Q.2] Create table Customer (id, name, address, ph_no). Create Application for performing the following operation on the table. (Using SQLite database). i] Insert new customer details (At least 5 records). ii] Show all the customer details.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <ListView
        android:id="@+id/list_view_customers"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@+id/btn_add_customer"
        android:dividerHeight="1dp" />

    <Button
        android:id="@+id/btn_add_customer"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="16dp"
        android:text="Add Customer" />

</RelativeLayout>
```

```
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class CustomerDatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "customer_database";
    private static final int DATABASE_VERSION = 1;

    // Table name and column names
    private static final String TABLE_NAME = "Customer";
    private static final String COLUMN_ID = "id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_ADDRESS = "address";
    private static final String COLUMN_PHONE_NUMBER = "ph_no";

    public CustomerDatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        // Create the Customer table
        String createTableQuery = "CREATE TABLE " + TABLE_NAME + "(" +
                COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT," +
```

```
            COLUMN_NAME + " TEXT," +
            COLUMN_ADDRESS + " TEXT," +
            COLUMN_PHONE_NUMBER + " TEXT" +
            ")";
    db.execSQL(createTableQuery);
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    // Drop older table if existed and recreate
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
    onCreate(db);
}

// Method to add a new customer
public void addCustomer(Customer customer) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COLUMN_NAME, customer.getName());
    values.put(COLUMN_ADDRESS, customer.getAddress());
    values.put(COLUMN_PHONE_NUMBER, customer.getPhoneNumber());
    db.insert(TABLE_NAME, null, values);
    db.close();
}

// Method to get all customers
public List<Customer> getAllCustomers() {
```

```
        List<Customer> customerList = new ArrayList<>();
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
        if (cursor.moveToFirst()) {
            do {
                Customer customer = new Customer();
                customer.setId(cursor.getInt(cursor.getColumnIndex(COLUMN_ID)));
                customer.setName(cursor.getString(cursor.getColumnIndex(COLUMN_NAME)));
                customer.setAddress(cursor.getString(cursor.getColumnIndex(COLUMN_ADDRESS)));

customer.setPhoneNumber(cursor.getString(cursor.getColumnIndex(COLUMN_PHONE_NUMB
ER)));
                customerList.add(customer);
            } while (cursor.moveToNext());
        }
        cursor.close();
        db.close();
        return customerList;
    }
}
```