**Одновременное вращение и перемещение bitmap на canvas**

**package** tva.knastu.com.igra001;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
**import** android.graphics.Canvas;  
**import** android.graphics.Matrix;  
**import** android.view.View;  
  
*/\*\*  
 \* Created by Vladimir on 27.10.2016.  
 \*/***public class** Board **extends** View {  
 **private** Bitmap **mBallBitmap**;  
 **private int mPosX**, **mPosY**;  
 **private** Matrix **m**;  
 Matrix **mb**;  
 Matrix **mc**;  
 **float alfa**=0;  
  
  
  
 **public** Board(Context context) {  
 **super**(context);  
 **mBallBitmap** = BitmapFactory.*decodeResource*(getResources(), R.drawable.***boll2***);  
 **mPosX** = **mPosY** = 0;  
 **m** = **new** Matrix();  
 **mb** = **new** Matrix();  
 **mc** = **new** Matrix();  
 }  
  
 @Override  
 **protected void** onDraw(Canvas canvas) {  
 **super**.onDraw(canvas);  
 **mPosX** += 2;*//координата Х объекта* **alfa** +=1; *//угол поворота объекта*

*/\**

*//матрица вращения* **mb**.setRotate(**alfa**, **mBallBitmap**.getWidth()/2, **mBallBitmap**.getHeight()/2); *//матрица перемещения* **mc**.setTranslate(**mPosX**, canvas.getHeight() / 2 - **mBallBitmap**.getHeight() / 2);  
*//Объединение матриц* **m**.setConcat(**mc**, **mb**); *//Отображение объекта* canvas.drawBitmap(**mBallBitmap**, **m**, **null**);  
*\*/*

**m**.setRotate(**alfa**, **mBallBitmap**.getWidth()/2, **mBallBitmap**.getHeight()/2);  
**m**.postTranslate(**mPosX**, mPosY);  
canvas.drawBitmap(**mBallBitmap**, **m**, **null**);

*// или*

*//* **m**.setRotate(**alfa**, **mBallBitmap**.getWidth()/2, **mBallBitmap**.getHeight()/2);  
 // **m**.postTranslate(**mPosX**, **mPosY**);  
 // canvas.drawBitmap(**mBallBitmap**, **m**, **null**);

invalidate();  
 }  
}

**protected void** onDraw(Canvas canvas) {  
 **super**.onDraw(canvas);  
 **mPosX** += 2;*//координата Х объекта***if**(**mPosX**>canvas.getWidth()+ **mBallBitmap**.getWidth()/2) **mPosX** = -**mBallBitmap**.getWidth()/2;  
 **alfa** +=2; *//угол поворота объекта* **if**(**alfa**==360) **alfa** = 0;

*//матрица вращения* **m**.setRotate(**alfa**, **mBallBitmap**.getWidth()/2, **mBallBitmap**.getHeight()/2);  
*//матрица перемещения* **m**.postTranslate(**mPosX**, canvas.getHeight() / 2 - **mBallBitmap**.getHeight() / 2);

*//Отображение объекта* canvas.drawBitmap(**mBallBitmap**, **m**, **null**);

invalidate();  
 }