

[ 1 ] MyWourisu.java

/\*

-----  
                    ウォーリスの公式  
                    Android 4.4 (Kit Kat)  
                    Copyright(C) K.Niwa 2019.12.8  
-----

\*/

package jp.kiyo.wuena.mywourisu;

import android.content.Context;  
import android.graphics.Canvas;  
import android.graphics.Color;  
import android.graphics.Paint;  
import android.graphics.Rect;  
import android.util.AttributeSet;  
import android.view.View;  
import android.content.res.Resources; //画像用  
import android.graphics.\*;  
import android.view.\*;  
import jp.kiyo.wuena.mywourisu.MainActivity;

public class MyWourisu extends View {

    private Bitmap bitmap1 = null; //画像用  
    int flag=0; //自動識別子  
    int ct=0; //分子・分母の項の数  
    int count; //ループカウンター  
    double pai; //πの近似値  
    double sa=1; //πを求める過程で使用

    public MyWourisu(Context context) {  
        super(context);  
        init(context);  
    }

    public MyWourisu(Context context, AttributeSet attrs) {  
        super(context,attrs);  
        init(context);  
    }

```

    }

    public MyWourisu(Context context, AttributeSet attrs,int defStyle) {
        super(context,attrs,defStyle);
        init(context);
    }

    private void init(Context context) {
        Resources res = context.getResources(); //画像用
        bitmap1 = BitmapFactory.decodeResource(res, R.drawable.wallis); //画像用
    }

    @Override
    protected void onDraw(Canvas canvas) {
        // TODO 自動生成されたメソッド・スタブ

        float a=0;
        float b=0;

        super.onDraw(canvas);
        canvas.drawColor(Color.WHITE);
        Paint paint = new Paint();
        paint.setColor(Color.BLUE);
        paint.setAlpha(50);
        canvas.drawRect((getWidth()/2-240)+10,(getHeight()/2-343)+10,(getWidth()/2-240)
+470,(getHeight()/2-343)+675, paint);

        paint.setAlpha(10000);
        paint.setColor(Color.BLUE);

        for (int i=0;i<3;i++) {
            canvas.drawLine((getWidth()/2-240)+10+i,(getHeight()/2-343)+10+i,(getWidth
()/2-240)+10+i,(getHeight()/2-343)+675-i, paint);
            canvas.drawLine((getWidth()/2-240)+10+i,(getHeight()/2-343)+675-i,(getWidth
()/2-240)+470-i,(getHeight()/2-343)+675-i, paint);
            canvas.drawLine((getWidth()/2-240)+470-i,(getHeight()/2-343)+675-i,(getWidth
()/2-240)+470-i,(getHeight()/2-343)+10+i, paint);
            canvas.drawLine((getWidth()/2-240)+470-i,(getHeight()/2-343)+10+i,(getWidth
()/2-240)+10+i,(getHeight()/2-343)+10+i, paint);
        }
    }

```

```

        if (MainActivity.ritsu != 0) {
            a=(float) (0.7*320/MainActivity.ritsu); //----- < 画像の拡大・縮小の横の倍率を指定する >
            b=(float) (0.7*320/MainActivity.ritsu); //----- < 画像の拡大・縮小の縦の倍率を指定する >
        }
        else {
            a=(float) 1.0;
            b=(float) 1.0;
        }

        Matrix Mat = new Matrix (); //----- < 画像を拡大・縮小する >

        Mat.postScale (a, b); //-----
        Bitmap bitmap2 = Bitmap.createBitmap( //-----
            bitmap1,0,0, //-----
            bitmap1.getWidth (), //-----
            bitmap1.getHeight (), //-----
            Mat,true //-----
        ); //-----

        if (bitmap2 != null) {
            canvas.drawBitmap (bitmap2, (getWidth () /2-240)+190-20, (getHeight () /2-343)
+130, paint);
        }

        paint.setAlpha (10000);
        paint.setColor (Color.BLUE);
        paint.setTextSize (30.0f);
        canvas.drawText ("【ウォーリスの公式】", (getWidth () /2-240)+120-30+10, (getHeight () /2-343)+80, paint);
        paint.setTextSize (25.0f);
        canvas.drawText (" (  $\pi$  の近似値を求める ) ", (getWidth () /2-240)+110+10, (getHeight () /2-343)+110, paint);

        paint.setColor (Color.BLUE);
        paint.setTextSize (19.0f);
        canvas.drawText ("Copyright (C) K.Niwa 2019.11.12", (getWidth () /2-240)+100,

```

```

(getHeight()/2-343)+600+40, paint);

    ct++;
    sa=sa*((double)2*ct*2*ct)/((double)(2*ct-1)*(2*ct+1));
    pai=(double)2*sa;

    paint.setColor(Color.BLACK);
    paint.setTextSize(23.0f);
    canvas.drawText("分子・分母の項数 = "+2*ct+" 個ずつ" ,(getWidth()/2-240)
+60-20, (getHeight()/2-343)+310, paint);

    canvas.drawText("円周率  $\pi$ " , (getWidth()/2-240)+60-20, (getHeight()/2-343)
+300+50, paint);
    canvas.drawText("      2・2・4・4・6・6・8・8・・・", (getWidth()/2-240)+100-30,
(getHeight()/2-343)+340+50, paint);
    canvas.drawText("= 2", (getWidth()/2-240)+100-30, (getHeight()/2-343)+360+50,
paint);
    canvas.drawLine((getWidth()/2-240)+150-30,(getHeight()/2-343)+350+50,(getWidth()
()/2-240)+380+50,(getHeight()/2-343)+350+50,paint);
    canvas.drawText("      1・3・3・5・5・7・7・9・・・", (getWidth()/2-240)+100-30,
(getHeight()/2-343)+380+50, paint);

    paint.setColor(Color.BLUE);
    canvas.drawText("="+pai ,(getWidth()/2-240)+100-30, (getHeight()/2-343)+420+50,
paint);

    paint.setColor(Color.BLACK);
    paint.setTextSize(18.0f);
    canvas.drawText("※ 画面をタッチすると自動になります。" ,(getWidth()/2-240)
+50, (getHeight()/2-343)+470+40, paint);
    canvas.drawText("※ 画面をタッチすると自動が止まります。" ,(getWidth()/2-240)
+50, (getHeight()/2-343)+500+40, paint);
    canvas.drawText("※ 更に画面をタッチすると初期化されます。" ,(getWidth()
/2-240)+50, (getHeight()/2-343)+530+40, paint);
    canvas.drawText("※ 画面が暗くなったらタイトルバーをタッチ!" ,(getWidth()
/2-240)+50, (getHeight()/2-343)+560+40, paint);

    if (flag==1) {
        invalidate(); ///表示を更新する
    }

```

```

} //protected void onDraw(Canvas canvas)

@Override
public boolean onTouchEvent(MotionEvent event) {
    flag++;
    flag = flag % 3;
    if (flag==0) {
        ct=0;           //分子・分母の項の数
        sa=1;           //πを求める過程で使用
    }

    invalidate();     //表示を更新する
    return false;

} //public boolean onTouchEvent(MotionEvent event)

} //public class MyWourisu extends View

```

[ 2 ] activity\_main.xml

```

<?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_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

```

```

<jp.kiyo.wuena.mywourisu.MyWourisu
    android:id="@+id/myview1"
    android:layout_height="match_parent"
    android:layout_width="match_parent"/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

### [ 3 ] MainActivity.java

```

/*
-----
        ウォーリスの公式
        Android 4.4 (Kit Kat)
        Copyright (C) K.Niwa 2019.12.8
-----
*/

package jp.kiyo.wuena.mywourisu;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.DisplayMetrics;    //<画像の拡大・縮小に必要なライブラリ>
import android.app.Activity;
import android.view.Menu;

public class MainActivity extends AppCompatActivity {

    static int ritsu;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        DisplayMetrics metrics = new DisplayMetrics(); //<端末の情報を取得する>
        getWindowManager().getDefaultDisplay().getMetrics(metrics);
        StringBuilder buffer = new StringBuilder();
        buffer.append("densityDpi (ドット数 / インチ) : " + String.valueOf

```

```
(metrics.densityDpi) + "\n");  
    ritsu=metrics.densityDpi;  
    }  
}
```