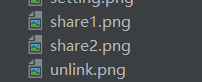
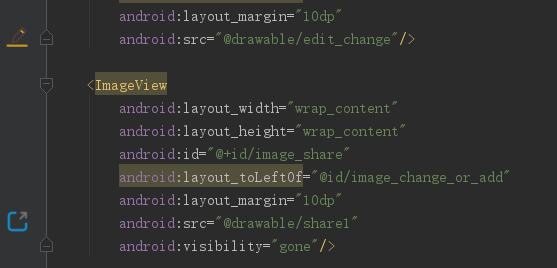
1. 实现步骤：

1、在res文件

### 1.1、在drawable-xxhdpi添加图片



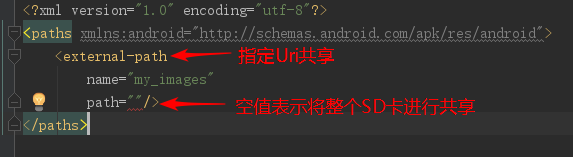
### 1.2、在title\_note\_category.xml定义ImageView控件



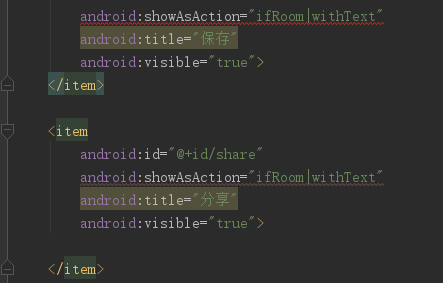
代码：

<ImageView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/image\_share"  
 android:layout\_toLeftOf="@id/image\_change\_or\_add"  
 android:layout\_margin="10dp"  
 android:src="@drawable/share1"  
 android:visibility="gone"/>

### 1.3、在xml目录下创建file\_path.xml文件



### 1.4、在layout目录下的option\_menu.xml添加分享功能选项



代码：

<item  
 android:id="@+id/share"  
 android:showAsAction="ifRoom|withText"  
 android:title="分享"  
 android:visible="true">  
  
</item>

1. 在AndroidManifest.xml文件中对内容提供器进行注册

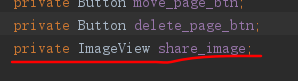


代码：

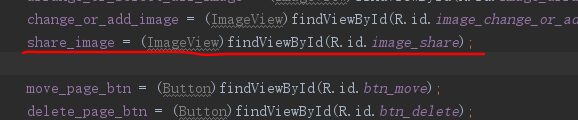
<provider  
 android:authorities="com.sonix.fileprovider"  
 android:name="android.support.v4.content.FileProvider"  
 android:exported="false"  
 android:grantUriPermissions="true">  
 <meta-data  
 android:name="android.support.FILE\_PROVIDER\_PATHS"  
 android:resource="@xml/file\_paths"/>  
</provider>

3、在Note\_category\_main().java文件

### 3.1、定义全局变量

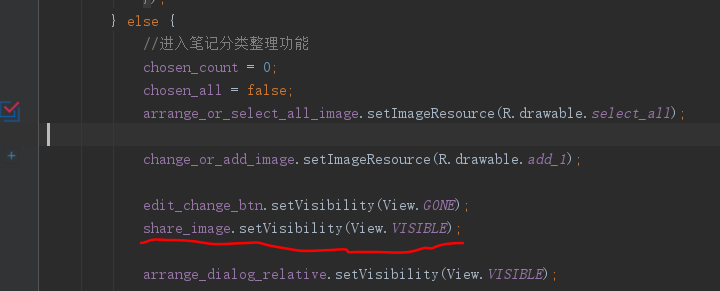


### 3.2、在onCreate()函数中初始化分享功能图标

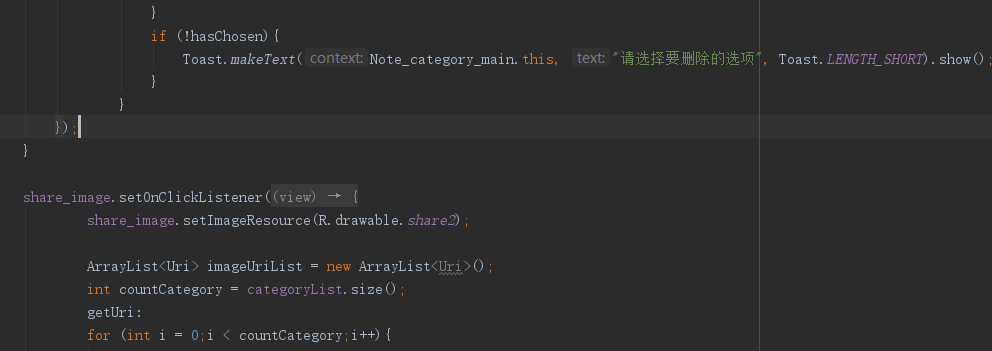


### 3.3、在arrange\_or\_change\_show()整理和更名功能初始化函数

#### **3.3.1、定义分享功能图标显示**



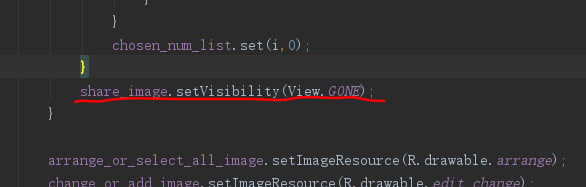
#### **3.3.2、定义分享功能图标点击监听**



代码：

share\_image.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 share\_image.setImageResource(R.drawable.*share2*);  
  
 ArrayList<Uri> imageUriList = new ArrayList<Uri>();  
 int countCategory = categoryList.size();  
 getUri:  
 for (int i = 0;i < countCategory;i++){  
 Category category = categoryList.get(i);  
 if (category.isChosen\_category()){  
 category.setChosen\_category(false);  
 }  
 for (Category.Page page : category.getPageList()){  
 if (page.isChosen\_page()){  
 page.setChosen\_page(false);  
 File image\_file = new File(page.getPage\_image());  
  
 Uri imageUri;  
 */\*\*  
 \* 低于Android7.0，调用Uri的fromFile()方法将File对象转换为Uri对象，Uri对象标识着本地真实路径。  
 \*  
 \* Android7.0开始直接使用本地这是路径Uri被认为是不安全的，会抛出一个FileUriExposedException异常。  
 \* 而FileProvider则是一种特殊的内容提供器，它使用了和内容提供器类似的机制对数据进行保护，可以选择性第将封装过的  
 \* Uri共享到外部，提高了应用的安全性。  
 \*  
 \* getUriForFile()方法接收3个参数，第一个为Context对象，第二个而任意唯一的字符串，第三个是File对象  
 \*/* if (Build.VERSION.*SDK\_INT* >= 24) {  
 imageUri = FileProvider.*getUriForFile*(Note\_category\_main.this,"com.sonix.fileprovider",image\_file);  
 } else {  
 imageUri = Uri.*fromFile*(image\_file);  
 }  
  
 imageUriList.add(imageUri);  
 if (imageUriList.size() == chosen\_count){  
 reduceChosen\_count(chosen\_count);  
 break getUri;  
 }  
 image\_file = null;  
 }  
 }  
 chosen\_num\_list.set(i,0);  
 }  
  
 Intent shareIntent = new Intent();  
 //ACTION\_SEND\_MULTIPLE发送多附件时用  
 shareIntent.setAction(Intent.*ACTION\_SEND\_MULTIPLE*);  
 shareIntent.setType("image/\*");  
 shareIntent.putParcelableArrayListExtra(Intent.*EXTRA\_STREAM*,imageUriList);  
 startActivity(Intent.*createChooser*(shareIntent,"页图分享到"));  
 share\_image.setImageResource(R.drawable.*share1*);  
 myExpandable.notifyDataSetChanged();  
 }  
});

### 3.4、在arrange\_or\_change\_quit()整理和更名功能退出函数定义分享功能图标隐藏



4、在OidActivity().java文件

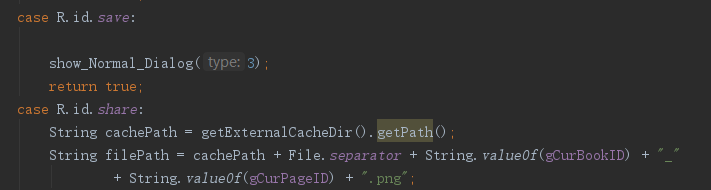
### 4.1、更改onCreateOptionsMenu()菜单初始化事件



代码：

if (!isConnected && !designation) {  
 menu.findItem(R.id.*flag1*).setVisible(false);  
 menu.findItem(R.id.*flag2*).setVisible(true);  
 menu.findItem(R.id.*clear*).setVisible(false);  
 menu.findItem(R.id.*color*).setVisible(false);  
 menu.findItem(R.id.*width*).setVisible(false);  
 menu.findItem(R.id.*replay*).setVisible(false);  
 menu.findItem(R.id.*setting*).setVisible(false);  
 menu.findItem(R.id.*offline*).setVisible(false);  
 menu.findItem(R.id.*LED*).setVisible(false);  
 menu.findItem(R.id.*OTA*).setVisible(false);  
 menu.findItem(R.id.*check\_txt*).setVisible(false);  
 menu.findItem(R.id.*recognition*).setVisible(false);  
 menu.findItem(R.id.*save*).setVisible(false);  
 menu.findItem(R.id.*share*).setVisible(false);  
} else if (!isConnected) {  
 menu.findItem(R.id.*flag1*).setVisible(false);  
 menu.findItem(R.id.*flag2*).setVisible(true);  
 menu.findItem(R.id.*clear*).setVisible(true);;  
 menu.findItem(R.id.*replay*).setVisible(true);  
 menu.findItem(R.id.*recognition*).setVisible(true);  
 menu.findItem(R.id.*save*).setVisible(true);  
 menu.findItem(R.id.*share*).setVisible(true);  
 menu.findItem(R.id.*color*).setVisible(false);  
 menu.findItem(R.id.*width*).setVisible(false);  
 menu.findItem(R.id.*setting*).setVisible(false);  
 menu.findItem(R.id.*offline*).setVisible(false);  
 menu.findItem(R.id.*LED*).setVisible(false);  
 menu.findItem(R.id.*OTA*).setVisible(false);  
 menu.findItem(R.id.*check\_txt*).setVisible(false);  
} else {  
 menu.findItem(R.id.*flag1*).setVisible(true);  
 menu.findItem(R.id.*flag2*).setVisible(false);  
 menu.findItem(R.id.*clear*).setVisible(true);  
 menu.findItem(R.id.*color*).setVisible(true);  
 menu.findItem(R.id.*width*).setVisible(true);  
 menu.findItem(R.id.*replay*).setVisible(true);  
 menu.findItem(R.id.*setting*).setVisible(true);  
 menu.findItem(R.id.*offline*).setVisible(true);  
 menu.findItem(R.id.*LED*).setVisible(false);  
 menu.findItem(R.id.*OTA*).setVisible(false);  
 menu.findItem(R.id.*check\_txt*).setVisible(false);  
 menu.findItem(R.id.*recognition*).setVisible(true);  
 menu.findItem(R.id.*save*).setVisible(true);  
 menu.findItem(R.id.*share*).setVisible(true);  
}

### 4.2、修改onOptionsItemSelected()菜单响应事件



代码：

case R.id.*share*:  
 String cachePath = getExternalCacheDir().getPath();  
 String filePath = cachePath + File.*separator* + String.*valueOf*(gCurBookID) + "\_"  
 + String.*valueOf*(gCurPageID) + ".png";  
 File image\_file = new File(filePath);  
 if (!image\_file.exists()){  
 save\_page\_image();  
 }  
  
 Uri imageUri;  
 if (Build.VERSION.*SDK\_INT* >= 24) {  
 imageUri = FileProvider.*getUriForFile*(OidActivity.this,"com.sonix.fileprovider",image\_file);  
 } else {  
 imageUri = Uri.*fromFile*(image\_file);  
 }  
 Intent shareIntent = new Intent();  
 shareIntent.setAction(Intent.*ACTION\_SEND*);  
 shareIntent.putExtra(Intent.*EXTRA\_STREAM*,imageUri);  
 shareIntent.setType("image/\*");  
 startActivity(Intent.*createChooser*(shareIntent,"页图分享到"));  
 return true;

5、在RecognitionActivity().java文件

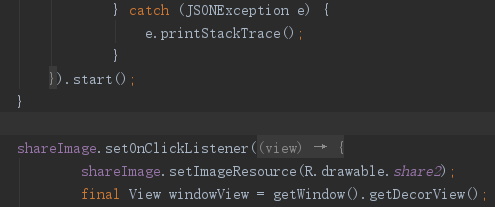
### 5.1、定义全局变量



### 5.2、在onCreate()分享功能图标实例化



### 5.3、定义分享功能图标点击事件



代码：

shareImage.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 shareImage.setImageResource(R.drawable.*share2*);  
 final View windowView = getWindow().getDecorView();  
 windowView.setDrawingCacheEnabled(true);  
 windowView.buildDrawingCache();  
 final Bitmap bitmap = windowView.getDrawingCache();  
  
 final AlertDialog.Builder normal\_dialog = new AlertDialog.Builder(RecognitionActivity.this);  
 AlertDialog alertDialog;  
 normal\_dialog.setMessage("分享识别结果");  
 normal\_dialog.setPositiveButton("图片", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int i) {  
 String cachePath = getExternalCacheDir().getPath();  
 String filePath = cachePath + File.*separator* + "recognition.png";  
 File file = new File(filePath);  
 if (file.exists()) {  
 file.delete();  
 getContentResolver().delete(MediaStore.Images.Media.*EXTERNAL\_CONTENT\_URI* , MediaStore.Images.Media.*DATA* + "=?", new String[]{filePath});  
 MediaScannerConnection.*scanFile*(RecognitionActivity.this, new String[]{filePath}, null, null);  
 }  
 try{  
 FileOutputStream outputStream = new FileOutputStream(filePath);  
 bitmap.compress(Bitmap.CompressFormat.*PNG*, 100, outputStream);  
 outputStream.flush();  
 outputStream.close();  
 } catch (Exception e){  
 e.printStackTrace();  
 }  
  
 Uri imageUri;  
 if (Build.VERSION.*SDK\_INT* >= 24) {  
 imageUri = FileProvider.*getUriForFile*(RecognitionActivity.this,"com.sonix.fileprovider",file);  
 } else {  
 imageUri = Uri.*fromFile*(file);  
 }  
 Intent shareIntent = new Intent();  
 shareIntent.setAction(Intent.*ACTION\_SEND*);  
 shareIntent.putExtra(Intent.*EXTRA\_STREAM*,imageUri);  
 shareIntent.setType("image/\*");  
 startActivity(Intent.*createChooser*(shareIntent,"页图分享到"));  
 shareImage.setImageResource(R.drawable.*share1*);  
 windowView.setDrawingCacheEnabled(false);  
 windowView.buildDrawingCache(false);  
  
 }  
  
 });  
  
 normal\_dialog.setNegativeButton("文字", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int i) {  
 Intent shareIntent = new Intent();  
 shareIntent.setAction(Intent.*ACTION\_SEND*);  
 shareIntent.putExtra(Intent.*EXTRA\_TEXT*,mTextView.getText().toString());  
 Log.*d*(*TAG*, "onClick: mTextView.getText()====" + mTextView.getText());  
 shareIntent.setType("text/plain");  
 startActivity(Intent.*createChooser*(shareIntent,"文字分享到"));  
 shareImage.setImageResource(R.drawable.*share1*);  
 windowView.setDrawingCacheEnabled(false);  
 windowView.buildDrawingCache(false);  
 }  
 });  
  
 alertDialog = normal\_dialog.create();  
 alertDialog.show();  
 }  
});

二、参考资料：

分享实现：https://blog.csdn.net/wanglining1987/article/details/52698535

三、改进建议：

实现自动适应屏幕大小定区域截图；实现以word或PDF文件进行分享。