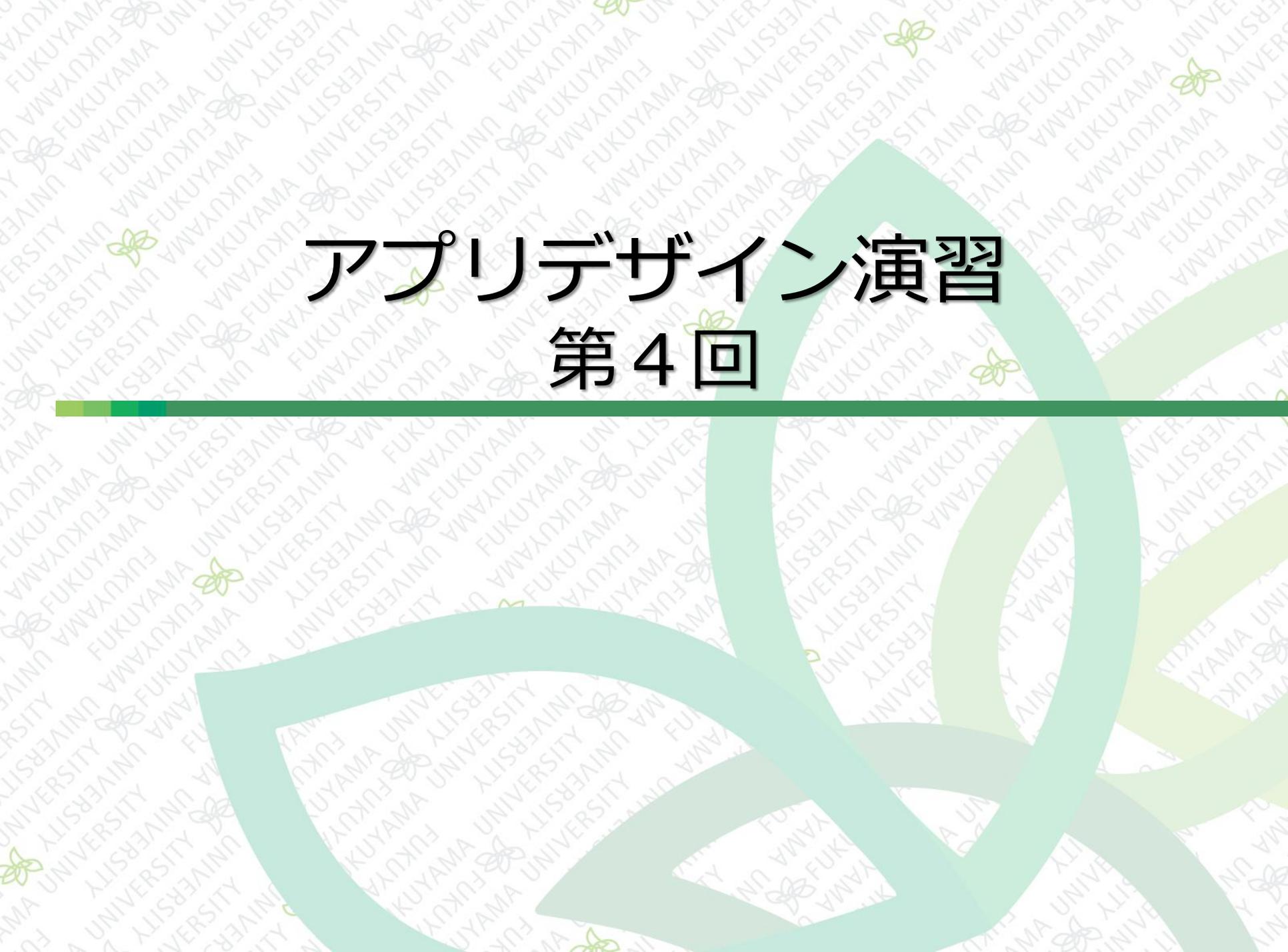


アプリデザイン演習 第4回

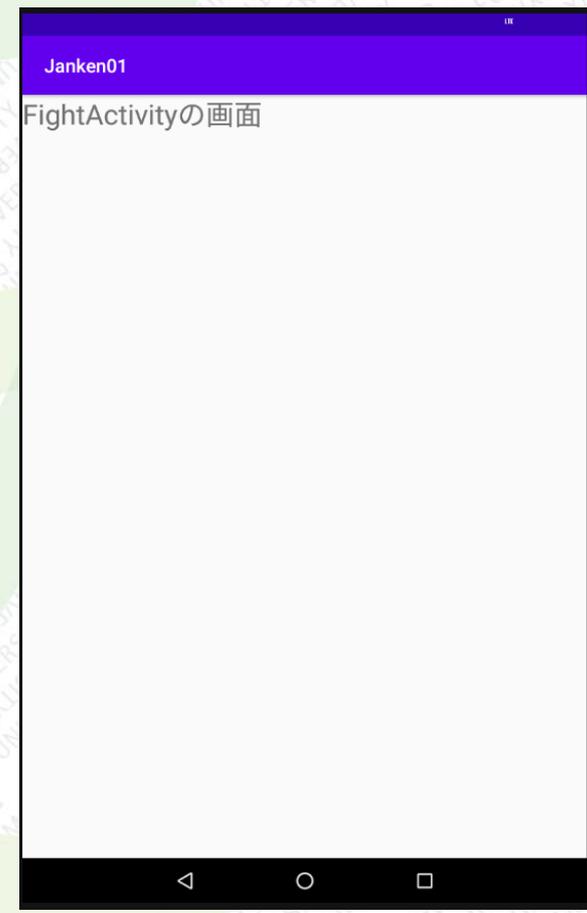


新しくアクティビティを追加する(前回まで)

勝負ボタンを押すと、新たに画面(アクティビティ)が表示されることを確認

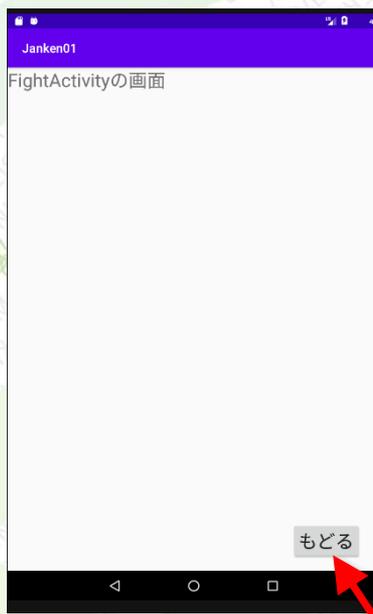


押すと、画面
が表示される



「もどる」ボタンを追加する

「勝負！」ボタンを押すと、新たに画面(アクティビティー)が表示される
「もどる」ボタンを押すと、もとの画面に戻る

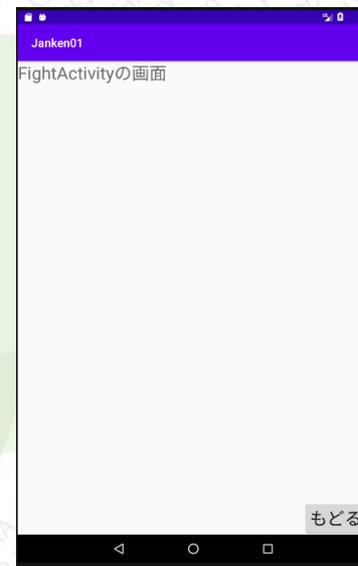


「もどる」ボタンを追加する

「もどる」ボタンを押すと、画面を閉じて元の画面に戻るようにする

1) 下記を「activity_fight.xml」に追記する

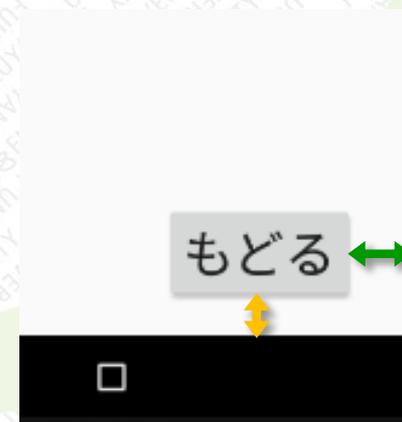
```
<Button
    android:id="@+id/buttonBack"
    android:text="もどる"
    android:textSize="30sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintRight_toRightOf="parent" />
```



2) さらに2行追記して、位置を調整

```
    android:layout_height="wrap_content"
    { android:layout_marginEnd="30sp"
    android:layout_marginBottom="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintRight_toRightOf="parent" />
```

ボタンの右端に
空白 ←→ を入れる



ボタンの下端に空白 ↕ を入れる

「もどる」ボタンを追加する

「もどる」ボタンを押すと、画面を閉じて元の画面に戻るようになる

(3) 「FightActivity.java」にボタンが押されたときの処理（アクティビティを破棄する処理）を追記する `finish()`;関数で破棄される

```
public class FightActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_fight);  
        Button btnBack = (Button)findViewById(R.id.buttonBack); // もどるボタン  
  
        btnBack.setOnClickListener( new View.OnClickListener(){  
            public void onClick( View view ){  
                finish();  
            }  
        });  
    }  
}
```

追加する

「もどる」ボタンを追加する

下記のような表示は、クリックすると展開される



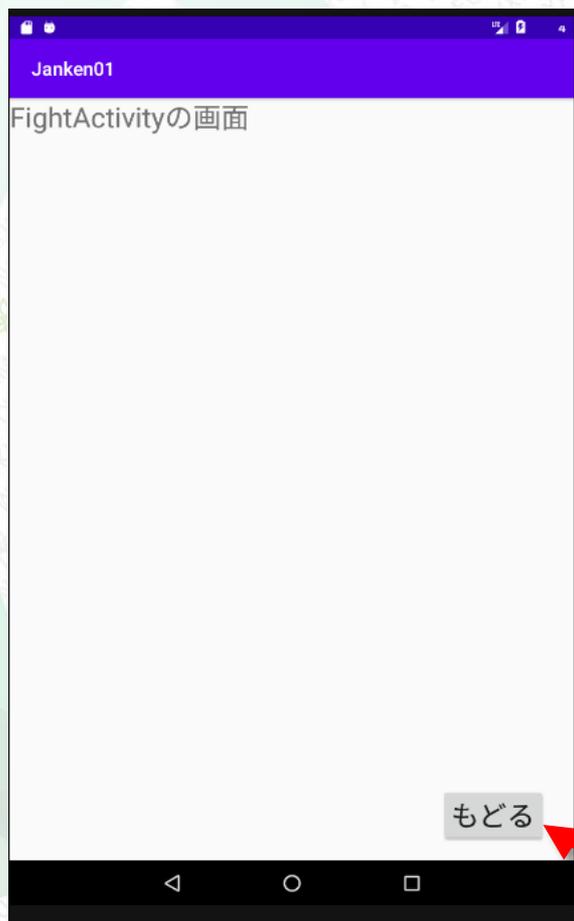
```
btnC.setOnClickListener( (view) → {  
    showResult( user: 3);  
    ImageView imageView = (ImageView) findViewById(R.id.imageView);  
    imageView.setImageResource(R.drawable.pa);  
});
```



```
btnC.setOnClickListener( new View.OnClickListener(){  
    public void onClick( View view ){  
        showResult( user: 3);  
        ImageView imageView = (ImageView) findViewById(R.id.imageView);  
        imageView.setImageResource(R.drawable.pa);  
    }  
});
```

「もどる」ボタンを追加する

「もどる」ボタンを押すと、元の画面に戻ることを確認する



戻る



勝敗結果を表示する . . .

まず、「FightActivityの画面」のところにグー、チョキ、パーの数値
1, 2, 3を表示する

ユーザーが選択したグー・チョキ・パーの情報を、新しく開いたアクティビティー
に渡す



ここに
グー : 1
チョキ : 2
パー : 3
を表示



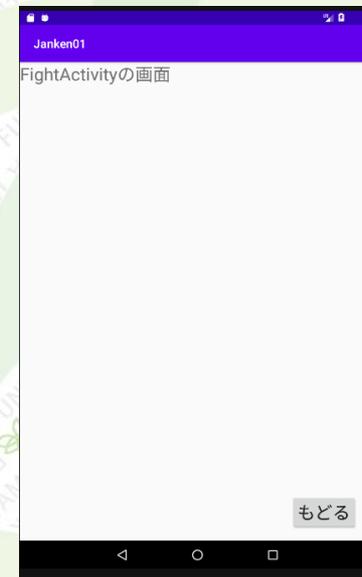
新しいアクティビティに値を渡す

新しいアクティビティに値を渡すには、
`putExtra()` メソッド と
`get ~ Extra()` メソッド
を使う

`putExtra()` → `getIntExtra()`
整数値を渡す場合



<値を渡す側>



<値を受け取る側>

新しいアクティビティに値を渡す

「MainActivity.java」に以下の内容を追記する

```
public class MainActivity extends AppCompatActivity {  
  
    TextView txtView01;  
    int user_hand;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {
```

追加する

```
        btnGo.setOnClickListener( new View.OnClickListener(){  
            public void onClick( View view ){  
                Intent intent = new Intent();  
                intent.setClassName("com.example.janken01","com.example.janken01.FightActivity");  
                intent.putExtra("USER_HAND", user_hand);  
                startActivity(intent);  
            }  
        });
```

```
        public void showResult(int user){ // 勝敗を決めて表示する  
            user_hand = user;  
            int b = 0;  
            Random rnd = new Random();  
            int c = rnd.nextInt(3) + 1; // 乱数 1 or 2 or 3  
            if( c==1 ){ // コンピュータの手を表示
```

新しいアクティビティに値を渡す

「FightActivity.java」に以下の内容を追記する

```
import android.widget.ImageView;
import android.content.Intent;
import android.widget.TextView;

public class FightActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_fight);
        Button btnBack = (Button)findViewById(R.id.buttonBack); // もどるボタン

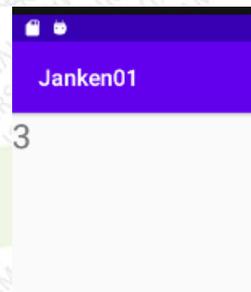
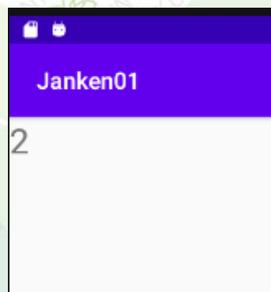
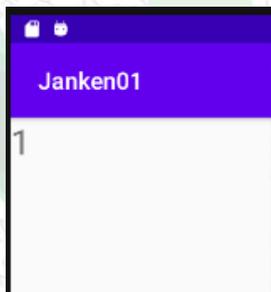
        Intent intent = getIntent();
        int user = intent.getIntExtra("USER_HAND",0);
        TextView txtView = (TextView)findViewById(R.id.text1);
        txtView.setText(String.valueOf(user));

        btnBack.setOnClickListener( new View.OnClickListener(){
            public void onClick( View view ){
                finish();
            }
        });
    }
}
```

追加する

新しいアクティビティに値を渡す

実行して動作を確認する
数値が表示されていればOK



勝敗結果の表示

「FightActivity」に勝敗結果を表示する

- ・ コンピュータの手、ユーザーの手を画像で表示する
- ・ 「勝ち・負け・あいこ」を表示する



勝敗結果の表示

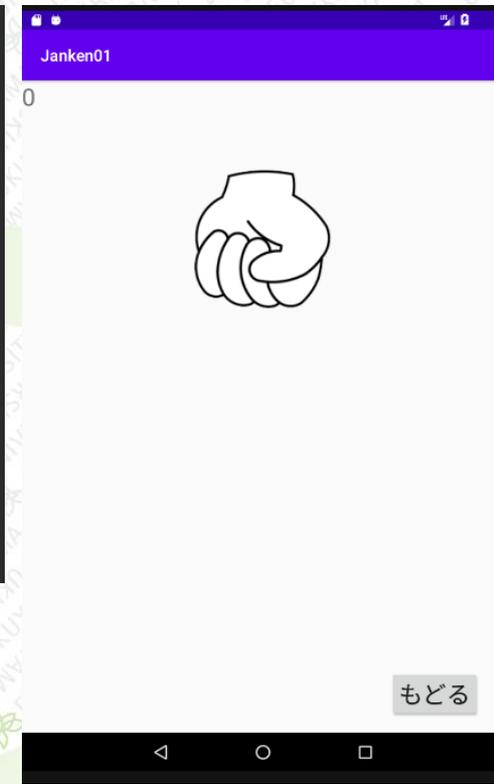
- ・コンピュータの手の画像を表示する（逆さまにする）
「activity_fight.xml」で下記を追記する

とりあえず、
グーの画像

```
<ImageView  
    android:id="@+id/imageView2"  
    android:layout_centerInParent="true"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:src="@drawable/gu"  
    android:scaleY="-1"  
    android:layout_marginTop="100sp"  
    app:layout_constraintLeft_toLeftOf="parent"  
    app:layout_constraintRight_toRightOf="parent"  
    app:layout_constraintTop_toTopOf="parent" />
```

この1行を入れると、
画像が上下反転する

画面の上端から画像までの
距離、数値が小さいと
画像が上方に配置される



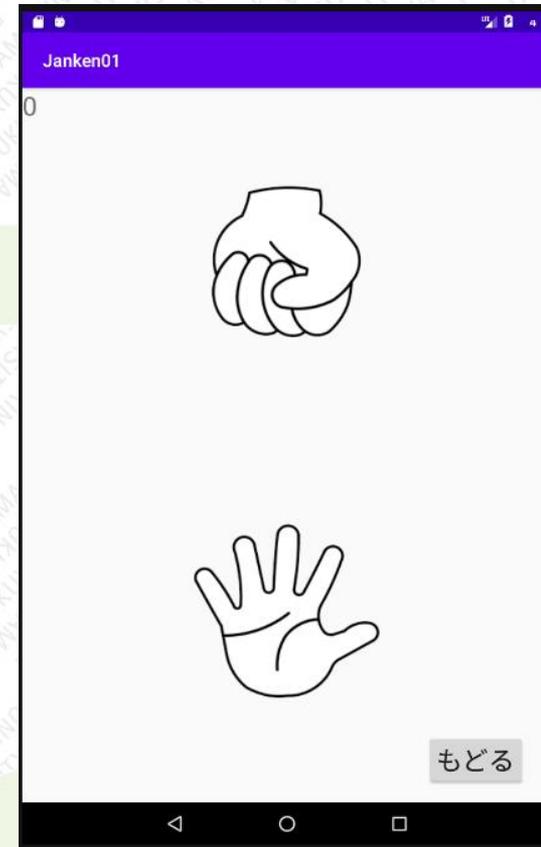
勝敗結果の表示

- ・ さらにユーザの手の画像を表示する
「activity_fight.xml」で下記を追記する

とりあえず、
パーの画像

```
<ImageView  
    android:id="@+id/imageView3"  
    android:layout_centerInParent="true"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:src="@drawable/pa"  
    android:layout_marginBottom="120sp"  
    app:layout_constraintLeft_toLeftOf="parent"  
    app:layout_constraintRight_toRightOf="parent"  
    app:layout_constraintBottom_toBottomOf="parent" />
```

前スライドとはちょっと異
なるので注意



勝敗結果の表示

- ・ ユーザが選んだグー・チョキ・パーの画像が表示されるようにする



勝敗結果の表示

- ・ ユーザが選んだグー・チョキ・パーの画像を表示されるようにする
「FightActivity.java」で下記を追記する

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_fight);
    Button btnBack = (Button)findViewById(R.id.buttonBack); // もどるボ

    Intent intent = getIntent();
    int user = intent.getIntExtra("USER_HAND",0);
    TextView txtView = (TextView)findViewById(R.id.text1);
    txtView.setText(String.valueOf(user));

    ImageView imageView = (ImageView) findViewById(R.id.imageView3);
    if( user==1 ) {
        imageView.setImageResource(R.drawable.gu);
    }else if( user==2 ) {
        imageView.setImageResource(R.drawable.choki);
    }else{
        imageView.setImageResource(R.drawable.pa);
    }

    btnBack.setOnClickListener(new View.OnClickListener(){
```

追加する

勝敗結果の表示

- 「MainActivity」から「FightActivity」へ Intent を使って
コンピュータの手 と 勝敗結果 の数値を渡す
「MainActivity.java」で下記を追記する

追加する

```
public class MainActivity extends AppCompatActivity {  
  
    TextView txtView01;  
    int user_hand;  
    int com_hand;  
    int result; // 1:ユーザの勝ち,0:あいこ,-1:COMの勝ち
```

追加する

```
btnGo.setOnClickListener( new View.OnClickListener(){  
    public void onClick( View view ){  
        Intent intent = new Intent();  
        intent.setClassName("com.example.janken01","com.  
        intent.putExtra("USER_HAND", user_hand);  
        intent.putExtra("COM_HAND", com_hand);  
        intent.putExtra("RESULT", result);  
        startActivity(intent);  
    }  
});
```

勝敗結果の表示

<続き>

```
public void showResult(int user){ // 勝敗を決めて表示する
    user_hand = user;
    int b = 0;
    Random rnd = new Random();
    int c = rnd.nextInt(3) + 1; // 乱数 1 or 2 or 3
    if( c==1 ){ // コンピュータの手を表示
        txtView01.setText("コンピュータ：グー");
    }else if( c==2 ){
        txtView01.setText("コンピュータ：チョキ");
    }else{
        txtView01.setText("コンピュータ：パー");
    }
    com_hand = c;
    if( user==c ){ // あいこだった
        result = 0;
        return;
    }
    if( user==1 ){ // 勝敗を判別
        if( c==2 ) b=1; else b=0;
    }else if( user==2 ){
        if( c==3 ) b=1; else b=0;
    }else{
        if( c==1 ) b=1; else b=0;
    }

    if( b==1 ){
        result = 1;
    }else{
        result = -1;
    }
}
```

追加する

追加する

勝敗結果の表示

「FightActivity.java」で下記を追記する

追加する

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_fight);
    Button btnBack = (Button)findViewById(R.id.buttonBack); // もどるボ

    Intent intent = getIntent();
    int user = intent.getIntExtra("USER_HAND",0);
    int com = intent.getIntExtra("COM_HAND",0);
    int res = intent.getIntExtra("RESULT",0);
    TextView txtView = (TextView)findViewById(R.id.text1);
    txtView.setText(String.valueOf(user));

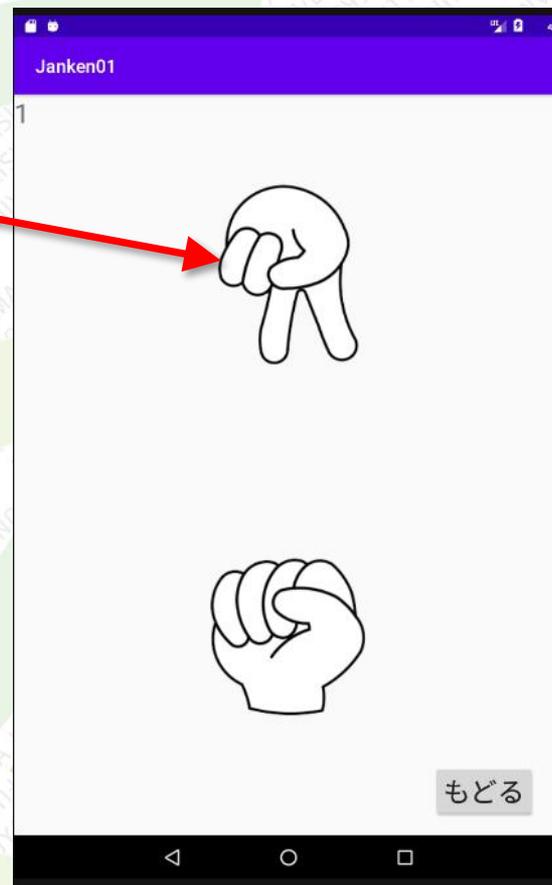
    ImageView imageView2 = (ImageView) findViewById(R.id.imageView2);
    if( com==1 ) {
        imageView2.setImageResource(R.drawable.gu);
    }else if( com==2 ) {
        imageView2.setImageResource(R.drawable.choki);
    }else{
        imageView2.setImageResource(R.drawable.pa);
    }

    ImageView imageView = (ImageView) findViewById(R.id.imageView3);
```

追加する

勝敗結果の表示

ここで、表示が正しいか確認してみる



勝敗結果の表示

- 「FightActivity」に **勝敗結果** の文字列を表示する
「activity_fight.xml」で下記を追記する

```
<TextView
    android:id="@+id/text2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="あなたの勝ち!"
    android:textSize="50sp"
    android:textColor="#ff0000"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintBottom_toBottomOf="parent" />
```

このTextViewを追加



勝敗結果の表示

「FightActivity.java」で下記を追記する

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_fight);  
    Button btnBack = (Button)findViewById(R.id.buttonBack); // もど  
  
    Intent intent = getIntent();  
    int user = intent.getIntExtra("USER_HAND",0);  
    int com = intent.getIntExtra("COM_HAND",0);  
    int res = intent.getIntExtra("RESULT",0);  
    TextView txtView = (TextView)findViewById(R.id.text1);  
    txtView.setText(String.valueOf(user));  
  
    TextView txtView2 = (TextView)findViewById(R.id.text2);  
    if( res==1 ) {  
        txtView2.setText("あなたの勝ち!");  
    }else if( res==-1 ){  
        txtView2.setText("コンピュータの勝ち!");  
    }else{  
        txtView2.setText("あいこ");  
    }  
  
    ImageView imageView2 = (ImageView) findViewById(R.id.imageView2);  
}
```

追加する

勝敗結果の表示

「MainActivity.java」で下記の部分をコメントアウトする

コメントに
する

```
public void showResult(int user){ // 勝敗を決めて表示する
    user_hand = user;
    int b = 0;
    Random rnd = new Random();
    int c = rnd.nextInt(3) + 1; // 乱数 1 or 2 or 3
    if( c==1 ){ // コンピュータの手を表示
        //txtView01.setText("コンピュータ：グー");
    }else if( c==2 ){
        //txtView01.setText("コンピュータ：チョキ");
    }else{
        //txtView01.setText("コンピュータ：パー");
    }
    com_hand = c;
    if( user==c ){ // あいこだった
```

勝敗結果の表示

動作を確認して完了

