

2016 年 10 月 20 日 (木) 実施

ウィジェット

ウィジェットとは

Android アプリの画面上で用いられるユーザインターフェイスの要素をウィジェットと呼ぶ。ウィジェットは `android.widget` パッケージに収納されている。

これまでの教材では、ウィジェットのうち、ボタンやテキストビューを扱ったが、他にも様々なウィジェットが用意されている。

* 詳細は次の Android Developers のページを参照

<http://developer.android.com/reference/android/widget/package-summary.html>

** 日本国内では、`android.appwidget` パッケージに収納されているアプリウィジェットもウィジェットと呼ばれることがあるので注意が必要である。アプリウィジェットは、新規にアクティビティを起動することなく、アプリデータやサービスに素早くアクセスするためのもので、アプリに埋め込んで利用出来る。詳細は次の Android Developers のページを参照

<http://developer.android.com/reference/android/appwidget/package-summary.html>

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_3rd」(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「MainActivity3」とする。

3) AVD の設定

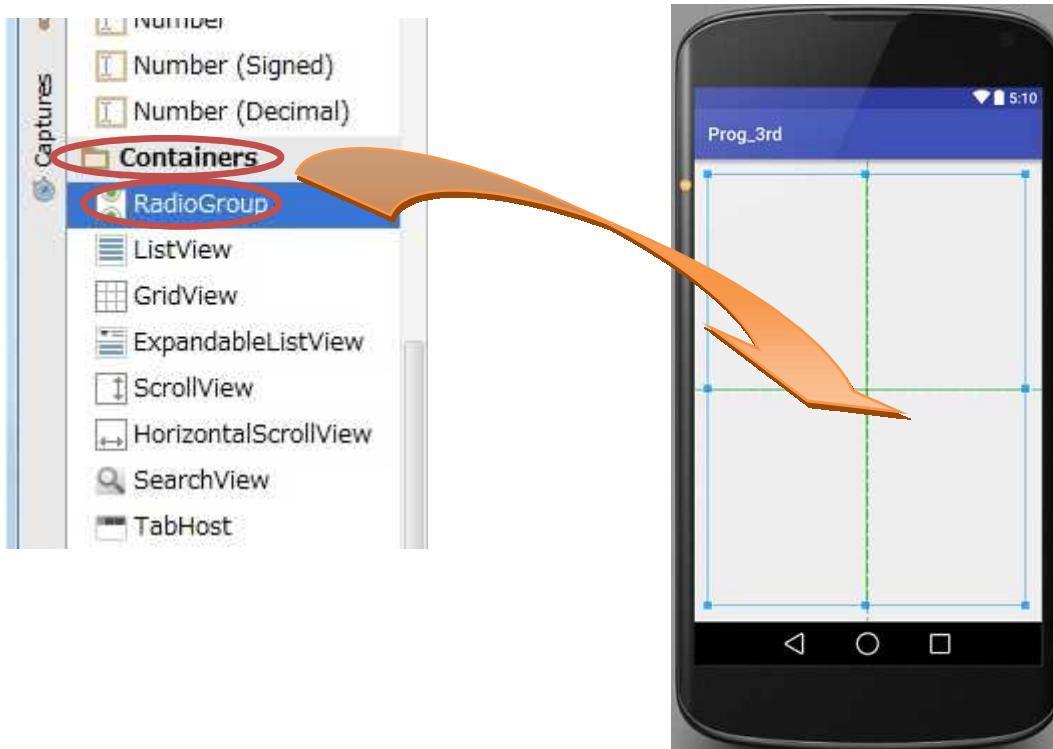
第 1 回の授業で作成した AVD の設定は H ドライブにあって残るが、SDK のシステムイメージは C ドライブにあるので、消失している。そこで、『Download』をクリックして、インストールし直す (第 2 回教材 p.4)。

課題

今回は、レイアウトにラジオボタングループ及びラジオボタンを配置し、選択するラジオボタンを変えると、トーストを用いて画面に一定時間メッセージを表示するアプリを作成することにより、ウィジェットのイベント処理の基本を学ぶ。

Android アプリの作成

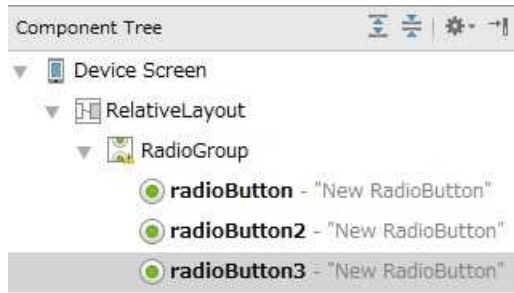
『activity_main3.xml』のタブを開く。先ず元から貼り付けられている「Hello World!」と表示されているテキストビューを選択し、『Delete』キーで削除する。続いて、『Palette』の『Containers』から『RadioGroup』をドラッグして配置する。



『Widgets』から『RadioButton』を上から順に 3 個ドラッグして配置する。



この時点での『Component Tree』の表示



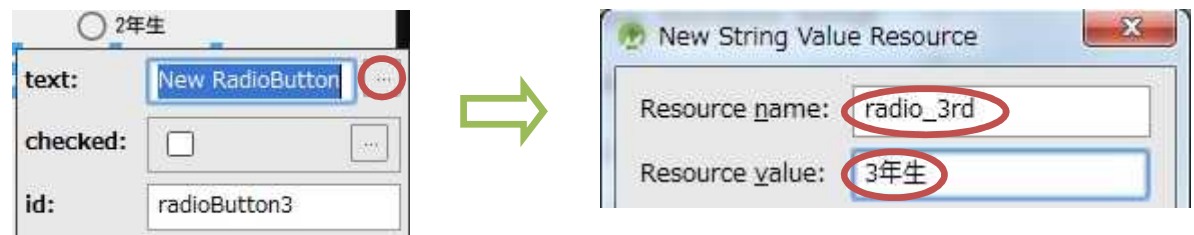
それぞれのラジオボタンをダブルクリックし、『…』ボタンを押して出てきた『Resources』では『New Resource』 → 『New string Value』を選択して、その値を変更していく。



radioButton Resource name: radio_1st Resource value: 1年生



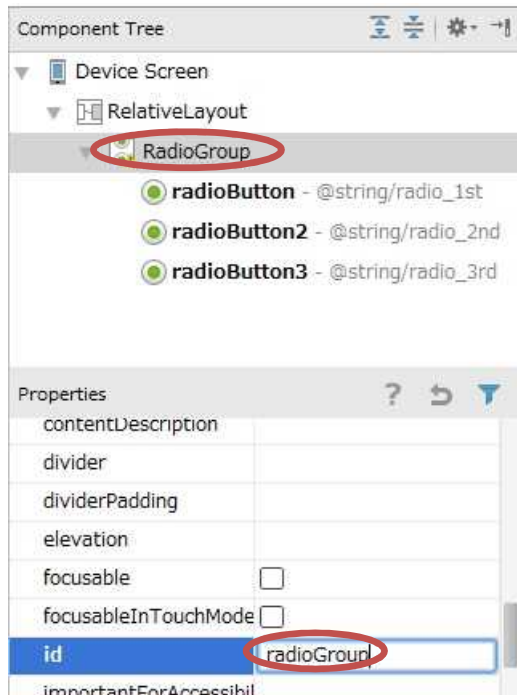
radioButton2 Resource name: radio_2nd Resource value: 2年生



radioButton3 Resource name: radio_3rd Resource value: 3年生

『Component Tree』で『RadioGroup』を選択して、『Properties』の『id』欄の値として『radioGroup』を入力する。

(次ページの図)



『保存』のアイコンをクリックして、**activity_main3.xml** を上書き保存する。

『**MainActivity3.java**』のタブを開いて、**選択するラジオボタンを変えると、トーストを用いて画面に一定時間メッセージを表示する機能**を付け加える。

```
final Activity activity = this;

RadioGroup rg = (RadioGroup) this.findViewById(R.id.radioGroup);
rg.setOnCheckedChangeListener(
    new OnCheckedChangeListener() {
        @Override
        public void onCheckedChanged(RadioGroup group, int checkedId) {
            RadioButton rbtn = (RadioButton) findViewById(checkedId);
            Toast tst = Toast.makeText(activity,
                "現在" + rbtn.getText(), Toast.LENGTH_SHORT);
            tst.show();
        }
    }
);
```

Toast, RadioGroup のインスタンス及びイベントリスナー

【Toast, RadioGroup のインスタンス及びイベントリスナーを付加】

このページの上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_3rd;
2
3  import android.app.Activity;
4  import android.os.Bundle;
5  import android.support.v7.app.AppCompatActivity;
6  import android.widget.RadioButton;
7  import android.widget.RadioGroup;
8  import android.widget.Toast;
9
10 public class MainActivity3 extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main3);
16         final Activity activity = this;
17
18         RadioGroup rg = (RadioGroup)this.findViewById(R.id.radioGroup);
19         rg.setOnCheckedChangeListener(
20             new OnCheckedChangeListener() {
21                 @Override
22                 public void onCheckedChanged(RadioGroup group, int checkedId) {
23                     RadioButton rbtn = (RadioButton)findViewById(checkedId);
24                     Toast tst = Toast.makeText(activity,
25                         "現在"+rbtn.getText(), Toast.LENGTH_SHORT);
26                     tst.show();
27                 }
28             });
29     }
30 }
31 }
32 }

```

Toastクラスのインスタンスは、画面に一定時間メッセージを表示するために用いる。makeTextメソッドの第1引数にはContextというアプリの環境についての情報へのインターフェイスを書くことになっているが、ここでは、『このアクティビティ』を表すthisをactivityと名付けた変数に代入して、指定している。makeTextメソッドの第2引数に記述した文字列を第3引数に指定した表示時間分、画面にメッセージを表示する。表示時間にはLENGTH_SHORT及びLENGTH_LONGの2種類がある。

この段階では、インポートの候補が複数あるので、『OnCheckedChageListner』にマウスポインタを当て、『Alt』のキーを抑えながら『Enter』のキーを打つ。出てきたインポートのオプションでは、android.widget.RadioGroupを選択する。

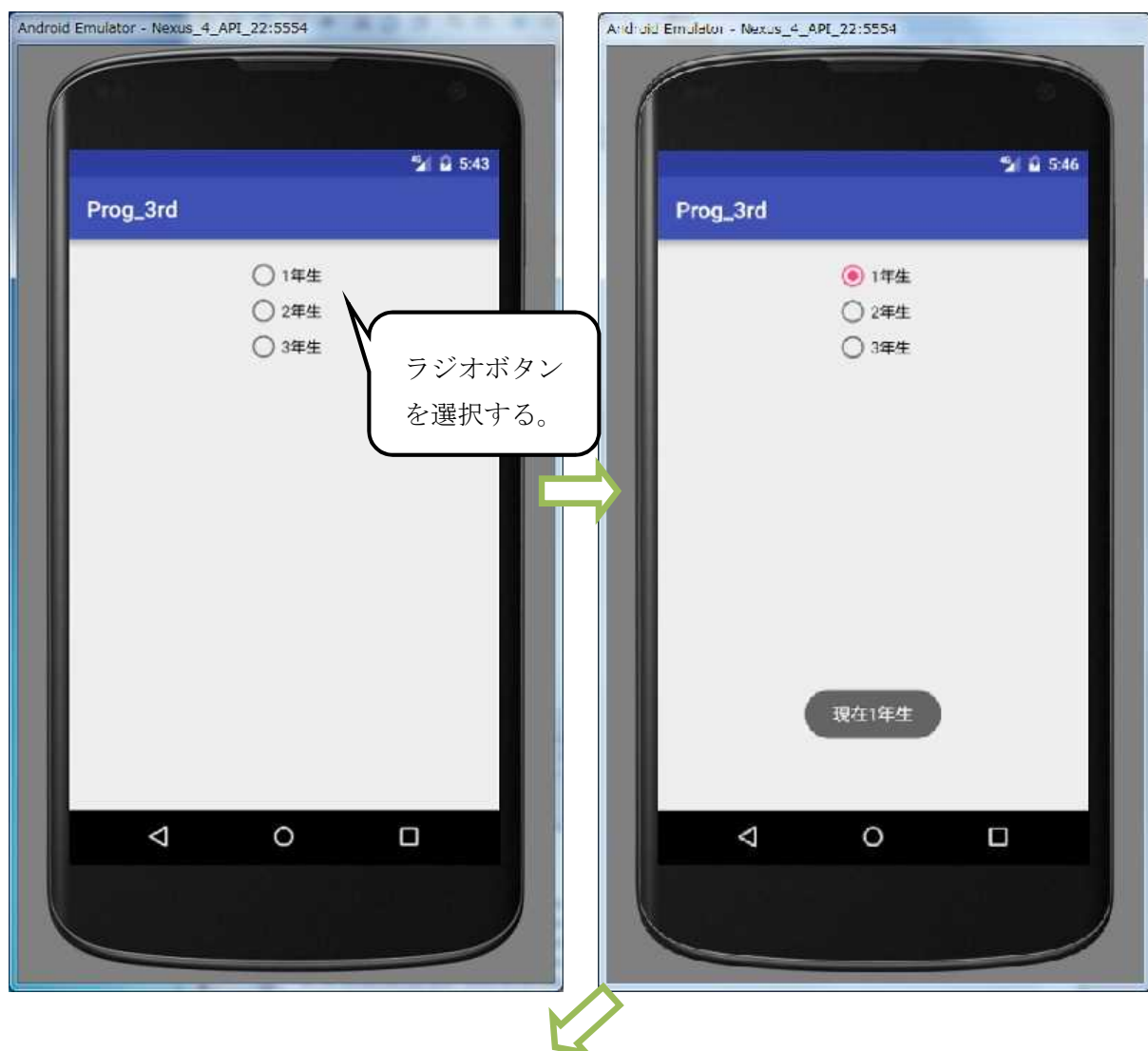
* これで、Android Studioでは、RadioGroup.OnCheckedChangeListenerに変わる。

(次ページの図)



『保存』のアイコンをクリックして、**MainActivity3.java** を上書き保存し、実行ボタンをクリックする。

起動したアプリのラジオボタンの状態を次々に変えて、画面に表示されるメッセージの内容を確認する。ここまでの動作確認が出来たら、完成となる。



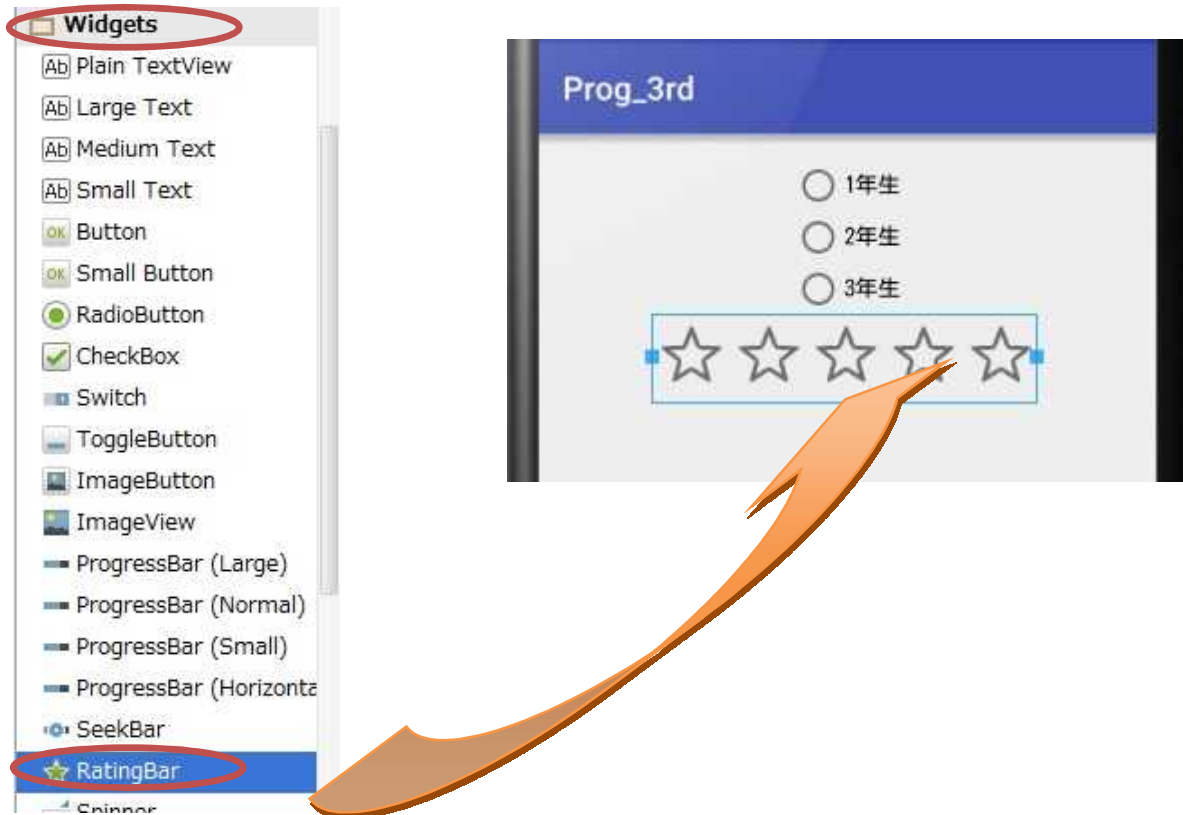


提出物：

- 1) 画面のレイアウト設定ファイル `activity_main3.xml`
- 2) アクティビティのソースファイル `MainActivity3.java`

【余裕のある人向けの課題】

`activity_main3.xml` 上に `RatingBar` (評価バー) を配置して、利用出来る様にアプリを拡張する。



`RatingBar` のイベントリスナーでは、評価バーをクリックして状態を変えたことを捉える。これを `MainActivity3.java` の `onCreate` メソッド内に加える。

```
RatingBar rtb = (RatingBar)this.findViewById(R.id.ratingBar);
rtb.setOnRatingBarChangeListener(
    new OnRatingBarChangeListener() {
        @Override
        public void onRatingChanged(RatingBar ratingBar, float rating,
            boolean fromUser) {
            Toast tst = Toast.makeText(activity,
                "評価が"+rating+"に変更されました", Toast.LENGTH_SHORT);
            tst.show();
        }
    }
);
```

`RatingBar` のインスタンス及びイベントリスナー


```

1 package jp.ac.cuc.b6x0004.prog_3rd;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.support.v7.app.AppCompatActivity;
6 import android.widget.RadioButton;
7 import android.widget.RadioGroup;
8 import android.widget.RatingBar;
9 import android.widget.Toast;
10
11 public class MainActivity3 extends AppCompatActivity {
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main3);
17         final Activity activity = this;
18
19         RadioGroup rg = (RadioGroup) this.findViewById(R.id.radioGroup);
20         rg.setOnCheckedChangeListener(
21             new RadioGroup.OnCheckedChangeListener() {
22                 @Override
23                 public void onCheckedChanged(RadioGroup group, int checkedId) {
24                     RadioButton rbtn = (RadioButton) findViewById(checkedId);
25                     Toast tst = Toast.makeText(activity,
26                         "現在" + rbtn.getText(), Toast.LENGTH_SHORT);
27                     tst.show();
28                 }
29             }
30         );
31
32         RatingBar rtb = (RatingBar) this.findViewById(R.id.ratingBar);
33         rtb.setOnRatingBarChangeListener(
34             new OnRatingBarChangeListener() {
35                 @Override
36                 public void onRatingChanged(RatingBar ratingBar, float rating,
37                     boolean fromUser) {
38                     Toast tst = Toast.makeText(activity,
39                         "評価が" + rating + "に変更されました", Toast.LENGTH_SHORT);
40                     tst.show();
41                 }
42             }
43         );
44     }
45 }
46

```

青色の枠内を付加

ここでは、『OnRatingBarChangeListener』にマウスポインタを当て、『Alt』のキーを抑えながら『Enter』のキーを打てば、適切なインポートが加えられる。

```

? android.widget.RatingBar.OnRatingBarChangeListener? Alt+Enter .id.ratingBar);
rtb.setOnRatingBarChangeListener(
34 new OnRatingBarChangeListener() {

```

『保存』のアイコンをクリックして、**MainActivity3.java** を上書き保存し、実行ボタンをクリックする。

起動したアプリの評価バーの状態を変えて、画面に表示されるメッセージの内容を確認する。ここまでの動作確認が出来たら、完成となる。

