5 ロケットの移動を繰り返す



- 5.1 ソースコード
- (1) テンプレート Stepの①の箇所に Step050View を入力してください。
- (2) 次のアプリケーションを新規作成してください。

Step040Viewをコピー&ペーストして、ファイル名を「Step050View」に変更してください。

プロジェクト名:StepPro????(年組席)

アプリケーション名 : Step050View

```
/*
     年
           組
                席 名前
* Step050View
      ロケットの移動を繰り返す
*
*/
public class Step050View extends SurfaceView
                                     implements SurfaceHolder.Callback
{
    . . .
   //コンストラクタ
   public Step050View(Context context)
   {
        . . .
   }
   //サーフェイスの生成
   public void surfaceCreated(SurfaceHolder holder)
   {
       executor.scheduleAtFixedRate(
           new Runnable()
           {
              public void run()
               ł
                  draw(canvas);
                  py--://py=py-1と同じ
                  px-=2;//px=px-2 と同じ
                  //斜め移動 端に着いたら反対側から出てくる
                  if(px<0) px=getWidth();
                  if(px>getWidth()) px=0;
                  if(py<0) py=getHeight();
                  if(py>getHeight()) py=0;
```

}, 0, 1, TimeUnit. MILLISECONDS);

# 5.2 実行結果

ロケットの進行方向に移動して、左端についたら右端からロケットが出て、下につ いたら上からロケットが出てきます。

# 5.3 解説

draw(canvas); py--; px-=2;//① //斜め移動 端に着いたら反対側から出てくる if(px < 0) px = getWidth(); //② if(px > getWidth()) px = 0;//③ if(py < 0) py = getHeight();//④ if(py > getHeight()) py = 0;//⑤

①px-=2

px=px-2 のことであり、x 座標を現在の座標より左へ2つ移動させる命令です。 つまり、py を-1 することにより、画像を左斜めに移動させます。

(2)if(px < 0) px = getWidth();

画像が左端に到達したら、px に getWidth メソッド(480 ドット)で取得した画面の横幅を代入します。そうすることによって画像が右端から表示されます。

(3)if(px > getWidth()) px = 0;

画像が右端に到達したら px に 0 を代入します。そうすることによって画像が 左端から表示されます。((1) px+=2 の場合に有効)

(4) if (py < 0) py = getHeight();

画像が上端に到達したら、py に getHeight メソッド(854 ドット)で取得した画面の縦幅を代入します。そうすることによって画像が下端から表示されます。

(5)if(py > getHeight()) py = 0;

画像が下端に到達したら py に 0 を代入します。そうすることによって画像が 上端から表示されます。((1) px+=2 の場合に有効) ※③と⑤は、①が px+=2 の場合に有効です。(右斜めに移動する)





### 5.4 演習

# 【演習51】

「宇宙船」の移動を繰り返すプログラムを作成してください。 プロジェクト名:StepPro????(年組席) アプリケーション名:Step051View 画像「r0.png」

# 【演習52】

「ようこそ Java へ」の文字の移動を繰り返すプログラムを作成してください。

プロジェクト名:StepPro????(年組席) アプリケーション名:Step052View

#### 【演習53】

「ようこそ Java へ」の文字が上下にゆっくり移動するプログラムを作成してください。

プロジェクト名: StepPro????(年組席) アプリケーション名: Step053View
 ■方法1 変数 flg に方向情報を設定する。



■方法2 スピードの演算子を反転させる。

private int speed=5;//■ロケットの速度

. . .

draw(canvas);

py-=speed;

if(py<TSIZE/2 || py>getHeight()) {speed=-speed;}//■演算子逆転

### 【演習54】

「ロケット」が左右にゆっくり移動するプログラムを作成してください。その際、ロ ケットが進行方向に向くようにしてください。

プロジェクト名:StepPro????(年組席) アプリケーション名:Step054View

画像「rocket.png」

### 【演習55】

「ようこそ Java へ」の文字が上下にゆっくり移動し、移動する際に文字の色が変化 するプログラムを作成してください。

プロジェクト名:StepPro????(年組席) アプリケーション名:Step055View

例 上昇する時は、黒色の文字で、下降するときは、赤色の文字に変化する。

ヒント: 10ミリ秒待機する。

#### 【演習56】 相互評価の対象課題

ロケット等の画像や「ようこそ Java へ」等の文字を使って、画面上をユニークな動きと色の変化をするプログラムを作成してください。なお、背景画像も工夫してください。

プロジェクト名:StepPro????(????:年組席)

アプリケーション名:Step056View????(????:年組席)

#### 【演習57】 相互評価対象の演習

<u>ロケット、うさぎ等の画像やテーマとなる文字</u>が、画面上をユニークな動きで動き回 り、背景や文字の色を変化させるプログラムを作成してください。なお、背景画像も 工夫してください。

※Step050Viewをコピー&ペーストして、ファイル名を「Step057View????」に変更 してください。

#### ■仕様

(1) プロジェクト名、アプリケーション名は次のとおりです。

プロジェクト名 : StepPro???? (????: 年組席)

アプリケーション名: Step057View???? (????: 年組席)

(2)動かす画像及び背景画像を作成して、「res」フォルダの「drawable-nodpi」フォル ダに保存してください。

動く画像のサイズ:最大 64ドット×64ドット 以下

背景画像のサイズ:480ドット×854ドット

ファイルの種類はGIF、PNGにしてください。

なお、透過GIFすると余分な部分が背景色になります。使用ソフトは「ウェブアートデザイナ」が使えます。

- (3) テーマとなる文字をプログラムに入れてください。なお、そのテーマに合った画 面構成になるようにしてください。
- (4) 文字色や背景色を、動かす動画に合わせて変化させてください。
- (5)動かす画像、文字、背景を変化させてもOKです。
- (6) ソースリストを提出してください。
- (7)評価項目(各10点)

①全体のデザイン性(色調等)
 ②画像(動かす画像、背景画像)のこだわり

③動きのこだわり
 ④組立てのこだわり(テーマ性)
 テーマとなる文字と画像の動きの親和性
 ⑤技術レベル

■文字の高さの取得

ポイントとなるのは下記の4つの座標でいずれもベースラインからの位置を示しています。

- FontMetrics.top
- FontMetrics.ascent
- FontMetrics. descent
- FontMetrics.bottom

図にすると下記のようなイメージになります。



●がテキストを描画する際に指定した座標です。

・中心線上に文字列を描画



FontMetricsの基本でみたように、指定した位置に文字を描画しようとしても指定 した座標はベースラインとして使用されるため、思っていたよりも上の位置に文字列 が描画されてしまいます。このような場合に Ascent, Descent から座標を計算して調 整する事で思った位置に文字列を描画する事ができます。

```
// 中心座標
float centerX = getWidth() / 2;
float centerY = getHeight() / 2;
// 中心線の描画(この中心に来るように文字列を描画したい)
Paint centerLinePaint = new Paint( Paint.ANTI_ALIAS_FLAG);>
centerLinePaint.setColor(Color.YELLOW);
canvas.drawLine(0, centerY, getWidth(), centerY, centerLinePaint);
canvas.drawLine(centerX, 0, centerX, getHeight(), centerLinePaint);
// 描画する文字列
String text = "hijkl";
// テキスト用ペイントの生成
Paint textPaint = new Paint( Paint.ANTI_ALIAS_FLAG);
textPaint.setTextSize( 35);
textPaint.setColor(Color.WHITE);
FontMetrics fontMetrics = textPaint.getFontMetrics();
// 文字列の幅を取得
float textWidth = textPaint.measureText( text);
// 中心にしたい X 座標から文字列の幅の半分を引く
float baseX = centerX - textWidth / 2;
// 中心にしたい Y 座標から Ascent と Descent の半分を引く
float baseY = centerY - (fontMetrics.ascent + fontMetrics.descent) / 2;
// テキストの描画
canvas.drawText( text, baseX, baseY, textPaint);
```