

2016 年 10 月 27 日 (木) 実施

レイアウト

レイアウトの位置付け

Android アプリで用いられる様々なレイアウト (Layout) は、それぞれ ViewGroup クラスを継承するクラスとして定義されているものであり、ViewGroup クラスは View クラスを継承するクラスである。View クラスはユーザインタフェースを構成する基本要素を表す。ビュー (View) は画面上に長方形の領域を占め、描画及びイベント操作を担う。ビューグループ (ViewGroup) は他のビューを含むことが出来る、不可視のコンテナ (容器) である。

授業の準備

1) Android Studio の初期設定

Android Studio を起動し、『Configure』→『設定のインポート』を選択し、第 3 回の教材の p.5 に従って設定をインポートする。

2) プロジェクトの新規作成

『Application name』(アプリ名)を「Prog_4th」(先頭は大文字,「_」は下線),『Company Domain』を「b6a0xxx.cuc.ac.jp」に書き換え,『Project Location』の先頭の「C:\Users\b6a0xxx」を『H:』に書き換えて,『次へ』ボタンを押す。

第 1 回と同様に『Minimum SDK』では『API 22』を選択する (第 1 回教材 p.7)。

『Activity name』は「MainActivity4」とする。

3) AVD の設定

第 1 回の授業で作成した AVD の設定は H ドライブにあって残るが, SDK のシステムイメージは C ドライブにあるので, 消失している。そこで, 『Download』をクリックして, インストールし直す (第 2 回教材 p.4)。

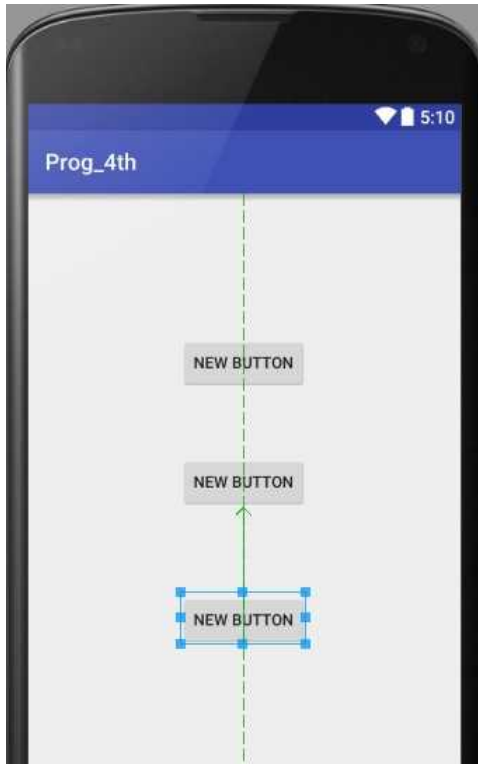
課題

今回は, これまで用いて来た RelativeLayout 以外のレイアウトとして, LinearLayout, GridLayout 及び TableLayout を用いた画面定義を作成し, それぞれのレイアウトの特性を学ぶ。

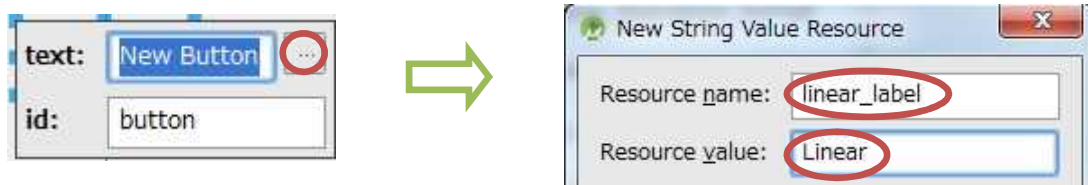
Android アプリの作成

『activity_main4.xml』のタブを開く。先ず元から貼り付けられている「Hello World!」と表示されているテキストビューを選択し, 『Delete』キーで削除する。続いて, 『Palette』の『Widgets』から『Button』を 3 個, 上から順にドラッグして配置する。

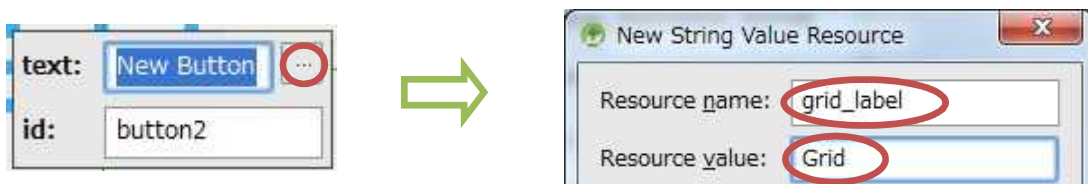
(図は次のページ)



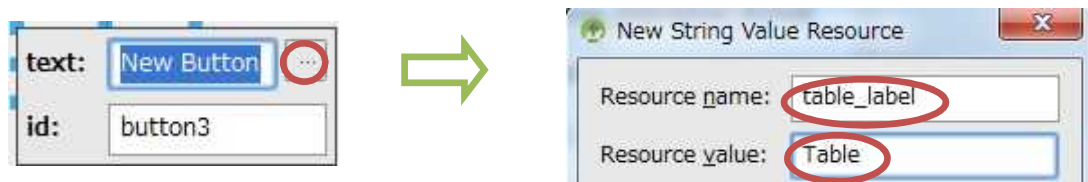
それぞれのボタンをダブルクリックし、『…』ボタンを押して出てきた『Resources』では『New Resource』 → 『New string Value』を選択して、その値を変更していく。



button Resource name: linear_label Resource value: Linear



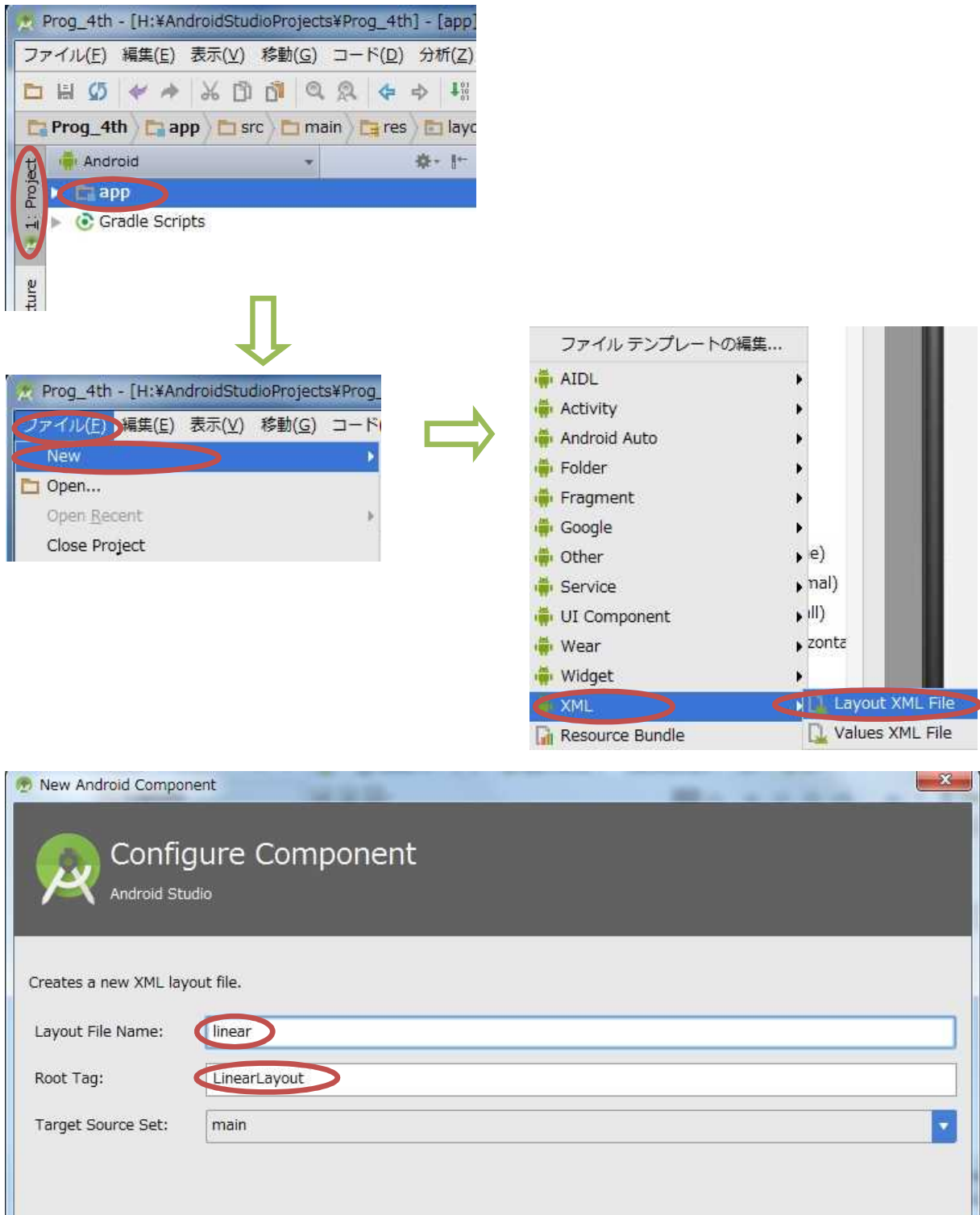
button2 Resource name: grid_label Resource value: Grid



button3 Resource name: table_label Resource value: Table

『XML layout file』を新規作成する。まずは『Project』タブを開き、『app』を選択する。

(図は次のページ)



『Layout File Name』の欄には「linear」と入力し、『Root Tag』の欄に「LinearLayout」が入力されていることを確認して、『終了』をクリックする。

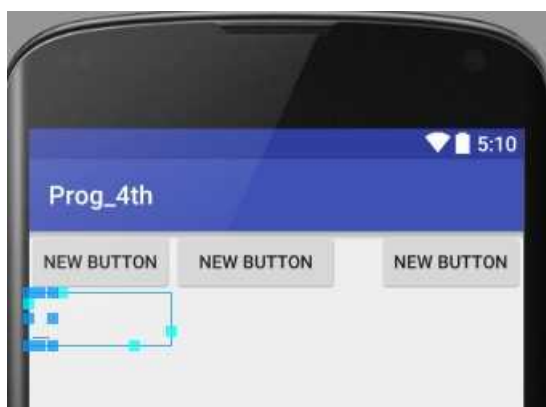
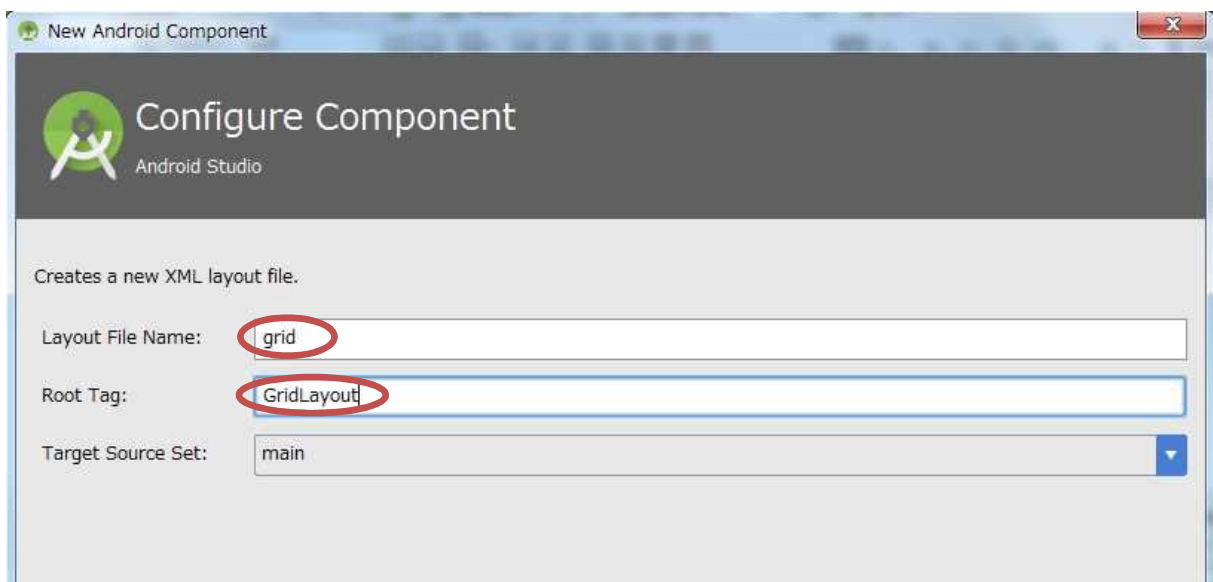
LinearLayout はウィジェットが直線状に並ぶレイアウトである。このことを確かめるために、ボタン 2 個とエディットテキスト 1 個を順に配置する。

(図は次のページ)



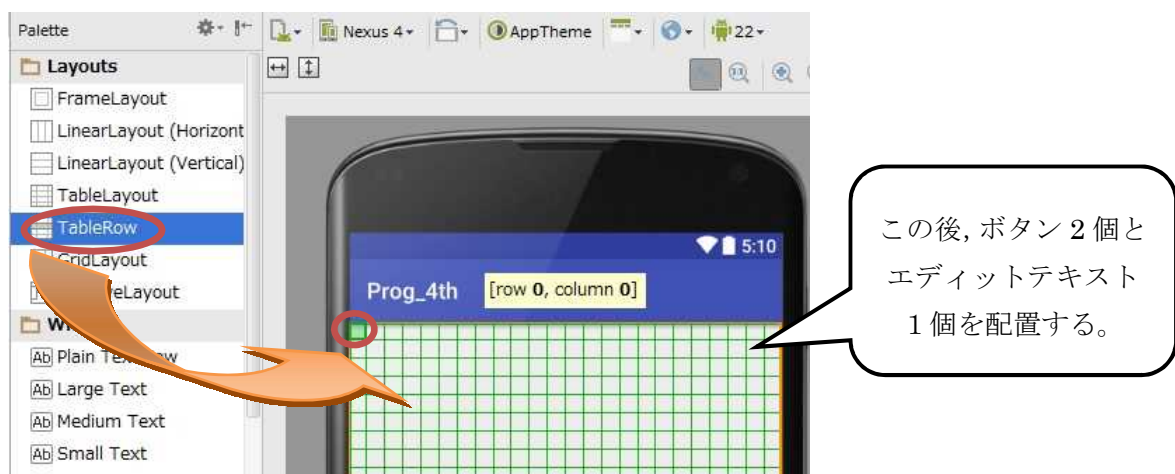
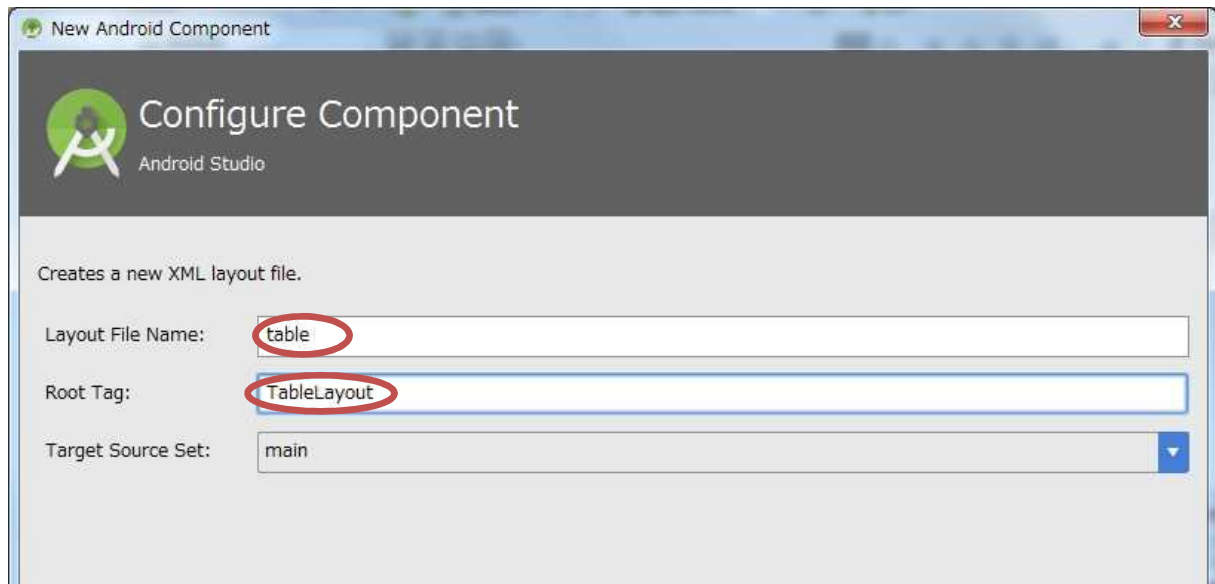
同様に、『XML layout file』を新規作成する。『Layout File Name』の欄には「grid」と入力し、『Root Tag』の欄に「GridLayout」と入力して、『終了』をクリックする。

GridLayout は格子状にウィジェットが並ぶレイアウトである。このことを確かめるために、ボタン 3 個とエディットテキスト 1 個を順に配置する。

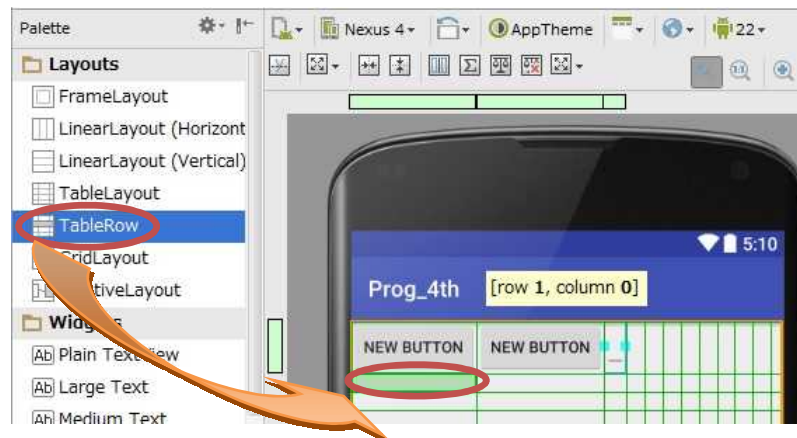


同様に、『XML layout file』を新規作成する。『Layout File Name』の欄には「table」と入力し、『Root Tag』の欄に「TableLayout」と入力して、『終了』をクリックする。

TableLayout は、TableRow を配置して、その中にウィジェットを配置するレイアウトである。このことを確かめるために、まず TableRow を『Layouts』からドラッグして配置し、次にその中にボタン 2 個とエディットテキスト 1 個を順に配置する。



更に、TableRow を配置して、その中にボタン 2 個とエディットテキスト 1 個を順に配置する。





『MainActivity4.java』のタブを開いて、ボタンをクリックするとそれぞれのレイアウトの例の画面が表示される機能を付け加える。

```
Button btn = (Button)this.findViewById(R.id.button);
btn.setOnClickListener(
    new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            setContentView(R.layout.linear);
        }
    }
);

Button btn2 = (Button)this.findViewById(R.id.button2);
btn2.setOnClickListener(
    new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            setContentView(R.layout.grid);
        }
    }
);

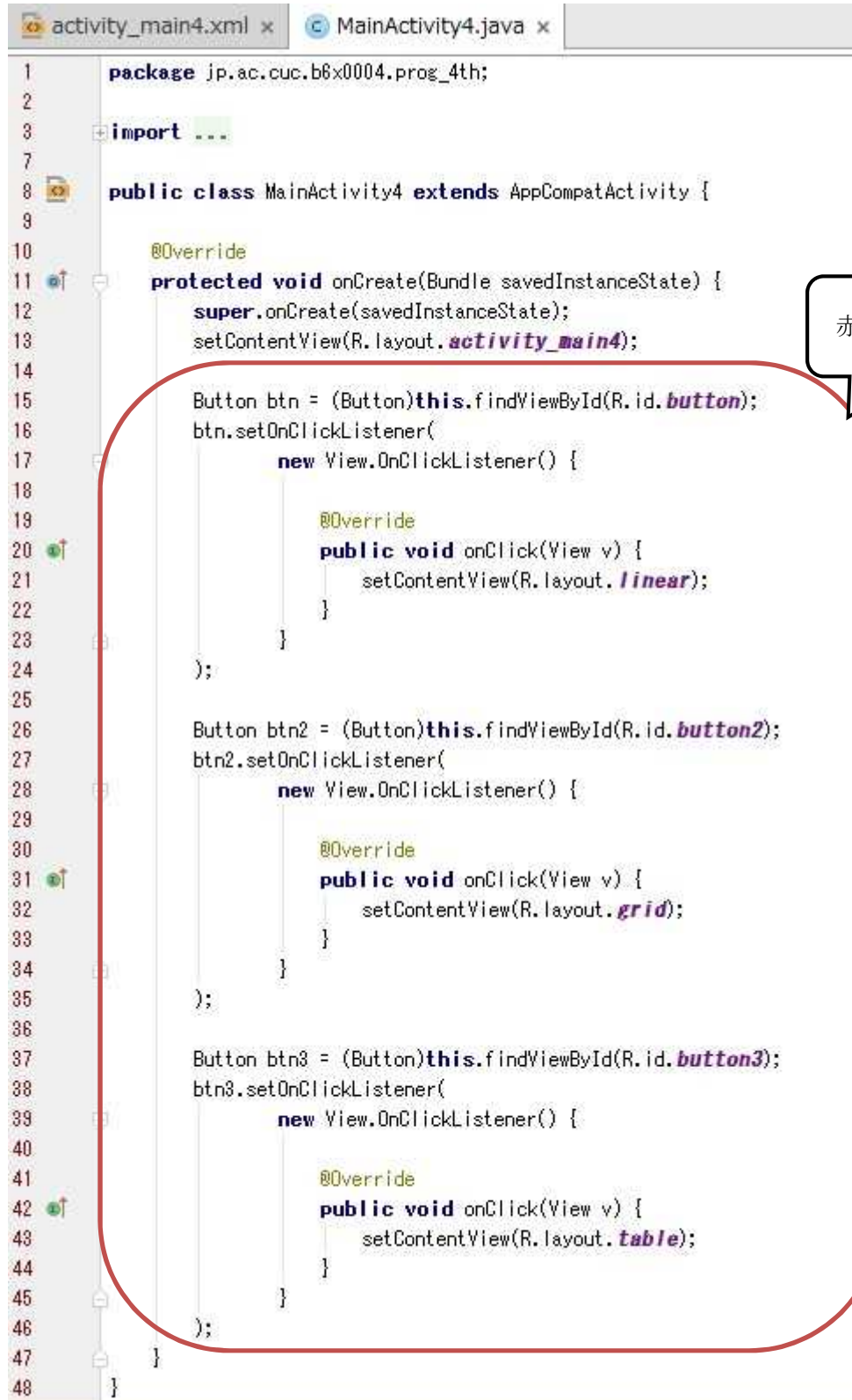
Button btn3 = (Button)this.findViewById(R.id.button3);
btn3.setOnClickListener(
    new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            setContentView(R.layout.table);
        }
    }
);
```

Button ウィジェットのインスタンス及びイベントリスナー

【Button ウィジェットのインスタンス及びイベントリスナーを付加】

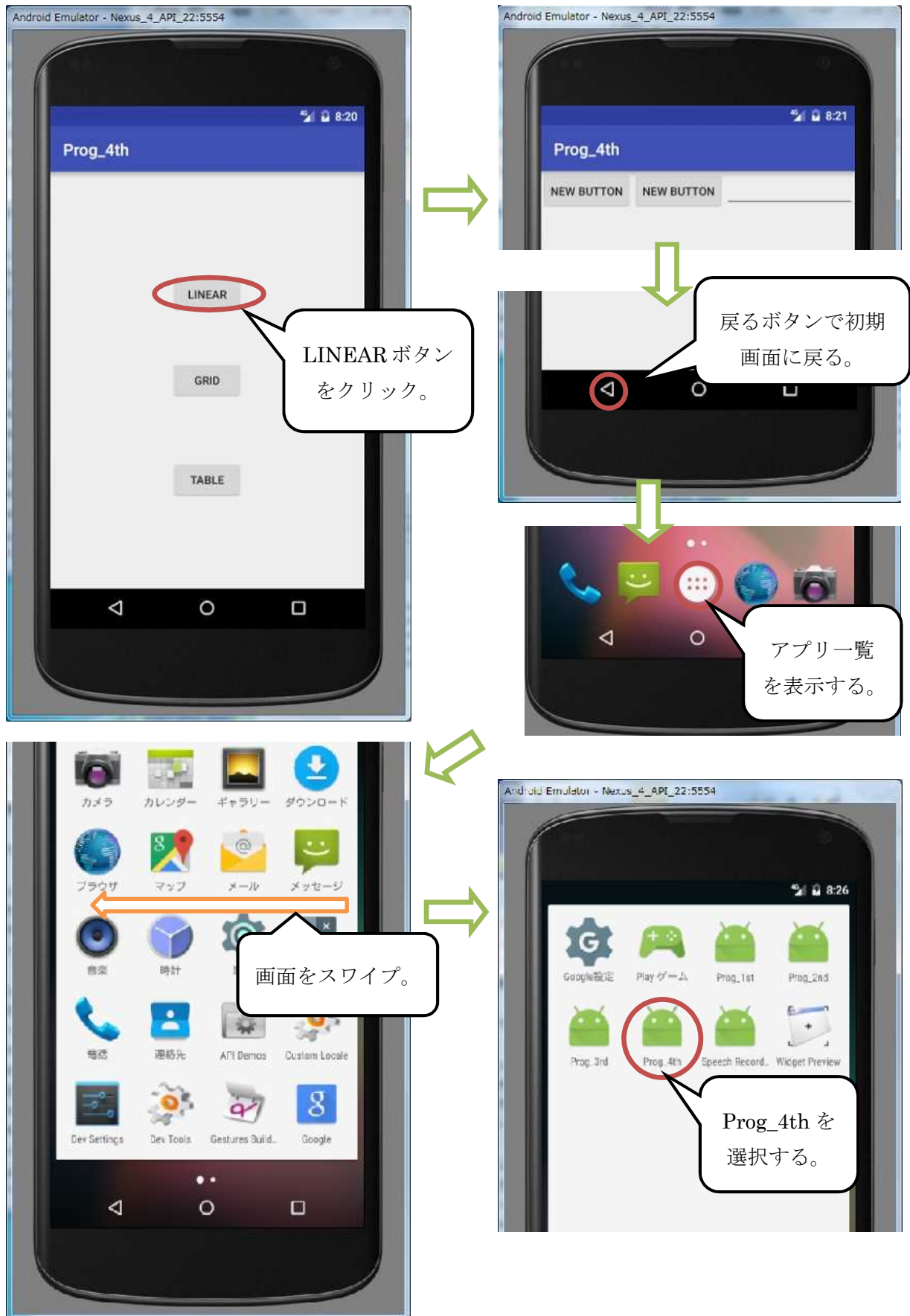
前のページの赤色の枠内にある内容を次ページの図の赤色の枠の位置 (onCreate メソッドの内部 ; 中括弧『{』と『}』との間) に入力する。



```
1 package jp.ac.cuc.b6x0004.prog_4th;
2
3 import ...
4
5
6
7
8 public class MainActivity4 extends AppCompatActivity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_main4);
14
15         Button btn = (Button) this.findViewById(R.id.button);
16         btn.setOnClickListener(
17             new View.OnClickListener() {
18
19                 @Override
20                 public void onClick(View v) {
21                     setContentView(R.layout.linear);
22                 }
23             }
24         );
25
26         Button btn2 = (Button) this.findViewById(R.id.button2);
27         btn2.setOnClickListener(
28             new View.OnClickListener() {
29
30                 @Override
31                 public void onClick(View v) {
32                     setContentView(R.layout.grid);
33                 }
34             }
35         );
36
37         Button btn3 = (Button) this.findViewById(R.id.button3);
38         btn3.setOnClickListener(
39             new View.OnClickListener() {
40
41                 @Override
42                 public void onClick(View v) {
43                     setContentView(R.layout.table);
44                 }
45             }
46         );
47     }
48 }
```

『保存』のアイコンをクリックして、**MainActivity4.java** を上書き保存し、実行ボタンをクリ

ックする。



初期画面に戻ったら、GRID ボタンをクリックする。(表示結果は下左図)

画面が切り替わったら、戻るボタンでアプリ一覧から『Prog_4th』を選択し、初期画面で TABLE ボタンをクリックする。(表示結果は下右図)



ここまでの動作確認が出来たら、完成となる。

提出物：

- 1) 画面のレイアウト設定ファイル **linear.xml**
- 2) 画面のレイアウト設定ファイル **grid.xml**
- 3) 画面のレイアウト設定ファイル **table.xml**
- 4) 画面のレイアウト設定ファイル **activity_main4.xml**
- 5) アクティビティのソースファイル **MainActivity4.java**