-------------------DBHelper----------------------------------

public class DBHelper extends SQLiteOpenHelper {  
  
 public DBHelper(@Nullable Context context) {  
 super(context, "KTTK5", null, 1);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase sqLiteDatabase) {  
 sqLiteDatabase.execSQL("create table Authors(" +  
 "id integer primary key," +  
 "name text," +  
 "address text," +  
 "email text)");  
 sqLiteDatabase.execSQL("create table Books(" +  
 "id integer primary key," +  
 "title text," +  
 "id\_author integer not null constraint id references " +  
 "Authors(id) on delete cascade on update cascade)");  
 }  
 @Override  
 public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
 sqLiteDatabase.execSQL("drop table if exists Books");  
 sqLiteDatabase.execSQL("drop table if exists Authors");  
 onCreate(sqLiteDatabase);  
 }  
 //Author  
 public int insertAuthor(Author author){  
 SQLiteDatabase db=this.getWritableDatabase();  
 ContentValues values= new ContentValues();  
 values.put("id",author.getId());  
 values.put("name",author.getName());  
 values.put("address", author.getAddress());  
 values.put("email", author.getEmail());  
 int re=(int)db.insert("Authors", null, values);  
 db.close();  
 return re;  
 }  
 public ArrayList<Author> getAllAuthors(){  
 ArrayList<Author> list=new ArrayList<>();  
 SQLiteDatabase db=this.getReadableDatabase();  
 Cursor cursor=db.rawQuery("select \* from authors",null);  
 if(cursor!=null){  
 cursor.moveToFirst();  
 while (!cursor.isAfterLast()){  
 list.add(new Author(cursor.getInt(0),  
 cursor.getString(1),  
 cursor.getString(2),  
 cursor.getString(3)));  
 cursor.moveToNext();  
 }  
 cursor.close();  
 db.close();  
 }  
 return list;  
 }  
 public Author getAuthorById(int id){  
 Author author= new Author();  
 SQLiteDatabase db=this.getReadableDatabase();  
 Cursor cursor=db.rawQuery("select \* from authors where id="+id,null);  
 if (cursor!=null){  
 cursor.moveToFirst();  
 author.setId(cursor.getInt(0));  
 author.setName(cursor.getString(1));  
 author.setAddress(cursor.getString(2));  
 author.setEmail(cursor.getString(3));  
 cursor.close();  
 db.close();  
 }  
 return author;  
 }  
 public Author getAuthorByName(String name){  
 Author author=new Author();  
 SQLiteDatabase db=this.getReadableDatabase();  
 Cursor cursor=db.rawQuery("select \* from authors where name='"+name+"'", null);  
 if (cursor!=null){  
 cursor.moveToFirst();  
 author.setId(cursor.getInt(0));  
 author.setName(cursor.getString(1));  
 author.setAddress(cursor.getString(2));  
 author.setEmail(cursor.getString(3));  
 cursor.close();  
 db.close();  
 }  
 return author;  
 }  
 public Integer deleteAuthor(int id){  
 SQLiteDatabase db=this.getWritableDatabase();  
 Integer re=db.delete("Authors", "id=?",new String[]{String.*valueOf*(id)});  
 return re;  
 }  
 public boolean updateAuthor(Author author){  
 SQLiteDatabase db=this.getWritableDatabase();  
 ContentValues values= new ContentValues();  
 values.put("id",author.getId());  
 values.put("name",author.getName());  
 values.put("address", author.getAddress());  
 values.put("email", author.getEmail());  
 db.update("Authors",values, "id=?", new String[]{String.*valueOf*(author.getId())});  
 return true;  
 }  
 //Book  
 public int insertBook(Book book){  
 SQLiteDatabase db=this.getWritableDatabase();  
 ContentValues values= new ContentValues();  
 values.put("id",book.getId());  
 values.put("title",book.getTitle());  
 values.put("id\_author", book.getAuthor().getId());  
 int re=(int)db.insert("Books", null, values);  
 db.close();  
 return re;  
 }

public ArrayList<Book> getAllBooks(){  
 ArrayList<Book> list=new ArrayList<>();  
 SQLiteDatabase db=this.getReadableDatabase();  
 Cursor cursor=db.rawQuery("select \* from books",null);  
 if(cursor!=null){  
 cursor.moveToFirst();  
 while (!cursor.isAfterLast()){  
 int id\_Author=cursor.getInt(2);  
 Author author=getAuthorById(id\_Author);  
 list.add(new Book(cursor.getInt(0),  
 cursor.getString(1),  
 author));  
 cursor.moveToNext();  
 }  
 cursor.close();  
 db.close();  
 }  
 return list;  
 }  
 public Book getBookById(int id){  
 Book book= new Book();  
 SQLiteDatabase db=this.getReadableDatabase();  
 Cursor cursor=db.rawQuery("select \* from books where id="+id, null);  
 if (cursor!=null){  
 cursor.moveToFirst();  
 book.setId(cursor.getInt(0));  
 book.setTitle(cursor.getString(1));  
 int id\_Author=cursor.getInt(2);  
 Author author=getAuthorById(id\_Author);  
 book.setAuthor(author);  
 cursor.close();  
 db.close();  
 }  
 return book;  
 }  
 public Integer deleteBook(int id){  
 SQLiteDatabase db=this.getWritableDatabase();  
 Integer re=db.delete("Books", "id=?",new String[]{String.*valueOf*(id)});  
 return re;  
 }  
 public boolean updateBook(Book book){  
 SQLiteDatabase db=this.getWritableDatabase();  
 ContentValues values= new ContentValues();  
 values.put("id",book.getId());  
 values.put("title",book.getTitle());  
 values.put("id\_author", book.getAuthor().getId());  
 db.update("books",values, "id=?", new String[]{String.*valueOf*(book.getId())});  
 return true;  
 }

------------------------------------Author-------------------------------------

public class MainActivity extends AppCompatActivity {  
  
 EditText ed\_id, ed\_name, ed\_address, ed\_email;  
 GridView gridView;  
 DBHelper dbHelper;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 ed\_id=findViewById(R.id.*ed\_idAuthor*);  
 ed\_name=findViewById(R.id.*ed\_nameAuthor*);  
 ed\_address=findViewById(R.id.*ed\_addressAuthor*);  
 ed\_email=findViewById(R.id.*ed\_emailAuthor*);  
 gridView=findViewById(R.id.*grid\_ListAuthor*);  
 dbHelper=new DBHelper(this);  
 select();  
 gridView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {  
 Integer id=Integer.*parseInt*(gridView.getItemAtPosition(i).toString());  
 Author author=dbHelper.getAuthorById(id);  
 ed\_id.setText(String.*valueOf*(author.getId()));  
 ed\_name.setText(author.getName().toString());  
 ed\_address.setText(author.getAddress().toString());  
 ed\_email.setText(author.getEmail().toString());  
 }  
 });  
 Button btn\_them=findViewById(R.id.*btn\_saveAuthor*);  
 btn\_them.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (ed\_id.getText().toString().equals("")||  
 ed\_name.getText().toString().equals("")||  
 ed\_address.getText().toString().equals("")||  
 ed\_email.getText().toString().equals("")){  
 Toast.*makeText*(getApplicationContext(), "Nhap du thong tin", Toast.*LENGTH\_SHORT*).show();  
 }else {  
 Author author= new Author();  
 author.setId(Integer.*parseInt*(ed\_id.getText().toString()));  
 author.setName(ed\_name.getText().toString());  
 author.setAddress(ed\_address.getText().toString());  
 author.setEmail(ed\_email.getText().toString());  
  
 if (dbHelper.insertAuthor(author)>0){  
 Toast.*makeText*(getApplicationContext(), "Luu thanh cong", Toast.*LENGTH\_SHORT*).show();  
 lamRong();  
 select();  
 }else Toast.*makeText*(getApplicationContext(), "Luu that bai", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 Button btn\_xoa=findViewById(R.id.*btn\_deleteAuthor*);  
 btn\_xoa.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (ed\_id.getText().toString().equals("")){  
 Toast.*makeText*(getApplicationContext(), "Chon author can xoa", Toast.*LENGTH\_SHORT*).show();  
 }else {  
 if (dbHelper.deleteAuthor(Integer.*parseInt*(ed\_id.getText().toString()))>0){  
 Toast.*makeText*(getApplicationContext(), "xoa thanh cong author", Toast.*LENGTH\_SHORT*).show();  
 lamRong();  
 select();  
 }  
 }  
 }  
 });  
 Button btn\_sua=findViewById(R.id.*btn\_updateAuthor*);  
 btn\_sua.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (ed\_id.getText().toString().equals("")){  
 Toast.*makeText*(getApplicationContext(), "Chon author can xoa", Toast.*LENGTH\_SHORT*).show();  
 }else {  
 Author author=dbHelper.getAuthorById(Integer.*parseInt*(ed\_id.getText().toString()));  
 author.setName(ed\_name.getText().toString());  
 author.setAddress(ed\_address.getText().toString());  
 author.setEmail(ed\_email.getText().toString());  
 if (dbHelper.updateAuthor(author)){  
 Toast.*makeText*(getApplicationContext(), "Cap nhat thanh cong", Toast.*LENGTH\_SHORT*).show();  
 lamRong();  
 select();  
 }else Toast.*makeText*(getApplicationContext(), "Cap nhat that bai", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 Button btn\_select=findViewById(R.id.*btn\_selectAuthor*);  
 btn\_select.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 select();  
 }  
 });  
 Button btn\_thoat=findViewById(R.id.*btn\_thoat*);  
 btn\_thoat.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });  
 Button btn\_book=findViewById(R.id.*btn\_book*);  
 btn\_book.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent=new Intent(MainActivity.this, BookActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
 private void select(){  
 if (!ed\_id.getText().toString().equals("")) {  
 Author author;  
 ArrayList<String> stringArrayList= new ArrayList<>();  
 author=dbHelper.getAuthorById(Integer.*parseInt*(ed\_id.getText().toString()));  
 stringArrayList.add(author.getId()+"");  
 stringArrayList.add(author.getName());  
 stringArrayList.add(author.getAddress());  
 stringArrayList.add(author.getEmail());  
 ArrayAdapter<String> adapter= new ArrayAdapter<>(MainActivity.this,  
 android.R.layout.*simple\_list\_item\_1*,stringArrayList);  
 gridView.setAdapter(adapter);  
 }else {  
 ArrayList<Author> list=dbHelper.getAllAuthors();  
 ArrayList<String> stringArrayList= new ArrayList<>();  
 for(Author author:list){  
 stringArrayList.add(author.getId()+"");  
 stringArrayList.add(author.getName());  
 stringArrayList.add(author.getAddress());  
 stringArrayList.add(author.getEmail());  
 }  
 ArrayAdapter<String> adapter= new ArrayAdapter<>(MainActivity.this,  
 android.R.layout.*simple\_list\_item\_1*,stringArrayList);  
 gridView.setAdapter(adapter);  
 }  
 }  
 private void lamRong(){  
 ed\_id.setText("");  
 ed\_email.setText("");  
 ed\_address.setText("");  
 ed\_name.setText("");  
 }

--------------------------------------------Book---------------------------------------

public class BookActivity extends AppCompatActivity {  
  
 EditText ed\_id, ed\_title;  
 Spinner spinnerAuthor;  
 ArrayAdapter<String> authorNameAdapter;  
 GridView gridView;  
 DBHelper dbHelper;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_book*);  
 ed\_id=findViewById(R.id.*ed\_idbook*);  
 ed\_title=findViewById(R.id.*ed\_titleBook*);  
 spinnerAuthor=findViewById(R.id.*spinnerAuthor*);  
 gridView=findViewById(R.id.*grid\_ListBook*);  
 dbHelper=new DBHelper(this);  
 select();  
 spinnerAuthor.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {  
 @Override  
 public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {  
 String authorName=spinnerAuthor.getSelectedItem().toString();  
 Author author=dbHelper.getAuthorByName(authorName);  
 Toast.*makeText*(getApplicationContext(), ""+author, Toast.*LENGTH\_SHORT*).show();  
 }  
  
 @Override  
 public void onNothingSelected(AdapterView<?> adapterView) {  
  
 }  
 });  
 loadAuhor();  
 select();  
 gridView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {  
 Integer id=Integer.*parseInt*(gridView.getItemAtPosition(i).toString());  
 Book book=dbHelper.getBookById(id);  
 ed\_id.setText(String.*valueOf*(book.getId()));  
 ed\_title.setText(book.getTitle().toString());  
 }  
 });  
 Button btn\_them=findViewById(R.id.*btn\_savebook*);  
 btn\_them.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (ed\_id.getText().toString().equals("")||  
 ed\_title.getText().toString().equals("")){  
 Toast.*makeText*(getApplicationContext(), "Nhap du thong tin", Toast.*LENGTH\_SHORT*).show();  
 }else {  
 int id=Integer.*parseInt*(ed\_id.getText().toString());  
 String title=ed\_title.getText().toString();  
 String authorName=spinnerAuthor.getSelectedItem().toString();  
 Author author=dbHelper.getAuthorByName(authorName);  
 if (id>0&&!title.equals("")){  
 Book book= new Book(id,title,author);  
 dbHelper.insertBook(book);  
 Toast.*makeText*(BookActivity.this," luu thanh cong"+book.getId(),Toast.*LENGTH\_SHORT*).show();  
 lamRong();  
 select();  
 }  
 else Toast.*makeText*(BookActivity.this," luu khong thanh cong",Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 Button btn\_xoa=findViewById(R.id.*btn\_deletebook*);  
 btn\_xoa.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (ed\_id.getText().toString().equals("")){  
 Toast.*makeText*(getApplicationContext(), "Chon book can xoa", Toast.*LENGTH\_SHORT*).show();  
 }else {  
 if (dbHelper.deleteBook(Integer.*parseInt*(ed\_id.getText().toString()))>0){  
 Toast.*makeText*(getApplicationContext(), "xoa thanh cong author", Toast.*LENGTH\_SHORT*).show();  
 lamRong();  
 select();  
 }  
 }  
 }  
 });  
 Button btn\_sua=findViewById(R.id.*btn\_updatebook*);  
 btn\_sua.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (ed\_id.getText().toString().equals("")){  
 Toast.*makeText*(getApplicationContext(), "Chon book can sua", Toast.*LENGTH\_SHORT*).show();  
 }else {  
 Book book=dbHelper.getBookById(Integer.*parseInt*(ed\_id.getText().toString()));  
 book.setTitle(ed\_title.getText().toString());  
 String authorName=spinnerAuthor.getSelectedItem().toString();  
 Author author=dbHelper.getAuthorByName(authorName);  
 book.setAuthor(author);  
 if (dbHelper.updateBook(book)){  
 Toast.*makeText*(getApplicationContext(), "Cap nhat thanh cong", Toast.*LENGTH\_SHORT*).show();  
 lamRong();  
 select();  
 }else Toast.*makeText*(getApplicationContext(), "Cap nhat that bai", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 Button btn\_select=findViewById(R.id.*btn\_selectbook*);  
 btn\_select.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 select();  
 }  
 });  
 Button btn\_thoat=findViewById(R.id.*btn\_thoat*);  
 btn\_thoat.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });  
 }  
 private void select(){  
 if (!ed\_id.getText().toString().equals("")) {  
 Book book;  
 ArrayList<String> stringArrayList= new ArrayList<>();  
 book=dbHelper.getBookById(Integer.*parseInt*(ed\_id.getText().toString()));  
 stringArrayList.add(book.getId()+"");  
 stringArrayList.add(book.getTitle());  
 stringArrayList.add(book.getAuthor().getId()+"");  
 ArrayAdapter<String> adapter= new ArrayAdapter<>(BookActivity.this,  
 android.R.layout.*simple\_list\_item\_1*,stringArrayList);  
 gridView.setAdapter(adapter);  
 }else {  
 ArrayList<Book> list=dbHelper.getAllBooks();  
 ArrayList<String> stringArrayList= new ArrayList<>();  
 for(Book book:list){  
 stringArrayList.add(book.getId()+"");  
 stringArrayList.add(book.getTitle());  
 stringArrayList.add(book.getAuthor().getId()+"");  
 }  
 ArrayAdapter<String> adapter= new ArrayAdapter<>(BookActivity.this,  
 android.R.layout.*simple\_list\_item\_1*,stringArrayList);  
 gridView.setAdapter(adapter);  
 }  
 }  
 private void lamRong(){  
 ed\_id.setText("");  
 ed\_title.setText("");  
 }  
 private void loadAuhor(){  
 ArrayList<Author> authors=dbHelper.getAllAuthors();  
 List<String> list=new ArrayList<>();  
 for (Author author:authors){  
 list.add(author.getName());  
 }  
 authorNameAdapter= new ArrayAdapter<>(this, android.R.layout.*simple\_spinner\_item*,list);  
 authorNameAdapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*);  
 spinnerAuthor.setAdapter(authorNameAdapter);  
 }