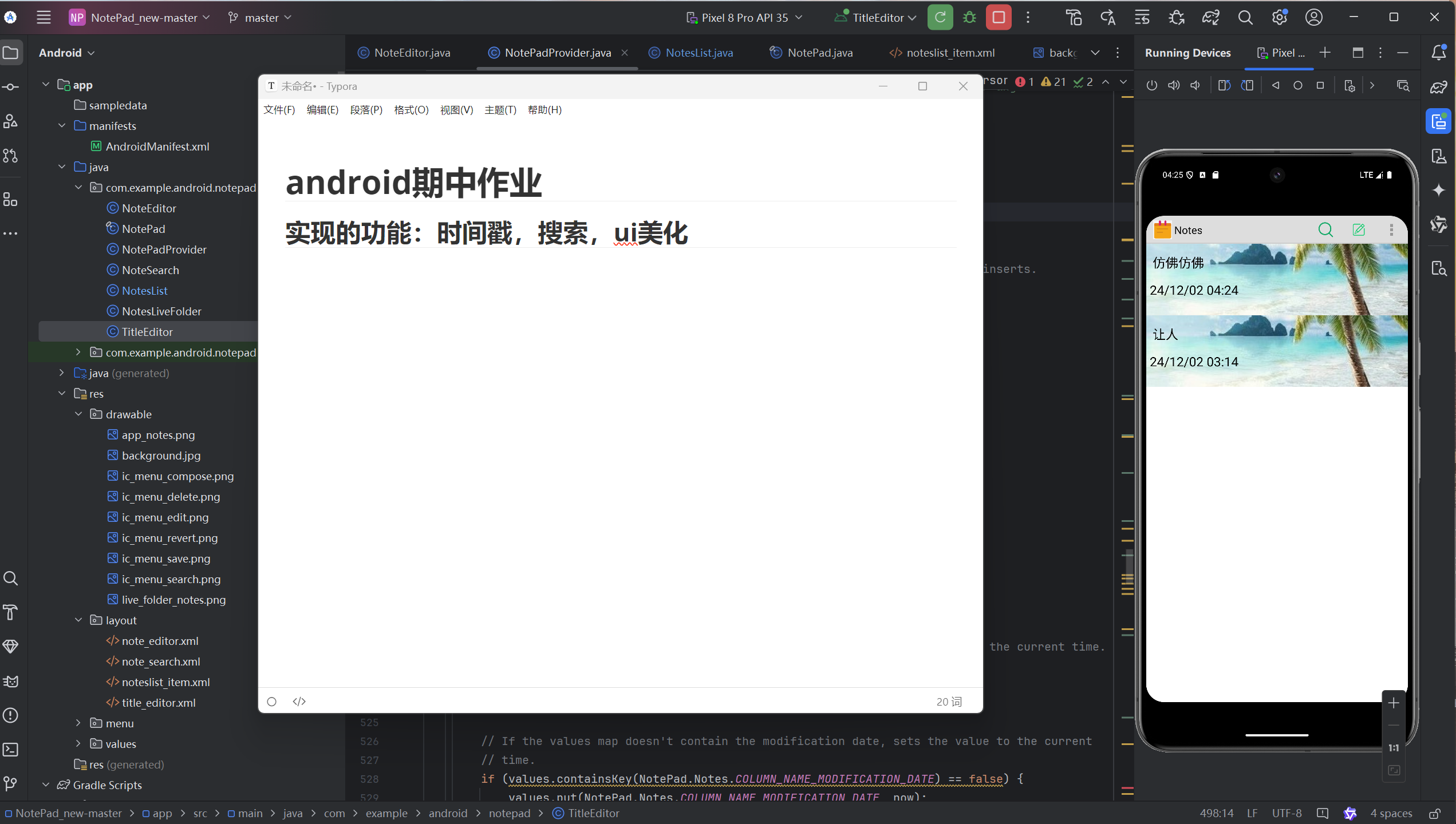
android期中作业

## 实现的功能：时间戳，搜索，ui美化

1、时间戳：

1.1、实现效果



1.2、实现过程

（1）在NoteList类中扩增dataColumns，以及对应的视图

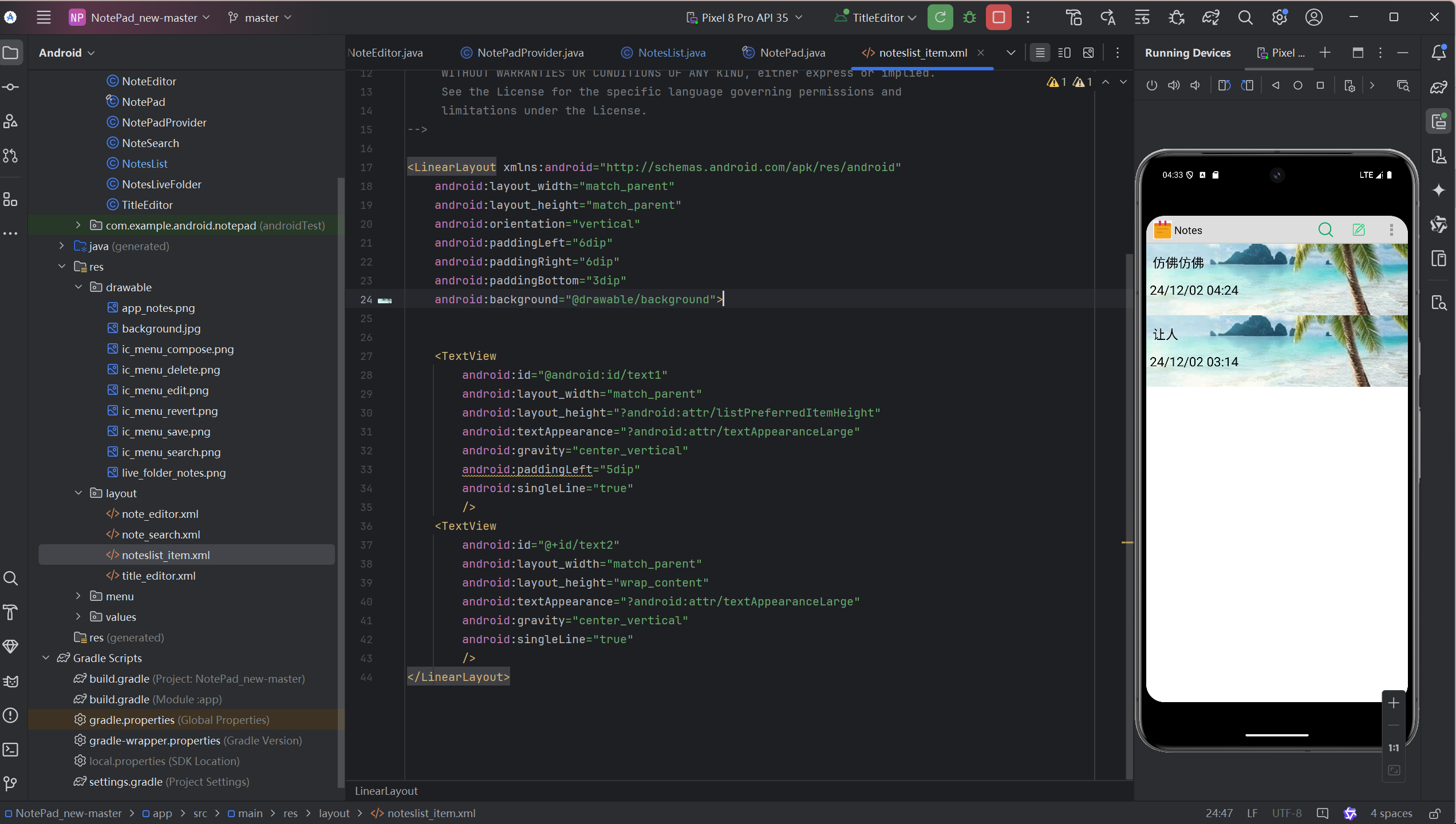
final String[] dataColumns = {

NotePad.Notes.COLUMN\_NAME\_TITLE , NotePad.Notes.COLUMN\_NAME\_MODIFICATION\_DATE} ;

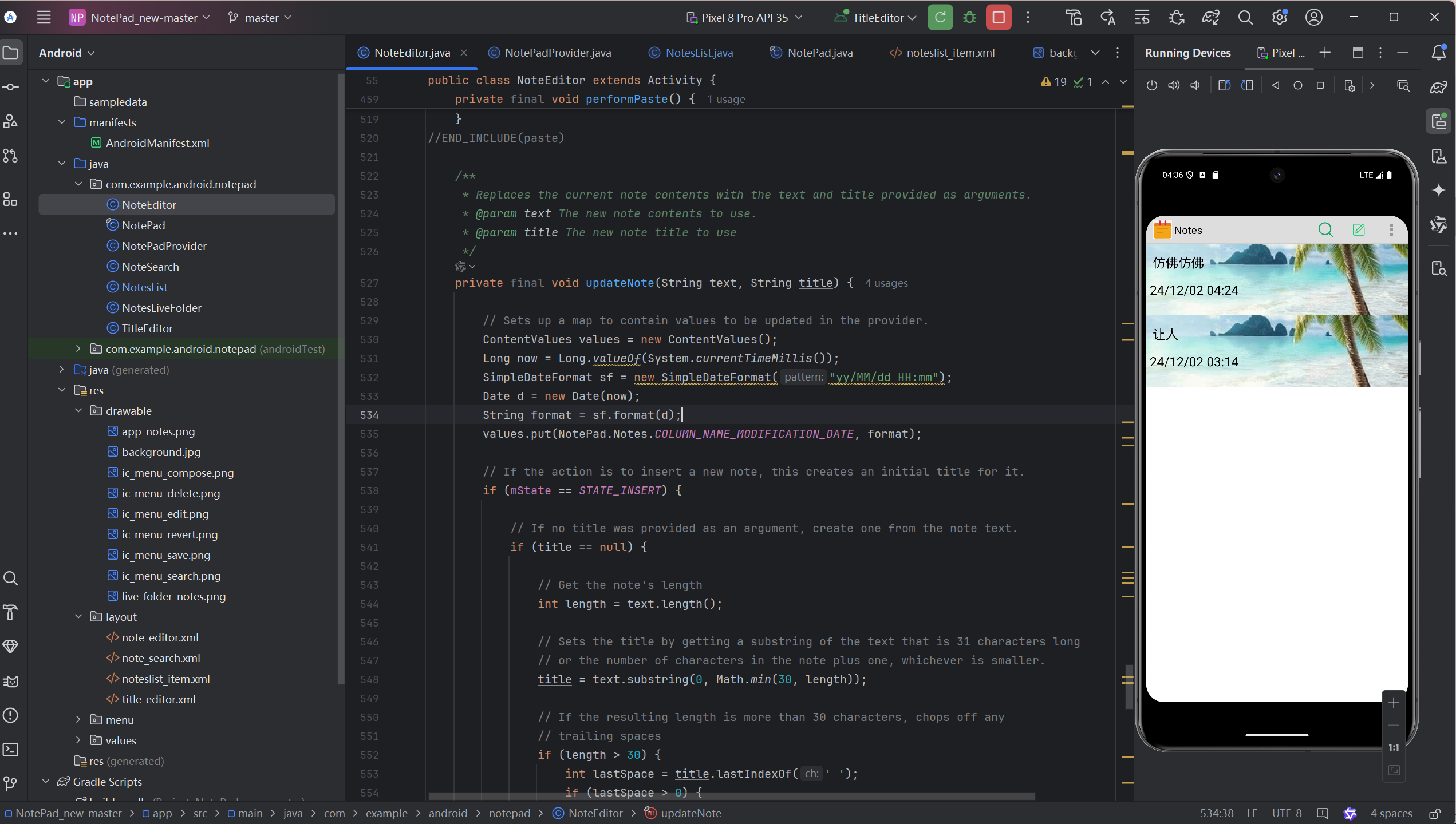
int[] viewIDs = {

android.R.id.text1, R.id.text2};

（2）对应的视图以及代码在notelist\_item.xml中完成,要比原先新增一个text2以及布局

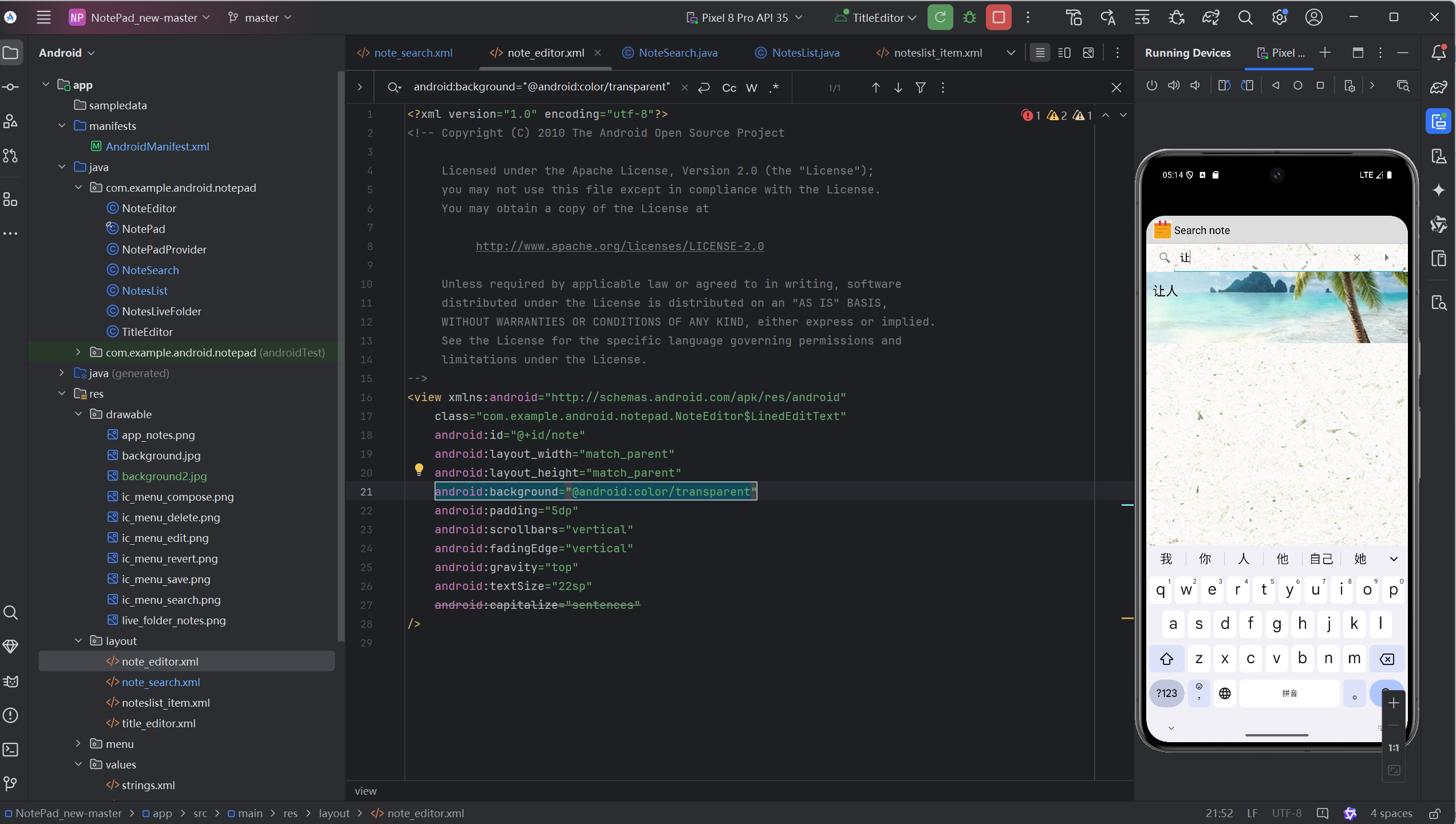


（3）在NoteEditor中修改显示时间的格式



2、搜索

2.1、实现效果



2.2：实现过程

（1）在list\_options\_menu.xml中添加搜索的按钮

<item

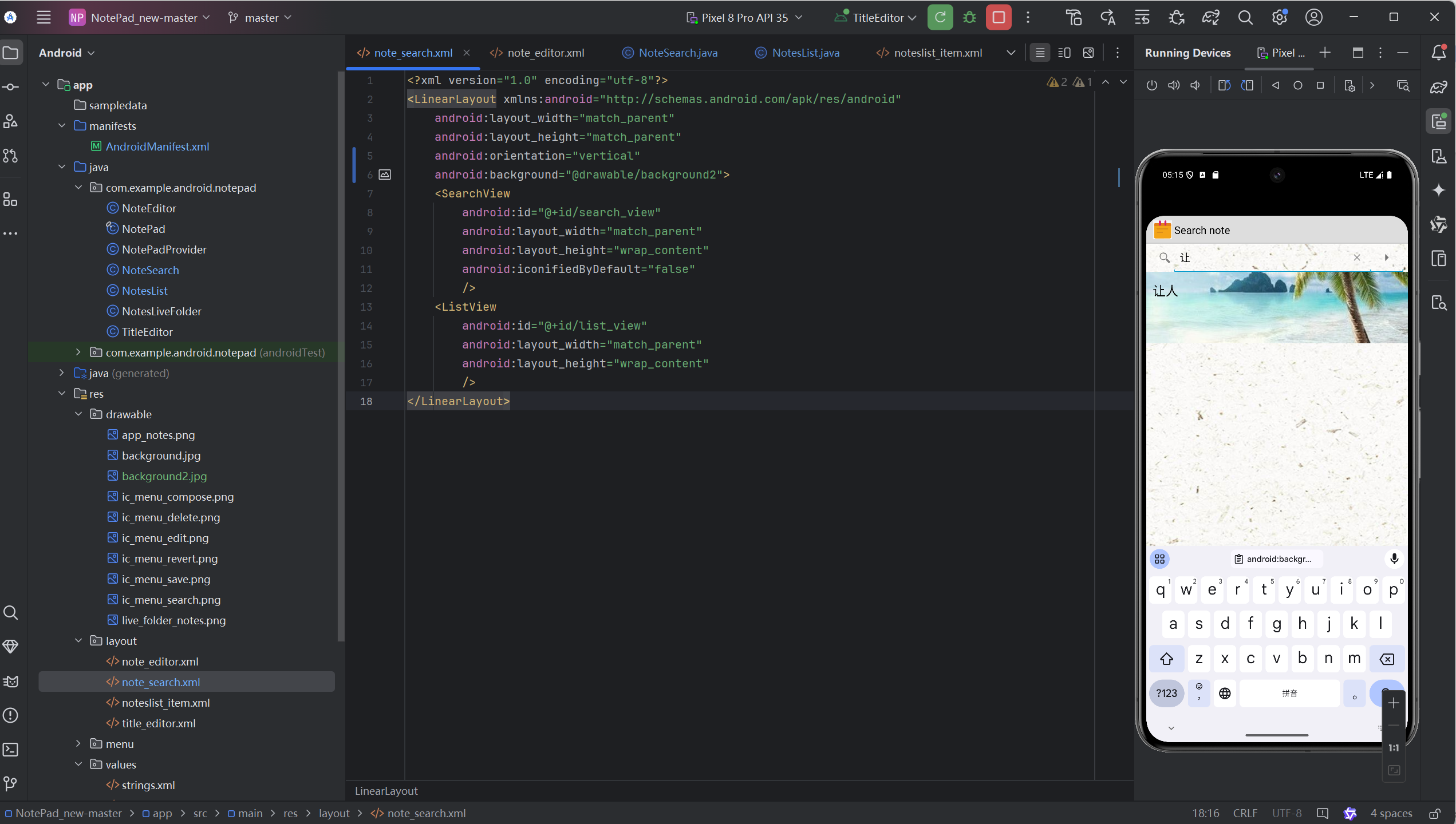
android:id="@+id/menu\_search"

android:icon="@drawable/ic\_menu\_search"

android:title="@string/menu\_search"

android:showAsAction="always" />

（2）创建一个笔记搜索的布局文件：note\_search.xml



(3)新建note\_search类实现查询功能

package com.example.android.notepad;  
  
import android.app.Activity;  
import android.content.Intent;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.os.Bundle;  
import android.widget.ListView;  
import android.widget.SearchView;  
import android.widget.SimpleCursorAdapter;  
import android.widget.Toast;  
  
public class NoteSearch extends Activity implements SearchView.OnQueryTextListener  
{  
  
 ListView listView;  
 SQLiteDatabase sqLiteDatabase;  
 */\*\*  
 \* The columns needed by the cursor adapter  
 \*/* private static final String[] *PROJECTION* = new String[]{  
  
 NotePad.Notes.*\_ID*,   
 NotePad.Notes.*COLUMN\_NAME\_TITLE*,   
 NotePad.Notes.*COLUMN\_NAME\_MODIFICATION\_DATE* };  
 @Override  
 public boolean onQueryTextSubmit(String query) {  
  
 Toast.*makeText*(this, "search:"+query, Toast.*LENGTH\_SHORT*).show();  
 return false;  
 }  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*note\_search*);  
 SearchView searchView = findViewById(R.id.*search\_view*);  
 Intent intent = getIntent();  
 if (intent.getData() == null) {  
  
 intent.setData(NotePad.Notes.*CONTENT\_URI*);  
 }  
 listView = findViewById(R.id.*list\_view*);  
 sqLiteDatabase = new NotePadProvider.DatabaseHelper(this).getReadableDatabase();  
 searchView.setSubmitButtonEnabled(true);  
   
 searchView.setQueryHint("search");  
 searchView.setOnQueryTextListener(this);  
  
 }  
 public boolean onQueryTextChange(String string) {  
  
 String selection1 = NotePad.Notes.*COLUMN\_NAME\_TITLE*+" like ? or "+NotePad.Notes.*COLUMN\_NAME\_NOTE*+" like ?";  
 String[] selection2 = {  
 "%"+string+"%","%"+string+"%"};  
 Cursor cursor = sqLiteDatabase.query(  
 NotePad.Notes.*TABLE\_NAME*,  
 *PROJECTION*,   
 selection1,  
 selection2,   
 null,   
 null,   
 NotePad.Notes.*DEFAULT\_SORT\_ORDER* );  
 String[] dataColumns = {  
  
 NotePad.Notes.*COLUMN\_NAME\_TITLE*,  
 NotePad.Notes.*COLUMN\_NAME\_MODIFICATION\_DATE* } ;  
 int[] viewIDs = {  
  
 android.R.id.*text1*,  
 android.R.id.*text2* };  
 SimpleCursorAdapter adapter  
 = new SimpleCursorAdapter(  
 this,   
 R.layout.*noteslist\_item*,   
 cursor,   
 dataColumns,  
 viewIDs  
 );  
 listView.setAdapter(adapter);  
 return true;  
 }  
}

（4）在NotesList类中的onOptionsItemSelected方法中增加搜索按钮的case

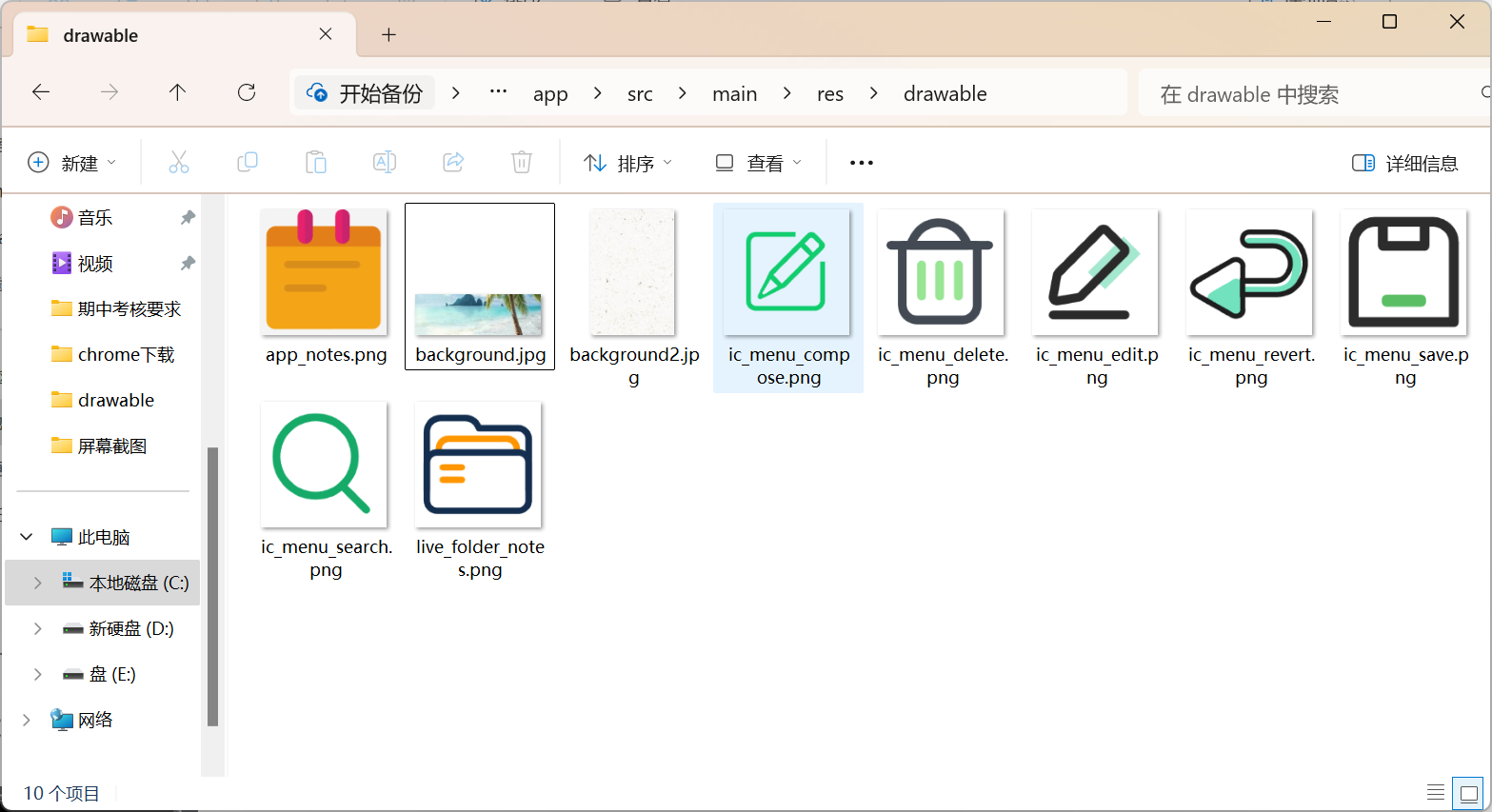
@Override  
public boolean onOptionsItemSelected(MenuItem item) {  
 switch (item.getItemId()) {  
 case R.id.*menu\_add*:  
 /\*  
 \* Launches a new Activity using an Intent. The intent filter for the Activity  
 \* has to have action ACTION\_INSERT. No category is set, so DEFAULT is assumed.  
 \* In effect, this starts the NoteEditor Activity in NotePad.  
 \*/  
 startActivity(new Intent(Intent.*ACTION\_INSERT*, getIntent().getData()));  
 return true;  
 case R.id.*menu\_paste*:  
 /\*  
 \* Launches a new Activity using an Intent. The intent filter for the Activity  
 \* has to have action ACTION\_PASTE. No category is set, so DEFAULT is assumed.  
 \* In effect, this starts the NoteEditor Activity in NotePad.  
 \*/  
 startActivity(new Intent(Intent.*ACTION\_PASTE*, getIntent().getData()));  
 return true;  
 case R.id.*menu\_search*:  
 Intent intent = new Intent(this, NoteSearch.class);  
 this.startActivity(intent);  
 return true;  
 default:  
 return super.onOptionsItemSelected(item);  
 }  
}

（5）在manifest文件中新增一个搜索的activity：<activity android:name=".NoteSearch"   
 android:label="@string/search\_note"  
 android:theme="@android:style/Theme.Holo.Light"/>

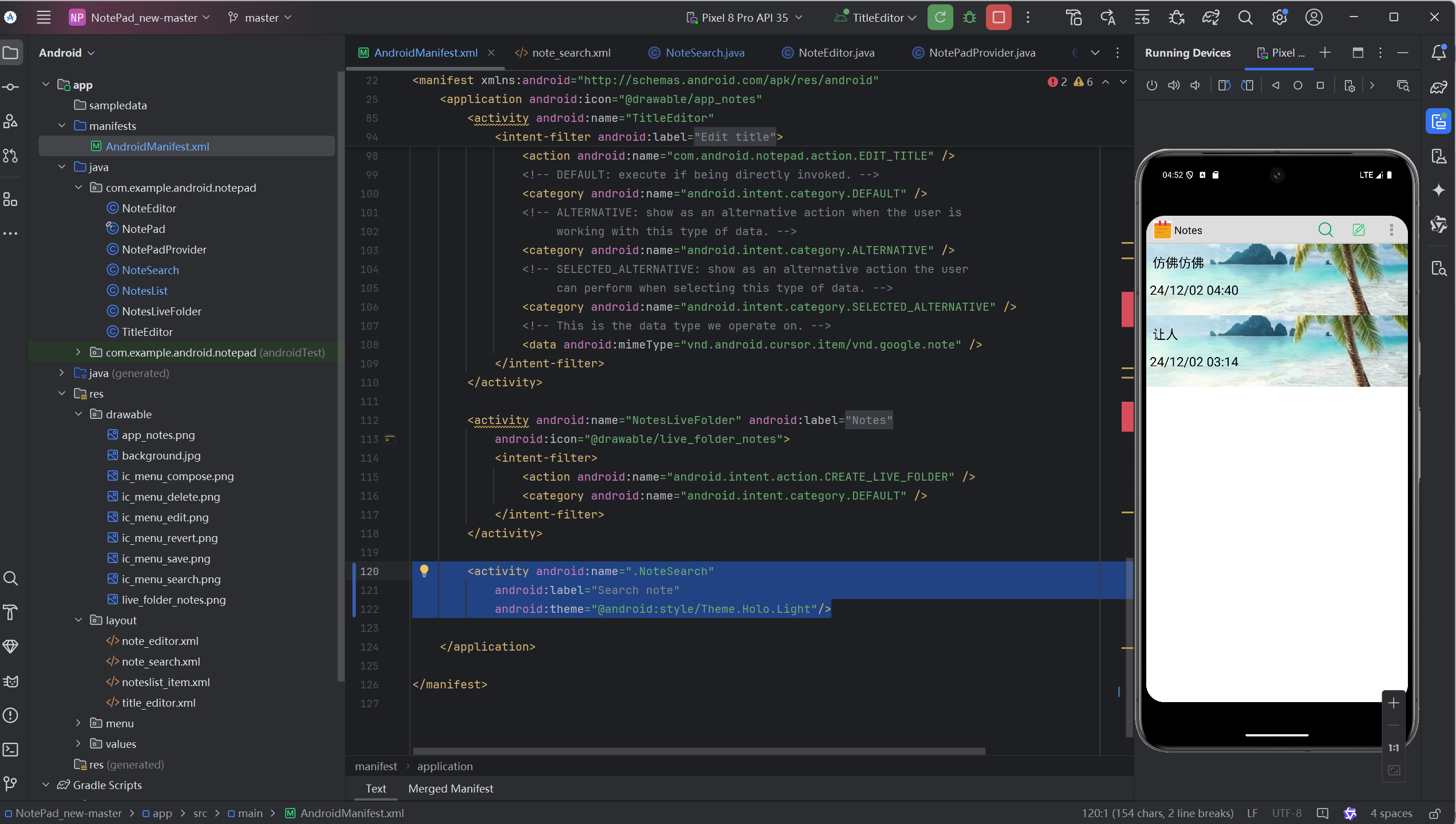
3、ui美化

3.1、效果预览

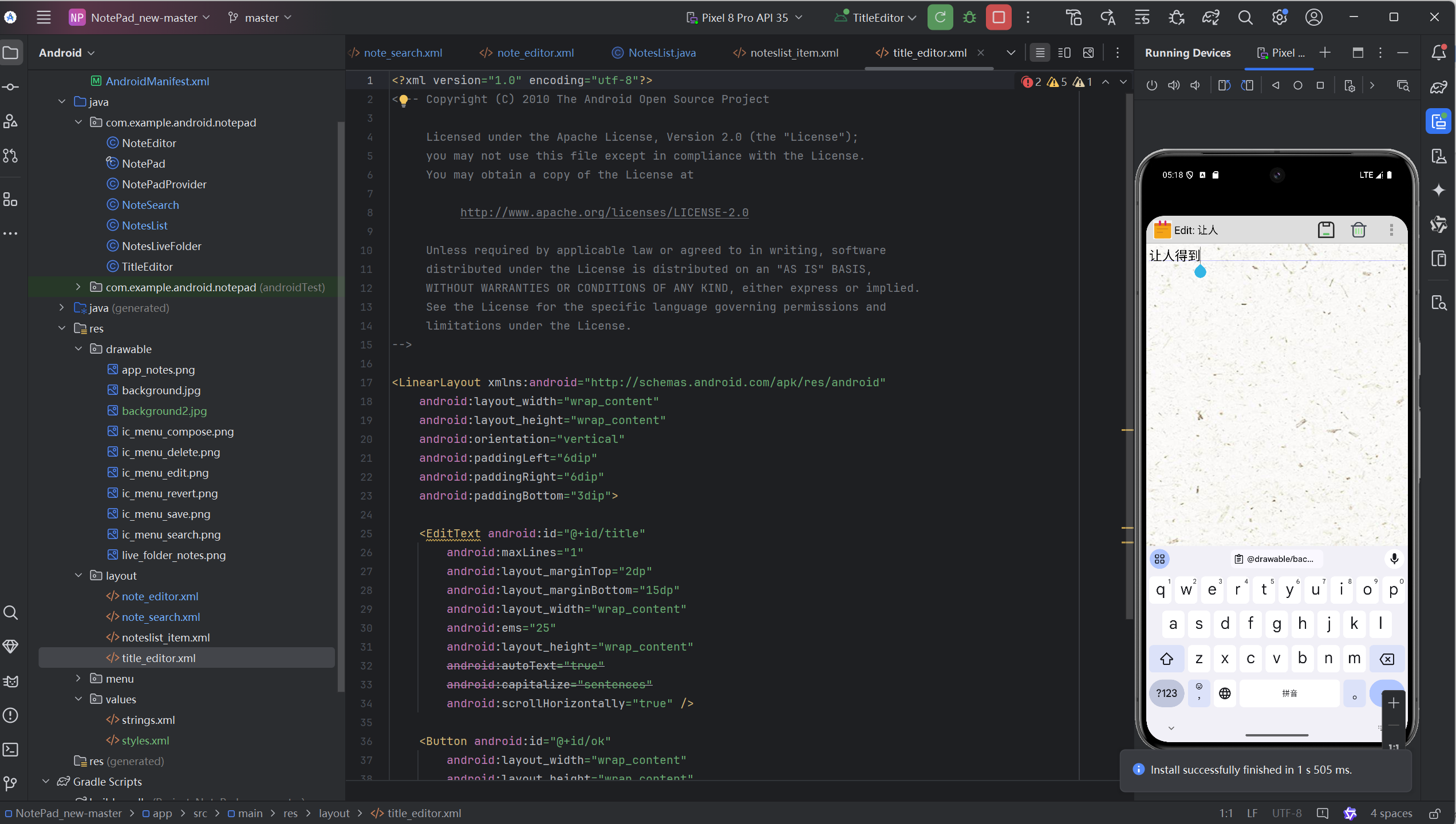
将所有图标，背景换成更为精致的图片

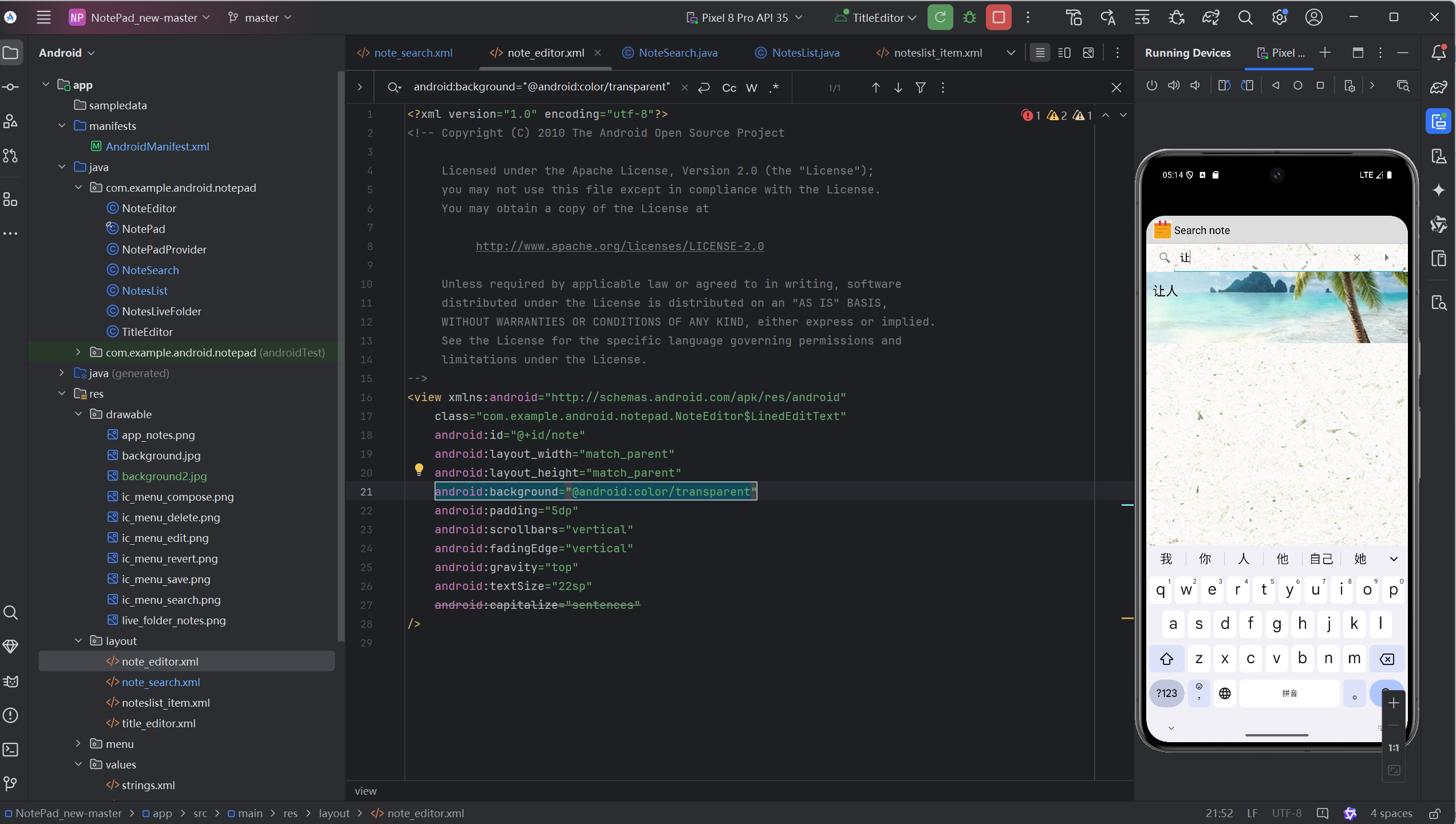


通体是简约的白色，每条笔记列有海滩作为背景



搜索和编辑页内拥有大理石背景





3.2实现过程

（1）在manifest文件中将所有activity的theme改为：@android:style/Theme.Holo.Light

（2）在网上搜索合适的图标以及背景放入drawable文件夹内，将原本丑陋的图标替换掉

（3）将notelist\_item的背景属性改为海滩图片

android:background="@drawable/background"

（4）将搜索以及编辑布局的背景属性改为大理石图片

android:background="@drawable/background2"