НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №6

з дисципліни

“Програмування мобільних систем”

Виконав:

студент групи ІВ-83

ЗК ІО-7303

Ву В'єт Тунг

Київ 2021

**Варіант № 7303%6+1=2**

**№ 7303%2+1=2**

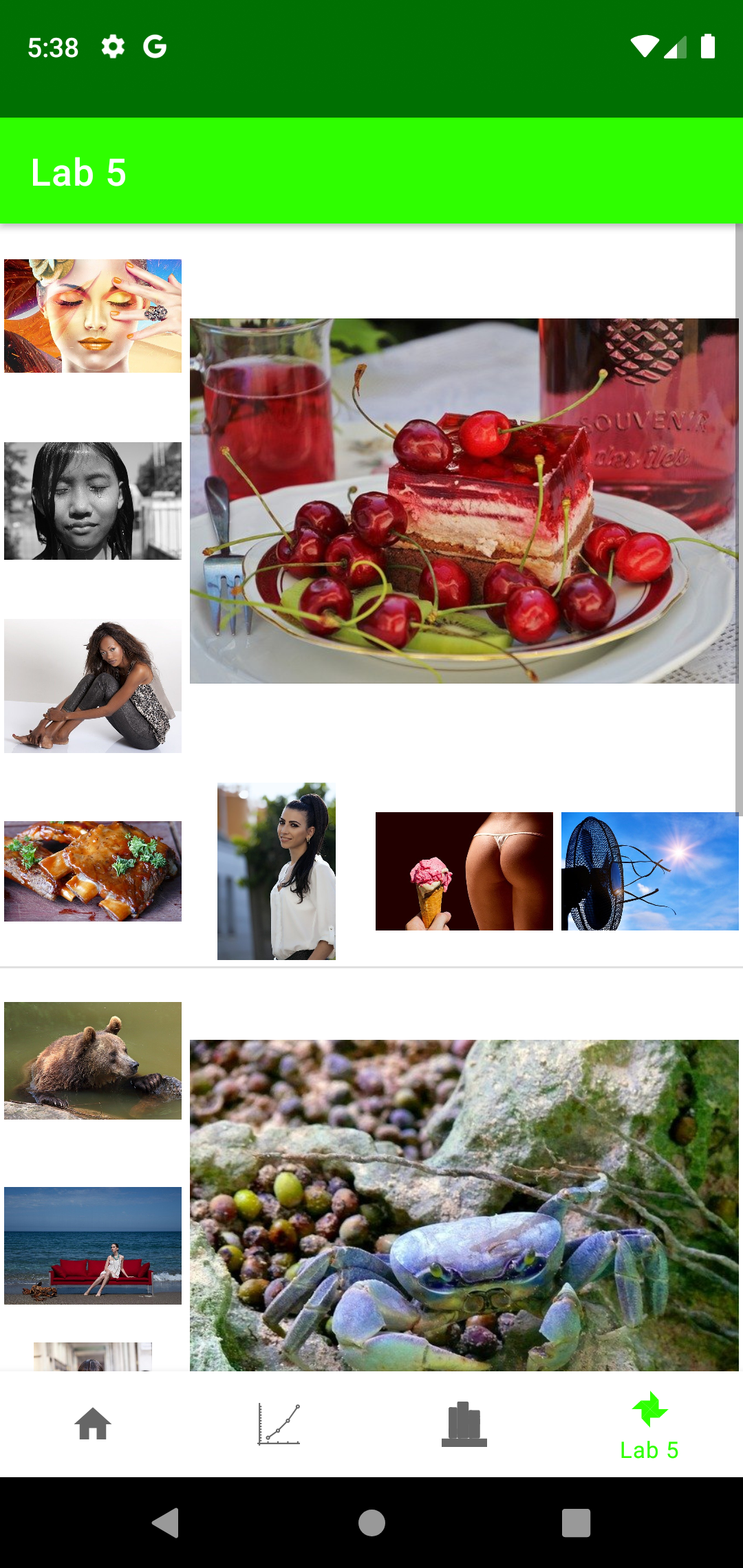
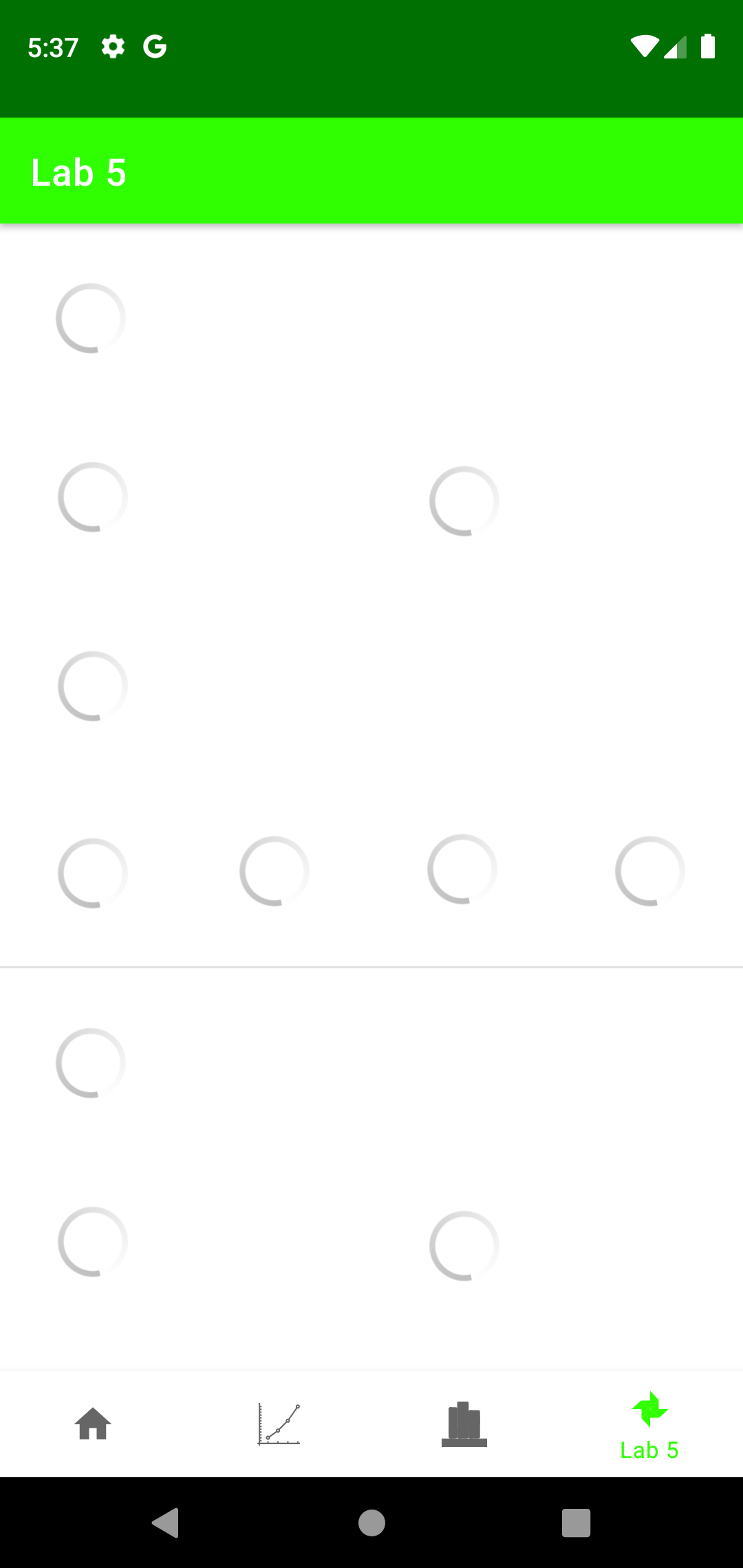
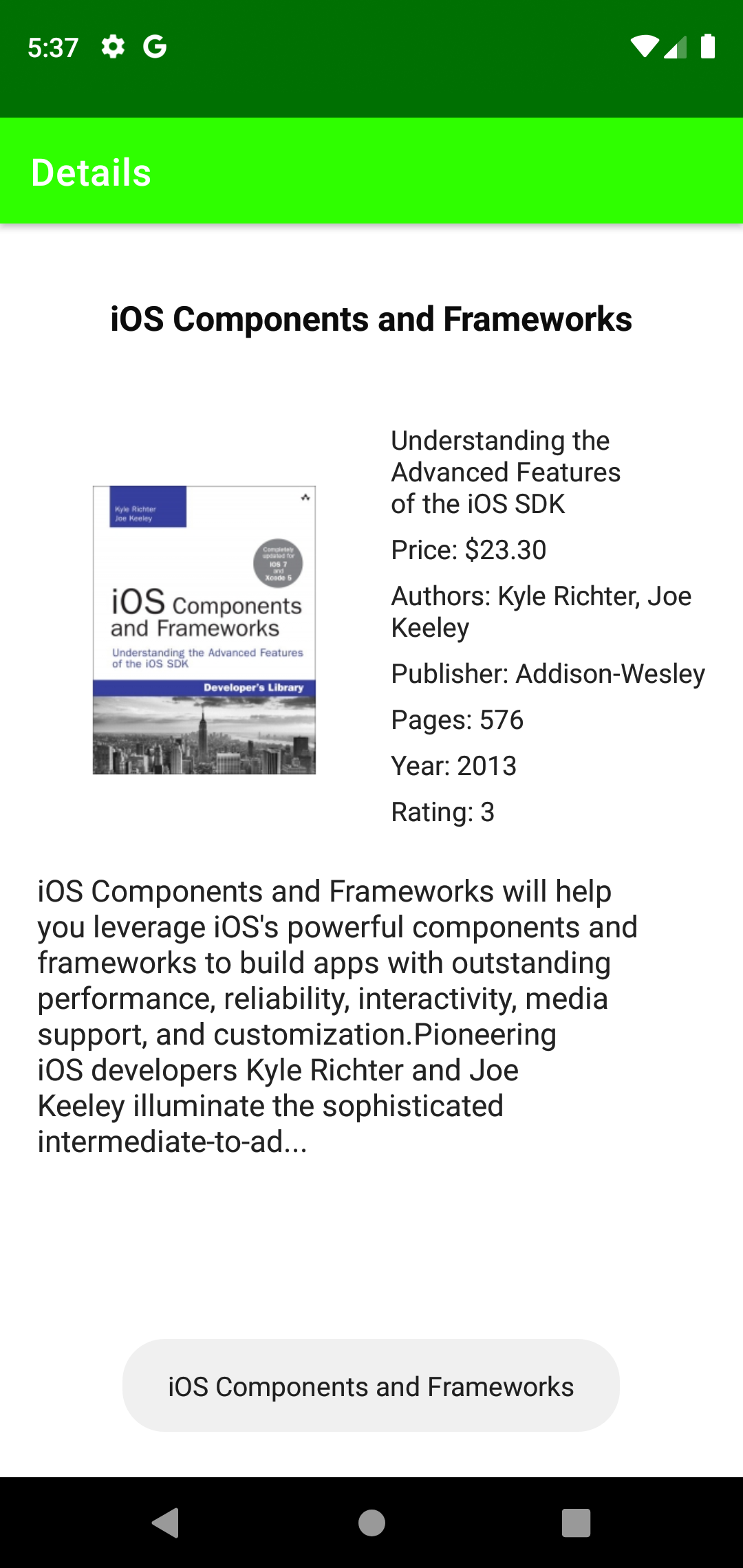
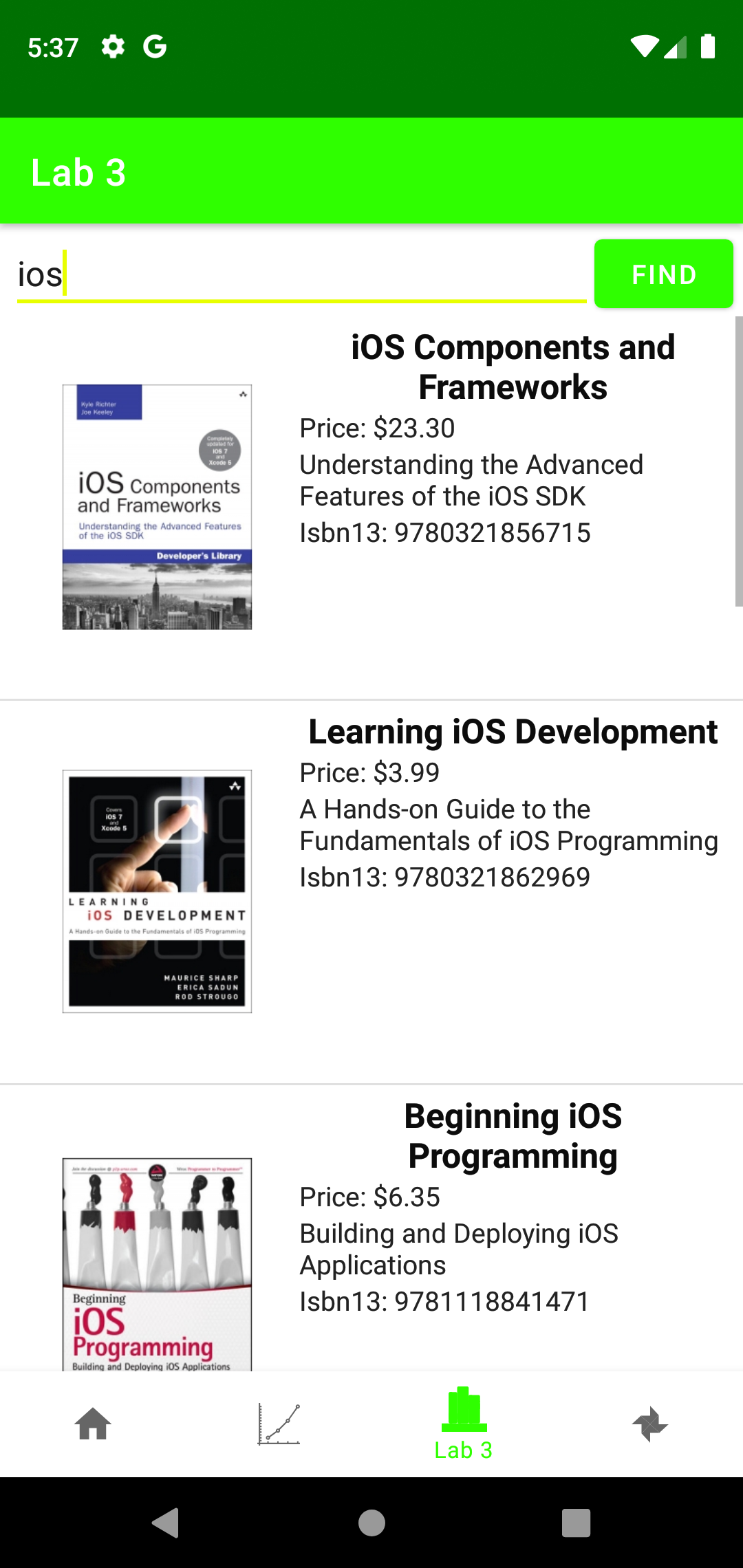
|  |
| --- |
| **Варіант 2** |
| Предметна область – книги  URL-адреса для отримання даних – https://api.itbook.store/1.0/search/**REQUEST**, де **REQUEST** – запит із поля для пошуку. |

|  |
| --- |
| **Варіант 2** |
| Предметна область – книги  URL-адреса для отримання даних – https://api.itbook.store/1.0/books/**IDENTIFIER**, де **IDENTIFIER** – ідентифікатор відповідної книги (поле **isbn13** із сутності книги) |

|  |
| --- |
| **Варіант 2** |
| **REQUEST** – “hot+summer”  **COUNT** – 24 |

**Скріншоти роботи додатка:**





**Лістинг коду**

**ImagesFragment.java**

package ua.kpi.comsys.IO7303.ui.images;  
  
import android.annotation.SuppressLint;  
import android.app.Activity;  
import android.content.Context;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.graphics.Point;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.view.Display;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ArrayAdapter;  
import android.widget.ImageView;  
import android.widget.LinearLayout;  
import android.widget.ListView;  
import android.widget.ProgressBar;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.fragment.app.Fragment;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.InputStreamReader;  
import java.net.MalformedURLException;  
import java.net.URL;  
import java.util.ArrayList;  
import java.util.List;  
  
import ua.kpi.comsys.IO7303.R;  
  
public class ImagesFragment extends Fragment {  
 View root;  
 static int *width*;  
 int height;  
 private ImagesListAdapter adapter;  
 ListView listView;  
 static LinearLayout *layout*;  
 String REQUEST = "\"hot+summer\"";  
 String imageUrlTarget="\"webformatURL\":\"";  
 int COUNT = 24;  
 String API\_KEY = "19193969-87191e5db266905fe8936d565";  
 View elemSet;  
 URL url;  
  
 @Override  
 public void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 Display screensize = getActivity().getWindowManager().getDefaultDisplay();  
 Point size = new Point();  
 screensize.getSize(size);  
 *width* = size.x;  
 height = size.y;  
 }  
  
  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 ViewGroup container, Bundle savedInstanceState) {  
 root = inflater.inflate(R.layout.*fragment\_four\_tab\_images*, container, false);  
 elemSet = inflater.inflate(R.layout.*images\_list*, container, false);  
  
 listView = root.findViewById(R.id.*imagesList*);  
  
 try {  
 url = new URL("https://pixabay.com/api/?key="+API\_KEY+"&q="+REQUEST+"&image\_type=photo&per\_page="+COUNT);  
 } catch (MalformedURLException e) {  
 e.printStackTrace();  
 }  
  
 new ParseJson("LoadImage").start();  
  
 return root;  
 }  
  
 static class ImagesListAdapter extends ArrayAdapter<List<String>> {  
 private final List<List<String>> taskImg;  
 Activity generalAct;  
  
 ImagesListAdapter(Context context, int textViewResourceId, List<List<String>> objects, Activity generalAct) {  
 super(context, textViewResourceId, objects);  
 this.taskImg = objects;  
 this.generalAct = generalAct;  
 }  
  
 @NonNull  
 @Override  
 public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {  
 LayoutInflater inflater = (LayoutInflater) getContext().getSystemService(Context.*LAYOUT\_INFLATER\_SERVICE*);  
 @SuppressLint("ViewHolder") View row = inflater.inflate(R.layout.*images\_list*, parent, false);  
  
 *layout* = row.findViewById(R.id.*imageSet*);  
 ViewGroup.LayoutParams params = *layout*.getLayoutParams();  
 params.height = *width*;  
 params.width = *width*;  
 *layout*.setLayoutParams(params);  
  
 List<ImageView> imagesListToShow = new ArrayList<>();  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img1*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img2*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img3*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img4*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img5*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img6*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img7*));  
 imagesListToShow.add(row.findViewById(R.id.*gal\_img8*));  
  
 List<ProgressBar> loadingStatusList = new ArrayList<>();  
 loadingStatusList.add(row.findViewById(R.id.*load1*));  
 loadingStatusList.add(row.findViewById(R.id.*load2*));  
 loadingStatusList.add(row.findViewById(R.id.*load3*));  
 loadingStatusList.add(row.findViewById(R.id.*load4*));  
 loadingStatusList.add(row.findViewById(R.id.*load5*));  
 loadingStatusList.add(row.findViewById(R.id.*load6*));  
 loadingStatusList.add(row.findViewById(R.id.*load7*));  
 loadingStatusList.add(row.findViewById(R.id.*load8*));  
  
 int imgNumber = taskImg.get(position).size();  
  
 for (int i=0; i<8; i++){  
 try {  
 if (i<imgNumber){  
 try {  
 new SetDownloadedImg(imagesListToShow.get(i)).execute(taskImg.get(position).get(i));  
 } catch (Exception e){}  
 }  
 else loadingStatusList.get(i).setVisibility(View.*INVISIBLE*);  
 } catch (Exception ignored){}  
 }  
  
 return row;  
 }  
 }  
  
 class ParseJson extends Thread {  
 ParseJson(String name){  
 super(name);  
 }  
  
 public void run(){  
 List<String> urls = new ArrayList<>();  
 List<List<String>> urls\_final = new ArrayList<>();  
  
 try {  
 BufferedReader br = new BufferedReader(new InputStreamReader(url.openStream()));  
 String inputLine;  
 String json = "";  
 while (true) {  
 if ((inputLine = br.readLine()) == null) break;  
 json += inputLine;  
 }  
  
 String S[] = json.split(imageUrlTarget);  
 for (String str : S){  
 if (str.substring(0, 4).equals("http")){  
 urls.add(str.split("\",\"")[0]);  
 }  
 }  
  
 for (String currentUrl : urls){  
 if (urls\_final != null){  
 if (urls\_final.size()==0){  
 List<String> tempImageList = new ArrayList<>();  
 urls\_final.add(tempImageList);  
 }  
 if (urls\_final.get(urls\_final.size()-1).size()>=8){  
 List<String> tempImageList = new ArrayList<>();  
 tempImageList.add(currentUrl);  
 urls\_final.add(tempImageList);  
 }  
 else {  
 urls\_final.get(urls\_final.size()-1).add(currentUrl);  
 }  
 }  
 }  
  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 getActivity().runOnUiThread(new Runnable() {  
 @Override  
 public void run() {  
 if(urls != null){  
 adapter = new ImagesListAdapter(getActivity(), R.layout.*images\_list*, urls\_final, getActivity());  
 listView.setAdapter(adapter);  
 }  
 else{  
 Toast.*makeText*(getContext(), "Failed data load", Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 });  
  
 System.*out*.println("URLS: " + urls\_final.toString());  
 }  
 }  
  
 private static class SetDownloadedImg extends AsyncTask<String, Void, Bitmap> {  
 ImageView imageViewObj;  
  
 public SetDownloadedImg(ImageView bmImage) {  
 this.imageViewObj = bmImage;  
 }  
  
 protected Bitmap doInBackground(String... urls) {  
 String urldisplay = urls[0];  
 Bitmap mIcon11 = null;  
 try {  
 InputStream in = new URL(urldisplay).openStream();  
 mIcon11 = BitmapFactory.*decodeStream*(in);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 return mIcon11;  
 }  
  
 protected void onPostExecute(Bitmap result) {  
 if (result == null)  
 imageViewObj.setImageResource(R.drawable.*no\_image*);  
 else imageViewObj.setImageBitmap(result);  
 }  
 }  
}

**BookDetail.java**

package ua.kpi.comsys.IO7303.ui.library;  
  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.InputStreamReader;  
import java.net.MalformedURLException;  
import java.net.URL;  
  
import ua.kpi.comsys.IO7303.R;  
  
public class BookDetail extends AppCompatActivity {  
 Book book;  
 String isbn13;  
 JsonHandler jsonHandler;  
  
 @Override  
 protected void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*book\_info*);  
  
 Bundle arguments = getIntent().getExtras();  
 isbn13 = arguments.get("Isbn13").toString();  
  
 jsonHandler = new JsonHandler();  
 jsonHandler.*setUserFileEnable*(false);  
  
 new LoadJson("LoadBook").start();  
 }  
  
 class LoadJson extends Thread {  
 LoadJson(String name){  
 super(name);  
 }  
  
 public void run(){  
 try {  
 URL oracle = new URL("https://api.itbook.store/1.0/books/"+isbn13);  
 BufferedReader in = new BufferedReader(new InputStreamReader(oracle.openStream()));  
 String result = in.readLine();  
 System.*out*.println("RESULT: "+result);  
  
 book = jsonHandler.*importBookFromString*(result);  
  
 System.*out*.println(book);  
  
 TextView title = findViewById(R.id.*bookDetailTitle*);  
 TextView subtitle = findViewById(R.id.*subtitleDetail*);  
 TextView price = findViewById(R.id.*priceDetail*);  
 TextView rating = findViewById(R.id.*ratingDetail*);  
 TextView publisher = findViewById(R.id.*publisherDetail*);  
 TextView authors = findViewById(R.id.*authorsDetail*);  
 TextView year = findViewById(R.id.*yearDetail*);  
 TextView pages = findViewById(R.id.*pagesDetail*);  
 TextView desc = findViewById(R.id.*desc*);  
  
 ImageView imageView = (ImageView) findViewById(R.id.*imageDetail*);  
  
 runOnUiThread(new Runnable() {  
 @Override  
 public void run() {  
 title.setText(book.getTitle());  
 subtitle.setText(book.getSubtitle());  
 price.setText("Price: "+ book.getPrice());  
 rating.setText("Rating: "+ book.getRating());  
 publisher.setText("Publisher: "+ book.getPublisher());  
 authors.setText("Authors: "+ book.getAuthors());  
 year.setText("Year: "+ book.getYear());  
 pages.setText("Pages: "+ book.getPages());  
 desc.setText(book.getDesc());  
  
 String posterUrl = book.getImage();  
 imageView.setImageResource(R.drawable.*no\_image*);  
  
 try {  
 new LoadAndSetImage(imageView).execute(posterUrl);  
 } catch (Exception e){  
 imageView.setImageResource(R.drawable.*no\_image*);  
 }  
 }  
 });  
 } catch (MalformedURLException e) {  
 e.printStackTrace();  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
  
 private class LoadAndSetImage extends AsyncTask<String, Void, Bitmap> {  
 ImageView bmImage;  
  
 public LoadAndSetImage(ImageView bmImage) {  
 this.bmImage = bmImage;  
 }  
  
 protected Bitmap doInBackground(String... urls) {  
 String urldisplay = urls[0];  
 Bitmap mIcon11 = null;  
 try {  
 InputStream in = new java.net.URL(urldisplay).openStream();  
 mIcon11 = BitmapFactory.*decodeStream*(in);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 return mIcon11;  
 }  
  
 protected void onPostExecute(Bitmap result) {  
 bmImage.setImageBitmap(result);  
 }  
 }  
  
 public void setBook(Book book) {  
 this.book = book;  
 }  
}

**LibraryFragment.java**

package ua.kpi.comsys.IO7303.ui.library;  
  
import android.content.Context;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.view.inputmethod.InputMethodManager;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ImageView;  
import android.widget.ListView;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.fragment.app.Fragment;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.InputStreamReader;  
import java.net.MalformedURLException;  
import java.net.URL;  
import java.util.ArrayList;  
  
import java.util.List;  
import android.app.Activity;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.os.AsyncTask;  
  
import ua.kpi.comsys.IO7303.R;  
  
public class LibraryFragment extends Fragment {  
 private List<Book> books;  
 private List<Book> foundBooks = new ArrayList<>();  
 private List<Book> booksToShow = new ArrayList<>();  
 private BookAdapter adapter;  
 ListView listView;  
 String customBooksList = "books\_list\_custom.txt";  
 String REQUEST\_BOOK\_TITLE;  
 JsonHandler jsonHandler;  
  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 ViewGroup container, Bundle savedInstanceState) {  
 View root = inflater.inflate(R.layout.*fragment\_third\_tab\_books*, container, false);  
  
 jsonHandler = new JsonHandler();  
 jsonHandler.*setFileUserName*(customBooksList);  
  
 books = jsonHandler.*importBookListFromJSON*(getContext());  
 listView = root.findViewById(R.id.*booksList*);  
  
 EditText searchRequest = root.findViewById(R.id.*searchField*);  
 Button searchBtn = root.findViewById(R.id.*buttonSearch*);  
  
 if(books != null){  
 adapter = new BookAdapter(getActivity(), R.layout.*book\_list*, books);  
  
 listView.setAdapter(adapter);  
// Toast.makeText(getContext(), "Loaded", Toast.LENGTH\_LONG).show();  
 }  
 else{  
 Toast.*makeText*(getContext(), "Load failed...", Toast.*LENGTH\_LONG*).show();  
 }  
  
  
 listView.setOnItemClickListener(new AdapterView.OnItemClickListener() { //DETAIL  
 @Override  
 public void onItemClick(AdapterView<?> parent, View itemClicked, int position,  
 long id) {  
 Toast.*makeText*(getContext(), booksToShow.get((int)id).getTitle(),  
 Toast.*LENGTH\_SHORT*).show();  
  
 startActivity(new Intent(getContext(), BookDetail.class).putExtra("Isbn13", booksToShow.get((int)id).getIsbn13()));  
 }  
 });  
  
 searchBtn.setOnClickListener(new View.OnClickListener() { // FIND  
 public void onClick(View view) {  
 String fieldText = searchRequest.getText().toString().toLowerCase();  
 if (books!=null)  
 books.clear();  
  
 if(!fieldText.equals("") & fieldText.length()>=3){  
 try {  
 REQUEST\_BOOK\_TITLE = fieldText;  
 new LoadJson("LoadJson").start();  
 } catch (Exception e){  
 e.printStackTrace();  
 }  
 }  
 else {  
 Toast.*makeText*(getContext(), "Uncorrected request", Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 });  
  
 return root;  
 }  
  
  
 class LoadJson extends Thread {  
 LoadJson(String name){  
 super(name);  
 }  
  
 public void run(){  
 try {  
 URL oracle = new URL("https://api.itbook.store/1.0/search/"+ REQUEST\_BOOK\_TITLE);  
 BufferedReader in = new BufferedReader(new InputStreamReader(oracle.openStream()));  
 String result = in.readLine();  
 System.*out*.println("RESULT: "+result);  
  
 foundBooks = jsonHandler.*importBookListFromString*(result);  
  
 jsonHandler.*exportToJSON*(getContext(), foundBooks);  
  
 BookAdapter adapter3 = new BookAdapter(getActivity(), R.layout.*book\_list*, foundBooks); // адаптер с новыми фильмами  
 try {  
 listView.setAdapter(adapter3);  
 } catch (Exception e2){}  
  
 // HIDE KEYBOARD  
 InputMethodManager imm = (InputMethodManager) getActivity().getSystemService(Activity.*INPUT\_METHOD\_SERVICE*);  
 View view = getActivity().getCurrentFocus();  
 if (view == null) {  
 view = new View(getActivity());  
 }  
 imm.hideSoftInputFromWindow(view.getWindowToken(), 0);  
 } catch (MalformedURLException e) {  
 e.printStackTrace();  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
  
 private class BookAdapter extends ArrayAdapter<Book>{  
 BookAdapter(Context context, int textViewResourceId, List<Book> objects) {  
 super(context, textViewResourceId, objects);  
 booksToShow = objects;  
 }  
  
 @NonNull  
 @Override  
 public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {  
 LayoutInflater inflater = getLayoutInflater();  
 View row = inflater.inflate(R.layout.*book\_list*, parent, false);  
 TextView title = row.findViewById(R.id.*bookTitle*);  
 TextView subtitle = row.findViewById(R.id.*bookSubtitle*);  
 TextView price = row.findViewById(R.id.*bookPrice*);  
 TextView isbn13 = row.findViewById(R.id.*bookIsbn13*);  
  
 title.setText(handle(booksToShow.get(position).getTitle()));  
 subtitle.setText(handle(booksToShow.get(position).getSubtitle()));  
 price.setText("Price: " + handle(booksToShow.get(position).getPrice()));  
 isbn13.setText("Isbn13: " + handle(booksToShow.get(position).getIsbn13()));  
  
 ImageView currImg = row.findViewById(R.id.*image*);  
 String imageName = booksToShow.get(position).getImage();  
  
 try {new LoadAndSetImage(currImg).execute(imageName);}  
 catch(Exception e) {currImg.setImageResource(R.drawable.*no\_image*);}  
 return row;  
 }  
  
 public String handle(String str){  
 if(str.equals("")) return "None";  
 else return str;  
  
 }  
 }  
  
 private class LoadAndSetImage extends AsyncTask<String, Void, Bitmap> {  
 ImageView bmImage;  
  
 public LoadAndSetImage(ImageView bmImage) {  
 this.bmImage = bmImage;  
 }  
  
 protected Bitmap doInBackground(String... urls) {  
 String urldisplay = urls[0];  
 Bitmap mIcon11 = null;  
 try {  
 InputStream in = new java.net.URL(urldisplay).openStream();  
 mIcon11 = BitmapFactory.*decodeStream*(in);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 return mIcon11;  
 }  
  
 protected void onPostExecute(Bitmap result) {  
 bmImage.setImageBitmap(result);  
 }  
 }  
}

**Висновок**

Виконано шосту лабораторну роботу. Модифіковано третю та четверту вкладки, додаток завантажує дані з мережі та відображає.