

فیءة سن ستعرض المدونة، هذه في

4. بمساعدة المدونة هذه كتابة تم:

مقدمة

تطبيقات في مخصوصة رسم واجهة لتنفيذ شاملاً مثال وهي `DrawActivity`، فئة سنستعرض المدونة، هذه في لتحقيق معًا عملها لكي فية مفصل شرح مع المستخدام، والخوارزميات مكون كل بتفكيك سنقوم [١٠].

المحاتويات جدول

- ١- الادارة التاريخية
- ٢- الادارة الخالصة
- ٣- تخصيصات ميزانية
- ٤- إعادات التراجع
- ٥- الأحداش معالجة
- ٦- إدارة إدارية
- ٧- الصوره العملية امتحانات
- ٨- نظرية امتحانات تهويه

نظرة عامة على علی

واجهة إدارة عن مسؤولية الفئات هذه بتطوريه. نقوم الذي الرسم تطبيق في رئيسية فئة هي DrawActivity، هذا في الرسومات. وتعديل إنشاء لـ مسخدم اللازم أدوات توفير بالرسم الخاصة المسخدم الرسم، هذا في الرسومات.

## الرئيسيّة المكونات

### 1. الـ :

هي Canvas كائن بواسطة تمثيلها يتم والخطوط. أشكال رسم فيها يتم التي المنسقة هي هذه الكود.

رسومات، لإنشاء الفأرة أو الـ لاستخدام الـ مع التفاعل للـ يمكّن

### 2. الـ :

الخطوط مثل مختلفة أشكال برسملـ تسمح التي الـ من مجموعة اـ يتصفح والـ والـ والـ .

مثل بـ , , .

### 3. الـ والـ :

الرسومات. لـ المختلفة والـ اختيار الـ لـ يمكّن في .

### 4. الـ والـ :

أو مثل مختلفة ملفات صيغة في بـ قام التي الرسومات حفظ لـ يمكّن في الصور مع الـ مكتبات عبر الـ فـ هذه تنفيذ يـ .

## المـ تفاعـ

الـ ، على بالـ بدأ ثم الـ من أداة باختيار المـ يقوم: الـ مثل الـ أدوات استخدام أو الـ تغيير طريق عن الرسومات تعديـ لـ يـ : الـ الـ .

الآخرين، مع مشاركته أو الـ العمل حفظ لـ يـ الرسـ، من الـ بـ : الـ .

## الـ عـ

```
public class DrawActivity extends AppCompatActivity {  
    private CanvasView canvasView;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_draw);  
    }  
}
```

```

        canvasView = findViewById(R.id.canvas);
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
    }

    public void clearCanvas() {
        canvasView.clearCanvas();
    }

    public void saveDrawing() {
        Bitmap bitmap = canvasView.getBitmap();
        // Save bitmap to file or share it
    }
}

```

اللوجة مسح مثل وظائف تنفيذية تم وال أدوات. اللوجة وإعداد `DrawActivity` تهيئ كي ف نرى الـ `m`، هذا في محددة. طرق عبر الرسم وحفظ

فهم خل ال من الرسومات. لإنشاء المـ `Service` معها يتـ `Activity` `DrawActivity` الخلاصة  
الاستخدام. وسـ `Service` رسم تطبيقي تطوير يمكننا تـ `Activity`، وكيفية الرؤية المـ `Service` المكونات  
المـ `Service` مع والـ `Activity` الصور، اقتصاص الرسم، عمليات مع يـ `Activity` التي الرؤية المـ `Service`  
الـ `Service`، الرسم، من المـ `Service` تـ `Service` مـ `Service` واجهة يـ `Service`. ورفع `Bitmap` الـ `Service` مثل الـ `Service`  
الصور. ومعالجة الـ `Service`، إعادة

```

public class DrawActivity extends Activity implements View.OnClickListener {
    // fragments
    public static final int CAMERA_RESULT = 1;
    public static final int CROP_RESULT = 2;
    public static final int DRAW_FRAGMENT = 0;
    public static final int RECOG_FRAGMENT = 1;
    public static final int RESULT_FRAGMENT = 2;
    public static final int WAIT_FRAGMENT = 3;
    public static final int MATERIAL_RESULT = 4;
    public static final String RESULT_JSON = "resultJson";
    public static final int INIT_FLOWER_ID = R.drawable.flower_b;
    public static final int LOGOUT = 0;
}

```

```
public static final int IMAGE_RESULT = 0;

//  
String baseUrl;  
DrawView drawView;  
Bitmap originImg;  
public static DrawActivity instance;  
View dir, clear, cameraView, materialView, scale;  
ImageView undoView, redoView;  
View upload;  
String cropPath;  
Tooltip toolTip;  
int curFragmentId = -1;  
int serverId = -1;  
private Bitmap resultBitmap;  
private RadioGroup radioGroup;  
Fragment curFragment;  
int curDrawMode;  
RadioButton drawBackBtn;  
private Activity ctxt;  
Uri curPicUri;  
}
```

النحو والصرف

الن شاط حيّة دورة استدعاء يتم، `onCreate` تطبيق في جديّد نشاط إنشاء عن دالن شاط. إن شاء عن داس تدع اوها يتم دالة أول وهي ()-ثعرف المرحلة هذه الاتهيه. بمراحله تبدأ والتي ال أوليه. ال متع غير رات وتعيي ن `onCreate` ال مستخد واجهه تهيهه مثل الأساسية الـ`MainActivity` التي تم الدالة، هذه في

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main); // (layout)
```

```
//  
}
```

المهام تنفيذ لضمان الأصلية الدالة يسندعى `super.onCreate(savedInstanceState)` أعلاه: الكود في الذي احتطيط ملف يعين `setContentView(R.layout.activity_main)` لاظام. بها يقوم الـتـي الأـسـاسـية ملف استخـدام يتم المـثـالـ، هـذاـ فـيـ الـمـسـتـخـدمـ، وـاجـهـةـ لـعـرـضـ اـسـتـخـدـامـ سـيـتـمـ.

و `onStart()` `onResume()` مثل النشاط حـيـاةـ دـورـةـ فـيـ الـأـحـرىـ الـدـوـالـ اـسـتـدـعـاءـ يـتـمـ، `onCreate()` تنفيذ بـعـدـ الـمـسـتـخـدمـ. معـ لـلـفـاعـلـ وجـاهـزاـ رـئـيـاـ أـصـبـحـ الـنـشـاطـ أـنـ تـعـنـيـ والـتـيـ

الـأـوـلـيـةـ، الـصـورـ تـحـمـيـلـ الـعـرـضـ، إـعـادـ مـثـلـ الـمـخـتـلـفـ الـتـهـيـةـ عـمـلـيـاتـ تـنـفـيـذـ يـتـمـ،  `setContentView(R.layout.draw_layout)` إنـشاءـ عـنـدـ الـأـحـدـاثـ، مـسـتـمـعـيـ وـتـهـيـةـ

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    instance = this;  
    ctxt = this;  
    cropPath = PathUtils.getCropPath();  
    setContentView(R.layout.draw_layout);  
    findView();  
    setSize();  
    initOriginImage();  
    toolTip = new Tooltip(this);  
    initUndoRedoEnable();  
    setIp();  
    initDrawmode();  
}
```

ترجمـةـ يـتـمـ لـاـ. تـطـبـيـقـ فـيـ `onCreate()` طـرـيـقـةـ منـ جـزـءـ وـهـوـ  `setContentView()` بـلـغـةـ مـكـتـوبـ الـمـقـدـمـ الـكـودـ مـلـاحـظـةـ: بـشـكـلـ الـتـطـبـيـقـ عـمـلـ لـضـمـانـ هـيـ كـمـاـ تـبـقـىـ أـنـ يـجـبـ الـأـسـمـاءـ هـذـهـ أـنـ حـيـثـ نـفـسـهـ، الـكـودـ أوـ الـمـتـغـيرـاتـ أوـ الـدـوـالـ أـسـمـاءـ صـحـيـحـ.

`setContentView(R.layout.draw_layout);`  
 `setContentView(R.layout.draw_layout);` الـنـشـاطـ فـيـ الـمـسـتـخـدمـ الـعـرـضـ لـتـهـيـةـ الـطـرـيـقـةـ هـذـهـ ثـسـتـخـدمـ.

```
private void findView() {  
    drawView = findViewById(R.id.drawView);  
    undoView = findViewById(R.id.undo);
```

```

redoView = findViewById(R.id.redo);
scale = findViewById(R.id.scale);
upload = findViewById(R.id.upload);
clear = findViewById(R.id.clear);
dir = findViewById(R.id.dir);
materialView = findViewById(R.id.material);
cameraView = findViewById(R.id.camera);

dir.setOnClickListener(this);
materialView.setOnClickListener(this);
undoView.setOnClickListener(this);
scale.setOnClickListener(this);
redoView.setOnClickListener(this);
clear.setOnClickListener(this);
cameraView.setOnClickListener(this);
upload.setOnClickListener(this);
initRadio();
}

```

تم تعيين للايقونات على النحو التالي: dir [] materialView [] undoView [] scale [] redoView [] clear [] cameraView [] upload

تم اسخدام دالة initRadio() لبيان راديو.

تم تعيين حجم الرسم عرض حجم

```

private void setSize() {
    setSizeByResourceSize();
    setViewSize(drawView);
}

```

تم ترجمة الكود الـ Java إلى:

```

private void setSize() {
    setSizeByResourceSize();
    setViewSize(drawView);
}

```

ترجمتها، يتم ولاء ممولة برمجة لغة وهي، بلغة مكتوب لأنه الكود في تغيير يوجد لها

```
private void setSizeByResourceSize() {  
    int width = getResources().getDimensionPixelSize(R.dimen.draw_width);  
    int height = getResources().getDimensionPixelSize(R.dimen.draw_height);  
    App.drawWidth = width;  
    App.drawHeight = height;  
}
```

إلى: أعلى الكود ترجمة تم

```
private void setSizeByResourceSize() {  
    int width = getResources().getDimensionPixelSize(R.dimen.draw_width);  
    int height = getResources().getDimensionPixelSize(R.dimen.draw_height);  
    App.drawWidth = width;  
    App.drawHeight = height;  
}
```

الموارد ملف في المحددة القيمة على بناء والارتفاع العرض حجم تعريف يتم الدالة، هذه في R.dimen.draw\_width و R.dimen.draw\_height، العامة القيم هذه تعريف يتم ثم، App.drawWidth و App.drawHeight.

```
private void setViewSize(View v) {  
    ViewGroup.LayoutParams lp = v.getLayoutParams();  
    lp.width = App.drawWidth;  
    lp.height = App.drawHeight;  
    v.setLayoutParams(lp);  
}
```

\_\_\_\_\_

رسم. استخدامها س يتم الاتي الأولى الصورة تحويل

```
private void initOriginImage() {  
    Bitmap bitmap = BitmapFactory.decodeResource(getResources(), INIT_FLOWER_ID);  
    String imgPath = PathUtils.getCameraPath();  
    BitmapUtils.saveBitmapToPath(bitmap, imgPath);  
    Uri uri1 = Uri.fromFile(new File(imgPath));  
    setImageByUri(uri1);  
}
```

العربية: إلى الكود ترجمة

```
private void initOriginImage() {  
    //  
    Bitmap bitmap = BitmapFactory.decodeResource(getResources(), INIT_FLOWER_ID);  
  
    //  
    String imgPath = PathUtils.getCameraPath();  
  
    //  
    BitmapUtils.saveBitmapToPath(bitmap, imgPath);  
  
    //      Uri  
    Uri uri1 = Uri.fromFile(new File(imgPath));  
  
    //      Uri  
    setImageByUri(uri1);  
}
```

الصورعملیات معالجة

وأقتصاص، **الصور** تعني مثل المخلفة، الصور عمليات بمعالجة **البيانات** النشاط يقوم

للسنة. وإعدادها الـ محدد **الـ** من الصورة بتحمـيل يـقوم

```
private void setImageByUri(final Uri uri) {  
    new Handler().postDelayed(new Runnable() {  
  
        @Override  
        public void run() {  
            curPicUri = uri;  
            Bitmap bitmap = null;
```

```

try {
    if (uri != null) {
        bitmap = BitmapUtils.getBitmapByUri(DrawActivity.this, uri);
    }
} catch (Exception e) {
    e.printStackTrace();
}

private void setImageByUri(final Uri uri) {
    new Handler().postDelayed(new Runnable() {
        @Override
        public void run() {
            curPicUri = uri;
            Bitmap bitmap = null;
            try {
                if (uri != null) {
                    bitmap = BitmapUtils.getBitmapByUri(DrawActivity.this, uri);
                }
            } catch (Exception e) {
                e.printStackTrace();
            }

            int originW = bitmap.getWidth();
            int originH = bitmap.getHeight();
            if (originW != App.drawWidth || originH != App.drawHeight) {
                float originRadio = originW * 1.0f / originH;
                float radio = App.drawWidth * 1.0f / App.drawHeight;
                if (Math.abs(originRadio - radio) < 0.01) {
                    Bitmap originBm = bitmap;
                    bitmap = Bitmap.createScaledBitmap(originBm, App.drawWidth, App.drawHeight, false);
                    originBm.recycle();
                } else {
                    cropIt(uri);
                    return;
                }
            }
            ImageLoader imageLoader = ImageLoader.getInstance();

```

```

imageLoader.addOrReplaceToMemoryCache("origin", bitmap);
originImg = bitmap;
serverId = -1;

العربية: إلى الكود ترجمة

int originW = bitmap.getWidth();
int originH = bitmap.getHeight();
if (originW != App.drawWidth || originH != App.drawHeight) {
    float originRadio = originW * 1.0f / originH;
    float radio = App.drawWidth * 1.0f / App.drawHeight;
    if (Math.abs(originRadio - radio) < 0.01) {
        Bitmap originBm = bitmap;
        bitmap = Bitmap.createScaledBitmap(originBm, App.drawWidth, App.drawHeight, false);
        originBm.recycle();
    } else {
        cropIt(uri);
        return;
    }
}
ImageLoader imageLoader = ImageLoader.getInstance();
imageLoader.addOrReplaceToMemoryCache("origin", bitmap);
originImg = bitmap;
serverId = -1;

```

عادةً، ترجمتها يتم ولـ تقنية أسماء لأنـ هي كما والوظائف الـ متغيرات أسماء عـ لـ الحفاظ تم ملاحظة:

```

drawView.setSrcBitmap(originImg);
showDrawFragment(App.ALL_INFO);
curDrawMode = App.DRAW_FORE;
}
}, 500);
}

```

الرسم لـ عنصر مـ صدرـ يـ كـ صـ وـ رـ ةـ الـ أـ صـ لـ يـ ةـ الـ أـ صـ وـ رـ ةـ تعـ يـ يـ نـ تمـ drawView.  
الـ مـ عـ لـ وـ مـ اـتـ جـ مـ يـ عـ مـ عـ الـ رـ سـ مـ حـ زـ ئـ لـ عـ رـ ضـ الـ دـ الـ اـ لـ اـ سـ تـ دـ عـ اـ ءـ تمـ showDrawFragment(App.ALL\_INFO).  
الـ أـ مـ اـ مـ يـ الـ رـ سـ مـ وـ ضـ عـ إـ لـىـ الـ حـ الـ يـ الـ رـ سـ مـ وـ ضـ عـ تعـ يـ يـ نـ تمـ curDrawMode = App.DRAW\_FORE.  
ثـانـيـةـ مـ لـ لـ لـ يـ 500ـ قـ درـهـ تـ أـ خـ يـ رـ بـ عـ دـ الـ أـ وـ اـ مـ رـ هـ ذـ هـ تـ نـ فـ يـ ذـ يـ تـ

## ٢-٣-١-١-١-١-١

الصورة. اقتصاص نشاط بدء

```
public void cropIt(Uri uri) {  
    Crop.startPhotoCrop(this, uri, cropPath, CROP_RESULT);  
}
```

## ٢-٣-١-١-١-١-٢

الخادم. إلى بتحميلا ويقوم ملف إلى المرسومة النقاطية الصورة يحفظ

```
public void saveBitmap() {  
    Bitmap handBitmap = drawView.getHandBitmap();  
    Bitmap originBitmap = drawView.getSrcBitmap();  
    saveBitmapToFileAndUpload(handBitmap, originBitmap);  
}
```

## ٢-٣-١-١-١-١-٢-١

متزامن. غير بشكل بتحميلا ويقوم ملف في النقاطية الصورة يحفظ

```
private void saveBitmapToFileAndUpload(Bitmap handBitmap, Bitmap originBitmap) {  
    final String originPath = PathUtils.getOriginPath();  
    BitmapUtils.saveBitmapToPath(originBitmap, originPath);  
    final String handPath = PathUtils.getHandPath();  
    BitmapUtils.saveBitmapToPath(handBitmap, handPath);  
    new AsyncTask<Void, Void, Void>() {  
        boolean res;  
        Bitmap foreBitmap;  
        Bitmap backBitmap;  
  
        private void saveBitmapToFileAndUpload(Bitmap handBitmap, Bitmap originBitmap) {  
            final String originPath = PathUtils.getOriginPath();  
            BitmapUtils.saveBitmapToPath(originBitmap, originPath);  
            final String handPath = PathUtils.getHandPath();  
            BitmapUtils.saveBitmapToPath(handBitmap, handPath);  
            new AsyncTask<Void, Void, Void> {  
                boolean res;  
                Bitmap foreBitmap;  
                Bitmap backBitmap;
```

```

@Override
protected void onPreExecute() {
    super.onPreExecute();
    showWaitFragment();
}

@Override
protected Void doInBackground(Void... params) {
    try {
        if (baseUrl == null) {
            throw new Exception("baseUrl is null");
        }
        String jsonRes = UploadImage.upload(baseUrl, serverId, Web.STATUS_CONTINUE, originPath, handPath, n
        getJsonData(jsonRes);
        res = true;
    } catch (Exception e) {
        res = false;
        e.printStackTrace();
    }
    return null;
}

private void getJsonData(String jsonRes) throws Exception {
    JSONObject json = new JSONObject(jsonRes);
    if (serverId == -1) {
        serverId = json.getInt(Web.ID);
    }
    String foreUrl = json.getString(Web.FORE);
    String backUrl = json.getString(Web.BACK);
    String resultUrl = json.getString(Web.RESULT);
    foreBitmap = Web.getBitmapFromUrlByStream1(foreUrl, 0);
    backBitmap = Web.getBitmapFromUrlByStream1(backUrl, 0);
    resultBitmap = Web.getBitmapFromUrlByStream1(resultUrl, 0);
}

@Override
protected void onPostExecute(Void aVoid) {

```

```

super.onPostExecute(aVoid);

if (res) {
    showRecogFragment(foreBitmap, backBitmap);
} else {
    Utils.toast(DrawActivity.this, R.string.server_error);
    recogNo();
}

}

}.execute();
}

```

---

## إدارة الرسم

والتعريف بالرسم مثل للتطبيق، المختلفة الحالات مع للتغييرات إدارة الرسم وإدارة التدوير والانتظار.

الرسم يعرض جزءاً من التطبيق.

```

private void showDrawFragment(int infoId) {
    curFragmentId = DRAW_FRAGMENT;
    curFragment = new DrawFragment(infoId);
    showFragment(curFragment);
}

```

الترجمة:

```

private void showDrawFragment(int infoId) {
    curFragmentId = DRAW_FRAGMENT;
    curFragment = new DrawFragment(infoId);
    showFragment(curFragment);
}

```

يُسّمى int نوع من معاملٍ تأخذ showDrawFragment ثسّمى خاصّة دالة تعريف يتمّ أعلاه، الّكود في النوع من جديـد كـائـن إـنشـاء يتمـ ثمـ curFragmentId قيمة تعـيـين يتمـ الدـالـةـ دـاخـلـ

دالة استدعاء يتم أخiera، infoId لـ curFragment. لمتغير تعريفه ويتم كمعلمات showFragment. الـ curFragment تمرينه كمعلمات.

الـ showWaitFragment()  
الـ showWaitFragment().

```
private void showWaitFragment() {  
    curFragmentId = WAIT_FRAGMENT;  
    showFragment(new WaitFragment());  
}
```

ترجمة:

```
private void showWaitFragment() {  
    curFragmentId = WAIT_FRAGMENT;  
    showFragment(new WaitFragment());  
}
```

باللغة ودوال متغيرات أسماء علی يحتوي لأنه ترجمته يتم لم البرمجية الـ `curFragmentId` الكود ملاحظة: البرمجة. في هي كما تبقى معاادة والتي الـ `WAIT_FRAGMENT`.

الـ showWaitFragment()  
الـ showWaitFragment().

```
private void showFragment(Fragment fragment) {  
    FragmentTransaction trans = getFragmentManager().beginTransaction();  
    trans.replace(R.id.rightLayout, fragment);  
    trans.commit();  
}
```

الـ `showFragment()`: إلى الكود ترجمة

```
private void showFragment(Fragment fragment) {  
    // Fragment  
    FragmentTransaction trans = getFragmentManager().beginTransaction();  
  
    // Fragment Layout Fragment  
    trans.replace(R.id.rightLayout, fragment);
```

```
//  
trans.commit();  
}
```

بآخر R.id.rightLayout المحدد الـ FragmentTransaction replace الدالة واستخدام نوع من لكيه إنشاء طريق عن ذلك يتم commit.

الإصدارات

القوائم. واحتياط الأذار على النقد مثل المخدم، تفاعلات مختلف مع **النقد** تتعامل  
المحتملة **النقد**  
المختلفة. العرض واجهات على النقد أحدها مع الاتصال

```
@Override
public void onClick(View v) {
    int id = v.getId();
    if (id == R.id.drawOk) {
        if (drawView.isDrawFinish()) {
            saveBitmap();
        } else {
            Utils.alertDialog(this, R.string.please_draw_finish);
        }
    } else if (id == R.id.recogOk) {
        recogOk();
    } else if (id == R.id.recogNo) {
        recogNo();
    } else if (id == R.id.dir) {
        Utils.getGalleryPhoto(this, IMAGE_RESULT);
    } else if (id == R.id.clear) {
        clearEverything();
    } else if (id == R.id.undo) {
        drawView.undo();
    } else if (id == R.id.redo) {
```

```
    drawView.redo();

} else if (id == R.id.camera) {
    Utils.takePhoto(cxt, CAMERA_RESULT);

} else if (id == R.id.material) {
    goMaterial();

} else if (id == R.id.upload) {
    com.lzw.commons.Utils.goActivity(cxt, PhotoActivity.class);

} else if (id == R.id.scale) {
    cropIt(curPicUri);

}

}
```

اقتصرت على إثبات ملائمة المدخلات، وبيان تأثيرها على الناتج، وذلك من خلال إثبات  
النحوين التاليين:

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (resultCode != RESULT_CANCELED) {
        Uri uri;
        switch (requestCode) {
            case IMAGE_RESULT:
                if (data != null) {
                    setImageByUri(data.getData());
                }
                break;
            case CAMERA_RESULT:
                setImageByUri(Utils.getCameraUri());
                break;
            case CROP_RESULT:
                uri = Uri.fromFile(new File(cropPath));
                setImageByUri(uri);
                break;
            case MATERIAL_RESULT:
                setImageByUri(data.getData());
        }
    }
}
```

---

## والإعادة للتراجع ميّزات

الرسومية. لـ `View` يوفر `onDraw()` لإعادة التراجع وظيفة `onDraw()`.  
الاستدعاء دوال تعين خلال من والإعادة للتراجع وظائف بـ `onDraw()`.

```
void initUndoRedoEnable() {
    drawView.history.setCallBack(new History.CallBack() {
        @Override
        public void onHistoryChanged() {
            setUndoRedoEnable();
            if (curFragmentId != DRAW_FRAGMENT) {
                showDrawFragment(curDrawMode);
            }
        }
    });
}
```

العربية: إلى الكود ترجمة

```
void initUndoRedoEnable() {
    drawView.history.setCallBack(new History.CallBack() {
        @Override
        public void onHistoryChanged() {
            setUndoRedoEnable();
            if (curFragmentId != DRAW_FRAGMENT) {
                showDrawFragment(curDrawMode);
            }
        }
    });
}
```

الكود: شرح

ـ `View` والإعادة للتراجع إمكانية لـ `onDraw()` دالة:

فی الـ**غاییرات** لـ**تتابع** نـ**داء رد تعيين** یـ**تم**:  
 الـ**سجل**.  
 الـ**سجل**. فـ**ي تغيير** حـ**دوث عن دالـة** هـ**ذه استدعاء** یـ**تم**:  
 الـ**سجل**. حـ**الة علـى بنـاء والـإعادة** الـ**تراجع** أـ**زرار تعـطـيل أو تـفعـيل**:  
 الـ**رسم** جـ**زء هو الـحالـي** الـ**جزـء يـكنـ لمـ إذا**:  
 لـ**رسم الـحالـي** الـ**وضـع** مع الـ**رسم** جـ**زء عـرض** یـ**تم**.

ترجمـهـا. یـ**تم** ولـ**برـمـجيـة** أـ**سمـاء** لـ**أنـها** هي كـ**ما** والـ**متـغيـرات** الـ**دوـال** أـ**سمـاء** عـ**لـى الـحـفـاظ** ـ**تم** مـ**لاحـظـة**:

```
void setUndoRedoEnable() {
    redoView.setEnabled(drawView.history.canRedo());
    undoView.setEnabled(drawView.history.canUndo());
}
```

---

## ـ**خاصـيـات**

لـ**لـخدمـيـن** تـ**سـمـح** تـ**فاعـلـيـة** مـ**سـتـخـدم** وـ**اجـهـة** لـ**إنشاء** **DrawView** تـ**خـصـيـص** كـ**يفـيـة** سـ**نـتـعـلـم** الـ**منـشـور**, هـ**ذا** فـ**ي**  
 ذـ**لكـ**. لـ**تحقـيق** وـ**جـعـل** سـ**نـتـخـدم** الـ**شـاشـة**. عـ**لـى** بـ**الـرـسـم**

قـ**الـاب** باـ**سـتـخـدام** **DrawView** فـ**ي** جـ**ديـد** مـ**شـروع** بـ**إنشاء** قـ**مـأـولـاـ**, جـ**ديـد** مـ**شـروع** إـ**نشـاء**: 1ـ**الـخطـوة**  
 بـ**رمـجـة**. كـ**لـغـة** **Swift** واـ**خـتـرـتـرـيـد**, كـ**ما** الـ**مـشـروع** بـ**تـسـمـيـة** قـ**مـأـولـاـ**.

الـ**تيـيـة** بـ**تعـريـف** وـ**قـمـ** **DrawView.swift** بـ**اسـمـ** جـ**ديـد** مـ**لـفـ** بـ**إنشاء** قـ**مـأـولـاـ** 2ـ**الـخطـوة**  
 منـ**ترـثـ** **UIView**.

```
import UIKit

class DrawView: UIView {

    var lines: [Line] = []
    var lastPoint: CGPoint!

    override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {
        lastPoint = touches.first!.location(in: self)
    }

    override func touchesMoved(_ touches: Set<UITouch>, with event: UIEvent?) {
        let touch = touches.first!
        let currentPoint = touch.location(in: self)

        let line = Line(lastPoint: lastPoint, currentPoint: currentPoint)
        lines.append(line)

        lastPoint = currentPoint
    }

    override func draw(_ rect: CGRect) {
        for line in lines {
            line.draw()
        }
    }
}
```

```

override func touchesMoved(_ touches: Set<UITouch>, with event: UIEvent?) {
    let newPoint = touches.first!.location(in: self)
    lines.append(Line(start: lastPoint, end: newPoint))
    lastPoint = newPoint
    self.setNeedsDisplay()
}

override func draw(_ rect: CGRect) {
    let context = UIGraphicsGetCurrentContext()
    context?.setStrokeColor(UIColor.black.cgColor)
    context?.setLineWidth(2)
    context?.setLineCap(.round)

    for line in lines {
        context?.move(to: line.start)
        context?.addLine(to: line.end)
        context?.strokePath()
    }
}

struct Line {
    var start: CGPoint
    var end: CGPoint
}

```

قم الواجهة. إلى UIView بـ إضافة وقم Main.storyboard افتح الواجهة إلى DrawView إضافة **3: الخطة** قسم فـ **الواجهة** إلى UIView بـ هذه الخاصية بـ **تغـ يـ يـ**.

على الرسم يمكنـ **الآن**، فـ **عـ لـ يـ**. جـ **هـ اـ زـ** أو مـ **حـ اـ كـ اـ** جـ **هـ اـ زـ** على التطبيق بـ **تشـ غـ يـ لـ قـ** **الـ تـ طـ بـ يـ قـ** **تـ شـ غـ يـ لـ**: **4: الخـ طـ وـ** على الشـ **اـ شـ**. على **إـ صـ بـ عـ كـ** **تـ حـ رـ يـ كـ** طـ **رـ يـ قـ** عن الشـ **اـ شـ**

سـ **مـ كـ هـ**. أو الخـ **طـ لـ وـ نـ** **تـ غـ يـ يـ** **رـ** مـ **ثـ لـ** خـ **صـ اـ ئـ صـ** بـ **إـ ضـ اـ فـ** **ةـ** **DrawView** **تـ خـ صـ يـ صـ** **يـ مـ كـ نـ كـ** **5: الخـ طـ وـ** **يـ لـ لـ يـ**: **كـ مـ اـ الـ كـ وـ دـ** **تـ عـ دـ يـ لـ** **يـ مـ كـ نـ كـ** **الـ خـ طـ**, **لـ وـ نـ** **تـ غـ يـ يـ** **رـ** **خـ اـ صـ يـ ةـ** **لـ** **إـ ضـ اـ فـ** **ةـ** **الـ مـ ثـ الـ**, سـ **بـ يـ لـ** **عـ لـ يـ**

```

class DrawView: UIView {
    var lines: [Line] = []
    var lastPoint: CGPoint!
    var strokeColor: UIColor = .black
    var strokeWidth: CGFloat = 2

    override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {
        lastPoint = touches.first!.location(in: self)
    }

    override func touchesMoved(_ touches: Set<UITouch>, with event: UIEvent?) {
        let newPoint = touches.first!.location(in: self)
        lines.append(Line(start: lastPoint, end: newPoint))
        lastPoint = newPoint
        self.setNeedsDisplay()
    }

    override func draw(_ rect: CGRect) {
        let context = UIGraphicsGetCurrentContext()
        context?.setStrokeColor(strokeColor.cgColor)
        context?.setLineWidth(strokeWidth)
        context?.setLineCap(.round)

        for line in lines {
            context?.move(to: line.start)
            context?.addLine(to: line.end)
            context?.strokePath()
        }
    }
}

```

خواص تعديل طريق عن وسمكه الخط لون تغييريمكنك الآن strokeColor و strokeWidth.

على بالرسم لخدمي يسمح مخصوص إنشاء كيفية تعلم منا المنشور، هذا في الخلاصة احتياجاتك. حسب والوظائف الخصائص من المزید بضافه الوظيفة هذه توسيع يمكنا الشاشة.

والتحجيم، الالمس، أحداش الرسم، عمليات لمعالجة يسخدم مخصوص عرض هو DrawView

## البرمجة المترافقية

والتحجيم. لرسم المساحة مع الاج

```
@Override  
public boolean onTouchEvent(MotionEvent event) {  
    if (!scaleMode) {  
        handleDrawTouchEvent(event);  
    } else {  
        handleScaleTouchEvent(event);  
    }  
    return true;  
}
```

:ترجمة

```
@Override  
public boolean onTouchEvent(MotionEvent event) {  
    if (!scaleMode) {  
        handleDrawTouchEvent(event);  
    } else {  
        handleScaleTouchEvent(event);  
    }  
    return true;  
}
```

ترجمت. يتم ل البرمجي والكود برمجي، كود لأنه تغييريده يتم لم البرمجية الكتلة في الموجود الكود ملاحظة:

```
private void handleDrawTouchEvent(MotionEvent event) {  
    int action = event.getAction();  
    float x = event.getX();  
    float y = event.getY();  
    if (action == MotionEvent.ACTION_DOWN) {  
        path.moveTo(x, y);  
    } else if (action == MotionEvent.ACTION_MOVE) {  
        path.quadTo(preX, preY, x, y);  
    } else if (action == MotionEvent.ACTION_UP) {  
        Matrix matrix1 = new Matrix();
```

```

        matrix.invert(matrix1);
        path.transform(matrix1);
        paint.setStrokeWidth(strokeWidth * 1.0f / totalRatio);
        history.saveToStack(path, paint);
        cacheCanvas.drawPath(path, paint);
        paint.setStrokeWidth(strokeWidth);
        path.reset();
    }

    setPrev(event);
    invalidate();
}

```

الترجمة:

```

private void handleDrawTouchEvent(MotionEvent event) {
    int action = event.getAction();
    float x = event.getX();
    float y = event.getY();
    if (action == MotionEvent.ACTION_DOWN) {
        path.moveTo(x, y);
    } else if (action == MotionEvent.ACTION_MOVE) {
        path.quadTo(preX, preY, x, y);
    } else if (action == MotionEvent.ACTION_UP) {
        Matrix matrix1 = new Matrix();
        matrix.invert(matrix1);
        path.transform(matrix1);
        paint.setStrokeWidth(strokeWidth * 1.0f / totalRatio);
        history.saveToStack(path, paint);
        cacheCanvas.drawPath(path, paint);
        paint.setStrokeWidth(strokeWidth);
        path.reset();
    }
    setPrev(event);
    invalidate();
}

```

## الشرح:

- `handleDrawTouchEvent`: لرسم الالمس أحدا ثعالج دالة
  - `MotionEvent`: الالمس. حدث يمثل
  - `action`: الرفع. الحركة، الضغط، مثل الحدث بنوع يحتفظ
  - `x, y`: الالمس. نقطة إحداثيات
  - `path.moveTo(x, y)`: الالنقطة من جديد مساري بدأ.
  - `path.quadTo(preX, preY, x, y)`: والنقطة `preX, preY` السابقة للنقطة بين منحنى يرسم، `x, y` الحالية.
  - `Matrix`: إحداثيات. لتحويل يستخدم
  - `paint.setStrokeWidth`: الفرشاة. عرض يحدده
  - `history.saveToStack`: للتاريخ. في والفرشاة الالمس اريحة حفظ
  - `cacheCanvas.drawPath`: المؤقتة. الرسم لوحة على الالمس اريرسم
  - `path.reset()`: الالمس اريتعيين يعيدي
  - `setPrev(event)`: السابقة. إحداثيات يحفظ
  - `invalidate()`: الواجهة. رسم يعيدي

```
private void handleScaleTouchEvent(MotionEvent event) {  
  
    switch (event.getActionMasked()) {  
  
        case MotionEvent.ACTION_POINTER_DOWN:  
            lastFingerDist = calFingerDistance(event);  
            break;  
  
        case MotionEvent.ACTION_MOVE:  
            if (event.getPointerCount() == 1) {  
                handleMove(event);  
            } else if (event.getPointerCount() == 2) {  
                handleZoom(event);  
            }  
            break;  
  
        case MotionEvent.ACTION_UP:  
  
        case MotionEvent.ACTION_POINTER_UP:  
            lastMoveX = -1;  
            lastMoveY = -1;  
            break;  
  
        default:  
            break;  
    }  
}
```

```
}
```

العربية: إلى الكود ترجمة

```
private void handleScaleTouchEvent(MotionEvent event) {  
    switch (event.getActionMasked()) {  
        case MotionEvent.ACTION_POINTER_DOWN:  
            lastFingerDist = calFingerDistance(event);  
            break;  
        case MotionEvent.ACTION_MOVE:  
            if (event.getPointerCount() == 1) {  
                handleMove(event);  
            } else if (event.getPointerCount() == 2) {  
                handleZoom(event);  
            }  
            break;  
        case MotionEvent.ACTION_UP:  
        case MotionEvent.ACTION_POINTER_UP:  
            lastMoveX = -1;  
            lastMoveY = -1;  
            break;  
        default:  
            break;  
    }  
}
```

ترجمتها. يتم ولاتقنية أسماء لأنها هي كما والمتغيرات الدوليين أسماء على الاحفاظ تم ملاحظة:

```
private void handleMove(MotionEvent event) {  
    float moveX = event.getX();  
    float moveY = event.getY();  
    if (lastMoveX == -1 && lastMoveY == -1) {  
        lastMoveX = moveX;  
        lastMoveY = moveY;  
    }  
    moveDistX = (int) (moveX - lastMoveX);  
    moveDistY = (int) (moveY - lastMoveY);
```

```

if (moveDistX + totalTranslateX > 0 || moveDistX + totalTranslateX + curBitmapWidth < width) {
    moveDistX = 0;
}
if (moveDistY + totalTranslateY > 0 || moveDistY + totalTranslateY + curBitmapHeight < height) {
    moveDistY = 0;
}
status = STATUS_MOVE;
invalidate();
lastMoveX = moveX;
lastMoveY = moveY;
}

```

إلى: أعلى إلكرود ترجمة تم ترجمة

```

private void handleMove(MotionEvent event) {
    float moveX = event.getX();
    float moveY = event.getY();
    if (lastMoveX == -1 && lastMoveY == -1) {
        lastMoveX = moveX;
        lastMoveY = moveY;
    }
    moveDistX = (int) (moveX - lastMoveX);
    moveDistY = (int) (moveY - lastMoveY);
    if (moveDistX + totalTranslateX > 0 || moveDistX + totalTranslateX + curBitmapWidth < width) {
        moveDistX = 0;
    }
    if (moveDistY + totalTranslateY > 0 || moveDistY + totalTranslateY + curBitmapHeight < height) {
        moveDistY = 0;
    }
    status = STATUS_MOVE;
    invalidate();
    lastMoveX = moveX;
    lastMoveY = moveY;
}

```

لـ `onTouchEvent`، يـ `handleMove` يـ `MotionEvent` يـ `getX` و `getY` يـ `float` يـ `moveX` و `moveY` يـ `float` يـ `lastMoveX` و `lastMoveY` يـ `float` يـ `-1` يـ `if` يـ `lastMoveX == -1 && lastMoveY == -1` يـ `lastMoveX = moveX` و `lastMoveY = moveY` يـ `int` يـ `moveDistX` و `moveDistY` يـ `(int) (moveX - lastMoveX)` و `(int) (moveY - lastMoveY)` يـ `if` يـ `moveDistX + totalTranslateX > 0 || moveDistX + totalTranslateX + curBitmapWidth < width` يـ `moveDistX = 0` يـ `if` يـ `moveDistY + totalTranslateY > 0 || moveDistY + totalTranslateY + curBitmapHeight < height` يـ `moveDistY = 0` يـ `STATUS_MOVE` يـ `status` يـ `invalidate` يـ `lastMoveX = moveX` و `lastMoveY = moveY`

```

private void handleZoom(MotionEvent event) {
    float fingerDist = calFingerDistance(event);
    calFingerCenter(event);
    if (fingerDist > lastFingerDist) {
        status = STATUS_ZOOM_OUT;
    } else {
        status = STATUS_ZOOM_IN;
    }
    scaledRatio = fingerDist * 1.0f / lastFingerDist;
    totalRatio = totalRatio * scaledRatio;
    if (totalRatio < initRatio) {
        totalRatio = initRatio;
    } else if (totalRatio > initRatio * 4) {
        totalRatio = initRatio * 4;
    }
    lastFingerDist = fingerDist;
    invalidate();
}

```

٠٠٠٠٠٠٠٠٠٠٠٠٠٠٠

لرسم يد يرسم الحالة الحال الحالية الحالية الحالية الحالية

```

@Override
protected void onDraw(Canvas canvas) {
    super.onDraw(canvas);
    if (scaleMode) {
        switch (status) {
            case STATUS_MOVE:
                move(canvas);
                break;
            case STATUS_ZOOM_IN:
            case STATUS_ZOOM_OUT:
                zoom(canvas);
                break;
            default:
                if (cacheBm != null) {
                    canvas.drawBitmap(cacheBm, matrix, null);
                }
        }
    }
}

```

```

        canvas.drawPath(path, paint);
    }
}
} else {
    if (cacheBm != null) {
        canvas.drawBitmap(cacheBm, matrix, null);
        canvas.drawPath(path, paint);
    }
}
}

```

## الرسوميات

الاتكبير [التصغير]. اثناء الحركة عملية مع يتعامل

```

private void move(Canvas canvas) {
    matrix.reset();
    matrix.postScale(totalRatio, totalRatio);
    totalTranslateX = moveDistX + totalTranslateX;
    totalTranslateY = moveDistY + totalTranslateY;
    matrix.postTranslate(totalTranslateX, totalTranslateY);
    canvas.drawBitmap(cacheBm, matrix, null);
}

```

العربى: إلى الكود ترجمة

```

private void move(Canvas canvas) {
    matrix.reset(); //
    matrix.postScale(totalRatio, totalRatio); //
    totalTranslateX = moveDistX + totalTranslateX; //
    totalTranslateY = moveDistY + totalTranslateY; //
    matrix.postTranslate(totalTranslateX, totalTranslateY); //
    canvas.drawBitmap(cacheBm, matrix, null); //
}

```

ترجمت. يتم لانفسه الكود أن حيث فقط، الوضعيية التعلقيات ترجمة تمت ملاحظة:

## الرسوميات

الاتكبير [التصغير]. عملية مع الاتصال

```

private void zoom(Canvas canvas) {
    matrix.reset();
    matrix.postScale(totalRatio, totalRatio);
    int scaledWidth = (int) (cacheBm.getWidth() * totalRatio);
    int scaledHeight = (int) (cacheBm.getHeight() * totalRatio);
    int translateX;
    int translateY;
    if (curBitmapWidth < width) {
        translateX = (width - scaledWidth) / 2;
    } else {
        translateX = (int) (centerPointX + (totalTranslateX - centerPointX) * scaledRatio);
        if (translateX > 0) {
            translateX = 0;
        } else if (scaledWidth + translateX < width) {
            translateX = width - scaledWidth;
        }
    }
    if (curBitmapHeight < height) {
        translateY = (height - scaledHeight) / 2;
    } else {
        translateY = (int) (centerPointY + (totalTranslateY - centerPointY) * scaledRatio);
        if (translateY > 0) {
            translateY = 0;
        } else if (scaledHeight + translateY < height) {
            translateY = height - scaledHeight;
        }
    }
    matrix.postTranslate(translateX, translateY);
    canvas.drawBitmap(cacheBm, matrix, null);
}

```

الرجم:

```

private void zoom(Canvas canvas) {
    matrix.reset();
    matrix.postScale(totalRatio, totalRatio);

```

```

int scaledWidth = (int) (cacheBm.getWidth() * totalRatio);
int scaledHeight = (int) (cacheBm.getHeight() * totalRatio);
int translateX;
int translateY;
if (curBitmapWidth < width) {
    translateX = (width - scaledWidth) / 2;
} else {
    translateX = (int) (centerPointX + (totalTranslateX - centerPointX) * scaledRatio);
    if (translateX > 0) {
        translateX = 0;
    } else if (scaledWidth + translateX < width) {
        translateX = width - scaledWidth;
    }
}
if (curBitmapHeight < height) {
    translateY = (height - scaledHeight) / 2;
} else {
    translateY = (int) (centerPointY + (totalTranslateY - centerPointY) * scaledRatio);
    if (translateY > 0) {
        translateY = 0;
    } else if (scaledHeight + translateY < height) {
        translateY = height - scaledHeight;
    }
}
matrix.postTranslate(translateX, translateY);
canvas.drawBitmap(cacheBm, matrix, null);
}

Y = height - scaledHeight;
}
}

totalTranslateX = translateX;
totalTranslateY = translateY;
curBitmapWidth = scaledWidth;
curBitmapHeight = scaledHeight;
matrix.postTranslate(translateX, translateY);
canvas.drawBitmap(cacheBm, matrix, null);

```

```
}
```

---

## التاريخ إدارة

والإعادة. التراجع وظائف لتمكين الرسم سجل تدريسي History

المحركات المترافق مع Android API، مما يتيح

المكتس. في والفرشة الحالياً المنسار يحفظ

```
public void saveToStack(Path path, Paint paint) {  
    Draw draw = new Draw();  
    draw.path = new Path(path);  
    draw.paint = new Paint(paint);  
    saveToStack(draw);  
}
```

يلى: كما العربية إلى أعلى الكود ترجمة تمت

```
public void saveToStack(Path path, Paint paint) {  
    Draw draw = new Draw();  
    draw.path = new Path(path);  
    draw.paint = new Paint(paint);  
    saveToStack(draw);  
}
```

مثل الكائنات أو المتغيرات أو الدوال أسماء ترجمة يتم ولياً، بلغة مكتوب المقدم الكود ملاحظة: Path، Paint، Draw، يتم ولي البرمجة لغة من جزء لأنها هي كما الأسماء هذه على الحفاظ يتم إلخ.

```
public void saveToStack(Draw draw) {  
    curPos++;  
    while (histroy.size() > curPos) {  
        histroy.pop();  
    }  
    histroy.push(draw);  
    if (callBack != null) {  
        callBack.onHistoryChanged();  
    }
```

```
    }  
}  
}
```

إلى: أعلاه الـ `Draw` تم ترجمة الكود.

```
public void saveToStack(Draw draw) {  
    curPos++;  
    while (histroy.size() > curPos) {  
        histroy.pop();  
    }  
    histroy.push(draw);  
    if (callBack != null) {  
        callBack.onHistoryChanged();  
    }  
}
```

كما في الأعلى، يُحفظ الكود المُحْدَّث في `histroy`. وظائف مثل `push` و `pop` مُطبقة على الكود المُحْدَث.

الرسوم سجل في المحددة النقطة حالي تمثل `histroy` نقطية صورة ثُرُج.

```
public Bitmap getBitmapAtDraw(int n) {  
    Canvas canvas = new Canvas();  
    Bitmap bm = Utils.getCopyBitmap(srcBitmap);  
    canvas.setBitmap(bm);  
    for (int i = 0; i <= n; i++) {  
        Draw draw = histroy.get(i);  
        canvas.drawPath(draw.path, draw.paint);  
    }  
    return bm;  
}
```

الرجوع. عمليّة بتنفيذ يقوم

```
public Bitmap undo() throws UnsupportedOperationException {  
    if (canUndo()) {
```

```

        curPos--;
        if (callBack != null) {
            callBack.onHistoryChanged();
        }
        return getBitmapAtDraw(curPos);
    } else {
        throw new UnsupportedOperationException(" ");
    }
}

```

## ٢-٣-١

الإعادة عملية بتنفيذية بعزم.

```

public Bitmap redo() throws UnsupportedOperationException {
    if (canRedo()) {
        curPos++;
        if (callBack != null) {
            callBack.onHistoryChanged();
        }
        return getBitmapAtDraw(curPos);
    } else {
        throw new UnsupportedOperationException(" ");
    }
}

```

## ٢-٣-٢-١

الأخير. الإرجاء عن التراجع بالإمكان كان إذا مما يتحقق

```

public boolean canUndo() {
    return curPos > 0;
}

```

الترجمة:

```

public boolean canUndo() {
    return curPos > 0;
}

```

تتغیر ولا عالمية برمجة لغة وهي **Java** بلغة مكتوب أنه حيث ترجمة إلى يحتج لـ المقدم الكود ملحوظة: الـ**language**.

## الـ**language**

الإجراء، إعادة بالإمكان كان إذا مما يتحقق

```
public boolean canRedo() {  
    return curPos + 1 < histroy.size();  
}
```

الترجمة:

```
public boolean canRedo() {  
    return curPos + 1 < histroy.size();  
}
```

الإجراءات تنفيذ إعادة مع يتطلب طبيق من جزء أنه ويبدو **language** بلغة مكتوب المقدم الكود ملحوظة: الـ**language** أن من التحقق طريق عن **language** الـ**language** إجراء إعادة تنفيذ بالمكان كان إذا مما يتحقق الكود. يعني فهذا صحيحًا، الشرط كان إذا **histroy.size()** تاريخ قائمة حجم من أقل واحد زائد **curPos** الحالي إجراء، إعادة تنفيذ يمكن أنه

---

## الخلاصة

نظام في مخصص رسم عرض لـ**DrawActivity** يوفر **language**. مكونات مع والتكامل الرسم، سجل وإدارة الملف، أحدا مع الـ**language** مثل في بما متتنوعة، تقنيات المثال هذا يعرض وخارزمية، مكون كل فهم خلال من . **language** المتسامنة غير والمهام **language** الـ**language** أجزاء مثل أخرى وتفاعلية. قوية رسم ميزات لـ**language** الخلاصات في التقنيات هذه من الاستفادة يمكن