## Task 7.1P

Video Link: <https://youtu.be/vG1cwPi3YiE>  
  
**Source Code**  
  
**MainActivity.java**

package com.example.lostfoundapp;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 private Button buttonCreateAdvert, buttonShowItems;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 buttonCreateAdvert = findViewById(R.id.*button\_create\_advert*);  
 buttonShowItems = findViewById(R.id.*button\_show\_items*);  
  
 buttonCreateAdvert.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(MainActivity.this, AddItemActivity.class);  
 startActivity(intent);  
 }  
 });  
  
 buttonShowItems.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(MainActivity.this, ListItemsActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

**AddItemActivity.java**

package com.example.lostfoundapp;  
  
import android.content.ContentValues;  
import android.database.sqlite.SQLiteDatabase;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.RadioButton;  
import android.widget.RadioGroup;  
import android.widget.Toast;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class AddItemActivity extends AppCompatActivity {  
  
 private EditText editTextName, editTextPhone, editTextDescription, editTextDate, editTextLocation;  
 private RadioGroup radioGroupType;  
 private Button buttonSave;  
 private DatabaseHelper dbHelper;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_add\_item*);  
  
 dbHelper = new DatabaseHelper(this);  
  
 radioGroupType = findViewById(R.id.*radioGroupType*);  
 editTextName = findViewById(R.id.*editTextName*);  
 editTextPhone = findViewById(R.id.*editTextPhone*);  
 editTextDescription = findViewById(R.id.*editTextDescription*);  
 editTextDate = findViewById(R.id.*editTextDate*);  
 editTextLocation = findViewById(R.id.*editTextLocation*);  
 buttonSave = findViewById(R.id.*buttonSave*);  
  
 buttonSave.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 int selectedTypeId = radioGroupType.getCheckedRadioButtonId();  
 String name = editTextName.getText().toString().trim();  
 String phone = editTextPhone.getText().toString().trim();  
 String description = editTextDescription.getText().toString().trim();  
 String date = editTextDate.getText().toString().trim();  
 String location = editTextLocation.getText().toString().trim();  
  
 if (selectedTypeId == -1) {  
 Toast.*makeText*(AddItemActivity.this, "Please select Lost or Found", Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
 if (name.isEmpty()) {  
 editTextName.setError("Name is required");  
 return;  
 }  
 if (phone.isEmpty()) {  
 editTextPhone.setError("Phone is required");  
 return;  
 }  
 if (description.isEmpty()) {  
 editTextDescription.setError("Description is required");  
 return;  
 }  
 if (date.isEmpty()) {  
 editTextDate.setError("Date is required");  
 return;  
 }  
 if (location.isEmpty()) {  
 editTextLocation.setError("Location is required");  
 return;  
 }  
  
 RadioButton selectedRadio = findViewById(selectedTypeId);  
 String type = selectedRadio.getText().toString();  
  
 saveToDatabase(type, name, phone, description, date, location);  
 }  
 });  
 }  
  
 private void saveToDatabase(String type, String name, String phone, String description, String date, String location) {  
 SQLiteDatabase db = dbHelper.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put(DatabaseHelper.*COLUMN\_TYPE*, type);  
 values.put(DatabaseHelper.*COLUMN\_NAME*, name);  
 values.put(DatabaseHelper.*COLUMN\_PHONE*, phone);  
 values.put(DatabaseHelper.*COLUMN\_DESCRIPTION*, description);  
 values.put(DatabaseHelper.*COLUMN\_DATE*, date);  
 values.put(DatabaseHelper.*COLUMN\_LOCATION*, location);  
  
 long newRowId = db.insert(DatabaseHelper.*TABLE\_NAME*, null, values);  
  
 if (newRowId != -1) {  
 Toast.*makeText*(this, "Advert Saved!", Toast.*LENGTH\_SHORT*).show();  
 finish();  
 } else {  
 Toast.*makeText*(this, "Error saving advert", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
}

**ListItemActivity.java**package com.example.lostfoundapp;  
  
import android.content.Intent;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
import androidx.appcompat.app.AppCompatActivity;  
import java.util.ArrayList;  
  
public class ListItemsActivity extends AppCompatActivity {  
  
 private ListView listView;  
 private DatabaseHelper dbHelper;  
 private ArrayList<String> itemsList;  
 private ArrayList<Integer> itemIds;  
 private ArrayList<String> itemDates;  
 private ArrayList<String> itemLocations;  
 private ArrayList<String> itemDescriptions;  
 private ArrayList<String> itemPhones;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_list\_items*);  
  
 listView = findViewById(R.id.*listViewItems*);  
 dbHelper = new DatabaseHelper(this);  
 itemsList = new ArrayList<>();  
 itemIds = new ArrayList<>();  
 itemDates = new ArrayList<>();  
 itemLocations = new ArrayList<>();  
 itemDescriptions = new ArrayList<>();  
 itemPhones = new ArrayList<>();  
  
 loadItems();  
  
 listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> adapterView, View view, int position, long id) {  
 openViewItemActivity(position);  
 }  
 });  
 }  
  
 private void loadItems() {  
 itemsList.clear();  
 itemIds.clear();  
 itemDates.clear();  
 itemLocations.clear();  
 itemDescriptions.clear();  
 itemPhones.clear();  
  
 SQLiteDatabase db = dbHelper.getReadableDatabase();  
 Cursor cursor = db.query(DatabaseHelper.*TABLE\_NAME*,  
 null, null, null, null, null, null);  
  
 while (cursor.moveToNext()) {  
 int id = cursor.getInt(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_ID*));  
 String type = cursor.getString(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_TYPE*));  
 String name = cursor.getString(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_NAME*));  
 String phone = cursor.getString(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_PHONE*));  
 String description = cursor.getString(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_DESCRIPTION*));  
 String date = cursor.getString(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_DATE*));  
 String location = cursor.getString(cursor.getColumnIndexOrThrow(DatabaseHelper.*COLUMN\_LOCATION*));  
  
 itemsList.add(type + ": " + name);  
 itemIds.add(id);  
 itemDates.add(date);  
 itemLocations.add(location);  
 itemDescriptions.add(description);  
 itemPhones.add(phone);  
 }  
 cursor.close();  
  
 ArrayAdapter<String> adapter = new ArrayAdapter<>(this,  
 android.R.layout.*simple\_list\_item\_1*, itemsList);  
 listView.setAdapter(adapter);  
 }  
  
 private void openViewItemActivity(int position) {  
 Intent intent = new Intent(ListItemsActivity.this, ViewItemActivity.class);  
 intent.putExtra("item\_id", itemIds.get(position));  
 intent.putExtra("item\_name", itemsList.get(position));  
 intent.putExtra("item\_description", itemDescriptions.get(position));  
 intent.putExtra("item\_phone", itemPhones.get(position));  
 intent.putExtra("item\_date", itemDates.get(position));  
 intent.putExtra("item\_location", itemLocations.get(position));  
 startActivityForResult(intent, 1);  
 }  
  
 @Override  
 protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
 if (requestCode == 1 && resultCode == *RESULT\_OK*) {  
 loadItems();  
 }  
 }  
}

**ViewItemActivity.java**  
package com.example.lostfoundapp;  
  
import android.content.Intent;  
import android.database.sqlite.SQLiteDatabase;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import android.widget.Toast;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class ViewItemActivity extends AppCompatActivity {  
  
 private TextView textViewName, textViewDate, textViewLocation;  
 private Button buttonRemove;  
 private int itemId;  
 private DatabaseHelper dbHelper;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_view\_item*);  
  
 dbHelper = new DatabaseHelper(this);  
  
 textViewName = findViewById(R.id.*textViewName*);  
 textViewDate = findViewById(R.id.*textViewDate*);  
 TextView textViewDescription = findViewById(R.id.*textViewDescription*);  
 TextView textViewPhone = findViewById(R.id.*textViewPhone*);  
 textViewLocation = findViewById(R.id.*textViewLocation*);  
 buttonRemove = findViewById(R.id.*buttonRemove*);  
  
 Intent intent = getIntent();  
 if (intent != null) {  
 itemId = intent.getIntExtra("item\_id", -1);  
 String name = intent.getStringExtra("item\_name");  
 String description = intent.getStringExtra("item\_description");  
 String phone = intent.getStringExtra("item\_phone");  
 String date = intent.getStringExtra("item\_date");  
 String location = intent.getStringExtra("item\_location");  
  
 textViewName.setText(name);  
 textViewDate.setText(date);  
 textViewDescription.setText(description);  
 textViewPhone.setText(phone);  
 textViewLocation.setText(location);  
 }  
  
 buttonRemove.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 deleteItem();  
 }  
 });  
 }  
  
 private void deleteItem() {  
 SQLiteDatabase db = dbHelper.getWritableDatabase();  
 db.delete(DatabaseHelper.*TABLE\_NAME*, DatabaseHelper.*COLUMN\_ID* + "=?", new String[]{String.*valueOf*(itemId)});  
 Toast.*makeText*(this, "Item Removed", Toast.*LENGTH\_SHORT*).show();  
 setResult(*RESULT\_OK*);  
 finish();  
 }  
}

**DatabaseHelper.java**

package com.example.lostfoundapp;  
  
import android.content.Context;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
public class DatabaseHelper extends SQLiteOpenHelper {  
  
 public static final String *DATABASE\_NAME* = "LostFoundDB.db";  
 public static final int *DATABASE\_VERSION* = 2;  
  
 public static final String *TABLE\_NAME* = "LostFoundItems";  
 public static final String *COLUMN\_ID* = "\_id";  
 public static final String *COLUMN\_TYPE* = "type";  
 public static final String *COLUMN\_NAME* = "name";  
 public static final String *COLUMN\_PHONE* = "phone";  
 public static final String *COLUMN\_DESCRIPTION* = "description";  
 public static final String *COLUMN\_DATE* = "date";  
 public static final String *COLUMN\_LOCATION* = "location";  
  
 private static final String *CREATE\_TABLE* = "CREATE TABLE " + *TABLE\_NAME* + " (" +  
 *COLUMN\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT, " +  
 *COLUMN\_TYPE* + " TEXT, " +  
 *COLUMN\_NAME* + " TEXT, " +  
 *COLUMN\_PHONE* + " TEXT, " +  
 *COLUMN\_DESCRIPTION* + " TEXT, " +  
 *COLUMN\_DATE* + " TEXT, " +  
 *COLUMN\_LOCATION* + " TEXT)";  
  
 public DatabaseHelper(Context context) {  
 super(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL(*CREATE\_TABLE*);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_NAME*);  
 onCreate(db);  
 }  
}

**Future Directions**

While the Lost & Found App built in this project covers the core functionality, there are several exciting directions it could take in the future.

One major improvement would be adding the ability for users to upload images of lost or found items. Visual details would make it much easier for owners to identify their belongings. This could be achieved by using Android's camera and gallery features, saving images either locally or to a cloud service (Android Developers, 2024).

Introducing user accounts is another key enhancement. Allowing people to sign up and manage their posts would add a layer of security and personalization. Implementing this with something like Firebase Authentication would be straightforward and highly effective (Firebase, 2024a).

Real-time notifications could also add a lot of value. For instance, users could get alerts when a new item is posted nearby. This would involve integrating a notification service like Firebase Cloud Messaging (Firebase, 2024b).

Location-based features are another smart addition. By using GPS and maps, users could filter items by proximity, making it easier to find lost things within their campus, neighbourhood, or city (Android Developers, 2024).

Lastly, moving to a cloud-based database like Firebase Firestore could make the app even more powerful, allowing for real-time updates across devices and better scalability (Firebase, 2024c).

Overall, these future improvements would take the app from a simple listing platform to a dynamic, user-friendly service that genuinely helps people reconnect with their lost belongings.  
  
References:  
  
Android Developers, 2024. Save data using files, databases, or cloud services. [online] Available at: [https://developer.android.com/training/data-storage](https://developer.android.com/training/data-storage%20) [Accessed 12 May 2025].

Firebase, 2024a. Firebase Authentication. [online] Available at: [https://firebase.google.com/products/auth](https://firebase.google.com/products/auth%20) [Accessed 12 May 2025].

Firebase, 2024b. Firebase Cloud Messaging. [online] Available at: [https://firebase.google.com/products/cloud-messaging](https://firebase.google.com/products/cloud-messaging%20) [Accessed 12 May 2025].

Firebase, 2024c. Cloud Firestore. [online] Available at: [https://firebase.google.com/products/firestore](https://firebase.google.com/products/firestore%20) [Accessed 12 May 2025].