

LAB 3 การจัดการ Layout

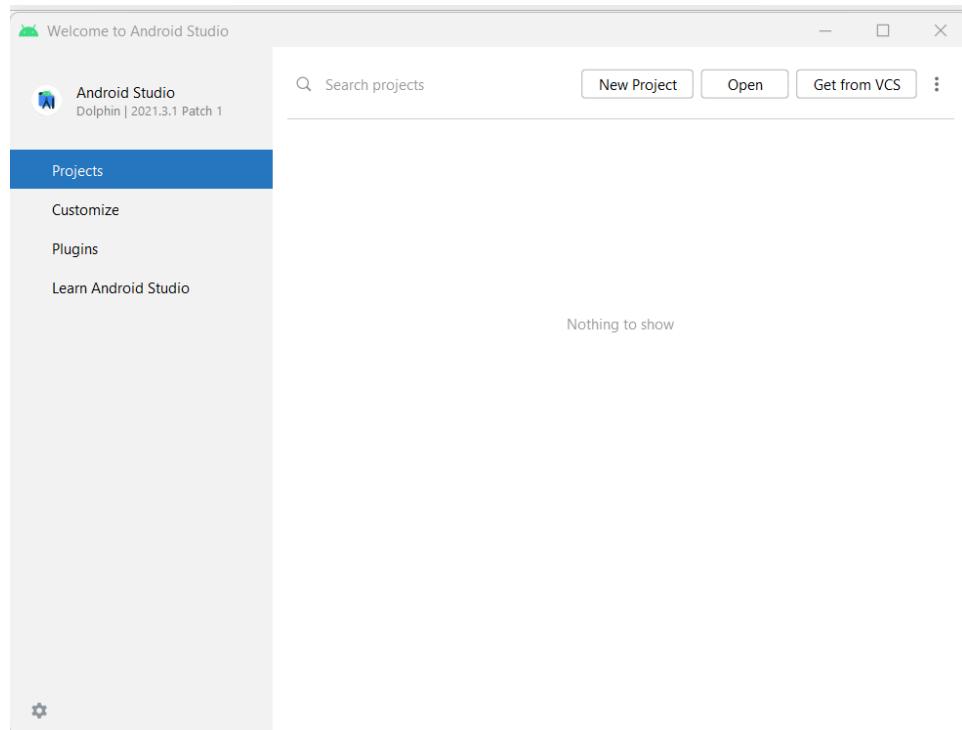
เอกสารประกอบการปฏิบัติการรายวิชา 342267 MOBILE DEVICE PROGRAMMING

วัตถุประสงค์

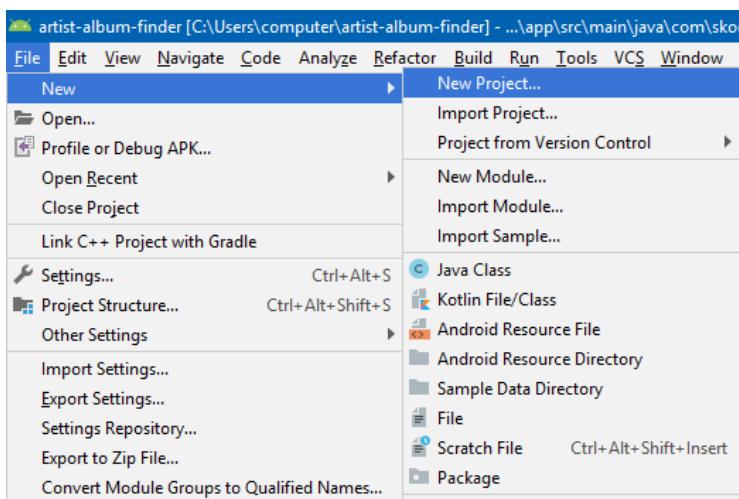
1. เพื่อให้นักศึกษาสามารถใช้การออกแบบหน้าจออุปกรณ์เคลื่อนที่และการใช้ LinearLayout และ RelativeLayout
2. เพื่อให้นักศึกษาสามารถใช้ Element ของ TextView และ ImageView

การสร้างโปรเจคใหม่

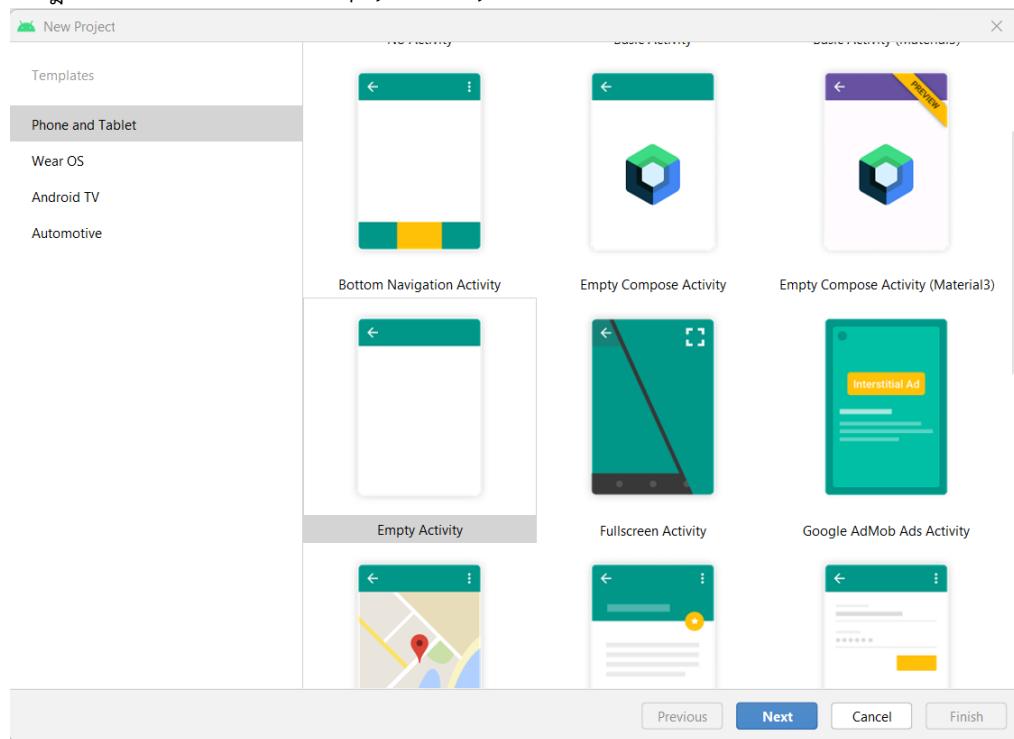
1. เมื่อเปิดโปรแกรม Android Studio แล้ว คลิกที่ New Project



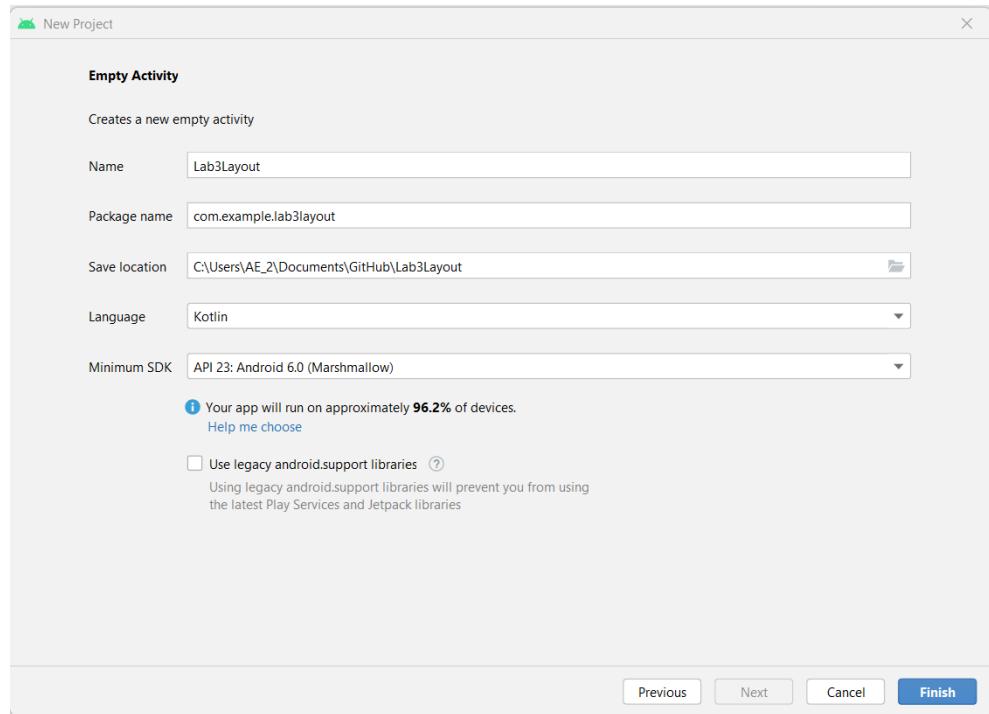
หรือถ้าเปิดโปรแกรมมาแล้วให้ไปที่ File >> New >> New Project...



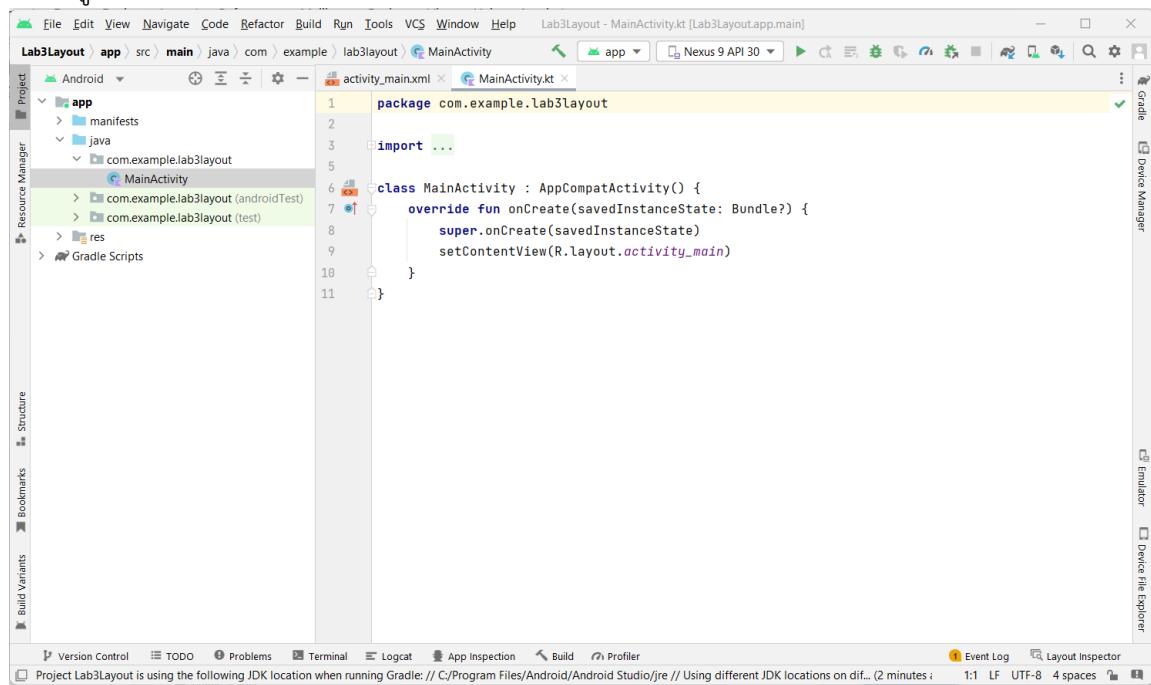
2. จากนั้นจะปรากฏหน้าจอ ดังนี้ ให้เลือก Empty Activity จากนั้นกด Next



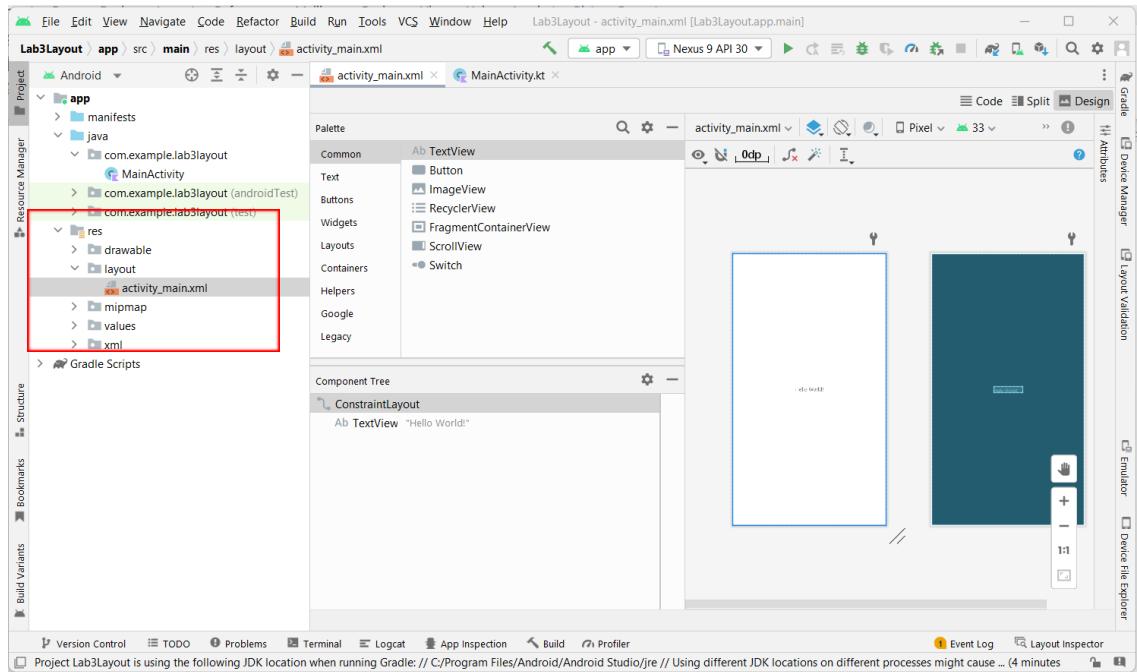
3. ถัดมาให้ตั้งชื่อ Application name คือ Lab3Layout และ Language เป็น Kotlin จากนั้นกด Finish



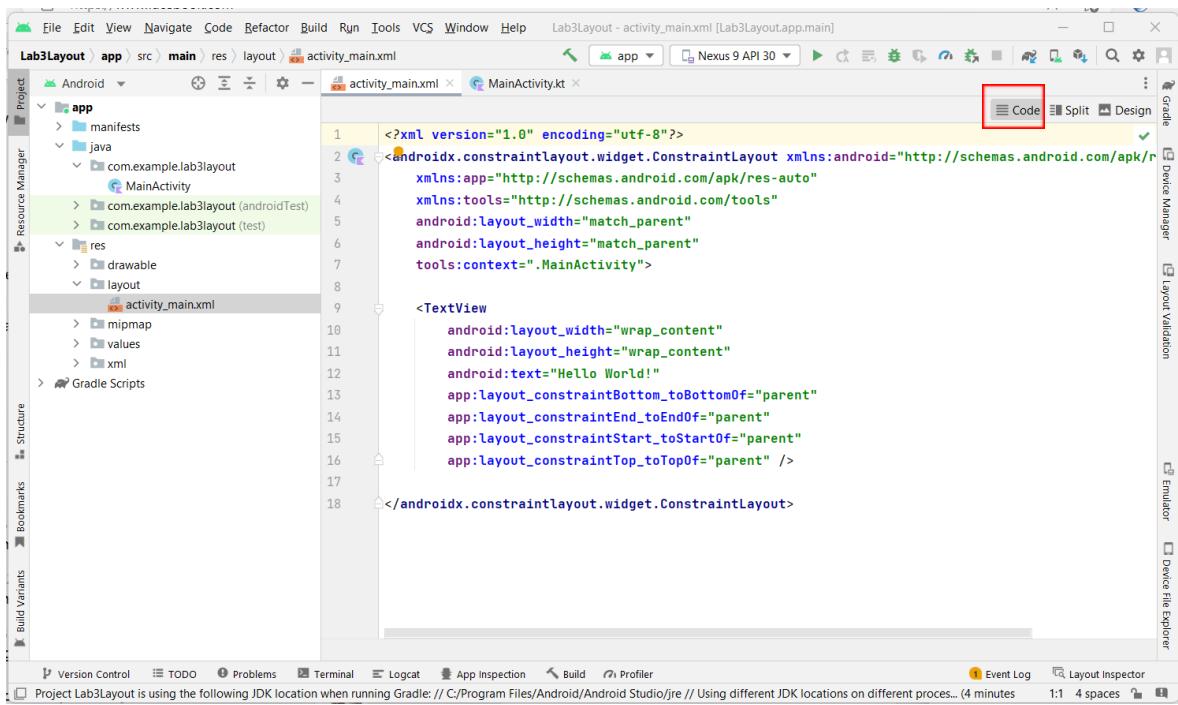
จะต้องรอสักครู่ เพื่อให้โปรแกรมสร้างโปรเจคเสร็จ



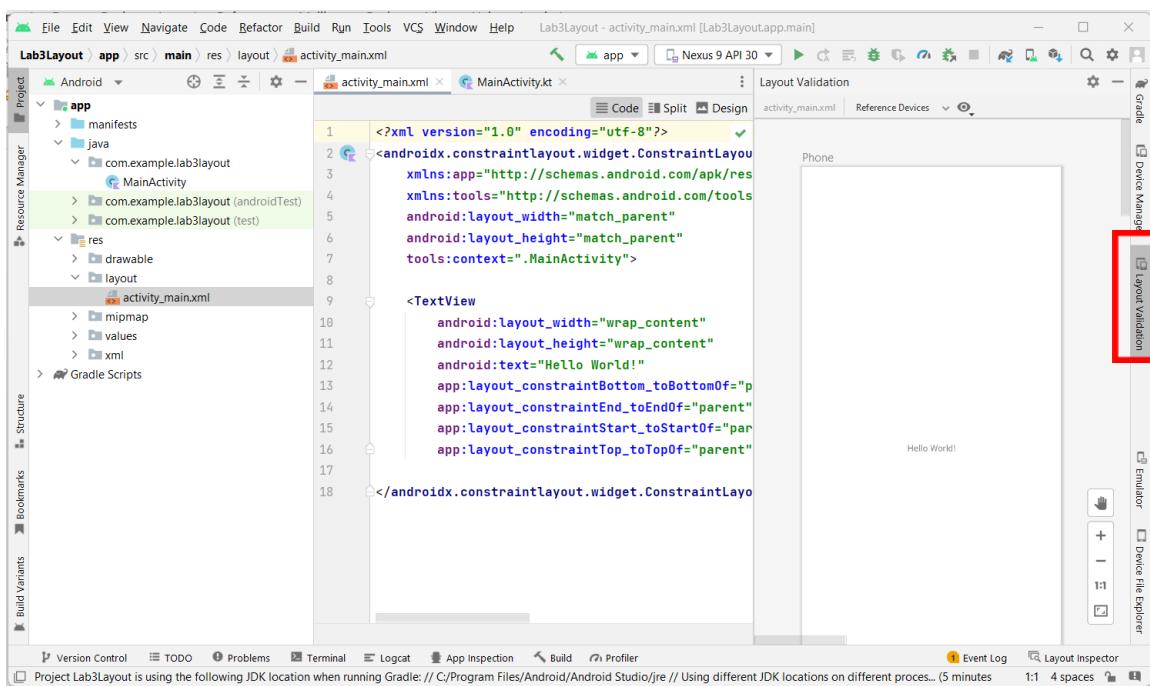
4. เมื่อ Layout ที่ถูกสร้างจะอยู่ที่แท็บ Project ที่ด้านซ้าย ให้คลิกไปที่ res >> laout >> activity_main.xml



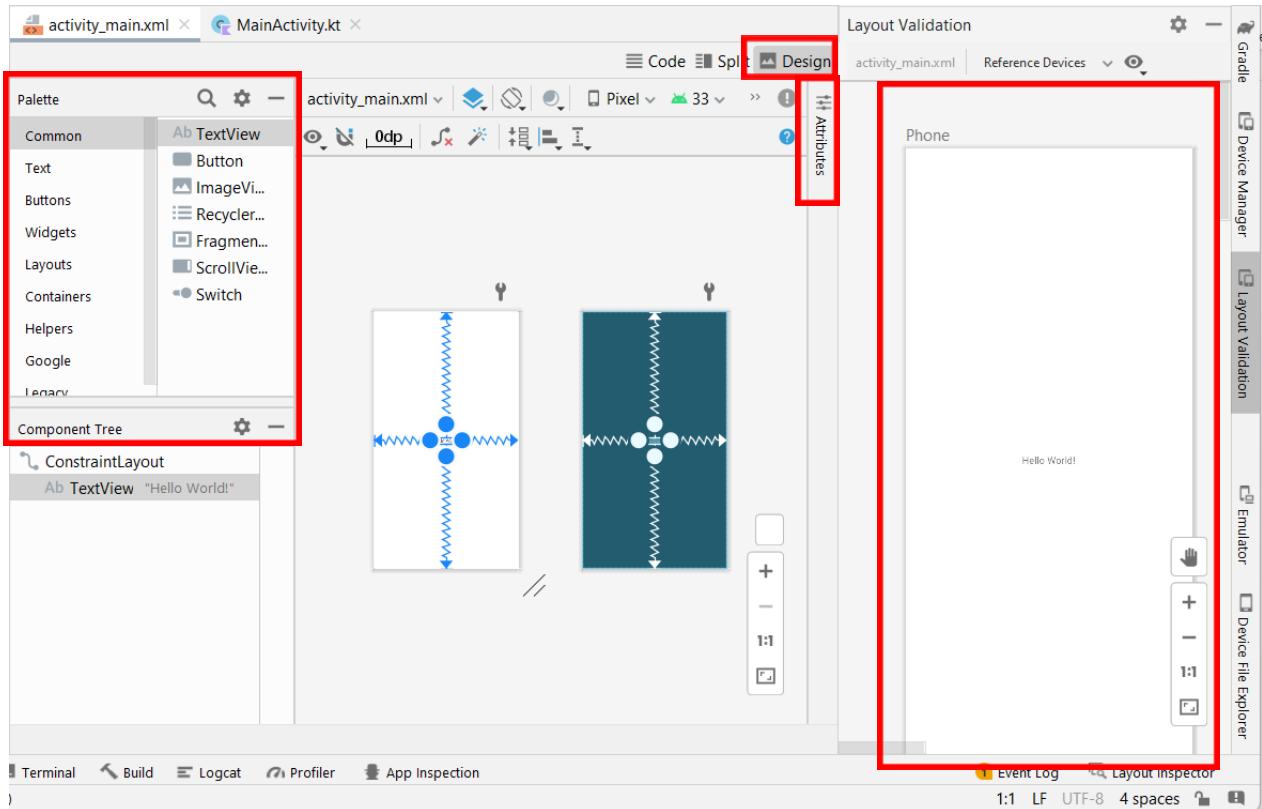
5. จักนั้นให้ลองเปลี่ยนแท็บเป็น Code จะแสดงคำสั่ง XML



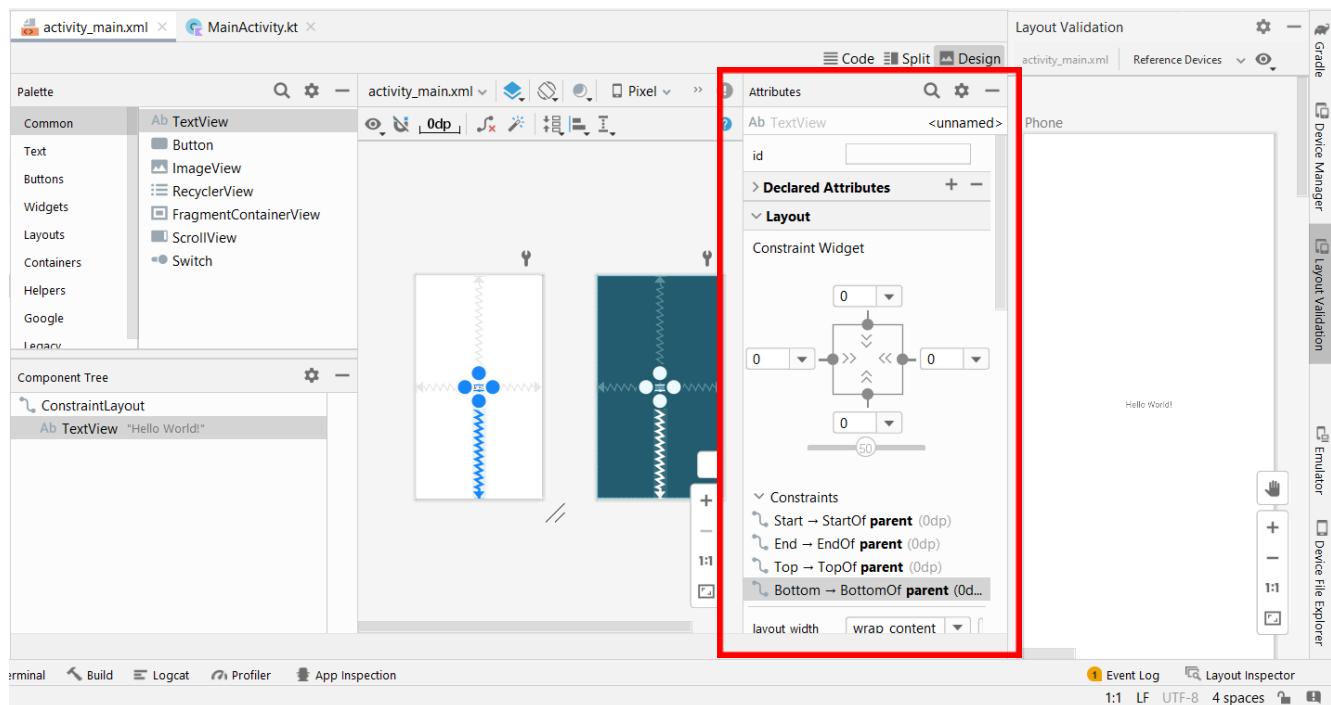
ส่วนของการแสดง preview ให้คลิกปุ่ม Layout Validation



สำหรับแท็บ Design จะสามารถถูก Object ต่างๆด้านซ้ายมือ แล้วนำมาวางที่ตัวหน้าจอ มีอีกหนึ่ง Tab คือ Preview



นอกเหนือไปนี้มี Attributes เพื่อเซตค่าของ object ด้วย



การจัดการ Layout

การจัดรูปแบบหน้าจอของ element ต่างๆ ในแอร์คุณควรการแสดงผลของหน้าจอจากตัวอย่างในภาพจะเป็น android.support.constraint.ConstraintLayout ซึ่งเป็นการลากแล้วนำ object มาวาง โดยเป็นค่าเริ่มต้นในการสร้างแอป ซึ่งมีการใช้งานที่ไม่ยากโดยที่ไม่ต้องพิมพ์คำสั่ง นักศึกษาสามารถฝึกใช้งานเองได้



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

ดังนั้นในปฏิบัติการนี้จะเป็นการเขียนคำสั่ง โดยใช้การจัดรูปแบบ 2 ประเภท คือ Linear Layout และ Relative Layout มีรายละเอียด ดังนี้

1. Linear Layout โดยกำหนดในส่วนของ Root เป็นการกำหนดในแนวตั้งหรือแนวนอน ต้องกำหนดคุณสมบัติ android:orientation เป็น vertical สำหรับแนวตั้ง หรือ horizontal สำหรับแนวนอน ในการเรียงการแสดงผลของ element

ตัวอย่างการใช้ android:orientation เป็น vertical

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:orientation="vertical"
8     tools:context=".MainActivity">
9
10    <TextView
11        android:layout_width="wrap_content"
12        android:layout_height="wrap_content"
13        android:text="Hello World!"
14        android:textSize="30sp" />
15
16    <TextView
17        android:layout_width="150dp"
18        android:layout_height="wrap_content"
19        android:text="We are cats."
20        android:textSize="20sp"
21        android:background="#FFEB3B" />
22
23    </LinearLayout>
```

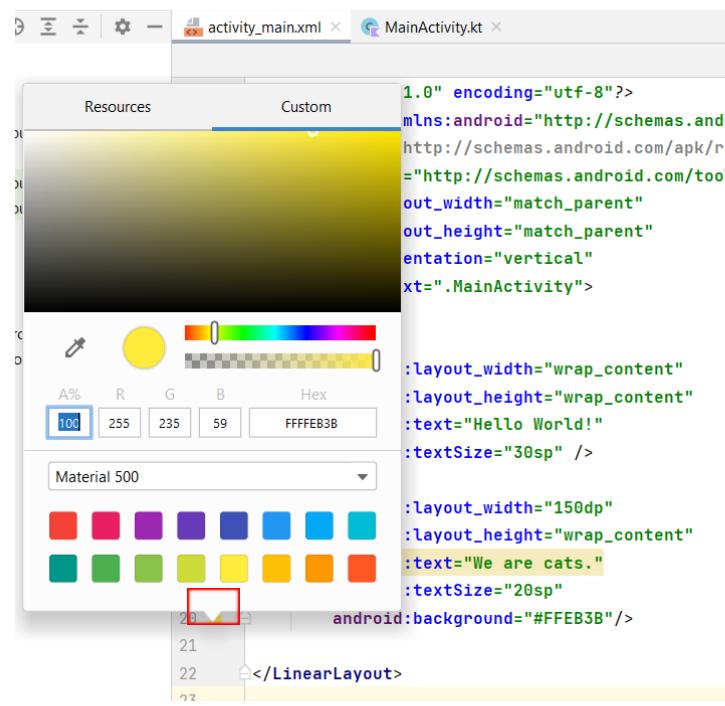
The screenshot shows an Android XML layout file in an IDE. The root element is a `<LinearLayout>`. Inside it, there are two `<TextView>` elements. The first `<TextView>` has its `text` attribute set to "Hello World!" and its `textSize` attribute set to "30sp". The second `<TextView>` has its `text` attribute set to "We are cats.", its `textSize` attribute set to "20sp", and its `background` attribute set to "#FFEB3B". The entire `<LinearLayout>` tag is highlighted with a red box.

อธิบายโค้ด

เริ่มแรกจะต้องเปลี่ยน Tag root ให้เป็น `LinearLayout` ดังภาพด้านบน และเขียนคำสั่ง `android:orientation = "vertical"` คือการจัดรูปแบบการวาง Object เรียงตามแนวตั้งลงมาทั้งหมด Object

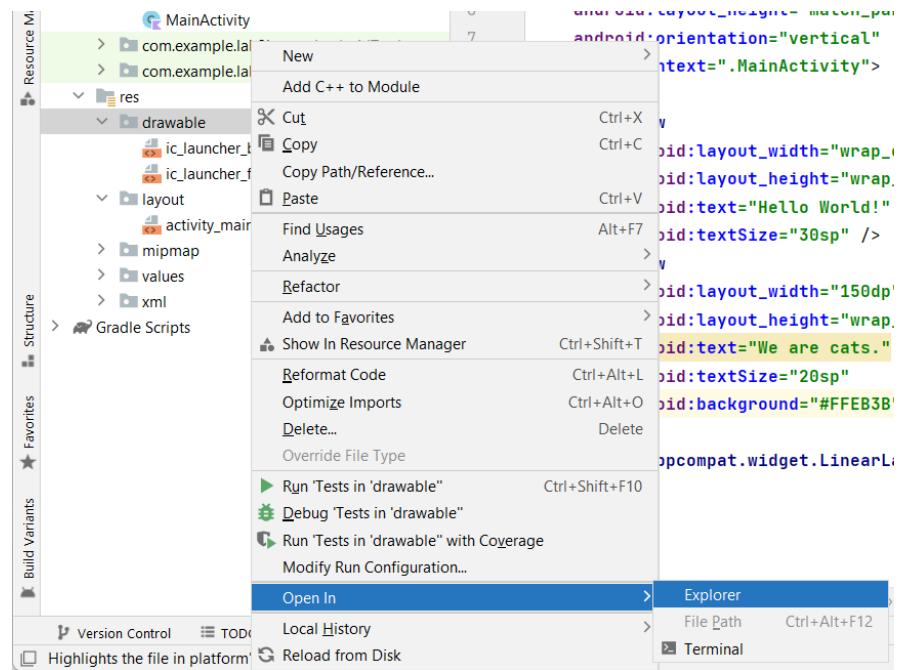
ข้อความที่ 1 `<TextView>` แรก ให้ลบคำสั่งในส่วนที่เป็น `app:` ออกให้หมด และเพิ่มคำสั่งกำหนดขนาดของข้อความ `android:textSize = "30sp"` ตามภาพด้านบน

ข้อความที่ 2 `<TextView>` ที่สอง ต้องการสร้าง `<TextView>` ขึ้นมา มีการกำหนดความกว้างของข้อความ 150 dp และความสูงเป็น `wrap_content` เพื่อให้ความสูงพอดีกับข้อความ และกำหนดสีของพื้นหลัง ใช้คำสั่ง `android:background = "#FFEB3B"` สำหรับตอนที่กำหนดสีจะมีจุดแต้มสีขึ้นที่ด้านขอบที่บรรทัดที่พิมพ์ เราสามารถคลิกแล้วเลือกสีได้ จากนั้นเพิ่มข้อความโดยใช้คำสั่ง `android:text = "We are cats."` และกำหนดขนาดของข้อความ `android:textSize = "20sp"`

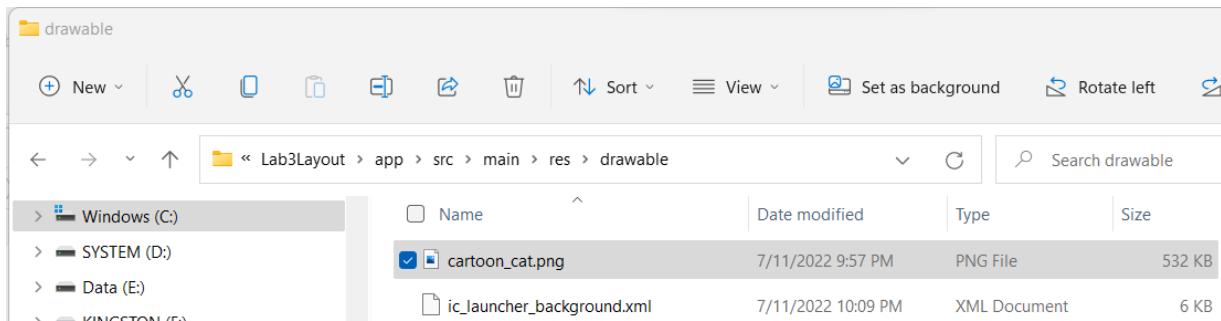


ถัดมาการใส่รูปภาพ ให้นำรูปภาพไปเก็บที่ drawable โดยให้คลิกขวาที่ drawable และเลือกที่ Open in >>

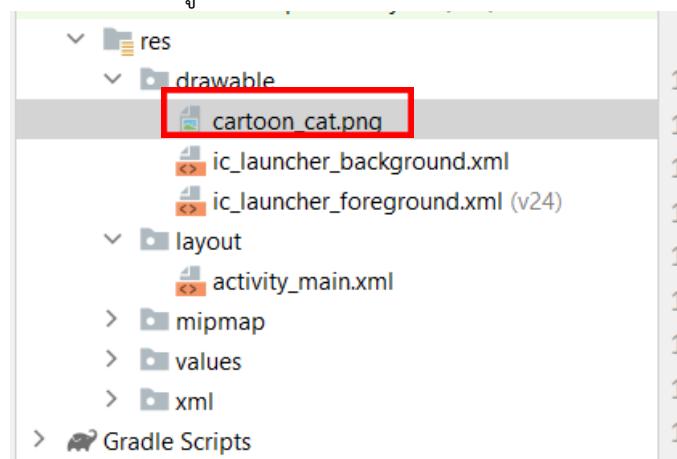
Explore



จากนั้น Copy รูปภาพที่ต้องการไปใส่ในโฟลเดอร์ที่เปิดไว้ ตัวอย่างที่ใช้ชื่อรูปภาพ คือ cartoon_cat.png



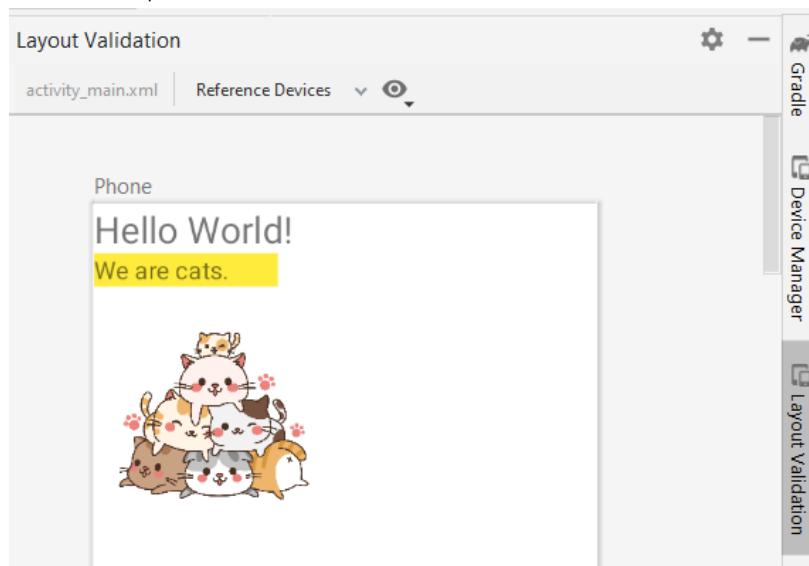
ส่วนของโปรแกรม android studio จะแสดงรูปภาพที่เพิ่มเข้าไป



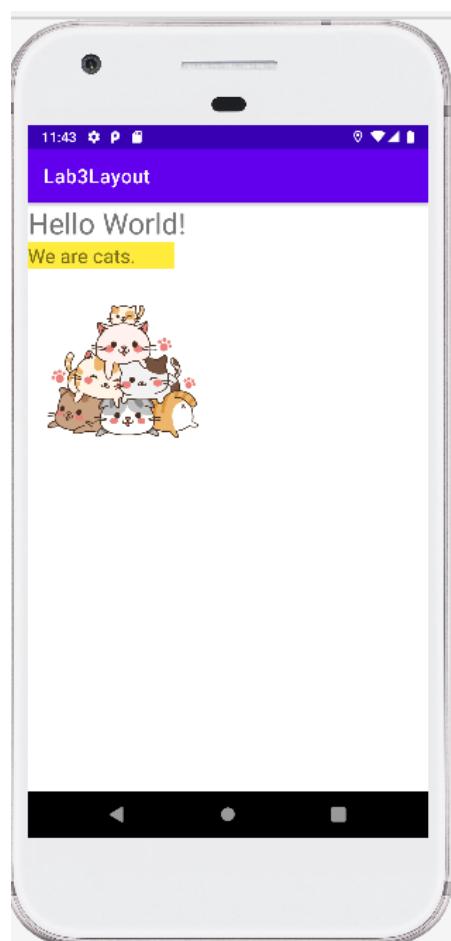
การเพิ่มคำสั่งการแสดงรูปภาพจะใช้ tag `<androidx.appcompat.widget.AppCompatImageView>`
โดยเพิ่มคำสั่งไปที่ไฟล์เดิม กำหนดให้เพิ่มคำสั่งดังนี้

```
10 <TextView  
11     android:layout_width="wrap_content"  
12     android:layout_height="wrap_content"  
13     android:text="Hello World!"  
14     android:textSize="30sp" />  
15 <TextView  
16     android:layout_width="150dp"  
17     android:layout_height="wrap_content"  
18     android:text="We are cats."  
19     android:textSize="20sp"  
20     android:background="#FFEB3B"/>  
21  
22 <androidx.appcompat.widget.AppCompatImageView  
23     android:layout_width="200dp"  
24     android:layout_height="200dp"  
25     android:src="@drawable/cartoon_cat"/>  
26  
27 </androidx.appcompat.widget.LinearLayoutCompat>
```

จากคำสั่งได้กำหนดขนาดของภาพคือ ความกว้าง 200dp และความสูง 200dp และใช้คำสั่ง android:src ในการอ้างอิงไปยังภาพที่เก็บไว้ ส่วนของ preview แสดงดังนี้



จากนั้นให้คลิกที่ปุ่ม เพื่อ Run โปรแกรม จะแสดงผลลัพธ์ดังนี้



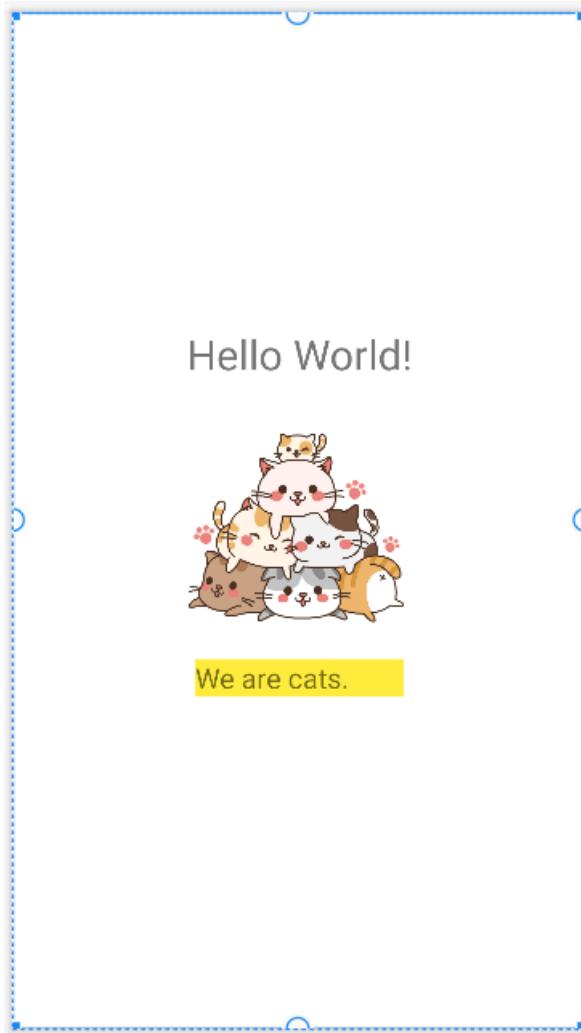
ถัดมาให้ลองเปลี่ยนคำสั่งส่วนของ android:orientation เป็น horizontal เรียง Element ในแนวนอนต่อไปจากข่ายไปขวา คำสั่งดังนี้

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:orientation="horizontal" highlighted
8     tools:context=".MainActivity">
9
10    <TextView
11        android:layout_width="wrap_content"
12        android:layout_height="wrap_content"
13        android:text="Hello World!"
14        android:textSize="30sp" />
15    <TextView
16        android:layout_width="150dp"
17        android:layout_height="wrap_content"
18        android:text="We are cats."
19        android:textSize="20sp"
20        android:background="#FFEB3B"/>
```

จากนั้นให้คลิกที่ปุ่ม  เพื่อ Run โปรแกรม จะแสดงผลลัพธ์ดังนี้



2. Relative Layout โดยกำหนดในส่วนของ Root เป็นการอ้างอิงตำแหน่งของ Element จากตำแหน่งของ Element อื่น ด้วยการแสดงผลแบบ RelativeLayout



ดังนั้นจากตัวอย่างเดิมเปลี่ยนเป็น RelativeLayout และมีการแก้ไขคำสั่งที่ต้องอ้างอิงตำแหน่งของ Element ดังนี้

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      tools:context=".MainActivity">
7
8      <TextView
9          android:layout_width="wrap_content"
10         android:layout_height="wrap_content"
11         android:text="Hello World!"
12         android:textSize="30sp"
13         android:layout_above="@+id/imageView"
14         android:layout_centerHorizontal="true"/>
```

```

15 <TextView
16     android:layout_width="150dp"
17     android:layout_height="wrap_content"
18     android:text="We are cats."
19     android:textSize="20sp"
20     android:background="#FFEB3B"
21     android:layout_below="@+id/imageView"
22     android:layout_centerHorizontal="true"/>
23 <androidx.appcompat.widget.AppCompatImageView
24     android:id="@+id/imageView" 1
25     android:layout_width="200dp"
26     android:layout_height="200dp"
27     android:src="@drawable/cartoon_cat"
28     android:layout_centerInParent="true"/>
29
30 </RelativeLayout>

```

อธิบายโค้ด

เริ่มแรกจะต้องเปลี่ยน tag root ให้เป็น RelativeLayout โดยใช้คำสั่ง <RelativeLayout> และลบส่วนของ android:orientation ออก ดังภาพด้านบน

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     tools:context=".MainActivity">

```

จากราฟที่เราสร้างหน้าจอจะต้องยังคงทำแน่นของภาพ ImageView ตั้งนี้เราจะกำหนดชื่อของ ImageView ก่อน โดยจะใช้คำสั่ง android:id="@+id/imageView" เพื่อเป็นการกำหนด id ให้กับ ImageView โดยตั้งชื่อว่า imageView

ตัดมาเพิ่มคำสั่ง android:layout_centerInParent="true" เป็นการกำหนดตำแหน่งในการจัดภาพให้อยู่กึ่งกลางของหน้าจอที่แสดงผล

```

23 <androidx.appcompat.widget.AppCompatImageView
24     android:id="@+id/imageView"
25     android:layout_width="200dp"
26     android:layout_height="200dp"
27     android:src="@drawable/cartoon_cat"
28     android:layout_centerInParent="true"/>

```

สำหรับข้อความที่ 1 <TextView> แรก ให้เพิ่มมีคำสั่ง android:layout_above="@+id/imageView" เพื่อกำหนดตำแหน่งของข้อความให้อยู่ด้านบนของภาพที่มี id = imageView ตัดมาเพิ่มคำสั่ง android:layout_centerHorizontal="true" เป็นการกำหนดให้ข้อความอยู่ตรงกลางของด้านซ้าย-ขวา

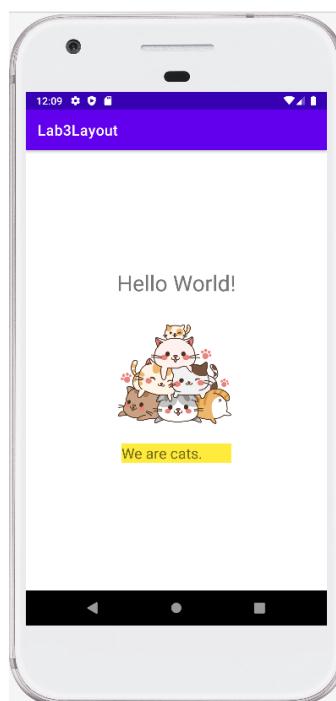
```
8 <TextView  
9     android:layout_width="wrap_content"  
10    android:layout_height="wrap_content"  
11    android:text="Hello World!"  
12    android:textSize="30sp"  
13    android:layout_above="@+id/imageView"  
14    android:layout_centerHorizontal="true"/>
```

ส่วนข้อความที่ 2 <TextView> ที่สอง มีการเพิ่มคำสั่ง android:layout_below="@+id/imageView" เพื่อกำหนดตำแหน่งของข้อความให้อยู่ด้านล่างของภาพที่มี id = imageView

ตัดมาเพิ่มคำสั่ง android:layout_centerHorizontal="true" เป็นการกำหนดให้ข้อความอยู่ตรงกลางของด้านซ้าย-ขวา เช่นเดียวกับข้อความแรก

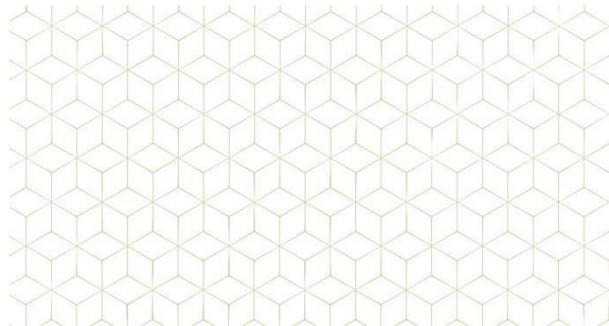
```
15 <TextView  
16     android:layout_width="150dp"  
17     android:layout_height="wrap_content"  
18     android:text="We are cats."  
19     android:textSize="20sp"  
20     android:background="#FFEB3B"  
21     android:layout_below="@+id/imageView"  
22     android:layout_centerHorizontal="true"/>
```

เมื่อ Run โปรแกรม จะแสดงผลลัพธ์ดังนี้



การใส่ภาพพื้นหลังให้อแอปพลิเคชัน

ถ้าหากแสดงภาพพื้นหลังในอุปกรณ์มือถือที่มีขนาดหน้าจอแตกต่างกัน จะทำให้การแสดงผลกับเครื่องที่มีอัตราส่วนหน้าจอที่ต่างกันออกไป ส่งผลให้ภาพที่ได้มีอัตราส่วนภาพผิดไปจากปกติ

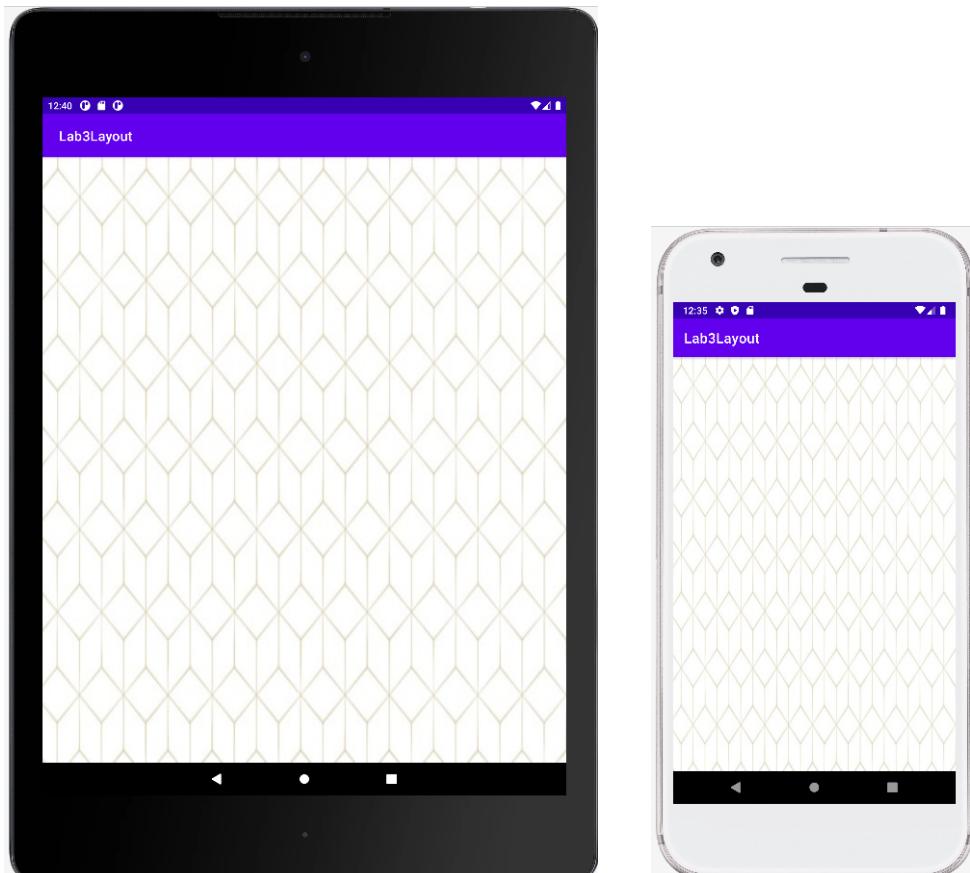


ภาพพื้นหลังต้นฉบับ bgnew.jpg

เมื่อใช้คำสั่งในการกำหนดพื้นหลังไว้ที่ Root View Group ดังนี้

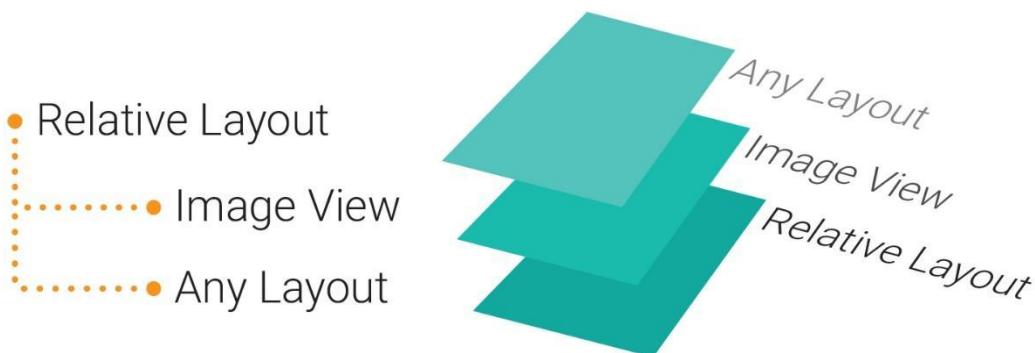
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:tools="http://schemas.android.com/tools"
4   android:layout_width="match_parent"
5   android:layout_height="match_parent"
6   android:background="@drawable/bgnew"
7   tools:context=".MainActivity">
```

เมื่อ Run โปรแกรมที่มีขนาดจอจะแสดงผลที่แตกต่างกัน จะแสดงผลลัพธ์ดังนี้



จากตัวอย่างข้างบน การใช้วิธีเตรียมภาพที่มีขนาดหน้าจอพอดีกับเครื่องนั้นๆ มาใช้ แต่เมื่อไปแสดงกับเครื่องที่มีอัตราส่วนหน้าจอที่ต่างกันออกໄປก็จะทำให้ภาพที่ได้มีอัตราส่วนภาพผิดไป เพราะมีการกำหนดภาพไว้เป็นภาพพื้นหลังของ Root View Group ที่ใช้วิธีขยายภาพตามขนาดของหน้าจอ ดังนั้นถ้าขนาดหน้าจอเปลี่ยนก็จะขยายให้สุดขอบ จึงเป็นที่มาของการบิดเบี้ยวของภาพ

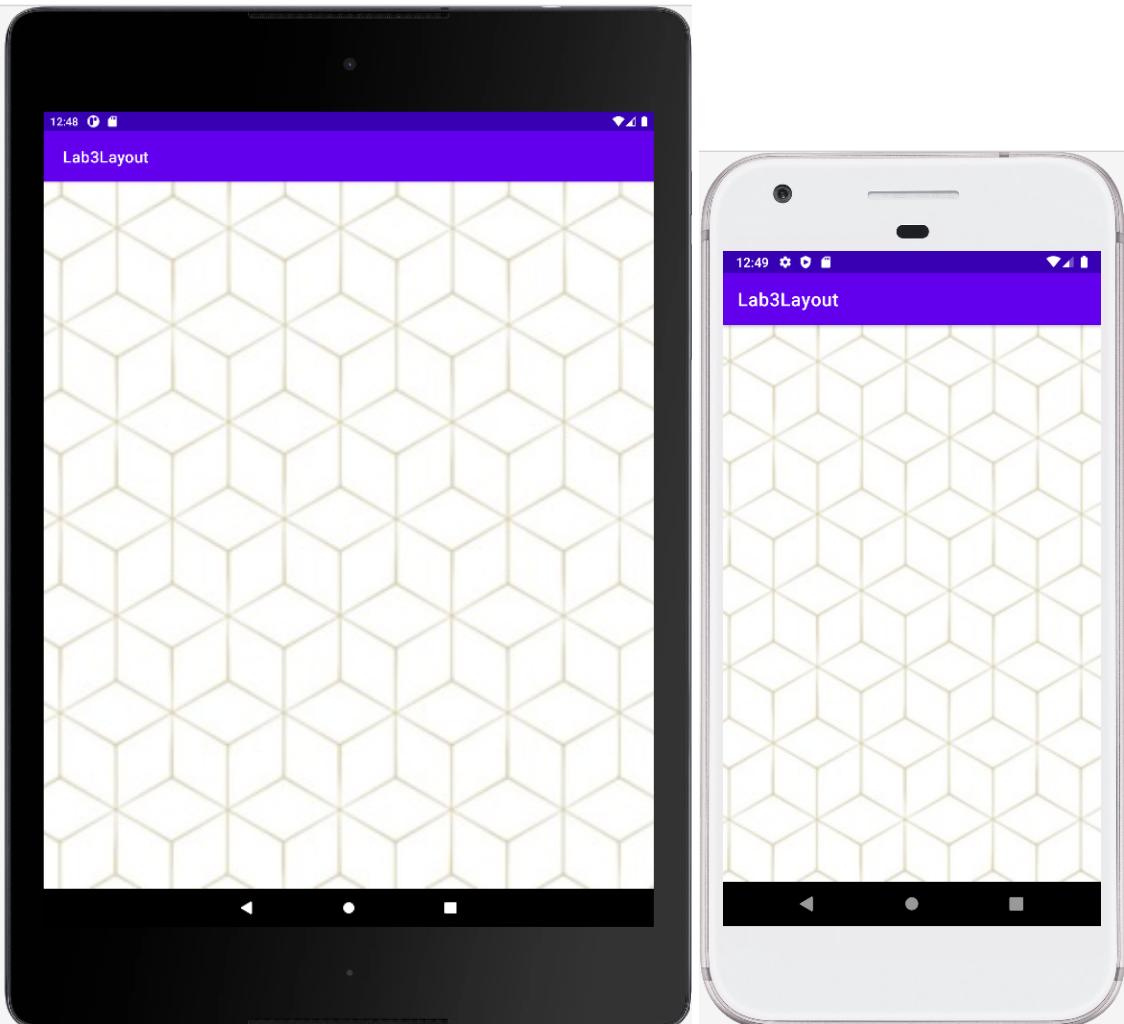
การแก้ปัญหานี้ด้วยเทคนิคการจัดレイ์เอาท์ โดยไม่กำหนดภาพพื้นหลังไว้ที่ Root View Group แต่กำหนดไว้ที่ส่วนใดก็ได้ ให้เหมือนภาพพื้นหลัง ให้ใช้ Relative Layout เป็น Root View Group และใช้ Image View ขยายเต็มหน้าจอแล้วกำหนดภาพพื้นหลังบนนี้แทน ส่วน View อื่นๆ สามารถกำหนดเป็น Layout ได้ก็ได้อีกที่ โดยจะเป็น Group View ตัวไหนก็ได้ ทั้งนี้เพราะว่า Image View มีข้อดีตรงที่มี Attribute ที่เรียกว่า **Scale Type** ที่มีข้อดี คือสามารถกำหนดเป็น Center Crop เพื่อให้ภาพขยายเต็มจอโดยที่อัตราส่วนภาพไม่เพี้ยน และส่วนที่เกินก็จะถูก Crop ทิ้งออกໄປ



คำสั่งในการจัดภาพพื้นหลังโดยใช้ Image View ดังนี้

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:tools="http://schemas.android.com/tools"
4   android:layout_width="match_parent"
5   android:layout_height="match_parent"
6   tools:context=".MainActivity">
7     <androidx.appcompat.widget.AppCompatImageView
8       android:layout_width="match_parent"
9       android:layout_height="match_parent"
10      android:src="@drawable/bgnew"
11      android:scaleType="centerCrop"/>
12   </RelativeLayout>
```

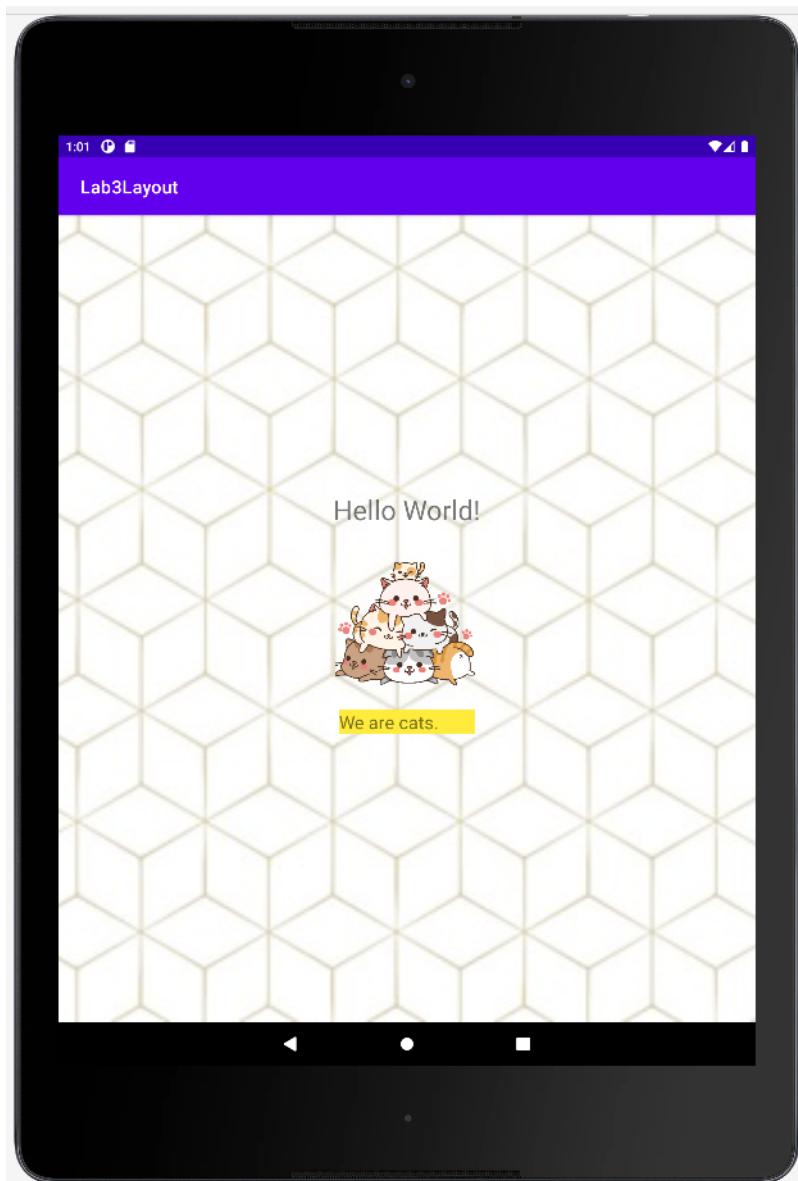
เมื่อ Run โปรแกรมที่มีขนาดจอจะแสดงผลที่แตกต่างกัน จะแสดงผลลัพธ์ดังนี้



เมื่อนำมาประยุกต์กับคำสั่งที่ได้สร้างหน้าจอที่ผ่านมาจะได้คำสั่ง ดังนี้

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     tools:context=".MainActivity">
7     <androidx.appcompat.widget.AppCompatImageView
8         android:layout_width="match_parent"
9         android:layout_height="match_parent"
10        android:src="@drawable/bgnew"
11        android:scaleType="centerCrop"/>
12     <RelativeLayout
13         android:layout_width="match_parent"
14         android:layout_height="match_parent">
15         <TextView
16             android:layout_width="wrap_content"
17             android:layout_height="wrap_content"
18             android:text="Hello World!"
19             android:textSize="30sp"
20             android:layout_above="@+id/imageView"
21             android:layout_centerHorizontal="true"/>
22         <TextView
23             android:layout_width="150dp"
24             android:layout_height="wrap_content"
25             android:text="We are cats."
26             android:textSize="20sp"
27             android:background="#FFEB3B"
28             android:layout_below="@+id/imageView"
29             android:layout_centerHorizontal="true"/>
30         <androidx.appcompat.widget.AppCompatImageView
31             android:id="@+id/imageView"
32             android:layout_width="200dp"
33             android:layout_height="200dp"
34             android:src="@drawable/cartoon_cat"
35             android:layout_centerInParent="true"/>
36     </RelativeLayout>
37 </RelativeLayout>
```

เมื่อ Run โปรแกรมที่มีขนาดจอจะแสดงผลที่แตกต่างกัน จะแสดงผลลัพธ์ดังนี้



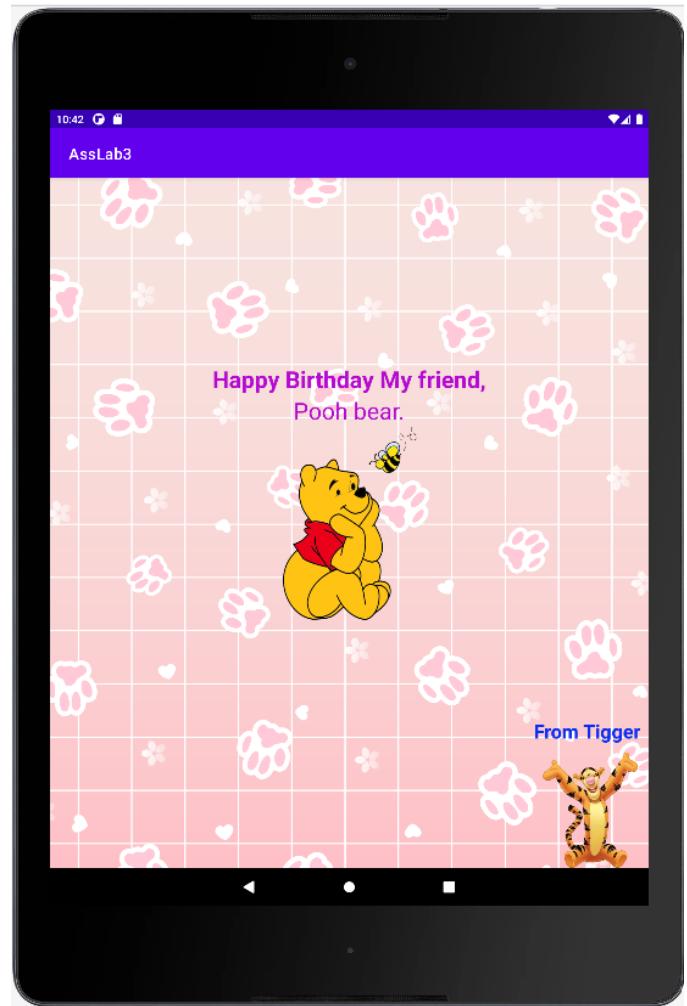
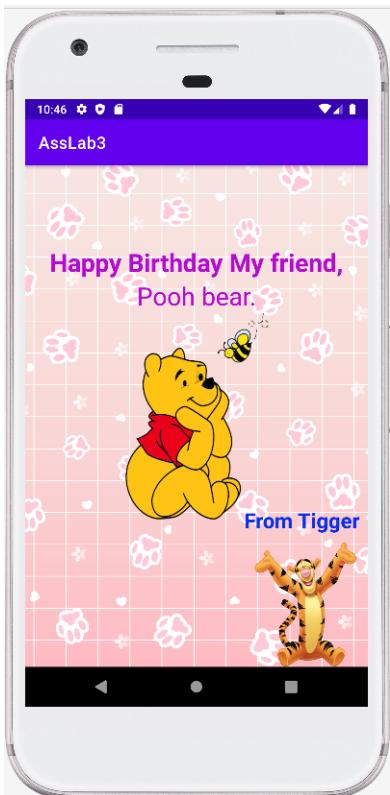
Assignment 3

*** Assignment ให้ทำเป็นการบ้าน

ให้นักศึกษาออกแบบการ์ดอายุพริวันเกิดให้เพื่อนภายใน Section ตัวเอง โดยกำหนดให้ต้องมีส่วนประกอบดังนี้

- ภาพพื้นหลัง,
- ข้อความอวยพรวันเกิด,
- รูปภาพของเพื่อน,
- ลงชื่อตัวเอง และ
- รูปภาพของตัวเอง

ตัวอย่าง



หรือ



