**import** android.content.ContentValues;  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.Canvas;  
**import** android.graphics.Color;  
**import** android.graphics.Paint;  
**import** android.graphics.Path;  
**import** android.os.Environment;  
**import** android.provider.MediaStore;  
**import** android.util.AttributeSet;  
**import** android.view.MotionEvent;  
**import** android.view.View;  
**import** android.widget.Toast;  
  
**import** java.io.File;  
**import** java.io.FileOutputStream;  
**import** java.io.OutputStream;  
**import** java.util.ArrayList;  
**import** java.util.Random;  
  
**import static** android.graphics.Color.***BLACK***;  
  
*/\*\*  
 \* Created by M on 2/13/2017.  
 \*/***public class** CanvasView **extends** View {  
  
 **public int width**, **colorIndex** =0;  
 **public int height**;  
 **private** Bitmap **mBitmap**;  
 **private** Canvas **mCanvas**;  
 **private** Path **mPath**;  
 Context **context**;  
 **private** Paint **mPaint**;  
 **private float mX**, **mY**;  
 **private static final float *TOLERANCE*** = 5;  
 **private** ArrayList<PaintPath> **storedPaths** = **new** ArrayList<>();  
 **private boolean colorSwitchOn** = **false**, **sizeSwitchOn** = **false**;  
  
 **public void** Switch(){  
 **if**(**colorSwitchOn**){  
 Random rand = **new** Random();  
 **int** rnd = rand.nextInt(8);  
 *//black,red,white,argb(255, 247, 148, 29),argb(255, 255, 242, 0),argb(255, 57, 181, 74), argb(255, 0, 174, 239),argb(255, 133, 96, 168)* **switch** (rnd){  
 **case** 0:  
 **mPaint**.setColor(Color.***WHITE***);  
 **break**;  
 **case** 1:  
 **mPaint**.setColor(Color.***BLACK***);  
 **break**;  
 **case** 2:  
 **mPaint**.setColor(Color.***RED***);  
 **break**;  
 **case** 3:  
 **mPaint**.setARGB(255, 247, 148, 29);  
 **break**;  
 **case** 4:  
 **mPaint**.setARGB(255, 255, 242, 0);  
 **break**;  
 **case** 5:  
 **mPaint**.setARGB(255, 57, 181, 74);  
 **break**;  
 **case** 6:  
 **mPaint**.setARGB(255, 0, 174, 239);  
 **break**;  
 **case** 7:  
 **mPaint**.setARGB(255, 133, 96, 168);  
 **break**;  
  
 }  
 *// 4f,7f,10f,13f,16f  
// mPaint* }  
 **if**(**sizeSwitchOn**){  
 Random rand = **new** Random();  
 **int** rnd = rand.nextInt(5);  
 **switch** (rnd) {  
 **case** 0:  
 **mPaint**.setStrokeWidth(4f);  
 **break**;  
 **case** 1:  
 **mPaint**.setStrokeWidth(7f);  
 **break**;  
 **case** 2:  
 **mPaint**.setStrokeWidth(10f);  
 **break**;  
 **case** 3:  
 **mPaint**.setStrokeWidth(13f);  
 **break**;  
 **case** 4:  
 **mPaint**.setStrokeWidth(16f);  
 **break**;  
 }  
 }  
 }  
 **public boolean** colorSwitch(**int** bttnClick){  
 **if**(bttnClick%2!=0){  
 **colorSwitchOn** = **true**;  
 }  
 **else if**(bttnClick%2==0){  
 **colorSwitchOn** = **false**;  
 }  
 **return colorSwitchOn**;  
 }  
 **public boolean** sizeSwitch(**int** bttnClick){  
 **if**(bttnClick%2!=0){  
 **sizeSwitchOn** = **true**;  
 }  
 **else if**(bttnClick%2==0){  
 **sizeSwitchOn** = **false**;  
 }  
 **return sizeSwitchOn**;  
 }  
 **class** PaintPath {  
 Path **path**;  
 Paint **paint**;  
 PaintPath(Path path, Paint paint) {  
 **this**.**path** = path;  
 **this**.**paint** = paint;  
 }  
 }  
  
  
 **public** CanvasView(Context context, AttributeSet attrs) {  
 **super**(context, attrs);  
 **mPath** = **new** Path();  
 **mPaint** = **new** Paint();  
 **mPaint**.setAntiAlias(**true**);  
 **mPaint**.setColor(***BLACK***);  
 **mPaint**.setStyle(Paint.Style.***STROKE***);  
 **mPaint**.setStrokeJoin(Paint.Join.***ROUND***);  
 **mPaint**.setStrokeWidth(4f);  
 }  
  
 @Override  
 **protected void** onSizeChanged(**int** w, **int** h, **int** oldw, **int** oldh) {  
 **super**.onSizeChanged(w, h, oldw, oldh);  
 *// your Canvas will draw onto the defined Bitmap* **mBitmap** = Bitmap.*createBitmap*(w, h, Bitmap.Config.***ARGB\_8888***);  
 **mCanvas** = **new** Canvas(**mBitmap**);  
 }  
  
 @Override  
 **protected void** onDraw(Canvas canvas) {  
 **super**.onDraw(canvas);  
 *// draw the mPath with the mPaint on the canvas when onDraw* **for** (PaintPath paintPath : **storedPaths**) {  
 canvas.drawPath(paintPath.**path**, paintPath.**paint**);  
 }  
 canvas.drawPath(**mPath**, **mPaint**);  
 }  
 **public void** change() {  
 **if** (**colorIndex**==0) {  
 **this**.setBackgroundColor(***BLACK***);  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==1) {  
 **this**.setBackgroundColor(Color.*argb*(255, 247, 148, 29));  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==2) {  
 **this**.setBackgroundColor(Color.*argb*(255, 255, 242, 0));  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==3) {  
 **this**.setBackgroundColor(Color.***RED***);  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==4) {  
 **this**.setBackgroundColor(Color.*argb*(255, 57, 181, 74));  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==5) {  
 **this**.setBackgroundColor(Color.*argb*(255, 0, 174, 239));  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==6) {  
 **this**.setBackgroundColor(Color.*argb*(255, 133, 96, 168));  
 **colorIndex**++;  
 }  
 **else if** (**colorIndex**==7) {  
 **this**.setBackgroundColor(Color.***WHITE***);  
 **colorIndex** = 0;  
 }  
 }  
 **public void** loadImageOntoCanvas(Bitmap bmp)  
 {  
 **mCanvas**.drawBitmap(bmp, 0, 0, **mPaint**);  
 }  
  
*// when ACTION\_DOWN start touch according to the x,y values* **private void** startTouch(**float** x, **float** y) {  
 **mPath** = **new** Path();  
 **mPath**.moveTo(x, y);  
 **mX** = x;  
 **mY** = y;  
 Switch();  
 }  
  
 *// when ACTION\_MOVE move touch according to the x,y values* **private void** moveTouch(**float** x, **float** y) {  
 **float** dx = Math.*abs*(x - **mX**);  
 **float** dy = Math.*abs*(y - **mY**);  
 **if** (dx >= ***TOLERANCE*** || dy >= ***TOLERANCE***) {  
 **mPath**.quadTo(**mX**, **mY**, (x + **mX**) / 2, (y + **mY**) / 2);  
 **mX** = x;  
 **mY** = y;  
 }  
 }  
  
 **public void** clearCanvas() {  
 **storedPaths**.clear();  
 **mPath**.reset();  
 invalidate();  
 startTouch(0, 0);  
 moveTouch(0, 0);  
 upTouch();  
 **this**.setBackgroundColor(Color.***WHITE***);  
 **colorIndex** = 0;  
 Switch();  
 }  
  
 **public void** drawCircle() {  
 **mPath**.addCircle(**mX**,**mY**,60, Path.Direction.***CCW***);  
 **mCanvas**.drawCircle(**mX**, **mY**, 60, **mPaint**);  
 **storedPaths**.add(**new** PaintPath(**mPath**, **new** Paint(){  
 {  
 **this**.setColor(**mPaint**.getColor());  
 **this**.setStyle(**mPaint**.getStyle());  
 **this**.setStrokeJoin(**mPaint**.getStrokeJoin());  
 **this**.setStrokeWidth(**mPaint**.getStrokeWidth());  
 Switch();  
 }  
 }));  
 invalidate();  
 }  
  
 **public void** drawRectangle() {  
 **mPath**.addRect(**mX**,**mY**, **mX**+50, **mY**+350, Path.Direction.***CCW***);  
*// 200 300 250 350* **mCanvas**.drawRect(**mX**,**mY**, **mX**+50, **mY**+350, **mPaint**);  
 **storedPaths**.add(**new** PaintPath(**mPath**, **new** Paint(){  
 {  
 **this**.setColor(**mPaint**.getColor());  
 **this**.setStyle(**mPaint**.getStyle());  
 **this**.setStrokeJoin(**mPaint**.getStrokeJoin());  
 **this**.setStrokeWidth(**mPaint**.getStrokeWidth());  
 Switch();  
 }  
 }));  
 invalidate();  
}  
 **public void** drawTriangle() {  
 **mPath**.moveTo(**mX**,**mY**);*//x,y* **float** tmpX = **mX**, tmpY = **mY**;  
 **mX** = **mX**+100;  
 **mY** = **mY**+200;  
 **mPath**.lineTo(**mX**,**mY**);  
 **mPath**.moveTo(**mX**,**mY**);*//x+100, y+200* **mX** = **mX**-200;  
 **mPath**.lineTo(**mX**,**mY**);*//x-100, y+200* **mPath**.moveTo(**mX**,**mY**);  
 **mX** = tmpX;  
 **mY** = tmpY;  
 **mPath**.lineTo(**mX**,**mY**);  
 **mCanvas**.drawPath(**mPath**, **mPaint**);  
 **storedPaths**.add(**new** PaintPath(**mPath**, **new** Paint(){  
 {  
 **this**.setColor(**mPaint**.getColor());  
 **this**.setStyle(**mPaint**.getStyle());  
 **this**.setStrokeJoin(**mPaint**.getStrokeJoin());  
 **this**.setStrokeWidth(**mPaint**.getStrokeWidth());  
 Switch();  
 }  
 }));  
 invalidate();  
 }  
 *// when ACTION\_UP stop touch* **private void** upTouch() {  
 **mPath**.lineTo(**mX**, **mY**);  
 **storedPaths**.add(**new** PaintPath(**mPath**, **new** Paint(){  
 {  
 **this**.setColor(**mPaint**.getColor());  
 **this**.setStyle(**mPaint**.getStyle());  
 **this**.setStrokeJoin(**mPaint**.getStrokeJoin());  
 **this**.setStrokeWidth(**mPaint**.getStrokeWidth());  
 Switch();  
 }  
 }));  
 }  
  
 *//override the onTouchEvent* @Override  
 **public boolean** onTouchEvent(MotionEvent event) {  
 **float** x = event.getX();  
 **float** y = event.getY();  
 **switch** (event.getAction()) {  
 **case** MotionEvent.***ACTION\_DOWN***:  
 startTouch(x, y);  
 invalidate();  
 Switch();  
 **break**;  
 **case** MotionEvent.***ACTION\_MOVE***:  
 moveTouch(x, y);  
 invalidate();  
 Switch();  
 **break**;  
 **case** MotionEvent.***ACTION\_UP***:  
 upTouch();  
 invalidate();  
 Switch();  
 **break**;  
 }  
 **return true**;  
 }  
  
 **public** Paint getmPaint() {  
 **return mPaint**;  
 }  
 **public void** save() {  
 **int**[] nums = {0,1,2,3,4,5,6,7,8,9};  
 String[] letters = {**"a"**,**"b"**,**"c"**,**"d"**,**"e"**,**"f"**,**"g"**,**"h"**,**"i"**,**"j"**,**"k"**,**"l"**,**"m"**,**"n"**,**"o"**,**"p"**,**"q"**,**"r"**,**"s"**,**"t"**,**"u"**,**"v"**,**"w"**,**"x"**,**"y"**,**"z"**};  
 OutputStream output;  
 File filepath = Environment.*getExternalStorageDirectory*();  
 File dir = **new** File(filepath.getAbsolutePath()  
 + **"/gallery/"**);  
 dir.mkdirs();  
 **this**.setDrawingCacheEnabled(**true**);  
 **this**.buildDrawingCache();  
 Bitmap bitmap1 = **this**.getDrawingCache();  
 Random rn = **new** Random();  
 **int** r = rn.nextInt(10), l = rn.nextInt(26);  
 File image = **new** File(dir + **"/img"**+(l+r)+**"jpeg"**);  
 **try** {  
 output = **new** FileOutputStream(image);  
 bitmap1.compress(Bitmap.CompressFormat.***JPEG***, 90, output);  
 output.flush();  
 output.close();  
 ContentValues values = **new** ContentValues();  
 values.put(MediaStore.Images.Media.***DATE\_TAKEN***, System.*currentTimeMillis*());  
 values.put(MediaStore.Images.Media.***MIME\_TYPE***, **"image/jpeg"**);  
 values.put(MediaStore.MediaColumns.***DATA***, filepath.toString());  
 Toast.*makeText*(getContext(), **"Canvas saved."**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **catch**(Exception e){  
 e.printStackTrace();  
 }  
 **this**.setDrawingCacheEnabled(**false**);  
 }  
}