android拍摄照片，并将照片保存至本地

|  |
| --- |
| **private static final int *REQUEST\_CODE\_IMAGE\_CAMERA*** = 1;  *//启动相机程序* Intent intent = **new** Intent(**"android.media.action.IMAGE\_CAPTURE"**);  *//将拍摄的图片保存到手机相册* ContentValues values = **new** ContentValues(1); values.put(MediaStore.Images.Media.***MIME\_TYPE***, **"image/jpeg"**); Uri uri = getContentResolver().insert(MediaStore.Images.Media.***EXTERNAL\_CONTENT\_URI***, values);  ((Application)(MainActivity.**this**.getApplicationContext())).setCaptureImage(uri);  intent.putExtra(MediaStore.***EXTRA\_OUTPUT***, uri);*//指定拍照的输出地址* startActivityForResult(intent, ***REQUEST\_CODE\_IMAGE\_CAMERA***); |
| @Override **protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  **super**.onActivityResult(requestCode, resultCode, data);   **if** (requestCode == ***REQUEST\_CODE\_IMAGE\_CAMERA*** && resultCode == ***RESULT\_OK***) {  Uri mPath = ((Application)(MainActivity.**this**.getApplicationContext())).getCaptureImage();  String file = getPath(mPath);  Bitmap bmp = Application.*decodeImage*(file);  startRegister(bmp, file);  } }  **private** String getPath(Uri uri) {  **if** (Build.VERSION.***SDK\_INT*** >= Build.VERSION\_CODES.***KITKAT***) {  **if** (DocumentsContract.*isDocumentUri*(**this**, uri)) {  *// ExternalStorageProvider* **if** (*isExternalStorageDocument*(uri)) {  **final** String docId = DocumentsContract.*getDocumentId*(uri);  **final** String[] split = docId.split(**":"**);  **final** String type = split[0];   **if** (**"primary"**.equalsIgnoreCase(type)) {  **return** Environment.*getExternalStorageDirectory*() + **"/"** + split[1];  }   *//* ***TODO handle non-primary volumes*** } **else if** (*isDownloadsDocument*(uri)) {   **final** String id = DocumentsContract.*getDocumentId*(uri);  **final** Uri contentUri = ContentUris.*withAppendedId*(  Uri.*parse*(**"content://downloads/public\_downloads"**), Long.*valueOf*(id));   **return** *getDataColumn*(**this**, contentUri, **null**, **null**);  } **else if** (*isMediaDocument*(uri)) {  **final** String docId = DocumentsContract.*getDocumentId*(uri);  **final** String[] split = docId.split(**":"**);  **final** String type = split[0];   Uri contentUri = **null**;  **if** (**"image"**.equals(type)) {  contentUri = MediaStore.Images.Media.***EXTERNAL\_CONTENT\_URI***;  } **else if** (**"video"**.equals(type)) {  contentUri = MediaStore.Video.Media.***EXTERNAL\_CONTENT\_URI***;  } **else if** (**"audio"**.equals(type)) {  contentUri = MediaStore.Audio.Media.***EXTERNAL\_CONTENT\_URI***;  }   **final** String selection = **"\_id=?"**;  **final** String[] selectionArgs = **new** String[] {  split[1]  };   **return** *getDataColumn*(**this**, contentUri, selection, selectionArgs);  }  }  }  String[] proj = { MediaStore.Images.Media.***DATA*** };  Cursor actualimagecursor = **this**.getContentResolver().query(uri, proj, **null**, **null**, **null**);  **int** actual\_image\_column\_index = actualimagecursor.getColumnIndexOrThrow(MediaStore.Images.Media.***DATA***);  actualimagecursor.moveToFirst();  String img\_path = actualimagecursor.getString(actual\_image\_column\_index);  String end = img\_path.substring(img\_path.length() - 4);  **if** (0 != end.compareToIgnoreCase(**".jpg"**) && 0 != end.compareToIgnoreCase(**".png"**)) {  **return null**;  }  **return** img\_path; } |