［１］MyTenencoin.java

```
/*
-----------------------------------------------------------------
      １０円玉を投げて円周率πを求める
            Android 4.4（Kit Kat）
       Copyright(C) K.Niwa 2019.12.7
-----------------------------------------------------------------
*/

package jp.kiyo.wuena.mytenencoin;

import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.Rect;
import android.graphics.RectF;
import android.util.AttributeSet;
import android.view.MotionEvent;
import android.view.View;

public class MyTenencoin extends View {        //View クラスを継承した MyMonte クラス

    int i;//変数宣言と初期化
    int sum=0;     //格子点と重なったコインの個数
    int px;        //コインの位置の x 座標を整数型にしたもの
    int py;        //コインの位置の y 座標を整数型にしたもの
    int N=1999;    //実験回数の最大値
    int flag=0;    //コインを投げたか(1)、否か(0)、初期化する(2)の識別子
    double pai;
    int k=0;       //投げたコインの個数
    double[] x=new double[2001];
    double[] y=new double[2001];

    public MyTenencoin(Context context, AttributeSet attrs, int defStyle) {  //コンストラクタ
        super(context, attrs, defStyle);
    }

    public MyTenencoin(Context context, AttributeSet attrs) {      //コンストラクタ
        super(context, attrs);
```

```
    }

    public MyTenencoin(Context context) {    //コンストラクタ
        super(context);
    }

    //onDraw メソッド----------------------------------------------------------------------------
    @Override
    protected void onDraw(Canvas canvas) {

        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<2;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);
        }

        paint.setColor(Color.BLACK);
        //格子の描画
        canvas.drawLine((getWidth()/2-240)+40,(getHeight()/2-343)+70,(getWidth()/2-240)
+40,(getHeight()/2-343)+530,paint);
            canvas.drawLine((getWidth()/2-240)+140,(getHeight()/2-343)+70,(getWidth()
/2-240)+140,(getHeight()/2-343)+530,paint);
            canvas.drawLine((getWidth()/2-240)+240,(getHeight()/2-343)+70,(getWidth()
/2-240)+240,(getHeight()/2-343)+530,paint);
            canvas.drawLine((getWidth()/2-240)+340,(getHeight()/2-343)+70,(getWidth()
```

```
/2-240)+340,(getHeight()/2-343)+530,paint);
            canvas.drawLine((getWidth()/2-240)+440,(getHeight()/2-343)+70,(getWidth()
/2-240)+440,(getHeight()/2-343)+530,paint);

            canvas.drawLine((getWidth()/2-240)+15,(getHeight()/2-343)+100,(getWidth()
/2-240)+465,(getHeight()/2-343)+100,paint);
            canvas.drawLine((getWidth()/2-240)+15,(getHeight()/2-343)+200,(getWidth()
/2-240)+465,(getHeight()/2-343)+200,paint);
            canvas.drawLine((getWidth()/2-240)+15,(getHeight()/2-343)+300,(getWidth()
/2-240)+465,(getHeight()/2-343)+300,paint);
            canvas.drawLine((getWidth()/2-240)+15,(getHeight()/2-343)+400,(getWidth()
/2-240)+465,(getHeight()/2-343)+400,paint);
            canvas.drawLine((getWidth()/2-240)+15,(getHeight()/2-343)+500,(getWidth()
/2-240)+465,(getHeight()/2-343)+500,paint);

        paint.setColor(Color.BLUE);
        //表題の表示
        paint.setTextSize(24.0f);
         canvas.drawText("【コインを投げて円周率πを求める！】", (getWidth()/2-240)
+50-20,(getHeight()/2-343)+45, paint);

        paint.setColor(Color.BLACK);
        paint.setTextSize(19.0f);

        if (k==0){
            canvas.drawText("※ 画面をﾀｯﾁするとコインを自動で投げます。",(getWidth
()/2-240)+50,(getHeight()/2-343)+90, paint);          //作者・作成年月の表示
             canvas.drawText("※ もう一度画面をﾀｯﾁすると止まります。", (getWidth()
/2-240)+50,(getHeight()/2-343)+120, paint);          //作者・作成年月の表示
             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)+150, paint);          //作者・作成年月の表示
             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)+180, paint);          //作者・作成年月の表示

             canvas.drawText("※ 格子点と重なったコインを赤い円で、重な",(getWidth()
/2-240)+50,(getHeight()/2-343)+425, paint);          //作者・作成年月の表示
             canvas.drawText("　らなかったコインを青い円で表します。", (getWidth()
/2-240)+50,(getHeight()/2-343)+445, paint);          //作者・作成年月の表示
             canvas.drawText("※（格子点と重なったコインの個数)を(投げた",(getWidth
()/2-240)+50,(getHeight()/2-343)+475, paint);          //作者・作成年月の表示
```

```
            canvas.drawText("　コインの総数)で割った値の４倍が円周率π", (getWidth
()/2-240)+50, (getHeight()/2-343)+495, paint);                    //作者・作成年月の表示
             canvas.drawText("　になっていることを観察してみよう。", (getWidth()
/2-240)+50, (getHeight()/2-343)+515, paint);                    //作者・作成年月の表示
        }

        paint.setColor(Color.BLUE);
          canvas.drawText( "Copyright( C)　K.Niwa　2019.11", (getWidth() /2-240) +110,
(getHeight()/2-343)+660, paint);                    //作者・作成年月の表示

        k=k+1;
        //投げたコインの個数を 1 個増やす
        x[k]=40+400*Math.random();
        //落ちるコインの位置（ｘ，ｙ）を乱数で求める
        y[k]=100+400*Math.random();

        if ((x[k]-40)*(x[k]-40)+(y[k]-100)*(y[k]-100)<2500) {                    //コイン
が格子点に重なった場合
            sum++;                                                                    //格子点
に重なったコインの個数を１つ増やす
        }
        else if ((x[k]-140)*(x[k]-140)+(y[k]-100)*(y[k]-100)<2500) {
            sum++;
        }
        else if ((x[k]-240)*(x[k]-240)+(y[k]-100)*(y[k]-100)<2500) {
            sum++;
        }
        else if ((x[k]-340)*(x[k]-340)+(y[k]-100)*(y[k]-100)<2500) {
            sum++;
        }
        else if ((x[k]-440)*(x[k]-440)+(y[k]-100)*(y[k]-100)<=2500) {
            sum++;
        }

        else if ((x[k]-40)*(x[k]-40)+(y[k]-200)*(y[k]-200)<2500) {
            sum++;
        }
        else if ((x[k]-140)*(x[k]-140)+(y[k]-200)*(y[k]-200)<2500) {
            sum++;
        }
```

```
else if ((x[k]-240)*(x[k]-240)+(y[k]-200)*(y[k]-200)<2500) {
    sum++;
}
else if ((x[k]-340)*(x[k]-340)+(y[k]-200)*(y[k]-200)<2500) {
    sum++;
}
else if ((x[k]-440)*(x[k]-440)+(y[k]-200)*(y[k]-200)<2500) {
    sum++;
}

else if ((x[k]-40)*(x[k]-40)+(y[k]-300)*(y[k]-300)<2500) {
    sum++;
}
else if ((x[k]-140)*(x[k]-140)+(y[k]-300)*(y[k]-300)<2500) {
    sum++;
}
else if ((x[k]-240)*(x[k]-240)+(y[k]-300)*(y[k]-300)<2500) {
    sum++;
}
else if ((x[k]-340)*(x[k]-340)+(y[k]-300)*(y[k]-300)<2500) {
    sum++;
}
else if ((x[k]-440)*(x[k]-440)+(y[k]-300)*(y[k]-300)<2500) {
    sum++;
}

else if ((x[k]-40)*(x[k]-40)+(y[k]-400)*(y[k]-400)<2500) {
    sum++;
}
else if ((x[k]-140)*(x[k]-140)+(y[k]-400)*(y[k]-400)<2500) {
    sum++;
}
else if ((x[k]-240)*(x[k]-240)+(y[k]-400)*(y[k]-400)<2500) {
    sum++;
}
else if ((x[k]-340)*(x[k]-340)+(y[k]-400)*(y[k]-400)<2500) {
    sum++;
}
else if ((x[k]-440)*(x[k]-440)+(y[k]-400)*(y[k]-400)<2500) {
    sum++;
```

```
        }

        else if ((x[k]-40)*(x[k]-40)+(y[k]-500)*(y[k]-500)<2500) {
            sum++;
        }
        else if ((x[k]-140)*(x[k]-140)+(y[k]-500)*(y[k]-500)<2500) {
            sum++;
        }
        else if ((x[k]-240)*(x[k]-240)+(y[k]-500)*(y[k]-500)<2500) {
            sum++;
        }
        else if ((x[k]-340)*(x[k]-340)+(y[k]-500)*(y[k]-500)<2500) {
            sum++;
        }
        else if ((x[k]-440)*(x[k]-440)+(y[k]-500)*(y[k]-500)<2500) {
            sum++;
        }

        for (i=1;i<=k;i++) {
            if ((x[i]-40)*(x[i]-40)+(y[i]-100)*(y[i]-100)<2500) {
                //コインが格子点に重なった場合
                px=(int)x[i]+(getWidth()/2-240);
        //倍精度型変数を整数型変数にキャストする
                py=(int)y[i]+(getHeight()/2-343);
                paint.setColor(Color.RED);                          //コイン
の色を赤にする
                paint.setStyle(Paint.Style.STROKE);
                canvas.drawCircle(px, py,50,paint);                 //格子点に重なっ
たコインを描く
            }
            else if ((x[i]-140)*(x[i]-140)+(y[i]-100)*(y[i]-100)<2500) {
                px=(int)x[i]+(getWidth()/2-240);
        //
                py=(int)y[i]+(getHeight()/2-343);
                paint.setColor(Color.RED);                          //
                paint.setStyle(Paint.Style.STROKE);
                canvas.drawCircle(px, py,50,paint);                 //
            }
            else if ((x[i]-240)*(x[i]-240)+(y[i]-100)*(y[i]-100)<2500) {
                        //
```

```
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if((x[i]-340)*(x[i]-340)+(y[i]-100)*(y[i]-100)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if((x[i]-440)*(x[i]-440)+(y[i]-100)*(y[i]-100)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }

        else if((x[i]-40)*(x[i]-40)+(y[i]-200)*(y[i]-200)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if((x[i]-140)*(x[i]-140)+(y[i]-200)*(y[i]-200)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
```

```
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-240)*(x[i]-240)+(y[i]-200)*(y[i]-200)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-340)*(x[i]-340)+(y[i]-200)*(y[i]-200)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-440)*(x[i]-440)+(y[i]-200)*(y[i]-200)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }

        else if ((x[i]-40)*(x[i]-40)+(y[i]-300)*(y[i]-300)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
```

```
        }
        else if ((x[i]-140)*(x[i]-140)+(y[i]-300)*(y[i]-300)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-240)*(x[i]-240)+(y[i]-300)*(y[i]-300)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-340)*(x[i]-340)+(y[i]-300)*(y[i]-300)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-440)*(x[i]-440)+(y[i]-300)*(y[i]-300)<2500)  {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }

        else if ((x[i]-40)*(x[i]-40)+(y[i]-400)*(y[i]-400)<2500)  {
                        //
```

```
            px=(int) x[i]+(getWidth()/2-240);
//
            py=(int) y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if ((x[i]-140)*(x[i]-140)+(y[i]-400)*(y[i]-400)<2500)  {
                        //
            px=(int) x[i]+(getWidth()/2-240);
//
            py=(int) y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if ((x[i]-240)*(x[i]-240)+(y[i]-400)*(y[i]-400)<2500)  {
                        //
            px=(int) x[i]+(getWidth()/2-240);
//
            py=(int) y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if ((x[i]-340)*(x[i]-340)+(y[i]-400)*(y[i]-400)<2500)  {
                        //
            px=(int) x[i]+(getWidth()/2-240);
//
            py=(int) y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                           //
        }
        else if ((x[i]-440)*(x[i]-440)+(y[i]-400)*(y[i]-400)<2500)  {
                        //
            px=(int) x[i]+(getWidth()/2-240);
//
            py=(int) y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                    //
```

```
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                    //
        }

        else if((x[i]-40)*(x[i]-40)+(y[i]-400)*(y[i]-400)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                    //
        }
        else if((x[i]-140)*(x[i]-140)+(y[i]-400)*(y[i]-400)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                    //
        }
        else if((x[i]-240)*(x[i]-240)+(y[i]-400)*(y[i]-400)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                    //
        }
        else if((x[i]-340)*(x[i]-340)+(y[i]-400)*(y[i]-400)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                    //
        }
```

```
        else if ((x[i]-440)*(x[i]-440)+(y[i]-400)*(y[i]-400)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }

        else if ((x[i]-40)*(x[i]-40)+(y[i]-500)*(y[i]-500)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-140)*(x[i]-140)+(y[i]-500)*(y[i]-500)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-240)*(x[i]-240)+(y[i]-500)*(y[i]-500)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
//
            py=(int)y[i]+(getHeight()/2-343);
            paint.setColor(Color.RED);                                          //
            paint.setStyle(Paint.Style.STROKE);
            canvas.drawCircle(px, py,50,paint);                         //
        }
        else if ((x[i]-340)*(x[i]-340)+(y[i]-500)*(y[i]-500)<2500) {
                        //
            px=(int)x[i]+(getWidth()/2-240);
```

```
        //
                py=(int)y[i]+(getHeight()/2-343);
                paint.setColor(Color.RED);                                    //
                paint.setStyle(Paint.Style.STROKE);
                canvas.drawCircle(px, py,50,paint);                    //
            }
            else if ((x[i]-440)*(x[i]-440)+(y[i]-500)*(y[i]-500)<2500) {
                        //
                px=(int)x[i]+(getWidth()/2-240);
        //
                py=(int)y[i]+(getHeight()/2-343);
                paint.setColor(Color.RED);                                    //
                paint.setStyle(Paint.Style.STROKE);
                canvas.drawCircle(px, py,50,paint);                    //
            }
            else {
                        //コインが格子点に重ならなかった場合
                px=(int)x[i]+(getWidth()/2-240);                    //落ちたコインの
位置 x 座標
                py=(int)y[i]+(getHeight()/2-343);                    //落ちたコインの
位置 y 座標
                paint.setColor(Color.BLUE);                                    //コイン
の色を青にする
                paint.setStyle(Paint.Style.STROKE);
                canvas.drawCircle(px, py,50,paint);                    //格子点に重なら
なかったコイン描く
            }
        }

        pai=(double)4*sum/k;                            //πの近似値の計算
        paint.setColor(Color.BLUE);
        paint.setTextSize(25.0f);
         canvas.drawText("円周率π＝"+pai, (getWidth()/2-240)+60, (getHeight()/2-343)
+570, paint);                    //テキストと数値を絵として描く
        paint.setTextSize(20.0f);
        paint.setColor(Color.BLACK);
         canvas.drawText("格子点と重なったコインの個数＝"+sum+"　個", (getWidth()
/2-240)+60, (getHeight()/2-343)+600, paint);                //テキストと数値を絵として描く
        paint.setTextSize(20.0f);
          canvas.drawText("投げたコインの総数＝"+k+"　個", (getWidth()/2-240)+60,
```

```
(getHeight()/2-343)+630, paint);                    //テキストと数値を絵として描く

        if (k<=N && flag==1) {    //投げる最大のコインの個数以下で、かつコインを投げる識別子が１の場合
            invalidate(); //再描画する（clear & goto onDraw）
        }

        if (k==N) {
            flag=2;         //コインを投げる識別子が２（初期化）の場合
        }

    }//protected void onDraw(Canvas canvas)

        //画面にタッチしたときのイベント処理
--------------------------------------------------------------------------------
    @Override
    public boolean onTouchEvent(MotionEvent event) {

        flag=flag+1;        //flag に１を加える
        flag=flag % 3;      //flag に１、２、０を代入する

        if (flag==0){       //コインを投げる識別子が０（停止）の場合
            sum=0;                  //格子点と重なったコインの個数を０にする
            k=0;            //投げたコインの個数を０にする
        }

        invalidate();                                   //再描画する（clear & goto onDraw）
        return false;

    }
}
```

［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.mytenencoin.MyTenencoin
        android:id="@+id/myfview1"
        android:layout_height="match_parent"
        android:layout_width="match_parent"/>

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

［３］MainActivity.java

```
/*
-----------------------------------------------------------------------
    １０円玉を投げて円周率πを求める
          Android 4.4（Kit Kat）
     Copyright(C) K.Niwa 2019.12.7
-----------------------------------------------------------------------
*/

package jp.kiyo.wuena.mytenencoin;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
```

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