกกำหนดการเข้าถึงด้วย private public และ protected (ข้อมูลและส่วนประมวลผลทั้งหมด ควรถูกซ่อนไว้เบื้องหลัง interface)

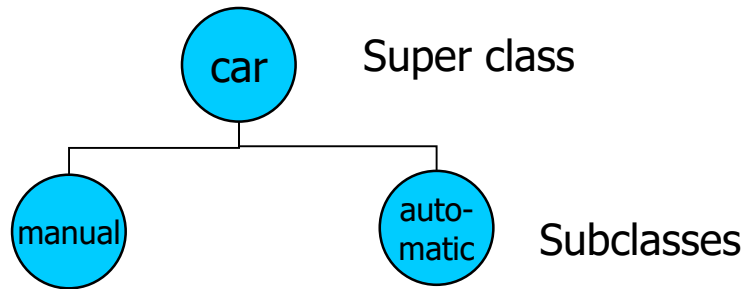
โค้ดตัวอย่าง ของ Encapsulation

```
private class CelsiusToFahrenheit{
    private double celsius;
    public double Celsius(double value){
        celsius = value;
    }

    public double MakeFahrenheit(){
        return (celsius * 9 / 5) + 32;
    }
}

private class EncapsulationTest{
    static void Main(string[] args){
        CelsiusToFahrenheit objCelsiusToFahren= new CelsiusToFahrenheit();
        objCelsiusToFahren.Celsius(37);
        System.out.println(objCelsiusToFahren.MakeFahrenheit());
    }
}
```

Inheritance



- ❑ ใน Java จะ inherit ได้เพียงแค่ class เดียว (Single inheritance) เหมือนกันกับ C# แต่แก้ปัญหานี้โดย สามารถ inherit ได้หลาย Interface (Multiple inheritance)
- ❑ Subclass มีคุณสมบัติมากกว่าหรือเท่ากับ Superclass

โค้ดตัวอย่าง ของ Inheritance: extends

```
public class SuperClass{
    public int getNb(){
        //specify what must happen
        return 1;
    }
    public int getNb2(){
        //specify what must happen
        return 2;
    }
}

public class SubClass extends SuperClass{
    public int getNb3(){
        return 3;
    }
}
```

```
Subclass s = new SubClass();
s.getNb(); //returns 1
s.getNb2(); //returns 2
s.getNb3(); //returns 3
```

โค้ดตัวอย่าง ของ Inheritance: interface

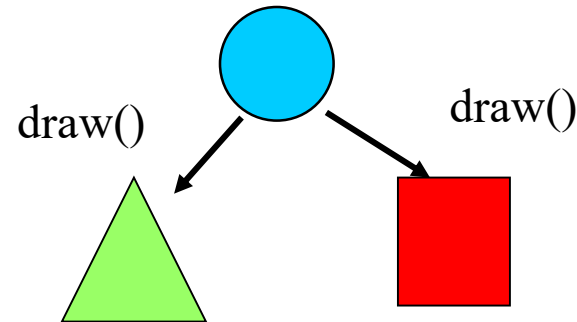
```
public interface ExampleInterface{  
    public void do();  
    public String doThis(int number);  
}
```

```
public class sub implements ExampleInterface{  
    public void do(){  
        //specify what must happen  
    }  

```

```
    public String doThis(int number){  
        //specfiy what must happen  
    }  
}
```

Polymorphism



❑ คือหลักการของ Method overriding ประโยชน์คือช่วยให้เรานำคลาสมาใช้ได้โดยสามารถเปลี่ยนแปลง method บาง method ของ base class ให้ตรงกับความต้องการ

โค้ดตัวอย่าง ของ Polymorphism

```
public class SuperClass{
    public int getNb(){
        //specify what must happen
        return 1;
    }
    public int getNb2(){
        //specify what must happen
        return 2;
    }
}

public class SubClass extends SuperClass{
    //you can override the implementation
    @Override
    public int getNb2(){
        return 3;
    }
}
```

```
SubClass sup = new SubClass();
sup.getNb(); //returns 1
sup.getNb2(); //returns 3
```

ภาคผนวก ข : การพัฒนา RESTful API

ตัวอย่างของ RESTful API ประกอบด้วย

- ❑ RESTful API ภาษา PHP
- ❑ RESTful API ภาษา C#.NET

RESTful API ภาษา PHP

ตัวอย่าง RESTful API นี้พัฒนาด้วยภาษา PHP โดยใช้
CodeIgniter Framework และ RestServer Framework ของ
Chris Kacerguis

สามารถ Download ตัวอย่างได้ที่
https://github.com/pichaiku/ci_api/

การติดตั้งฐานข้อมูล

ทำการ import database จากไฟล์ saledb.sql ลงไปใน MySQL หรือ MariaDB

โดยใช้ phpMyAdmin หรือเครื่องมืออื่น ๆ

การติดตั้งระบบ

1. ติดตั้ง XAMPP หรือ AppServ
2. Download โค้ดทั้งหมดไปเก็บไว้ใน Path ที่ต้องการ เช่น
C:\xampp\htdocs\ci_api
3. Config ฐานข้อมูล ได้แก่ hostname, username, password, และ
ชื่อฐานข้อมูล ที่ไฟล์
C:\xampp\htdocs\ci_api\application\config\database.php

ตัวอย่างการ Config ฐานข้อมูลในไฟล์ database.php

```
$db['default'] = array(  
    'dsn'       => '',  
    'hostname'  => 'localhost',  
    'username'  => 'root',  
    'password'  => '',  
    'database'  => 'saledb',  
);
```

การใช้งาน

1. ใช้ GET method เพื่อดูรายการข้อมูล (พนักงาน) ทั้งหมด
เช่น http://localhost/ci_api/index.php/api/employee/
2. ใช้ GET method เพื่อดูรายละเอียดข้อมูล (พนักงาน) โดยการระบุรหัส
ต่อท้าย เช่น
http://localhost/ci_sale/index.php/api/employee/000002
3. ใช้ POST method เพื่อเพิ่มข้อมูล
เช่น http://localhost/ci_api/index.php/api/employee/create
4. ใช้ PUT method เพื่อปรับปรุงข้อมูล
เช่น http://localhost/ci_api/index.php/api/employee/update/000002
5. ใช้ DELETE method เพื่อลบข้อมูล
เช่น http://localhost/ci_api/index.php/api/employee/delete/000002

การสร้าง API

1. สร้างตารางข้อมูลในฐานข้อมูล เช่น ตารางสินค้า (product)
2. copy ไฟล์ C:\xampp\htdocs\ci_api\application\controllers\api\Employee.php ไปเก็บไว้ที่ C:\xampp\htdocs\ci_api\application\controllers\api\Product.php
3. แก้ไขโค้ด Product.php
 - 3.1 ส่วนหลักของ Class แก้ไขดังนี้

```
//แก้ไขชื่อ Class ของ Controller เช่น จาก Employee เป็น Product
class Employee extends \Restserver\Libraries\REST_Controller {
    function __construct(){
        parent::__construct();
        //แก้ไขชื่อ Class ของ Model เช่น จาก EmployeeModel
        //เป็น ProductModel
        $this->load->model('EmployeeModel');
        $this->load->database();
    }
    ...
    ...
}
```

3.2 ส่วนของ GET method เพื่อดูรายการข้อมูล (พนักงาน) แก้ไขดังนี้

```
public function index_get($id=-1)
{
    if($id!=-1){
        //แก้ไขชื่อ Class ของ Model เช่น
        //จาก EmployeeModel เป็น ProductModel
        $data=$this->EmployeeModel->selectOne($id);
    }else{
        //แก้ไขชื่อตารางข้อมูล เช่น product
        $query = $this->db->get("employee");
        $data = $query->result();
    }
    ...
    ...
}
```

3.3 ส่วนของ POST method เพื่อเพิ่มข้อมูล แก้ไขดังนี้

```
public function create_post()
{
    $data = array(
        //แก้ไขชื่อ Field ของตารางข้อมูล
        'username' => $this->post('username'),
        'password' => $this->post('password'),
        'empID' => $this->post('empID'),
        ...
        ...
    );
    //แก้ไขชื่อ Class ของ Model เช่น จาก EmployeeModel
    //เป็น ProductModel
    if($this->EmployeeModel->insert($data)){
        $this->response(array(
            'message' => 'success',
            'status' => 'true'));
        ...
        ...
    }
}
```

3.4 ส่วนของ PUT method เพื่อปรับปรุงข้อมูล แก้ไขดังนี้

```
public function update_put($id)
{
    $data = array(
        //แก้ไขชื่อ Field ของตารางข้อมูล
        'username' => $this->put('username'),
        'password' => $this->put('password'),
        'firstName' => $this->put('firstName'),
        'lastName' => $this->put('lastName')
    );
    //แก้ไขชื่อ Class ของ Model เช่น จาก EmployeeModel
    //เป็น ProductModel
    if($this->EmployeeModel->update($data,$id)){
        $this->response(array(
            'message' => 'success',
            'status' => 'true'));
        ...
        ...
    }
}
```

3.5 ส่วนของ DELETE method เพื่อลบข้อมูล แก้ไขดังนี้

```
public function delete_delete($id)
{
    //แก้ไขชื่อ Class ของ Model เช่น
    //จาก EmployeeModel เป็น ProductModel
    if($this->EmployeeModel->delete($id)){
        $this->response(array(
            'message' => 'success',
            'status' => 'true'))
        ...
        ...
    }
```


4. copy ไฟล์ C:\xampp\htdocs\ci_api\application\models\EmployeeModel.php ไปเก็บไว้ที่ C:\xampp\htdocs\ci_api\application\models\ProductModel.php

5. แก้ไขโค้ด ProductModel.php

5.1 ส่วนหลักของ Class แก้ไขดังนี้

//แก้ไขชื่อ Class ของ Model เช่น จาก EmployeeModel เป็น ProductModel

```
class EmployeeModel extends CI_Model {
```

```
    function __construct() {  
        parent::__construct();  
    }
```

5.2 ส่วนของการดูรายการข้อมูล แก้ไขดังนี้

```
public function selectOne($id){  
    //แก้ไขชื่อตารางข้อมูล เช่น จาก employee เป็น product  
    //และแก้ไขรหัสข้อมูล เช่น จาก empID เป็น productID  
    $query = $this->db->get_where("employee",array("empID"=>$id));  
    $data = $query->result();  
    return $data;  
}
```

5.3 ส่วนของการเพิ่มข้อมูล แก้ไขดังนี้

```
public function insert($data) {  
    //แก้ไขชื่อตารางข้อมูล เช่น จาก employee เป็น product  
    if ($this->db->insert("employee", $data)) {  
        return true;  
    }  
}
```

5.4 ส่วนของการปรับปรุงข้อมูล แก้ไขดังนี้

```
public function update($data,$id) {  
    $this->db->set($data);  
    //แก้ไขรหัสข้อมูล เช่น จาก empID เป็น productID  
    //และแก้ไขชื่อตารางข้อมูล เช่น จาก employee เป็น product  
    $this->db->where("empID", $id);  
    $this->db->update("employee", $data);  
    return true;  
}
```

5.5 ส่วนของการลบข้อมูล แก้ไขดังนี้

```
public function delete($id) {  
    //แก้ไขชื่อตารางข้อมูล เช่น จาก employee เป็น product  
    //และแก้ไขรหัสข้อมูล เช่น จาก empID เป็น productID  
    if ($this->db->delete("employee", "empID = ".$id."")) {  
        return true;  
    }  
}
```

RESTful API ภาษา C#.NET

ตัวอย่าง RESTful API นี้พัฒนาด้วยภาษา C#.NET โดยใช้ Web API Framework

สามารถ Download ตัวอย่างได้ที่

https://github.com/pichaiku/csharp_sale

การติดตั้งฐานข้อมูล

1. unzip ไฟล์ SaleDB.zip ซึ่งจะพบไฟล์ SaleDB.bak ใน format ของ MSSQL Server 2019
2. 2. ทำการ restore database จากไฟล์ SaleDB.bak โดยใช้ MSSQL Server Management Studio 18.2 หรือสูงกว่า

การติดตั้งระบบ

1. เปิด Solution จากไฟล์ Cshap.sln ด้วย Visual Studio.NET version 2017 หรือสูงกว่า ใน Solution จะประกอบไปด้วย 2 projects คือ API project กับ Web project
2. Config ฐานข้อมูล ได้แก่ datasource, username, password, และชื่อฐานข้อมูลหรือ catalog ที่ไฟล์ Web.config ตัวอย่างการ Config ฐานข้อมูลในไฟล์ Web.config

```
data source=localhost;
initial catalog=saledb;
user id=sa;
password=123
```
3. ทำการ Publish เฉพาะ API project ไปยัง Server (ชื่อเว็บไซต์หรือหมายเลข IP) ที่ต้องการ

#การใช้งาน

1. ใช้ GET method เพื่อดูรายการข้อมูล (พนักงาน) ทั้งหมด
เช่น http://localhost/csharp_api/api/employee
2. ใช้ GET method เพื่อดูรายละเอียดข้อมูล (พนักงาน) โดยการระบุรหัส
ต่อท้าย เช่น
http://localhost/csharp_api/api/employee/2
3. ใช้ POST method เพื่อเพิ่มข้อมูล
เช่น http://localhost/csharp_api/api/employee
4. ใช้ PUT method เพื่อปรับปรุงข้อมูล
เช่น http://localhost/csharp_api/api/employee/2
5. ใช้ DELETE method เพื่อลบข้อมูล
เช่น http://localhost/csharp_api/api/employee/2

ดูรายละเอียดการใช้งานได้ที่
http://localhost/csharp_api/help

การสร้าง API

1. สร้างตารางข้อมูลในฐานข้อมูล เช่น ตารางสินค้า (product)
2. สร้าง Controller class
 - ในโฟลเดอร์ Controllers ให้ทำการ copy ไฟล์ EmployeeController.cs ไปเก็บไว้เป็นไฟล์ ProductController.cs
 - แก้ไขชื่อ Class, ชื่อตารางข้อมูล, และชื่อ Field ข้อมูล
3. สร้าง Model class
 - ในโฟลเดอร์ Models ให้ Double click ที่ไฟล์ *.edmx จะแสดงหน้า Diagram ของ Model class ขึ้นมา
 - ที่หน้า Diagram ทำการ Right mouse click แล้วเลือก "Update model from database"
 - ที่ Tab "Add" ให้คลิกเลือกตารางสินค้าVS.NET จะทำการสร้าง Product.cs ให้อัตโนมัติ