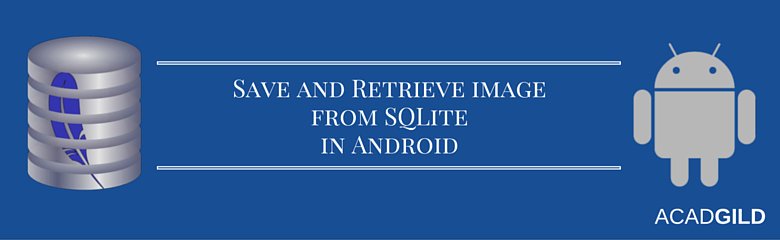
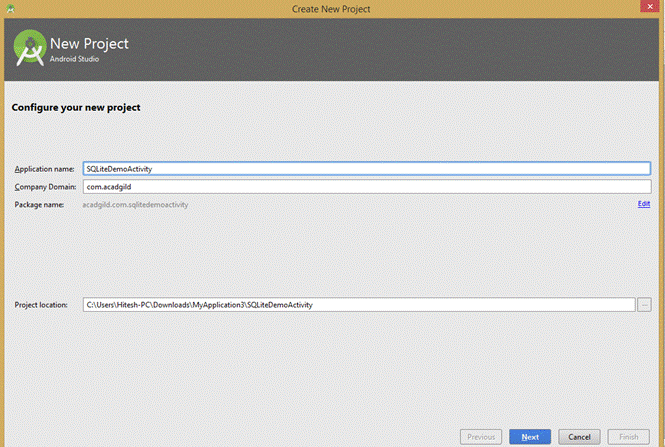
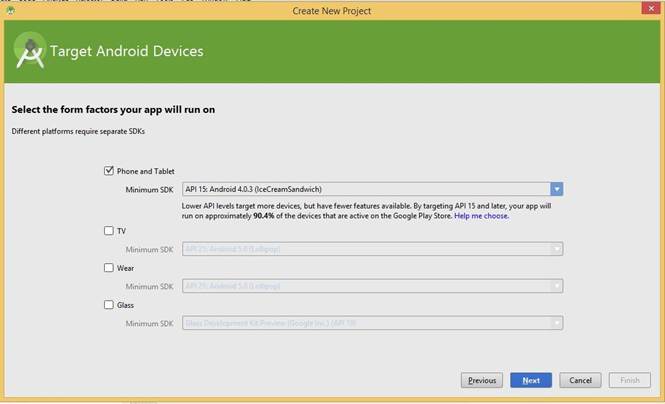
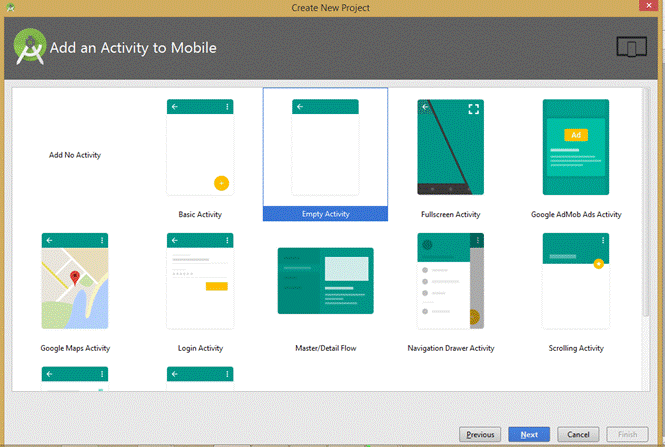
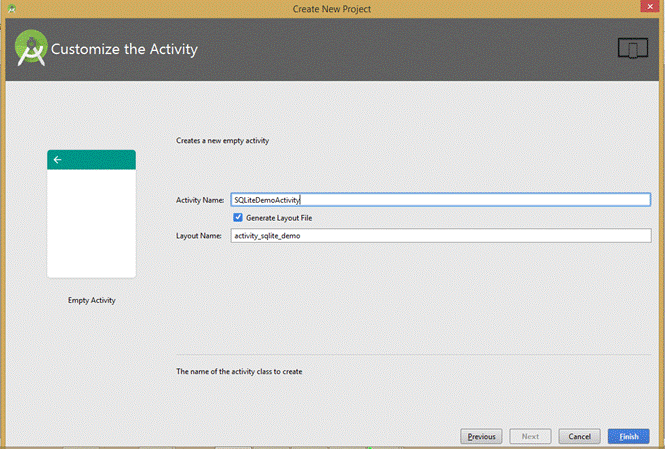
**How to Store and Retrieve Image from SQLite in Android**

[[](https://acadgild.com/blog/author/pushpa) **Pushpa**](https://acadgild.com/blog/author/pushpa) July 27, 2017

[12](https://acadgild.com/blog/save-retrieve-image-sqlite-database-android#comments) 37,244



This blog is for beginners who wants to work with SQLite for storing images & retrieving the same in Android device.  
Here, in this blog, we will be inserting and displaying an image from SQLite database.  
**SQLite:**SQLite is an open source SQL Database in local devices. It stores data to a text file onto a device. SQLite is a light weight database which comes inbuilt with [Android OS.](https://acadgild.com/web-development/android-development-training-certification)  
Let’s go through each and every step to acquire the required results  
1. Create a New Project and give a name as ‘SQLiteDemoActivity’ and click Next.  
  
2. Select the form factor on which you want to run your app and click on Next.  
  
3. Add a blank activity to Mobile and click on Next.  
  
4. Customize your activity and click on Finish.  
  
5. Building “SQLiteDemoActivity” Gradle Project Info.  
  
6. In activity\_main.[XML](https://acadgild.com/web-development/android-development-training-certification), Add one ListView.

<ListView

android:id="@+id/list"

android:layout\_width="fill\_parent"

android:layout\_height="0dp"

android:layout\_weight="0.55" >

</ListView>

7. Add one more layout named as screen\_list.xml and add ImageView and TextView on it.

[Get Skilled in Android Development](https://acadgild.com/mobile-app-development/android-development-training-certification)

<ImageView

android:id="@+id/imgIcon"

android:layout\_width="0dp"

android:layout\_height="100dp"

android:layout\_weight="0.71"

android:gravity="center\_vertical" />

<TextView

android:id="@+id/txtTitle"

android:layout\_width="80dp"

android:layout\_height="fill\_parent"

android:gravity="center\_vertical"

android:textSize="14dp"

android:layout\_marginLeft="7dp" />

8. Add a code in MainActivity.java for Image Bitmap and store into the SQLite.

DataBaseHandler db = new DataBaseHandler(this);

// get image from drawable

Bitmap image = BitmapFactory.decodeResource(getResources(),

R.drawable.android);

// convert bitmap to byte

ByteArrayOutputStream stream = new ByteArrayOutputStream();

image.compress(Bitmap.CompressFormat.JPEG, 100, stream);

byte imageInByte[] = stream.toByteArray();

// Inserting Contacts

Log.d("Insert: ", "Inserting ..");

db.addContact(new Contact("Android", imageInByte));

// display main List view bcard and contact name

// Reading all contacts from database

List<Contact> contacts = db.getAllContacts();

for (Contact cn : contacts) {

String log = "ID:" + cn.getID() + " Name: " + cn.getName()

+ " ,Image: " + cn.getImage();

// Writing Contacts to log

Log.d("Result: ", log);

//add contacts data in arrayList

imageArry.add(cn);

}

adapter = new ContactImageAdapter(this, R.layout.screen\_list,

imageArry);

ListView dataList = (ListView) findViewById(R.id.list);

dataList.setAdapter(adapter);

}

9. Create a constructor Contact.

// getting ID

public int getID() {

return this.\_id;

}

// setting id

public void setID(int keyId) {

this.\_id = keyId;

}

You have to add Name and Image like above code in **Get()** and **Set()**.  
10. Create an Adapter for getting a view and set an Image in List.

ArrayList<Contact> data=new ArrayList<Contact>();

public ContactImageAdapter(Context context, int layoutResourceId, ArrayList<Contact> data) {

super(context, layoutResourceId, data);

this.layoutResourceId = layoutResourceId;

this.context = context;

this.data = data;

}

Add a code for getView() below onCreate().

@Override

public View getView(int position, View convertView, ViewGroup parent) {

View row = convertView;

ImageHolder holder = null;

if(row == null)

{

LayoutInflater inflater = ((Activity)context).getLayoutInflater();

row = inflater.inflate(layoutResourceId, parent, false);

holder = new ImageHolder();

holder.txtTitle = (TextView)row.findViewById(R.id.txtTitle);

holder.imgIcon = (ImageView)row.findViewById(R.id.imgIcon);

row.setTag(holder);

}

else

{

holder = (ImageHolder)row.getTag();

}

Contact picture = data.get(position);

holder.txtTitle.setText(picture.\_name);

//convert byte to bitmap take from contact class

byte[] outImage=picture.\_image;

ByteArrayInputStream imageStream = new ByteArrayInputStream(outImage);

Bitmap theImage = BitmapFactory.decodeStream(imageStream);

holder.imgIcon.setImageBitmap(theImage);

return row;

}  
11. Create a DatabaseHandler class for Handling a Database. For this, you should extend SQLiteOpenHandler class.  
In **onCreate()**, create a Table consisting of 3 columns – id, name, photo/image  
**// Creating Tables**

@Override

public void onCreate(SQLiteDatabase db) {

String CREATE\_CONTACTS\_TABLE = "CREATE TABLE " + TABLE\_CONTACTS + "("

+ KEY\_ID + " INTEGER PRIMARY KEY," + KEY\_NAME + " TEXT,"

+ KEY\_IMAGE + " BLOB" + ")";

db.execSQL(CREATE\_CONTACTS\_TABLE);

}

In **onUpgrade()**, upgrade a Table

// Upgrading database

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

// Drop older table if existed

db.execSQL("DROP TABLE IF EXISTS " + TABLE\_CONTACTS);

// Create tables again

onCreate(db);

}

**Adding a new Contact**

public void addContact(Contact contact) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues values = new ContentValues();

values.put(KEY\_NAME, contact.\_name); // Contact Name

values.put(KEY\_IMAGE, contact.\_image); // Contact Phone

// Inserting Row

db.insert(TABLE\_CONTACTS, null, values);

db.close(); // Closing database connection

}

**Getting a single Contact**

Contact getContact(int id) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = db.query(TABLE\_CONTACTS, new String[] { KEY\_ID,

KEY\_NAME, KEY\_IMAGE }, KEY\_ID + "=?",

new String[] { String.valueOf(id) }, null, null, null, null);

if (cursor != null)

cursor.moveToFirst();

Contact contact = new Contact(Integer.parseInt(cursor.getString(0)),

cursor.getString(1), cursor.getBlob(1));

// return contact

return contact;

}

**Getting All Contacts**

public List<Contact> getAllContacts() {

List<Contact> contactList = new ArrayList<Contact>();

// Select All Query

String selectQuery = "SELECT \* FROM contacts ORDER BY name";

SQLiteDatabase db = this.getWritableDatabase();

Cursor cursor = db.rawQuery(selectQuery, null);

// looping through all rows and adding to list

if (cursor.moveToFirst()) {

do {

Contact contact = new Contact();

contact.setID(Integer.parseInt(cursor.getString(0)));

contact.setName(cursor.getString(1));

contact.setImage(cursor.getBlob(2));

// Adding contact to list

contactList.add(contact);

} while (cursor.moveToNext());

}

// close inserting data from database

db.close();

// return contact list

return contactList;

}

**Updating Single Contact**

public int updateContact(Contact contact) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues values = new ContentValues();

values.put(KEY\_NAME, contact.getName());

values.put(KEY\_IMAGE, contact.getImage());

// updating row

return db.update(TABLE\_CONTACTS, values, KEY\_ID + " = ?",

new String[] { String.valueOf(contact.getID()) });

}

**Deleting Single Contact**

public void deleteContact(Contact contact) {

SQLiteDatabase db = this.getWritableDatabase();

db.delete(TABLE\_CONTACTS, KEY\_ID + " = ?",

new String[] { String.valueOf(contact.getID()) });

db.close();

}

**Getting Contact Counts**

public int getContactsCount() {

String countQuery = "SELECT \* FROM " + TABLE\_CONTACTS;

SQLiteDatabase db = this.getReadableDatabase();

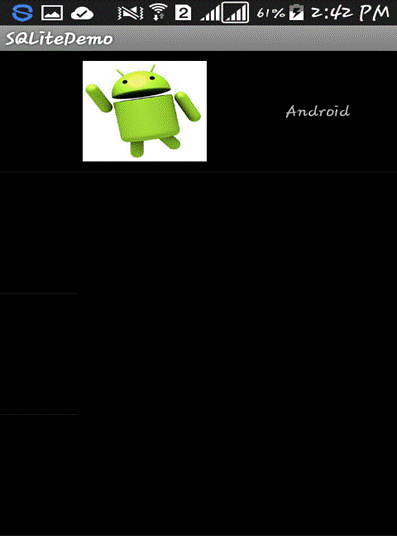
Cursor cursor = db.rawQuery(countQuery, null);

cursor.close();

// return count

return cursor.getCount();

}

12. Run your app and you will get an output like below:  


**Source Code:** [Save and Retrieve Image in Android](https://github.com/hiteshbpatel/Android_Blog_Projects/tree/master/Android-SQLite-master)  
Keep visiting our site [www.acadgild.com](https://acadgild.com/) for more updates on Android and other technologies.