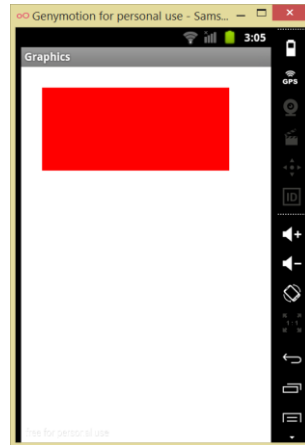


BÀI TẬP ANDROID 5

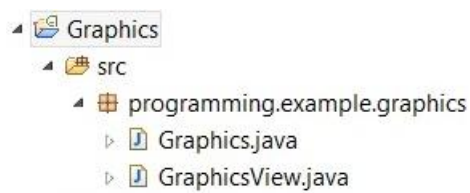
Bài tập 1:

Xây dựng ứng dụng sử dụng giao diện đồ họa



Hướng dẫn:

-Tạo 2 lớp theo cấu trúc



- Lớp Graphics kế thừa từ Activity, trong hàm onCreate sẽ hiển thị một View (lớp GraphicsView trong GraphicsView.java)

```
package programming.example.graphics;

import android.app.Activity;
import android.os.Bundle;

public class Graphics extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new GraphicsView(this));
    }
}
```

-Lớp GraphicsView kế thừa từ View

và thực hiện các thao tác vẽ

```
import android.content.Context;
import android.graphics.Canvas;
import android.view.View;

public class GraphicsView extends View {

    public GraphicsView(Context context) {
        super(context);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        // Drawing commands go here

        invalidate();
    }
}
```

-Run chương trình, sẽ hiển thị màn hình màu trắng của canvas.



-Các thao tác vẽ sẽ thực hiện trong hàm onDraw

```

public class GraphicsView extends View {

    public GraphicsView(Context context) {
        super(context);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        // Drawing commands go here
        Rect r=new Rect(40,40,400,200);
        Paint paint=new Paint();
        paint.setStyle(Paint.Style.FILL);
        paint.setColor(Color.RED);
        canvas.drawRect(r, paint);

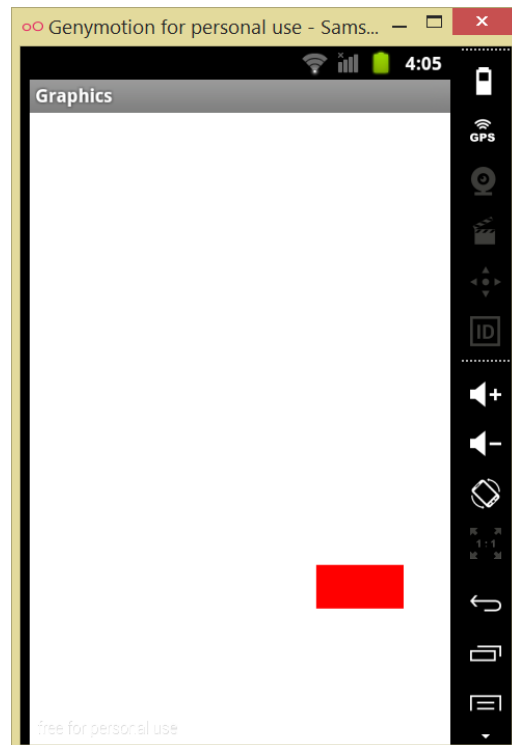
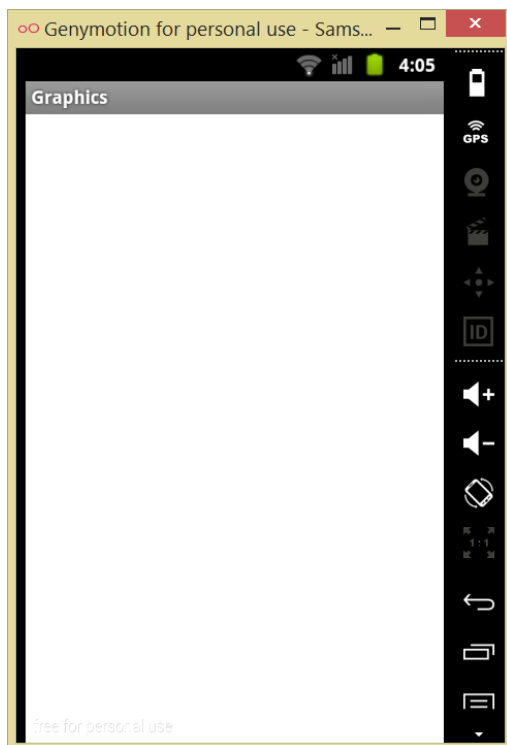
        invalidate();
    }
}

```

-Chạy chương trình và xem kết quả

Bài 2:

Viết chương trình xử lý sự kiện Touch trên canvas. Khi người dùng touch vào 1 vị trí trên màn hình, sẽ hiển thị hình chữ nhật ở vị trí đó.



Hướng dẫn:

-Thêm các thuộc tính vị trí x,y, chiều dài và chiều rộng của hình chữ nhật

```
public class GraphicsView extends View {
    int x=0;
    int y=0;
    int d=100;
    int r=50;
```

-Sửa hàm onDraw, thực hiện vẽ khi người dùng đã touch vào vị trí x,y trên màn hình

```
@Override
protected void onDraw(Canvas canvas) {
    // Drawing commands go here
    if(x!=0&&y!=0){
        int right=x+d;
        int bottom=y+r;
        Rect r=new Rect(x,y,right,bottom);
        Paint paint=new Paint();
        paint.setStyle(Paint.Style.FILL);
        paint.setColor(Color.RED);
        canvas.drawRect(r, paint);
    }
    invalidate();
}
```

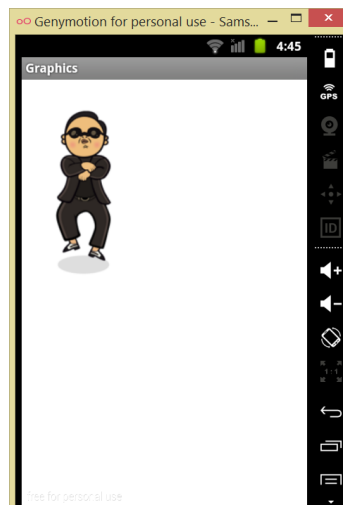
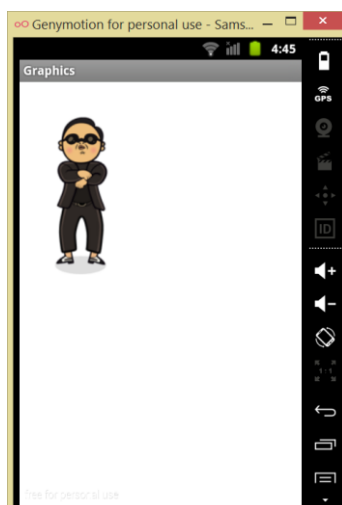
-Thêm hàm xử lý sự kiện touch trên view

```
@Override
public boolean onTouchEvent(MotionEvent event) {
    // TODO Auto-generated method stub
    x=(int)event.getX();
    y=(int)event.getY();
    return super.onTouchEvent(event);
}
```

-Chạy và xem kết quả

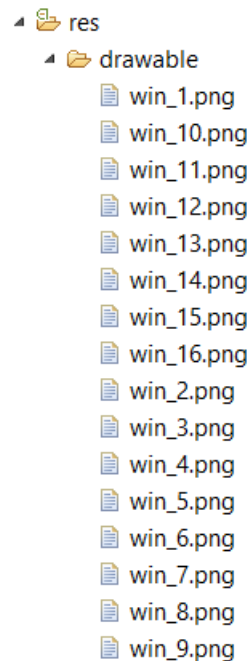
Bài 3:

Viết ứng dụng hiển thị một bitmap, khi touch màn hình, sẽ chuyển bitmap chuyển động tiếp theo như hình.



Hướng dẫn:

-Chép các file hình ảnh vào thư mục res/drawable



-Thêm thuộc tính gồm một mảng các bitmap và biến đếm i, khởi tạo các bitmap trong constructor.

```
public class GraphicsView extends View {  
    Bitmap[] frames = new Bitmap[16]; //16 frames  
    int i=0;  
    public GraphicsView(Context context) {  
        super(context);  
        frames[0] = BitmapFactory.decodeResource(getResources(), R.drawable.win_1);  
        frames[1] = BitmapFactory.decodeResource(getResources(), R.drawable.win_2);  
        frames[2] = BitmapFactory.decodeResource(getResources(), R.drawable.win_3);  
        frames[3] = BitmapFactory.decodeResource(getResources(), R.drawable.win_4);  
        frames[4] = BitmapFactory.decodeResource(getResources(), R.drawable.win_5);  
        frames[5] = BitmapFactory.decodeResource(getResources(), R.drawable.win_6);  
        frames[6] = BitmapFactory.decodeResource(getResources(), R.drawable.win_7);  
        frames[7] = BitmapFactory.decodeResource(getResources(), R.drawable.win_8);  
        frames[8] = BitmapFactory.decodeResource(getResources(), R.drawable.win_9);  
        frames[9] = BitmapFactory.decodeResource(getResources(), R.drawable.win_10);  
        frames[10] = BitmapFactory.decodeResource(getResources(), R.drawable.win_11);  
        frames[11] = BitmapFactory.decodeResource(getResources(), R.drawable.win_12);  
        frames[12] = BitmapFactory.decodeResource(getResources(), R.drawable.win_13);  
        frames[13] = BitmapFactory.decodeResource(getResources(), R.drawable.win_14);  
        frames[14] = BitmapFactory.decodeResource(getResources(), R.drawable.win_15);  
        frames[15] = BitmapFactory.decodeResource(getResources(), R.drawable.win_16);  
    }  
}
```

-Sửa lại hàm onDraw và onTouchEvent

```

@Override
protected void onDraw(Canvas canvas) {
    // Drawing commands go here
    if(i<16){
        canvas.drawBitmap(frames[i],40,40,new Paint());
    }
    else{
        i=0;
    }
    invalidate();
}
@Override
public boolean onTouchEvent(MotionEvent event) {
    // TODO Auto-generated method stub
    i++;
    return true;
}

```

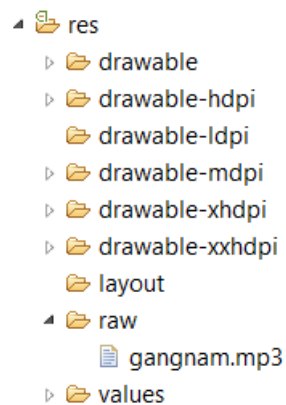
-Chạy chương trình và xem kết quả.

Bài 4:

Viết lại ứng dụng của bài 3, đối tượng tự động chuyển động kèm theo play file nhạc gangnam style

Hướng dẫn:

-Thêm file nhạc vào project



-Thêm các thuộc tính vào lớp View

```

public class GraphicsView extends View {
    Bitmap[] frames = new Bitmap[16]; //16 frames

    int i=0;
    long last_tick=0;
    long period=200;
    Context ctext;

    MediaPlayer mPlayer;

    public GraphicsView(Context context) {
        super(context);
        ctext=context;
        frames[0] = BitmapFactory.decodeResource(getResources(), R.drawable.win_1);
        frames[1] = BitmapFactory.decodeResource(getResources(), R.drawable.win_2);
        frames[2] = BitmapFactory.decodeResource(getResources(), R.drawable.win_3);
        frames[3] = BitmapFactory.decodeResource(getResources(), R.drawable.win_4);
    }
}

```

-Play nhạc từ resource (gọi trong onCreate)

```

mPlayer=MediaPlayer.create(ctext,R.raw.gangnam);
mPlayer.start();

```

-Viết lại onDraw

```

protected void onDraw(Canvas canvas) {
    // Drawing commands go here
    if(i<16){
        long time = (System.currentTimeMillis() - last_tick);
        if (time >= period) //the delay time has passed. set next frame
        {
            last_tick = System.currentTimeMillis();
            canvas.drawBitmap(frames[i],40,40,new Paint());
            i++;
            // Again call onDraw method
            postInvalidate();
        }
        else //still within delay. redraw current frame
        {
            canvas.drawBitmap(frames[i],40,40,new Paint());
            // Again call onDraw method
            postInvalidate();
        }
    }
    else{
        i=0;
        postInvalidate();
    }
}
}

```

-Lưu ý, khi kết thúc ứng dụng release đối tượng MediaPlayer.