Построение игрового поля с масштабированием под экран

*//Построение игровой зоны с масштабированием под экран***package** tva.knastu.com.igra\_002;  
  
**import** android.app.Activity;  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
**import** android.graphics.Canvas;  
**import** android.graphics.Color;  
**import** android.graphics.Matrix;  
**import** android.graphics.Paint;  
**import** android.graphics.Rect;  
**import** android.os.Bundle;  
**import** android.view.View;  
  
**public class** MainActivity **extends** Activity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(**new** DrawView(**this**));  
 }  
  
  
 **class** DrawView **extends** View {  
 **private float mPosX**, **mPosY**; *// координаты клубка* Paint **p**;  
 Rect **rect**;  
 **float sclX**, **sclY**;  
 **private** Matrix **m**;  
 **private** Bitmap[] **kubik** = **new** Bitmap[8];  
 **private int**[][] **layer** = {  
 { 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 },  
 { 1, 5, 1, 3, 5, 1, 1, 2, 1, 1, 1, 5 },  
 { 0, 0, 0, 1, 2, 1, 1, 2, 0, 0, 0, 4 },  
 { 0, 1, 3, 0, 2, 0, 0, 2, 0, 0, 0, 3 },  
 { 0, 5, 0, 0, 1, 1, 1, 2, 0, 1, 0, 1 },  
 { 6, 0, 1, 3, 4, 0, 5, 2, 0, 0, 0, 1 },  
 { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1},  
 };  
  
  
  
  
  
 **public** DrawView(Context context) {  
 **super**(context);  
 **kubik**[0] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***background***);  
 **kubik**[1] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***wall***);  
 **kubik**[2] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***ladder***);  
 **kubik**[3] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***bonus1***);  
 **kubik**[4] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***bonus2***);  
 **kubik**[5] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***enemy***);  
 **kubik**[6] = BitmapFactory.*decodeResource*(getResources(), R.drawable.***me***);  
  
 **mPosY** = 0;  
 **mPosX** = 0;  
  
 **m** = **new** Matrix();  
 **p** = **new** Paint();  
 **rect** = **new** Rect();  
  
  
 }  
  
 @Override  
 **protected void** onDraw(Canvas canvas) {  
 *// заливка канвы цветом* canvas.drawARGB(80, 102, 204, 255);  
 **sclX** = (**float**)((**float**)canvas.getWidth()/(**float**)(12\***kubik**[1].getWidth() ));  
 **sclY** = (**float**)((**float**)canvas.getHeight()/(**float**)(7\***kubik**[1].getHeight() ));  
**for** (**int** i = 0; i<7; i++) {  
 **for** (**int** j = 0; j < 12; j++) {  
  
 **m**.reset();  
 **m**.setScale(**sclX**, **sclY**, **mPosX**, **mPosY**);  
 **m**.postTranslate(**mPosX**\***sclX**, **mPosY**\***sclY**);  
 **if** (**layer**[i][j]>0) canvas.drawBitmap(**kubik**[**layer**[i][j]], **m**, **null**);  
 **mPosX** += (**float**)**kubik**[1].getWidth()\***sclX**;  
 }  
  
 **mPosY** += (**float**)**kubik**[1].getHeight()\***sclY**;  
 **mPosX** = 0;  
}}  
  
 }  
  
}

