Департамент образования Воронежской области

Государственное бюджетное профессиональное образовательное учреждение

Воронежской области «Борисоглебский техникум промышленных

и информационных технологий»

**ОТЧЕТ**

**по учебной практике**

**УП.05.01 Проектирование и разработка информационных систем**

|  |  |
| --- | --- |
| Выполнил студент \_\_3\_\_ курса  спец. 09.02.07 «Информационные системы и программирование»  Группа 3.2ИСИП\_\_  Сусликов К. В.  (ФИО)  Дата сдачи \_\_31.05.2023 г.\_\_\_\_ | Проверил  преподаватель  Пеньков А.С.  (ФИО)  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Борисоглебск

**2022 –2023 учебный год**

**Оглавление**

[1. ВЫБОР СРЕДСТВ РАЗРАБОТКИ 3](#_Toc135822938)

[2. РАЗРАБОТКА ТЕХНИЧЕСКОГО ЗАДАНИЯ 5](#_Toc135822939)

[3. СИСТЕМА КОНТРОЛЯ ВЕРСИЙ 9](#_Toc135822940)

[5. СОЗДАНИЕ МАКЕТА ИНФОРМАЦИОННОЙ СИСТЕМЫ 14](#_Toc135822941)

[6. СОЗДАНИЕ ЭЛЕМЕНТОВ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА 16](#_Toc135822942)

[7. MATERIAL DESIGN 18](#_Toc135822943)

[8. РАБОТА СО СПИСОЧНЫМИ ЭЛЕМЕНТАМИ ИНТЕРФЕЙСА 19](#_Toc135822944)

[9. ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД В РАЗРАБОТКЕ КОДА ИНФОРМАЦИОННОЙ СИТЕМЫ 23](#_Toc135822945)

[10. СОЗДАНИЕ ПРОЦЕДУР ОБРАБОТКИ ИНФОРМАЦИИ. 25](#_Toc135822946)

[11. ОБЕСПЕЧЕНИЕ ДОПОЛНИТЕЛЬНОЙ ФУНКЦИОНАЛЬНОСТИ СИСТЕМЫ. 28](#_Toc135822947)

[12. INTENTS 34](#_Toc135822948)

[13. ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ 41](#_Toc135822949)

[14. ОТЛАДКА ПРИЛОЖЕНИЯ. 44](#_Toc135822950)

[15. ЗАЩИТА ПРИЛОЖЕНИЯ. ОЦЕНКА КАЧЕСТВА И НАДЕЖНОСТИ ПРИЛОЖЕНИЯ 46](#_Toc135822951)

[СОПРОВОЖДЕНИЕ ПРОГРАММ 50](#_Toc135822952)

[СОПРОВОЖДЕНИЕ ПРОГРАММ 53](#_Toc135822953)

**1 ВЫБОР СРЕДСТВ РАЗРАБОТКИ**  
  
1) Определение ОС Android – Android открытая операционная система для мобильных телефонов, смартфонов, коммуникаторов, планшетных компьютеров, электронных книг, цифровых проигрывателей, научных часов, ноутбуков и смартбуков, основанная на ядре Linux.

2) Что входит в инструментарий Android разработчика -  
  
  
Для разработки Android-приложений, в первую очередь, необходимо установить **Java Development Kit (JDK)**. После установки JDK разработчику нужно установить IDE.  
  
**Android Software Development Kit (SDK)** – содержит множество инструментов и утилит для создания и тестирования приложений.  
  
А также потребуются:  
**Приложения, Платформа приложений, библиотеки и выполняемая среда Android**  
  
  
  
  
  
  
  
3) Перечислите обязательные элементы в структуре приложения  
  
gen – Файлы, сгенерированные самой Java. Здесь находится ткой важный файл как R.java  
  
AndroidManifest.xml -–Файл манифеста AndroidManifest.xml предоставляет системе основную информацию о программе. Каждое приложение должно иметь свой файл манифеста  
  
src – Каталог, в котором содержится исходный код приложения  
  
res – Каталог, содержащий ресурсы приложения. В данном каталоге могут находиться подпапки drawable, anim, layout, menu, values, xml и raw

4) Перечислите опциальные элементы в структуре приложения  
  
  
assets – Произвольное собрание каталогов и файлов  
  
  
Activities, Services, Broadcast receivers и Content providers

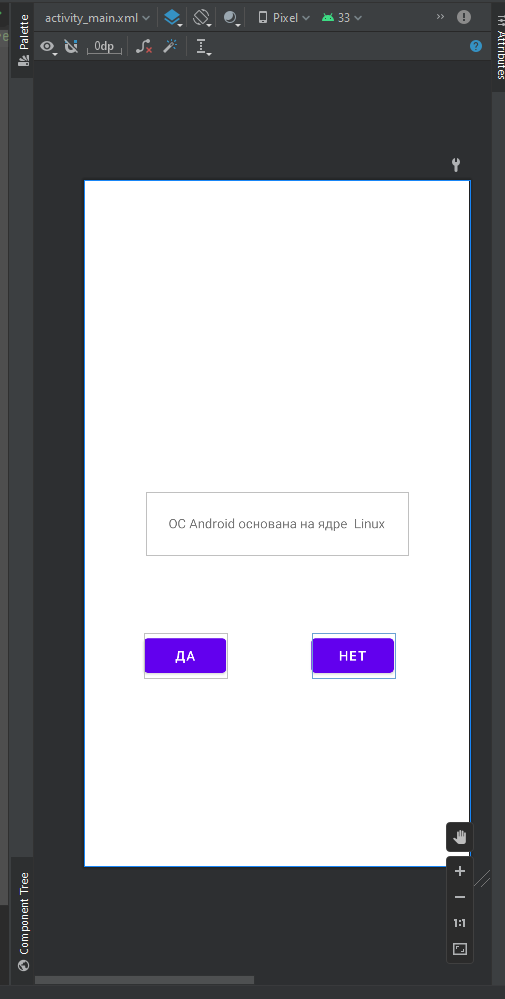
5) Изобразите жизненный цикл android приложения



**2 задание:** «Первое ANDROID-Приложение»  
  
1. Создал проект Android и назвал его «DroidQestion»  
  
2.Заполнил класс activity\_main

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".QuestActivity">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:padding="24dp"  
 android:text="@string/question\_text"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/true\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/true\_button"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.198"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.707" />  
  
 <Button  
 android:id="@+id/false\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/false\_button"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.752"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.707" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. Расположил кнопки и главное окно на экране

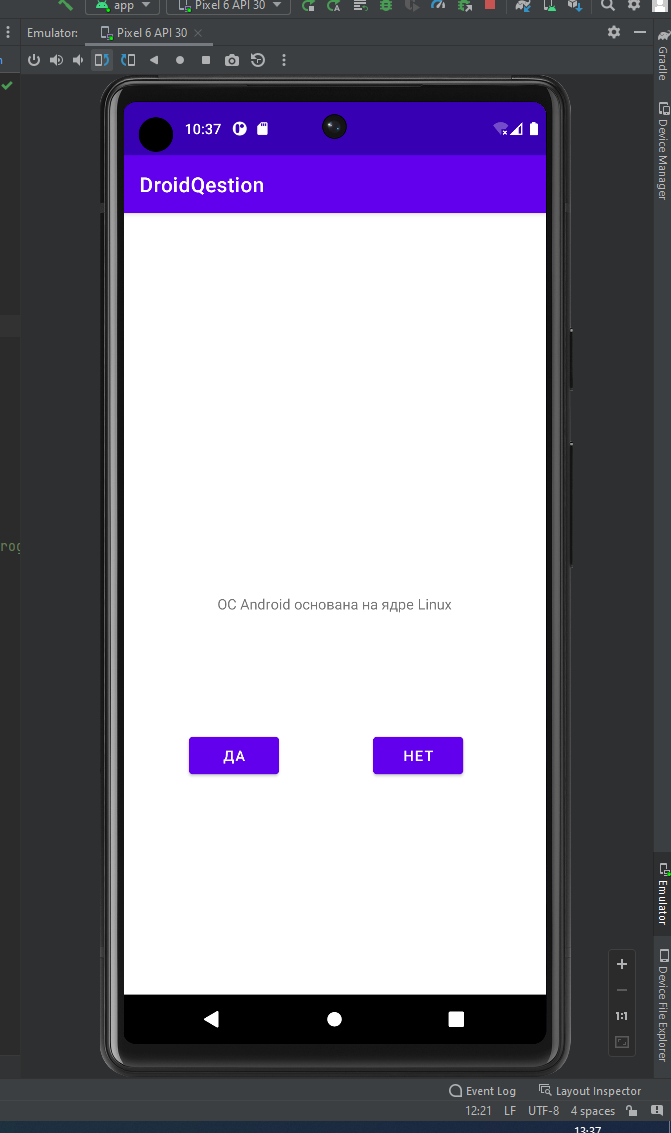


1. Добавил строковые ресурсы в strings.xml
2. <resources>  
    <string name="app\_name">DroidQestion</string>  
    <string name="question\_text">OC Android основана на ядре Linux</string>  
    <string name="true\_button">Да</string>  
    <string name="false\_button">Нет</string>  
    <string name="action\_settings">Settings</string>  
    <string name="correct\_toast">Верно!</string>  
    <string name="incorrect\_toast">Не верно!</string>  
     
   </resources>

5.Заполнил QuestActivity.java

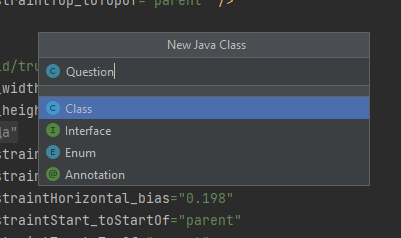
package com.example.droidqestion;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.annotation.SuppressLint;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
public class QuestActivity extends AppCompatActivity {  
private Button mTrueButton;  
private Button mFalseButton;  
 @SuppressLint("MissingInflatedId")  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 mTrueButton= (Button) findViewById(R.id.*true\_button*);  
 mFalseButton= (Button) findViewById(R.id.*false\_button*);  
 mTrueButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void  
 onClick(View view) {  
 Toast.*makeText*(QuestActivity.this,R.string.*incorrect\_toast*,Toast.*LENGTH\_SHORT*).show();  
 }  
 });mFalseButton=(Button) findViewById(R.id.*false\_button*);  
 mFalseButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Toast.*makeText*(QuestActivity.this,R.string.*correct\_toast*,Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
 }  
}

### 6. Выполнил запуск в эмуляторе

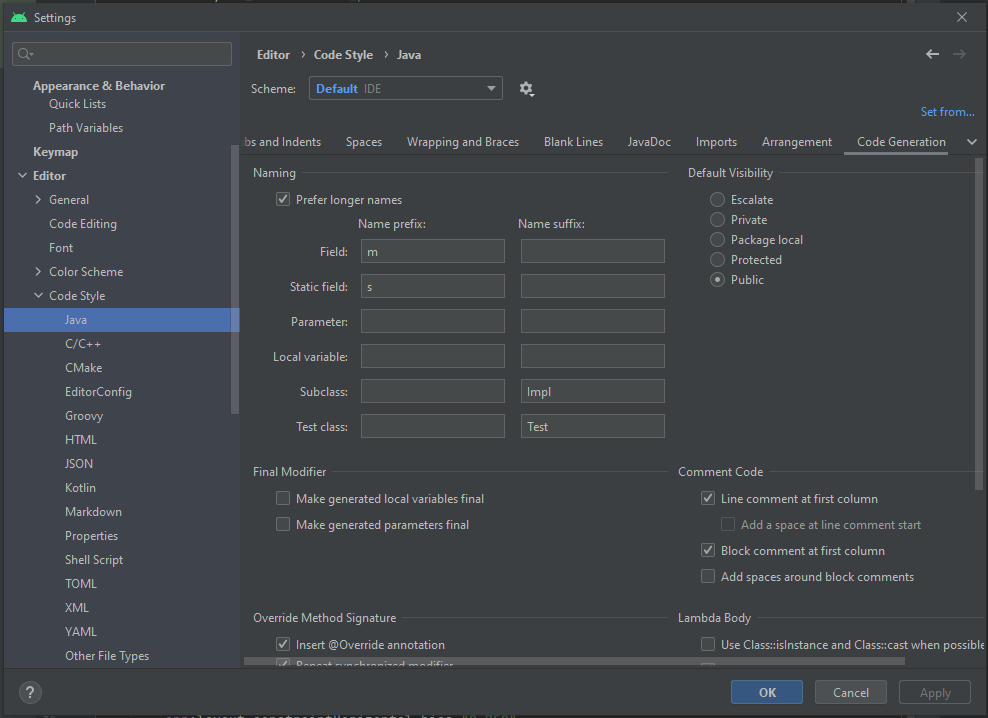
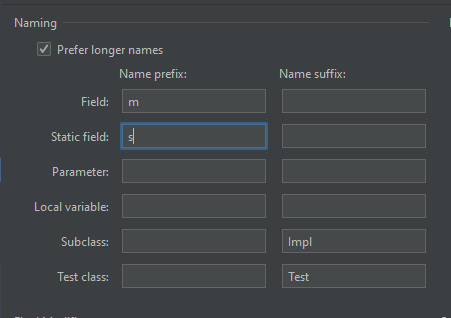


**3 задание:** «ANDROID И МОДЕЛЬ MVC»

1. Создал новый класс Question

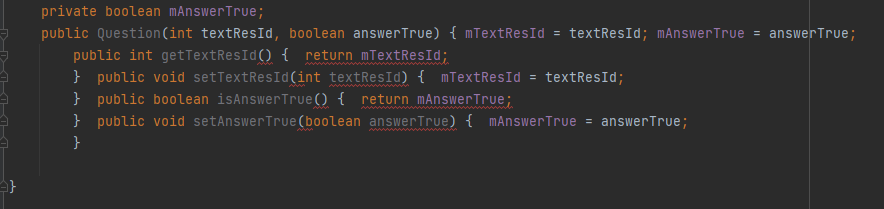
  
  
  
2. Заполнил код в классе: (Question.java)

package com.example.droidqestion;  
  
public class Question { private int mTextResId;  
 private boolean mAnswerTrue;  
 public Question(int textResId, boolean answerTrue) { mTextResId = textResId; mAnswerTrue = answerTrue;  
}  
}

3. Открыл настройки Android Studio, зашёл в раздел Editor и в Code Style выбрал категорию Java и перешёл на вкладку Code Generation  
  
  
  
4. В таблице Naming Prefix ввёл префикс m для полей и добавил префикс s для статических полей в строке Static field  
****

5. С помощью команды Generate сгенерировал get- и set метод для каждой переменной и получил такой код:

package com.example.droidqestion;  
  
public class Question { private int mTextResId;  
 private boolean mAnswerTrue;  
 public Question(int textResId, boolean answerTrue) { mTextResId = textResId; mAnswerTrue = answerTrue;  
}  
  
 public boolean isAnswerTrue() {  
 return mAnswerTrue;  
 }  
  
 public void setAnswerTrue(boolean answerTrue) {  
 mAnswerTrue = answerTrue;  
 }  
}

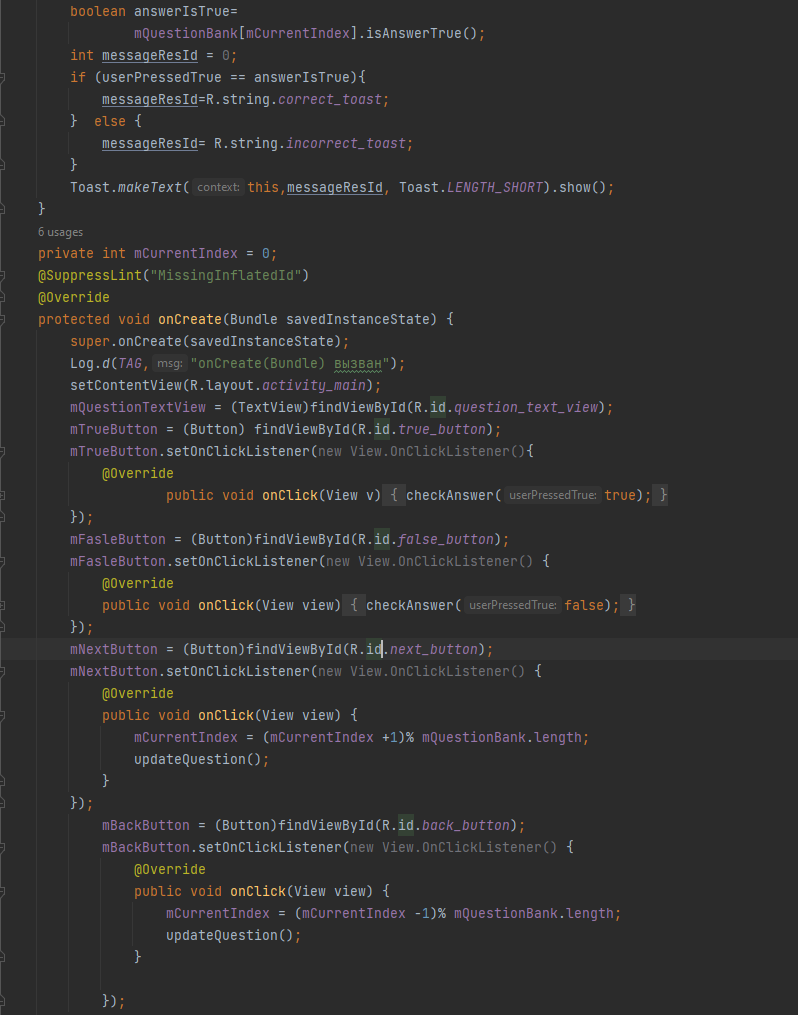
6. Прописал в классе public Question  
  
  
  
  
  
7. Обновил уровень представления DroidQuest и влючил в него кнопку «Далее»

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".QuestActivity">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:padding="24dp"  
 android:id="@+id/question\_text\_view"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/true\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/true\_button"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.198"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.707" />  
  
 <Button  
 android:id="@+id/false\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/false\_button"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.752"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.707" />  
  
 <Button  
 android:id="@+id/next\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Далее"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.805" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

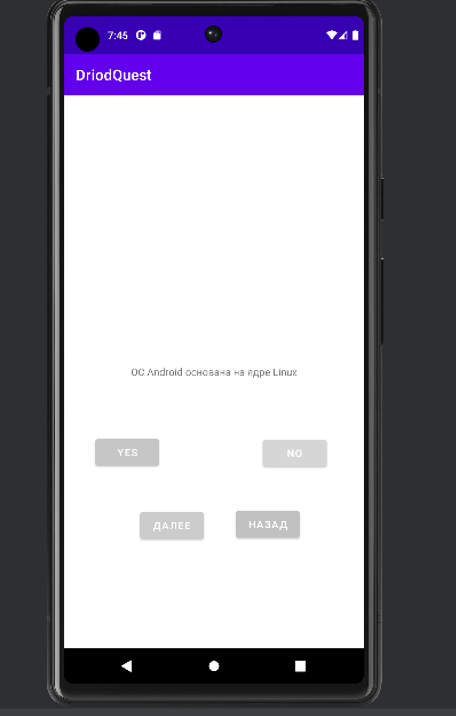
8. Обновил строки в (strings.xml) добавив дополнительные вопросы и удалил лишние строки

<resources>  
 <string name="app\_name">DroidQestion</string>  
 <string name="true\_button">Да</string>  
 <string name="false\_button">Нет</string>  
 <string name="next\_button">Далее</string>  
 <string name="action\_settings">Settings</string>  
 <string name="question\_android">OC Android основана на ядре Linux.</string>  
 <string name="question\_linear">Разметка LinearLayout позиционирует свои дочерние элементы в строки и столбцы.</string>  
 <string name="question\_service">Компонент Android-приложения Service предоставляют доступ к данным.</string>  
 <string name="question\_res">Ресурсы приложения содержатся в отдельном файле XML.</string>  
 <string name="question\_manifest">AndroidManifest.xml предоставляет системе основную информацию о программе.</string>  
 <string name="correct\_toast">Верно!</string>  
 <string name="incorrect\_toast">Не верно!</string>  
  
</resources>

9. Добавил переменные массива и подключил виджет TextView



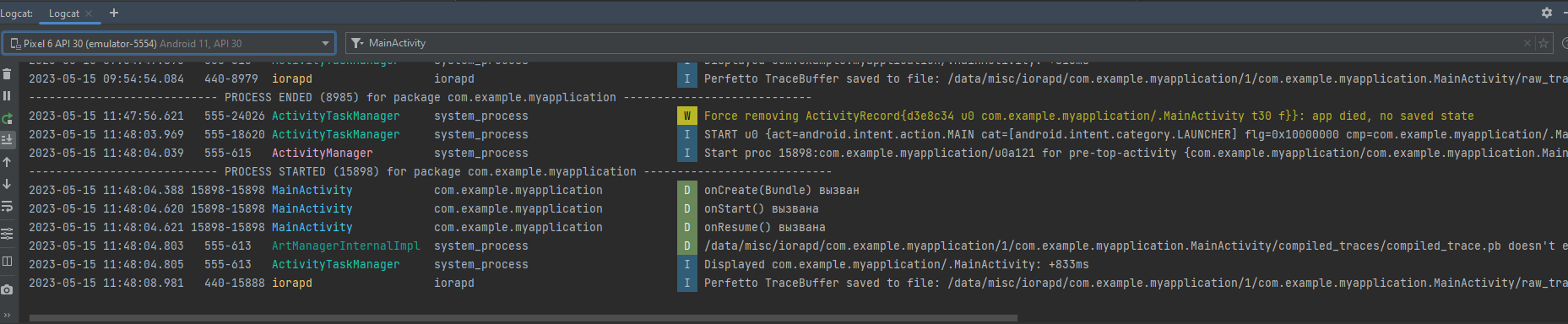
10. Тест программы. Всё работает



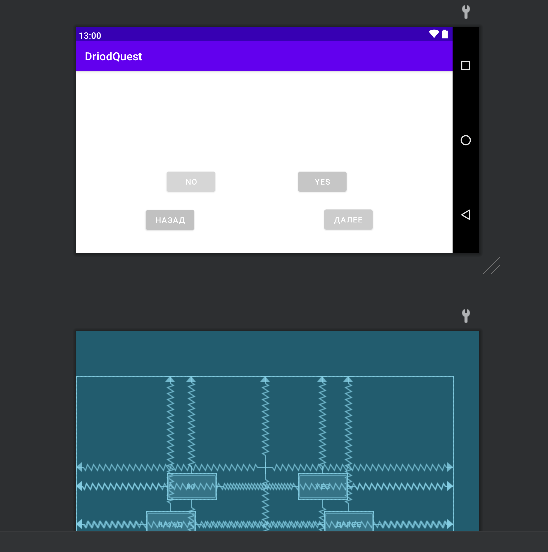
**4. СБОР ДАННЫХ ДЛЯ СОЗДАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ**1. Собрал данные для создания информационной системы

public class MainActivity extends AppCompatActivity {  
 private static final String *TAG* = "MainActivity";  
 @Override  
 public void onStart() {  
 super.onStart();  
 Log.*d*(*TAG*, "onStart() вызван");  
 }  
 @Override  
 public void onPause() {  
 super.onPause();  
 Log.*d*(*TAG*, "onPause() вызван");  
 }  
 @Override  
 public void onResume() {  
  
 super.onResume();  
 Log.*d*(*TAG*, "onResume() вызван");  
 }  
 @Override  
 public void onStop() {  
  
 super.onStop();  
 Log.*d*(*TAG*, "onStop() вызван");  
 }  
 @Override  
 public void onDestroy() {  
  
 super.onDestroy();  
 Log.*d*(*TAG*, "onDestroy() вызван");  
 }

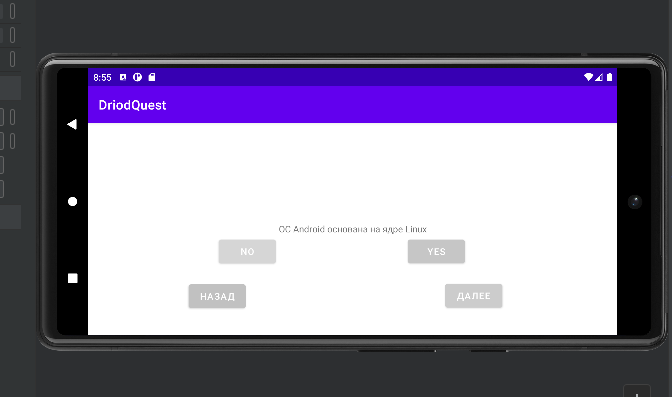
1. Переопределил методы жизненного цикла



3. Использовал LogCat



3.Создал макет для альбомной ориентации



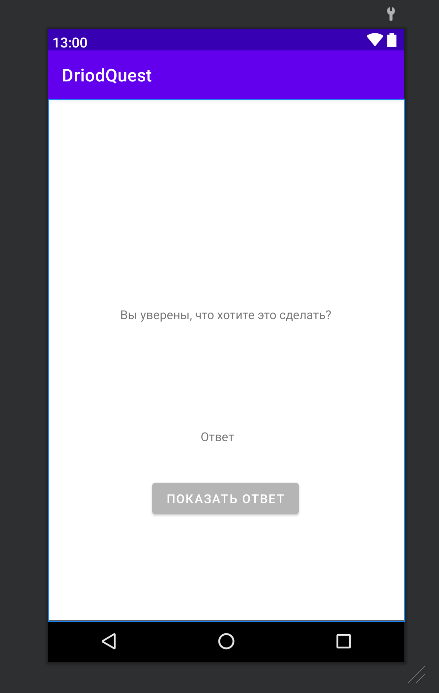
1. Альбомная ориентация

@Override  
protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 Log.*d*(*TAG*, "onCreate(Bundle) вызван");  
 setContentView(R.layout.*activity\_main*);  
 mQuestionTextView = (TextView)findViewById(R.id.*question\_text\_view*);  
 int question = mQuestionBank[mCurrentIndex].getTextResId();  
 mQuestionTextView.setText(question);  
 mTrueButton = (Button) findViewById(R.id.*true\_button*);  
 mFalseButton = (Button) findViewById(R.id.*false\_button*);  
 mTrueButton.setOnClickListener(new View.OnClickListener() {

1. Переопределение onSaveInstanceState(…)



# **5. СОЗДАНИЕ МАКЕТА ИНФОРМАЦИОННОЙ СИСТЕМЫ**

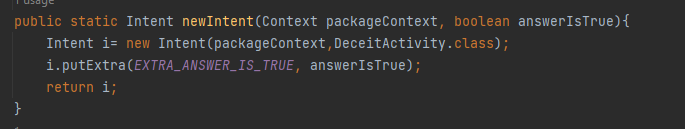


<Button  
 android:id="@+id/show\_answer\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="#B5B5B5"  
 android:text="@string/show\_answer\_button"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.789" />  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/warning\_text"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.41" />  
  
 <TextView  
 android:id="@+id/answer\_text\_view"  
 android:layout\_width="59dp"  
 android:layout\_height="15dp"  
 android:layout\_marginTop="380dp"  
 app:layout\_constraintBottom\_toBottomOf="@+id/show\_answer\_button"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.0"  
 tools:text="Ответ" />  
</androidx.constraintlayout.widget.ConstraintLayout>

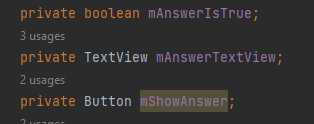
1.Заполнил макет второй активности

<string name="show\_answer\_button">Показать ответ</string>  
<string name="deceit\_button">Обмануть !</string>  
<string name="judgment\_toast">Обманывать нельзя !</string>

2. Добавил строки (strings.xml)



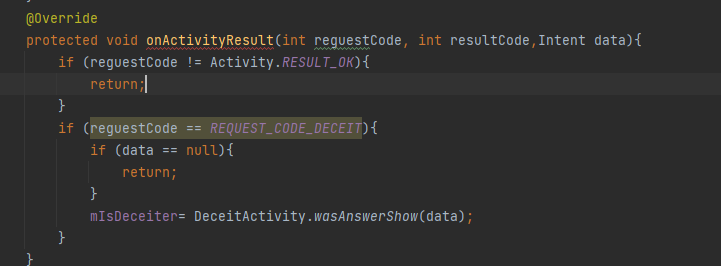
3. Метод newIntent(...) (DeceitActivity.java)

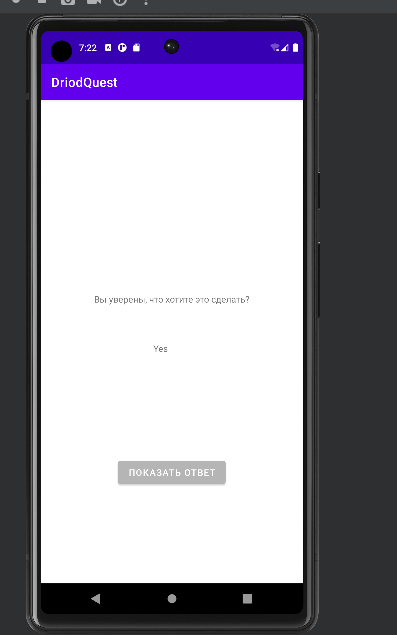


4. Добавил функциональность подсматривания ответов

@Override  
protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_deceit*);  
 mAnswerIsTrue = getIntent().getBooleanExtra(*EXTRA\_ANSWER\_IS\_TRUE*,false);  
 mAnswerTextView = (TextView) findViewById(R.id.*answer\_text\_view*);  
 mShowAnswer = (Button) findViewById(R.id.*show\_answer\_button*);  
 mShowAnswer.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (mAnswerIsTrue){  
 mAnswerTextView.setText(R.string.*true\_button*);  
 } else {  
 mAnswerTextView.setText(R.string.*false\_button*);  
 }  
 setAnswerShowResualt(true);  
 }  
 });

5. Реализовал onActivityResult(…)

  
6. В итоге это пункта:



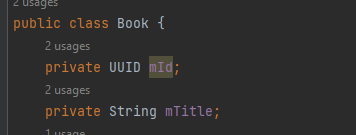
# **6. СОЗДАНИЕ ЭЛЕМЕНТОВ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА**

UI-ФРАГМЕНТЫ

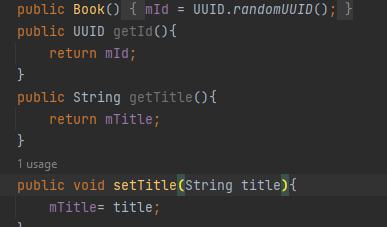
Настройка шаблонного кода (BookActivity.java)



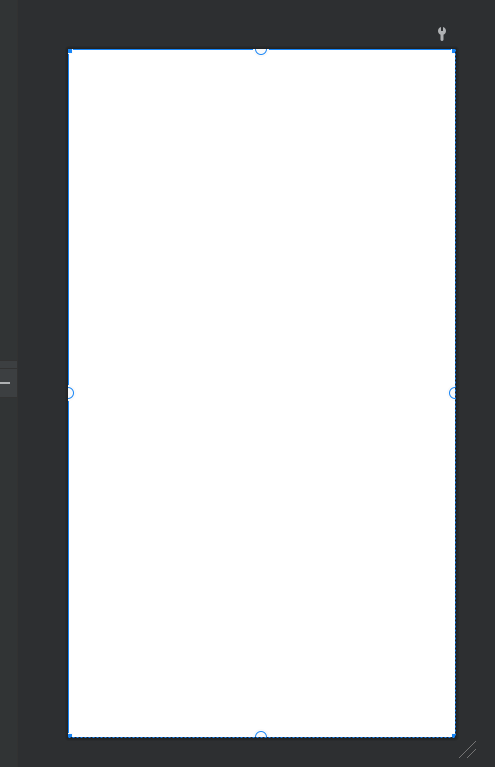
1.Добавил кода в класс Book (Book.java)



2. Сгенерированные get- и set-методы (Book.java)



3.Запустил приложения



4. Файл макета для представления фрагмента

(fragment\_book.xml)

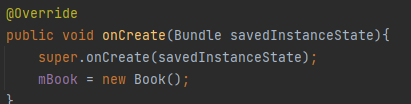
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="16dp"  
 android:layout\_marginRight="16dp"  
 android:layout\_marginTop="16dp">  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:layout\_marginRight="4dp">

5. Добавил и удалил некоторые строки

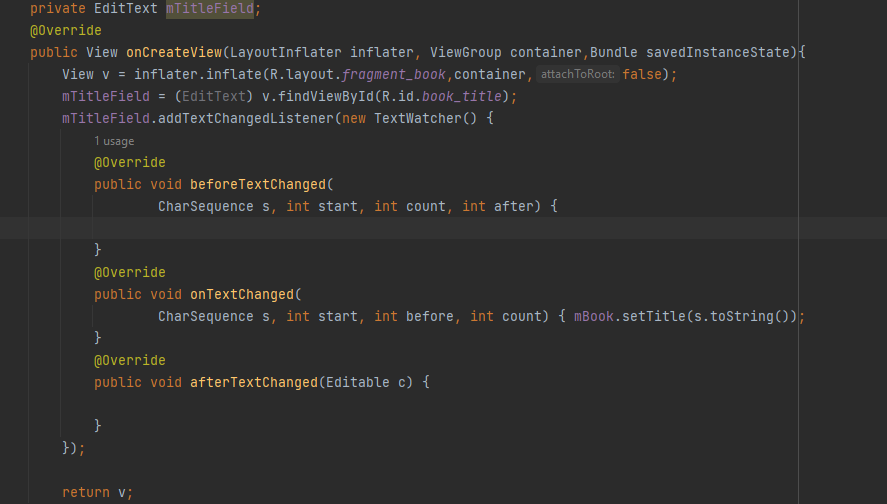
<resources>  
 <string name="app\_name">BookDepository</string>  
 <string name="menu\_settings">Settings</string>  
 <string name="title\_activity\_book">BookActivity</string>  
 <string name="book\_title\_hint">Введите название книги.</string>

6. Переопределил Fragment.onCreate(Bundle)

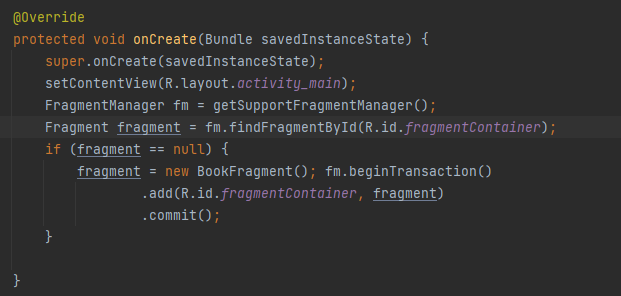
(BookFragment.java)



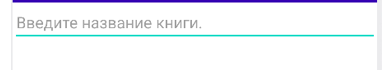
7. Настроил виджет EditText (BookFragment.java)



8. Получил объект FragmentManager

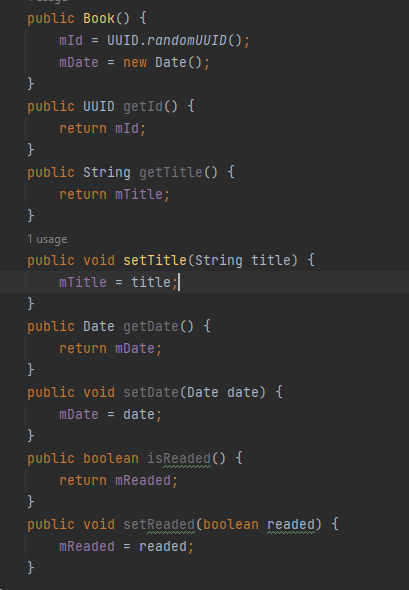
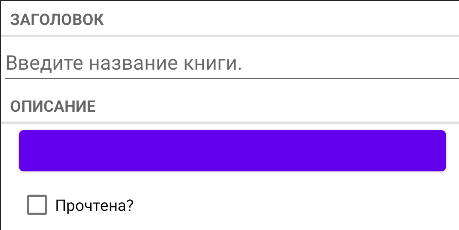


9. Сделал транзакцию фрагментов

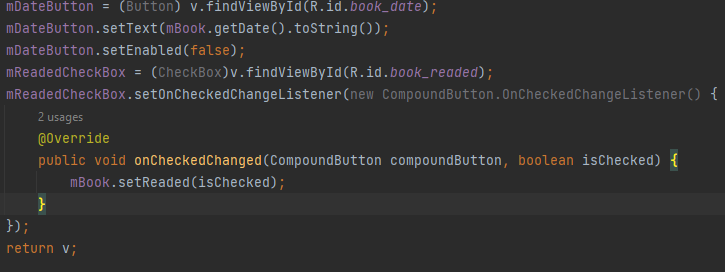


# **7. MATERIAL DESIGN**

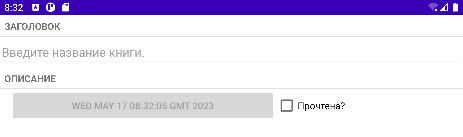
1. Добавил полей в класс Book и обновил макет

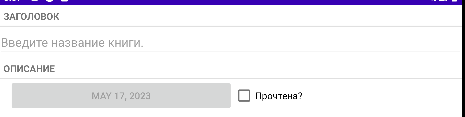
<string name="book\_title\_label">Заголовок</string>  
<string name="book\_details\_label">Описание</string>  
<string name="book\_readed\_label">Прочтена?</string>



2.Подключил виджеты

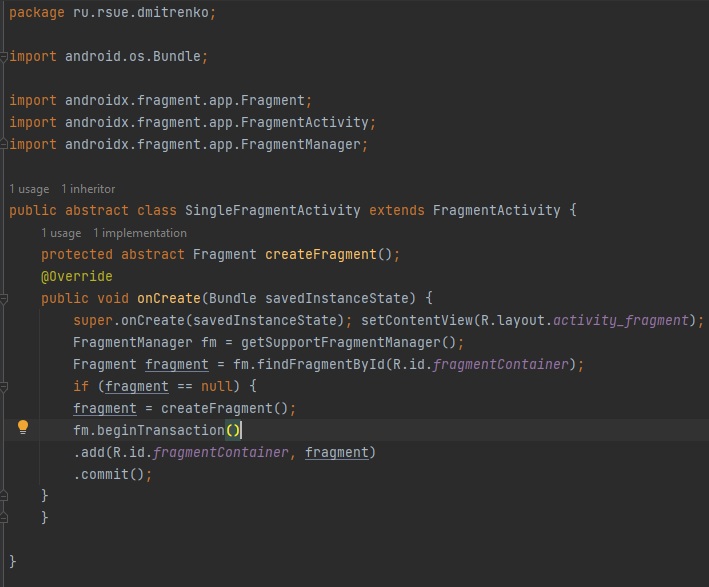


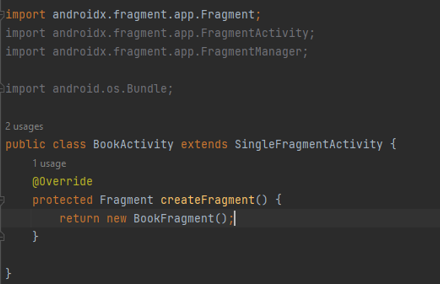
3. Сделал альбомную ориентацию

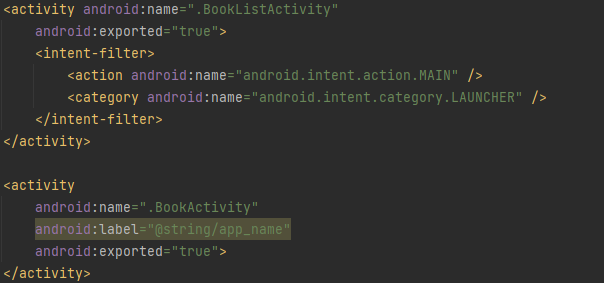


4. Сделал форматирование даты

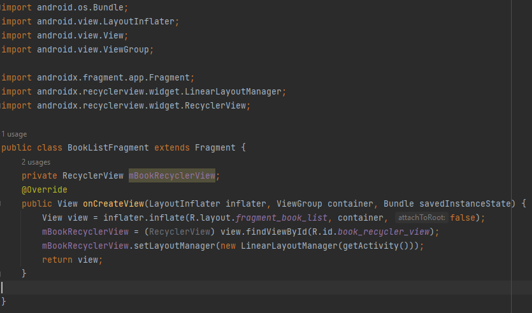
# **8. РАБОТА СО СПИСОЧНЫМИ ЭЛЕМЕНТАМИ ИНТЕРФЕЙСА**

1. Сделал новый класс  


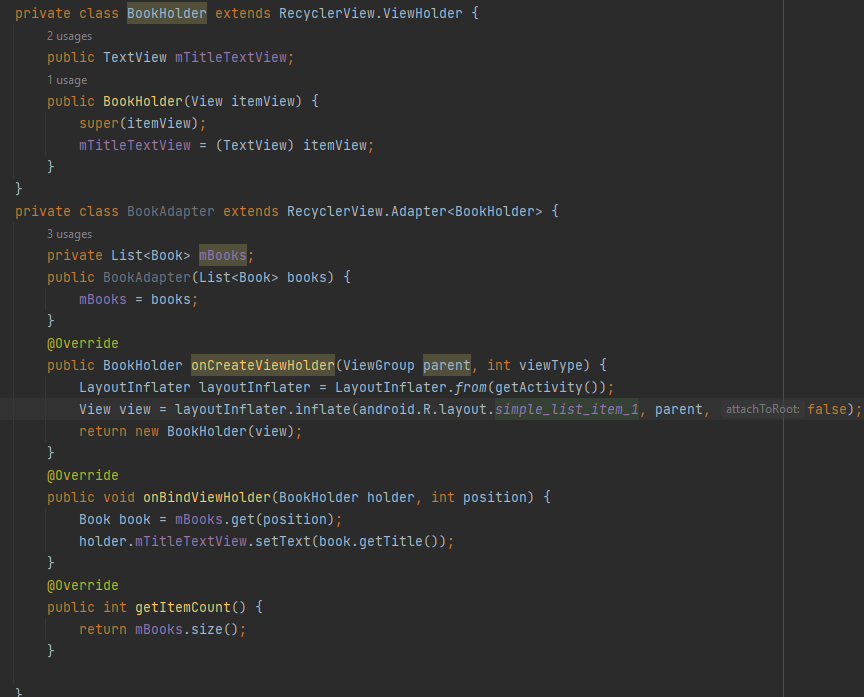
1. Переработал класс BookActivity  
   



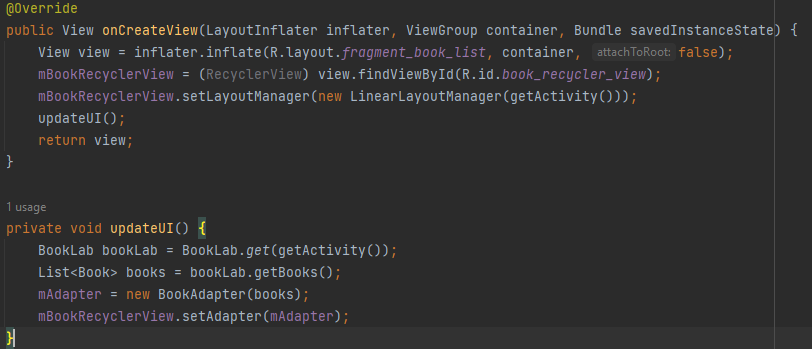
1. Приобрёл BookListActivity активностью лаунчера (AndroidManifest.xml)



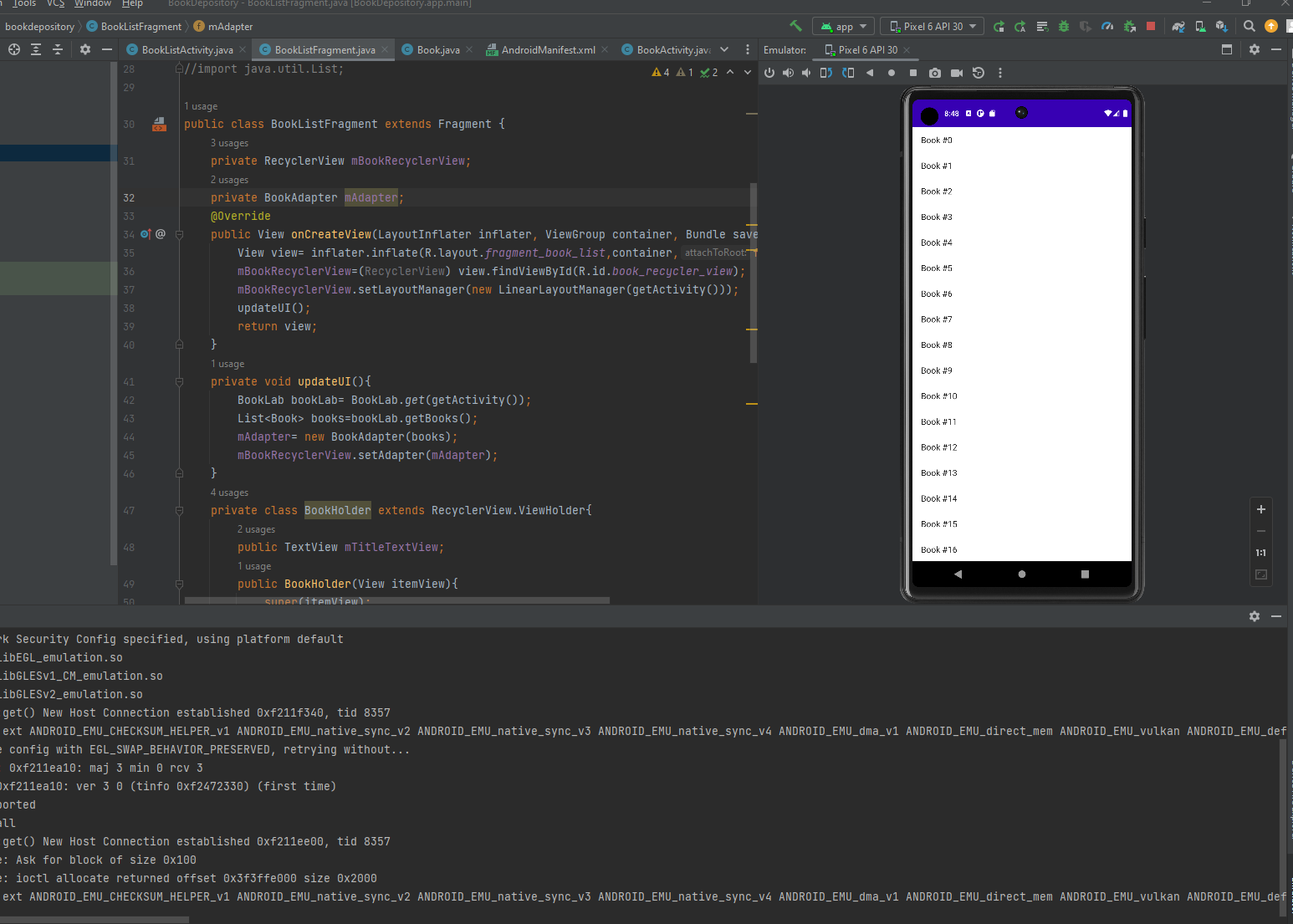
1. Включил RecyclerView



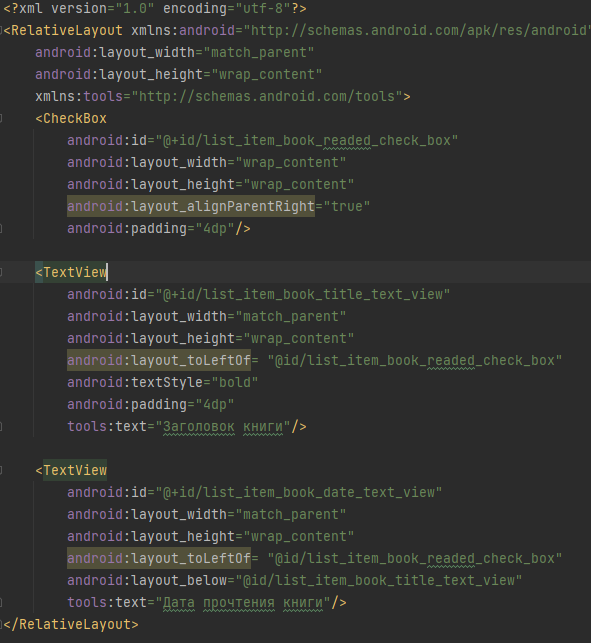
1. Создал адаптер



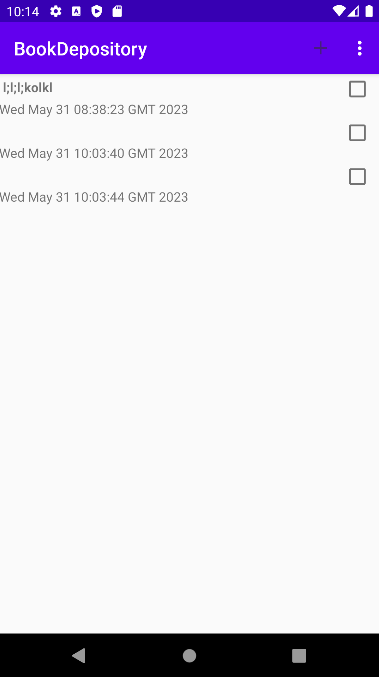
1. Подготовил адаптер (BookListFragment.java)



1. Реализовал новый список RecyclerView

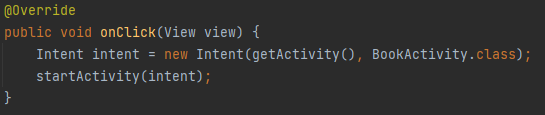


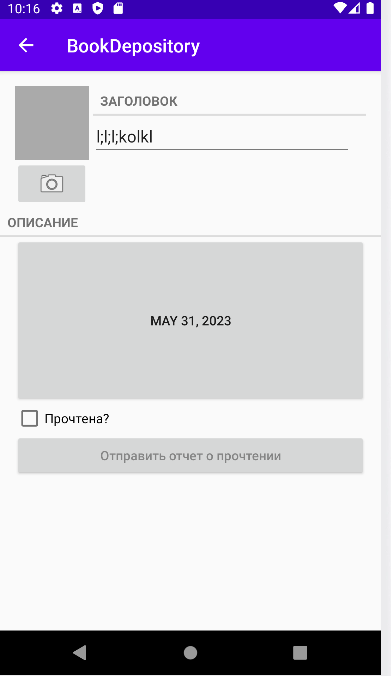
1. Сделал новый макет list\_item\_book.xml



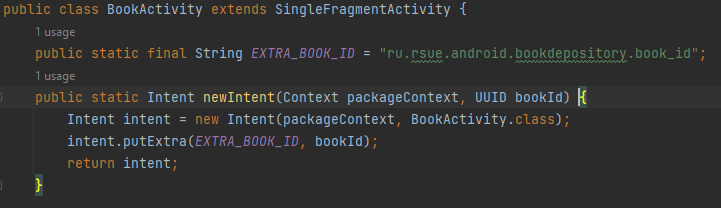
1. Сделал вывод списков и LISTFRAGMENT

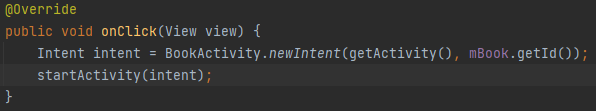
# **9. ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД В РАЗРАБОТКЕ КОДА ИНФОРМАЦИОННОЙ СИТЕМЫ**



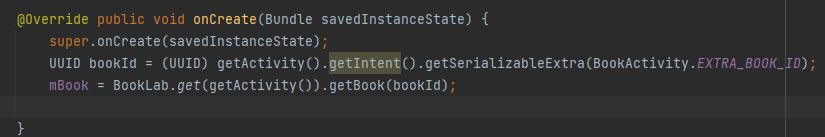


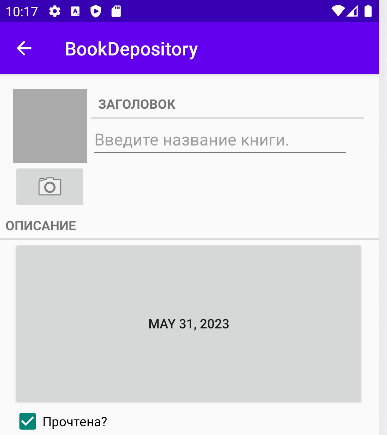
1. Запустил активность из фрагмента

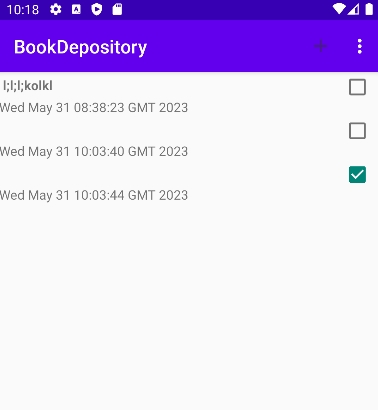




1. Использовал дополнения

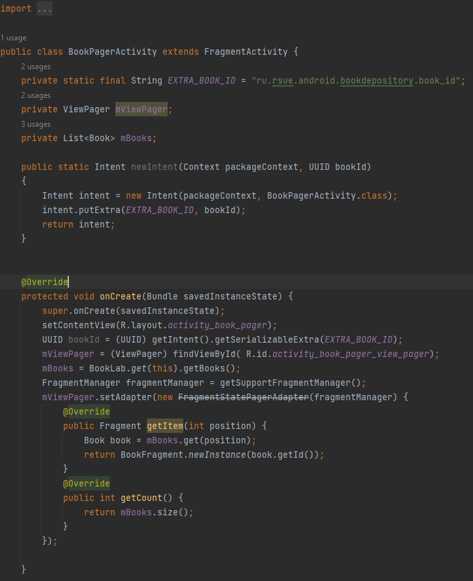


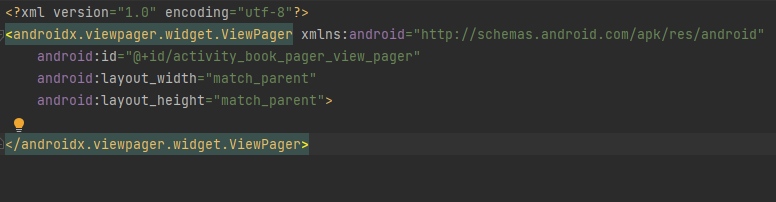
1. Чтение дополнений  
   
2. Появился экземпляр BookFragment с правильными данными книги



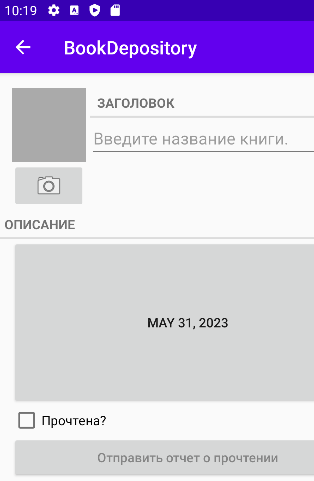
5. Сделал перезагрузку списка

# **10. СОЗДАНИЕ ПРОЦЕДУР ОБРАБОТКИ ИНФОРМАЦИИ.**

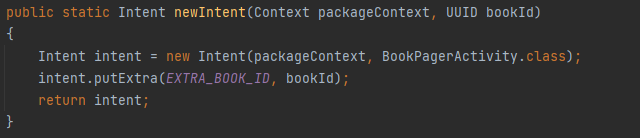
1.Создал класс BookPagerActivity и заполнил его  


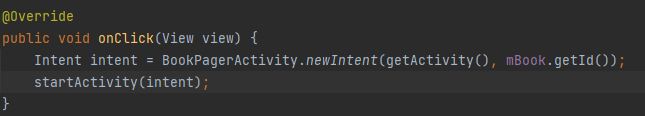
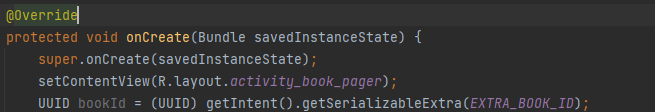


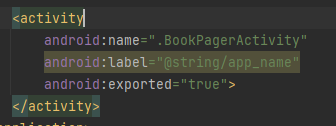
2.Создал ещё новый макет



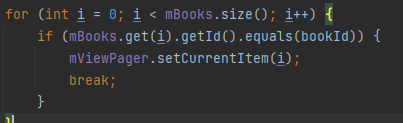








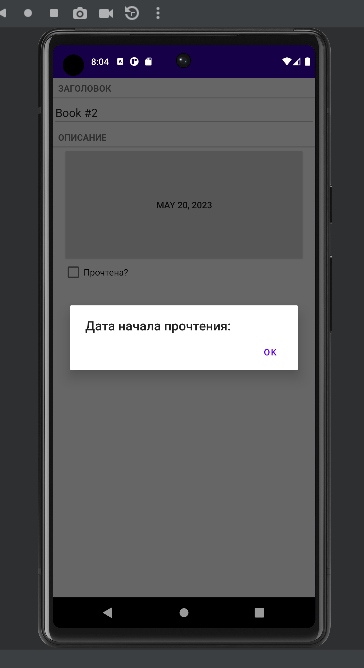
3. Сделал прокрутку страниц



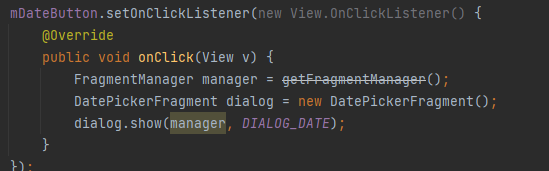
1. Сделал назначение исходного элемента (BookPagerActivity.java)

# **11. ОБЕСПЕЧЕНИЕ ДОПОЛНИТЕЛЬНОЙ ФУНКЦИОНАЛЬНОСТИ СИСТЕМЫ.**

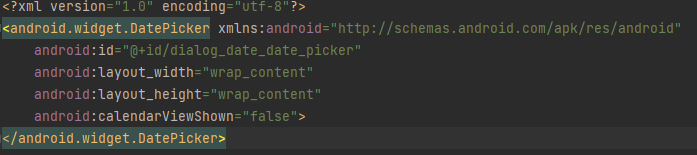
1.Создал новый класс  

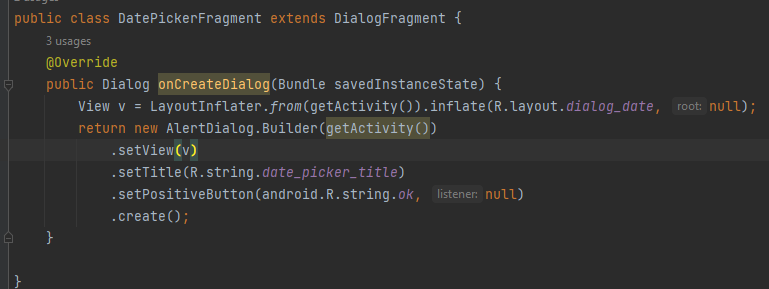


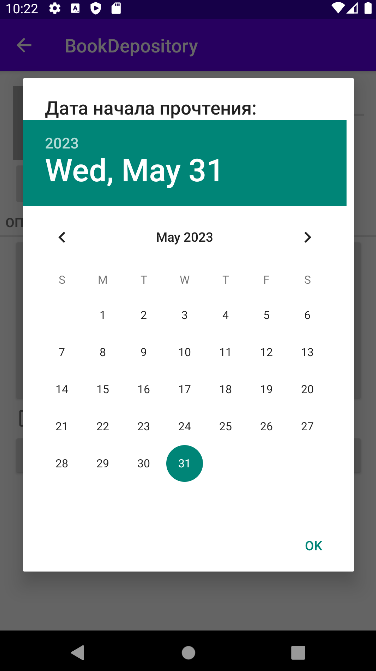


2. Сделал отображение DialogFragment

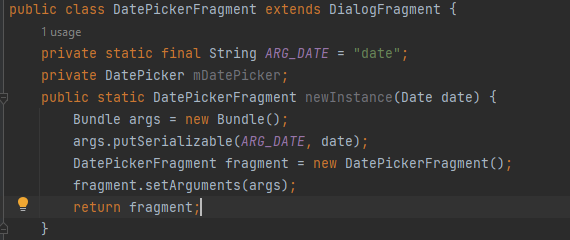


3. Добавил новый макет dialog\_date

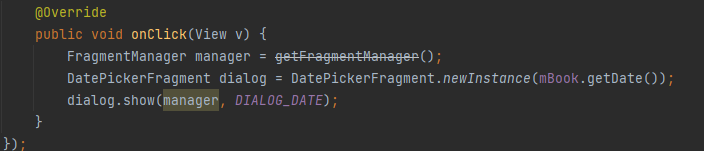
  
  
4. Проверил работает ли отображение дат



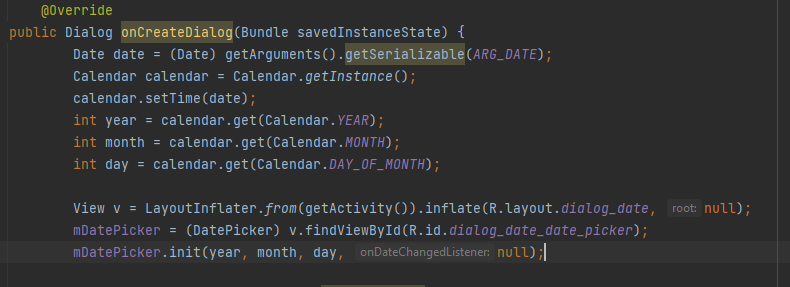
5. Подключил DatePicker в AlertDialog (DatePickerFragment.java)



6. Добавил метод newInstance(Date) (DatePickerFragment.java)

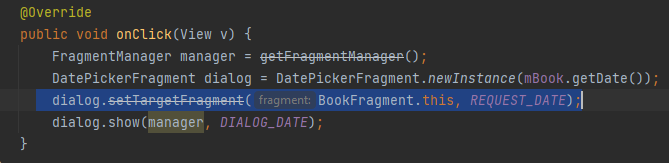


7.Добавил вызов newInstance() (BookFragment.java)



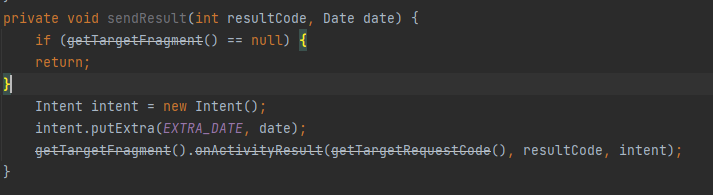
8. Дату и инициализацию DatePicker (DatePickerFragment.java)



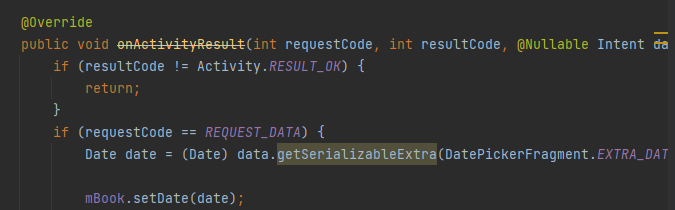


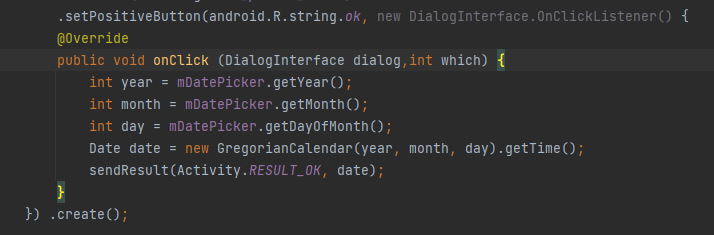
9.Назначил целевой фрагмент (BookFragment.java)



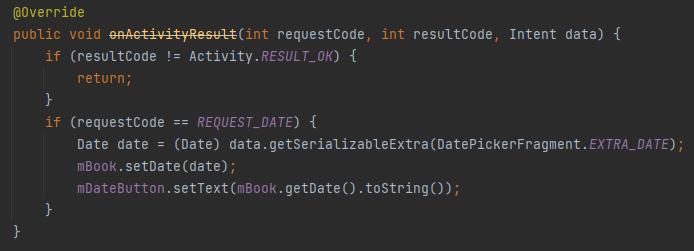


10. Сделал обратный вызов целевого фрагмента (DatePickerFragment.java)

  
11. Убедился в том, что в диалоговом окне теперь отображается DatePicker. На устройствах с версией Lollipop и выше отображается календарный виджет

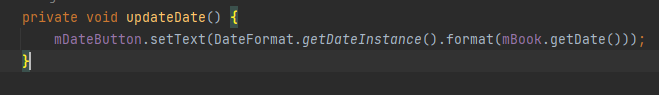


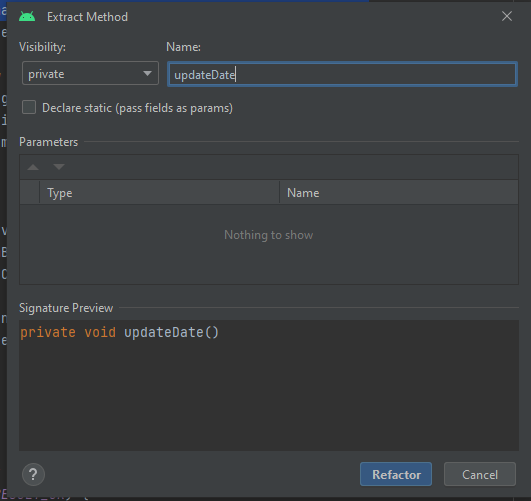
12. Передал информацию (DatePickerFragment.java)



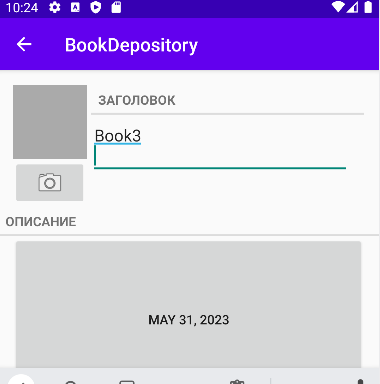
13. Реакция на получение данных от диалогового окна (BookFragment.java)





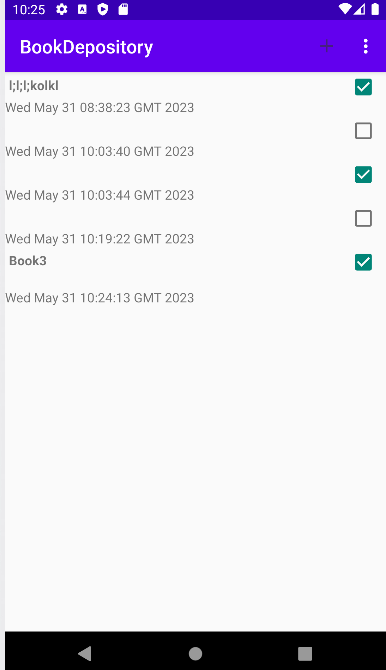


14. Сделал выделение кода в метод updateDate() (BookFragment.java)

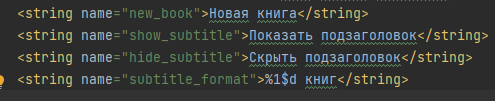


15.Запустил приложение и проверил, что я могу изменять дату

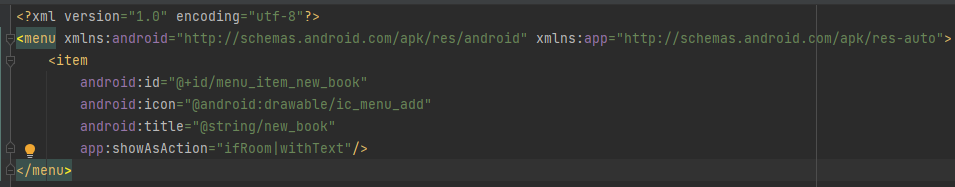
# **12. INTENTS**



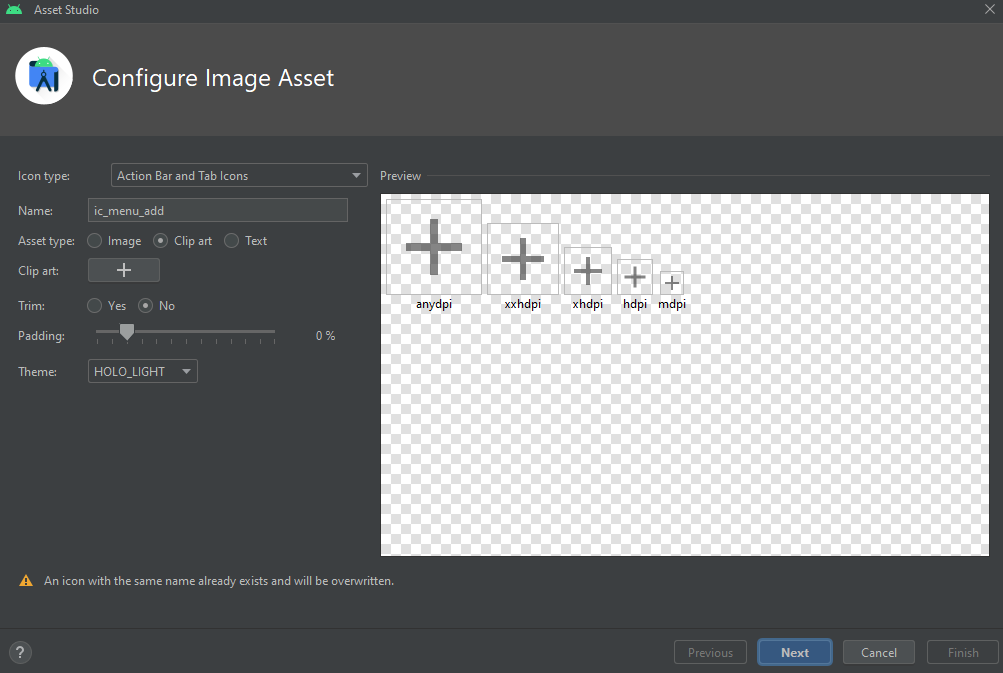
1. Сделал темную тему

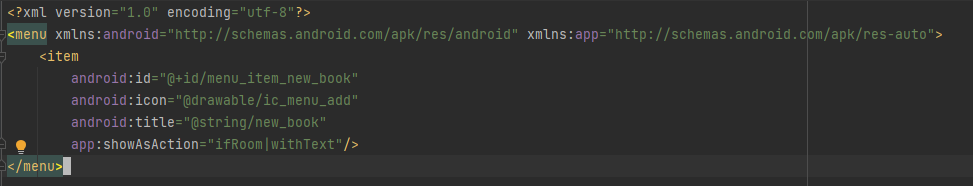


1. Добавил строки для меню

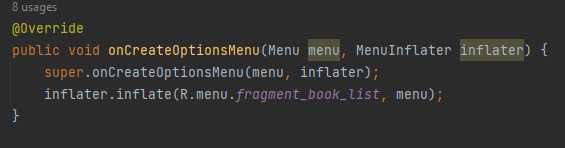


1. Создал файл меню

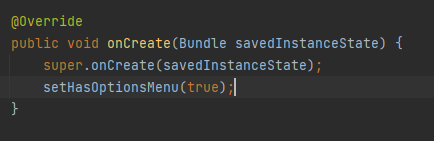


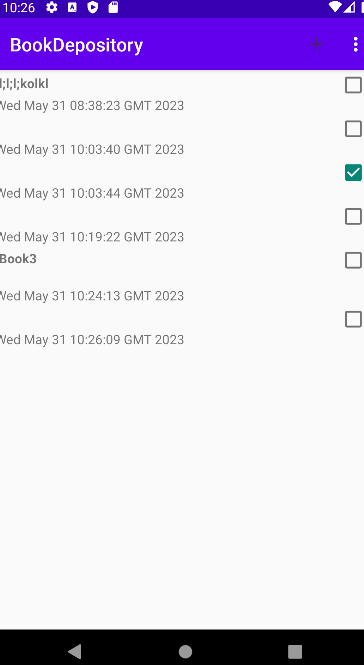


1. Добавил иконки и ссылку на локальный ресурс

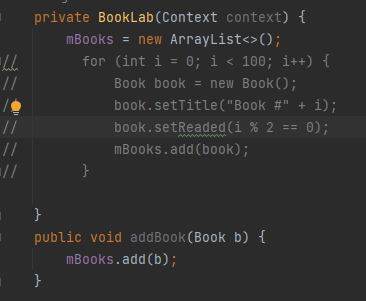


1. Заполнил ресурс меню (BookListFragment.java)

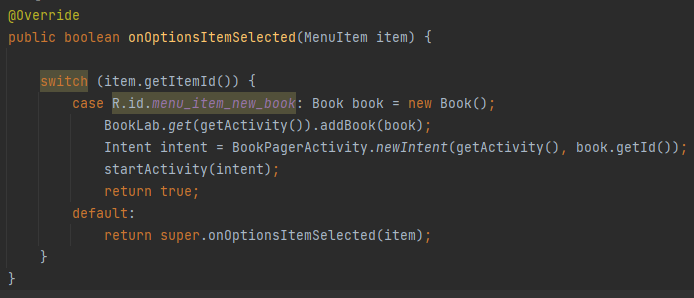




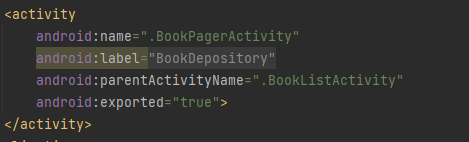
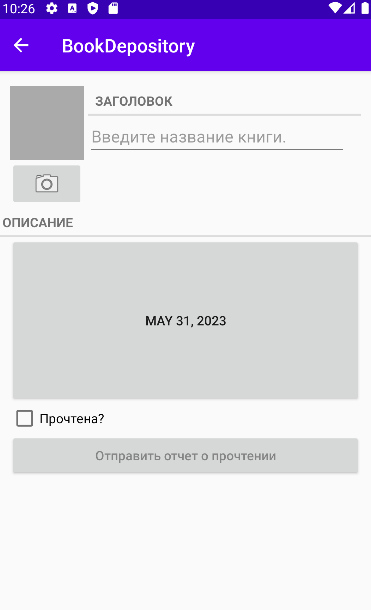
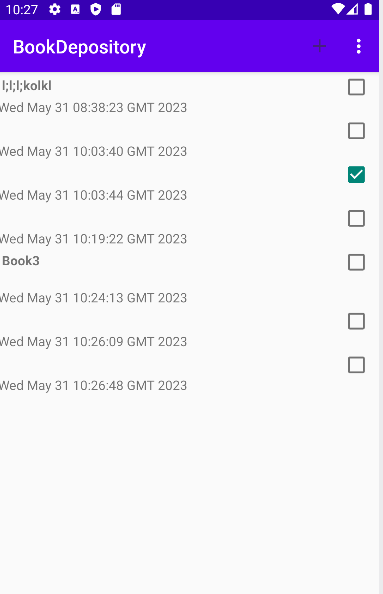
1. Получил обратные вызовы (BookListFragment.java)



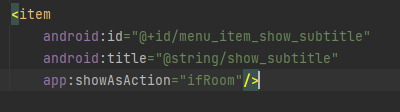
1. Добавил новый объект Book (BookLab.java)

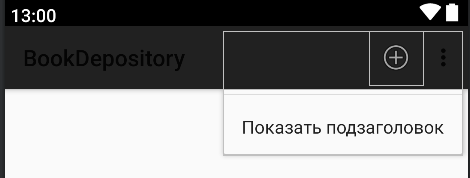


1. Добавил реакцию на выбор команды меню (BookListFragment.java)

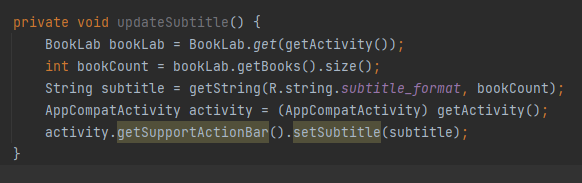
  
 

1. Создал добавление новой книги

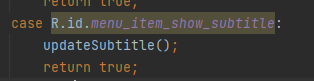




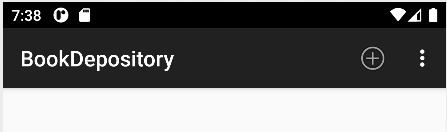
10. Добавил элемент действия Show Subtitle(res/menu/fragment\_book\_list.xml)



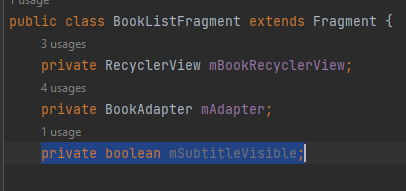
11. Назначил подзаголовок панели инструментов (BookListFragment.java)

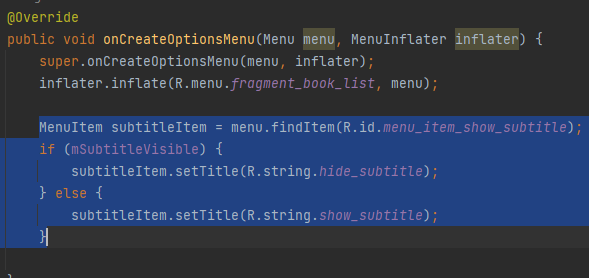


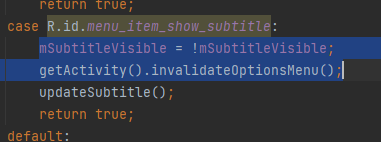
12. Обработал элемент действия Show Subtitle (BookListFragment.java)



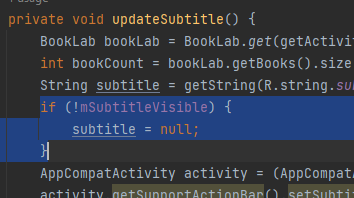
13.Убедилась в том, что в подзаголовке отображается количество книг



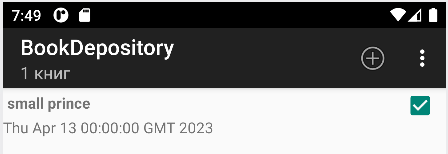


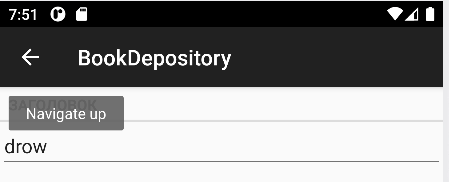


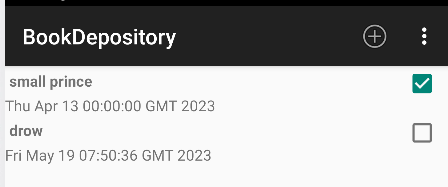
14.Обновил MenuItem (BookListFragment.java)



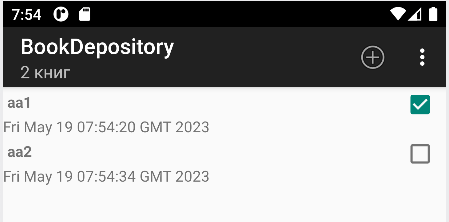
15.Отобразил подзаголовок (BookListFragment.java)

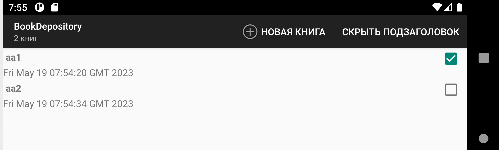






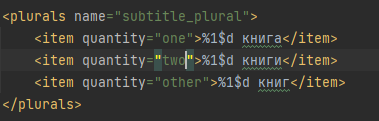
16.Отображается правильное количество книг

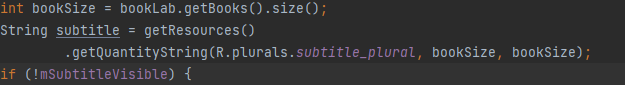


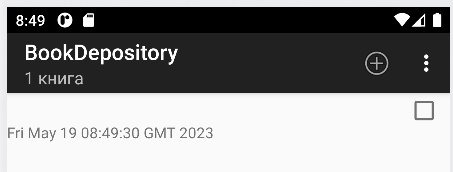


17. Подзаголовок сохранился при повороте приложения

**Задание2:**

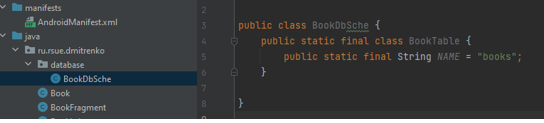




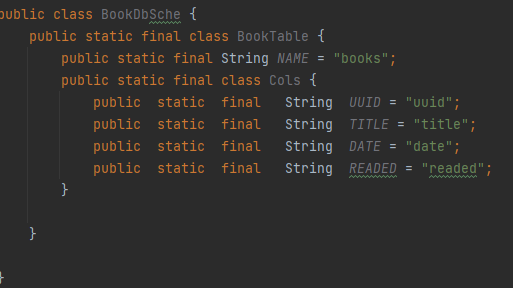


Множественное число в строках

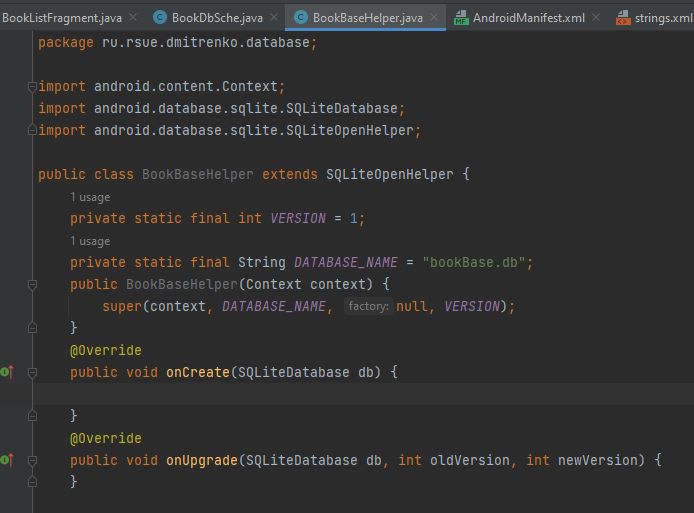
# **13. ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ**



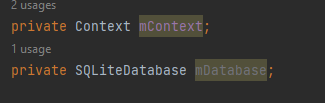
1. Создал новый класс

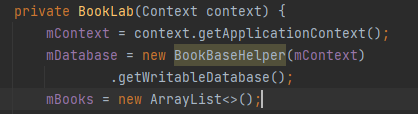


1. Определение столбцов таблицы (BookDbSchema.java)

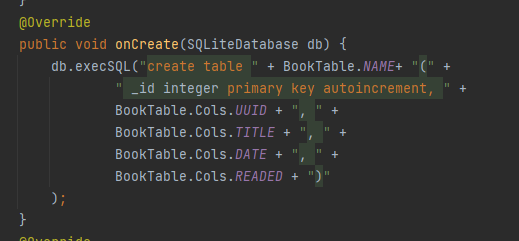


1. Добавил BookBaseHelper (BookBaseHelper.java)

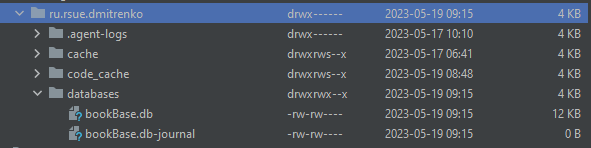




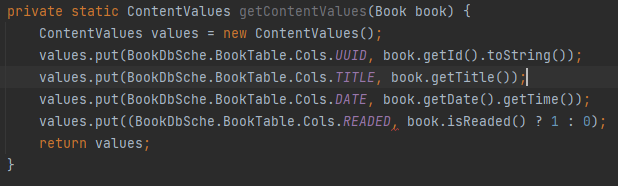
1. Открытие SQLiteDatabase (BookLab.java)



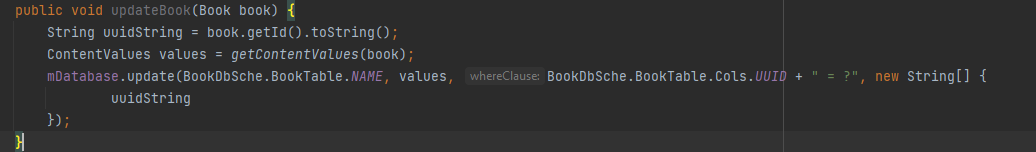
1. Создал таблицу (BookBaseHelper.java)



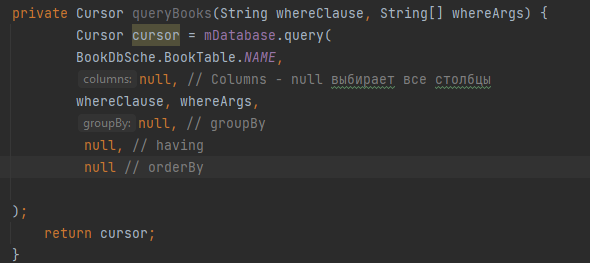
1. База данных добавлена



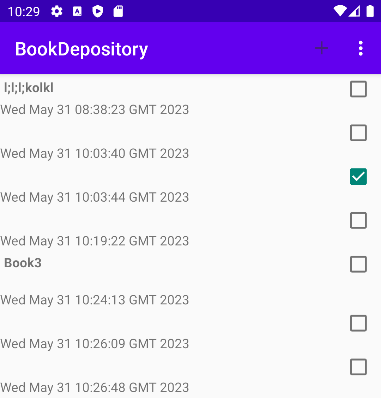
1. Сделал запись в базу данных



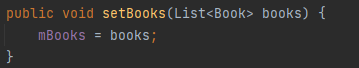
1. Сделал обновление записи (BookLab.java)

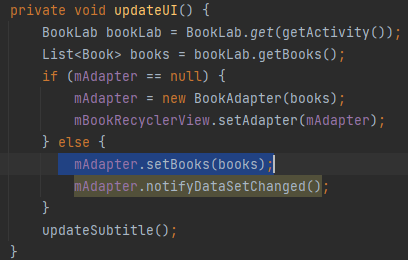


1. Чтение из базы данных



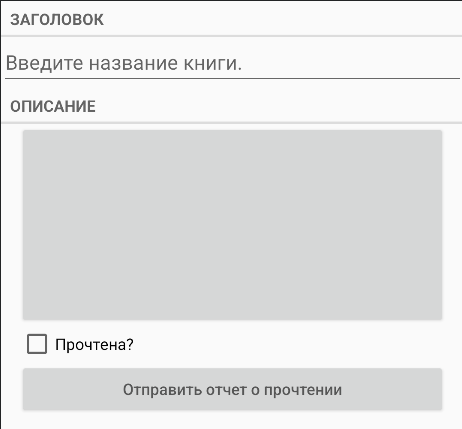
1. Приложение работает стабильно



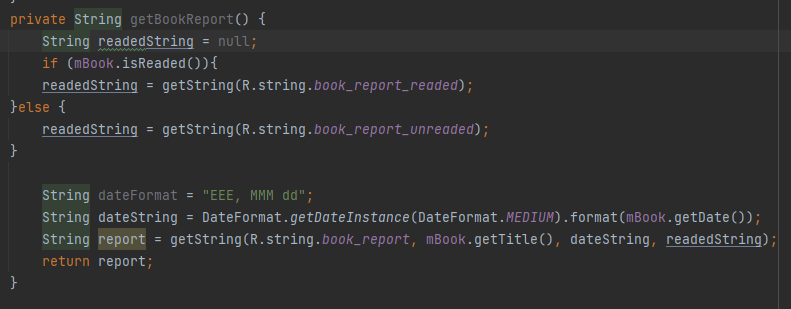


11. Добавил setBooks(List<Book>) и вызвал его в (BookListFragment.java)

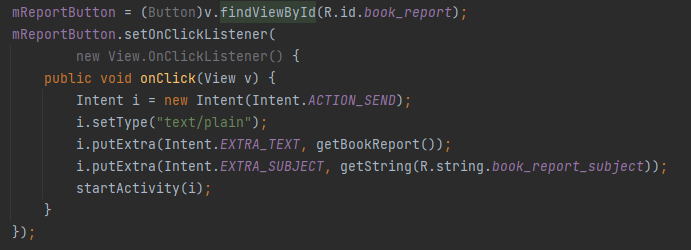
# **14. ОТЛАДКА ПРИЛОЖЕНИЯ.**



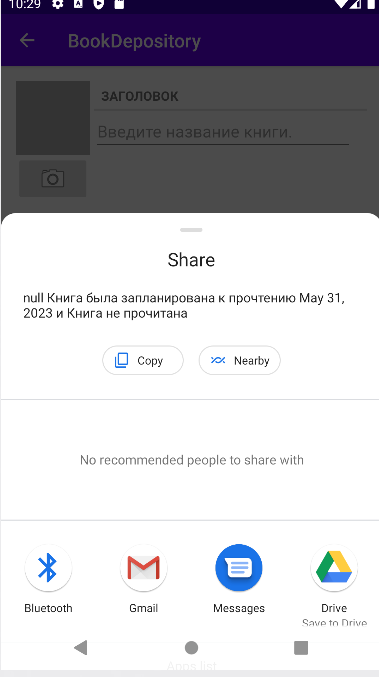
1. Добавил кнопку



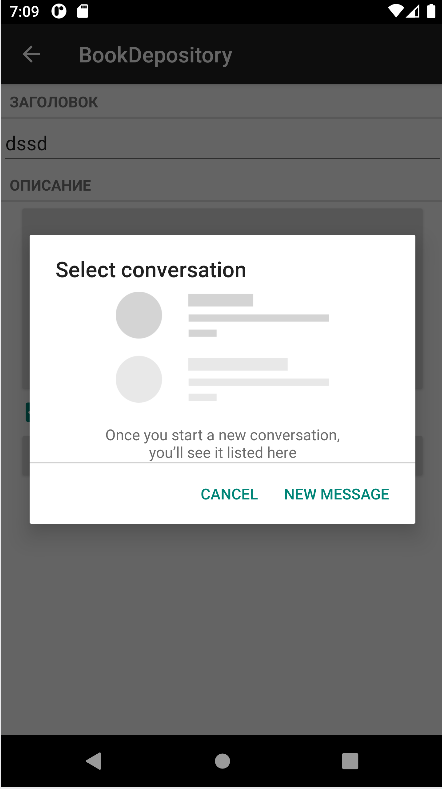
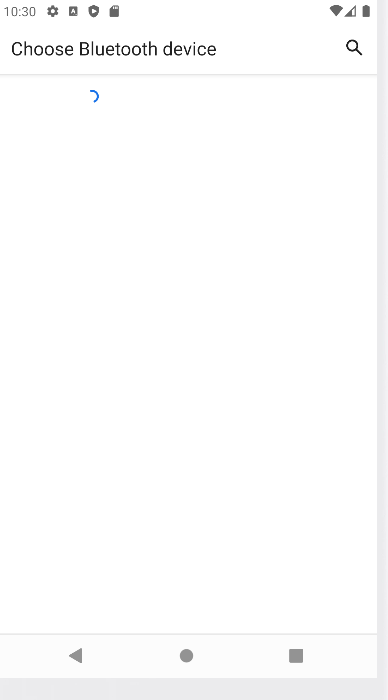
1. Добавил метод getBookReport() (BookFragment.java)



1. Отправка отчета о книге (BookFragment.java)

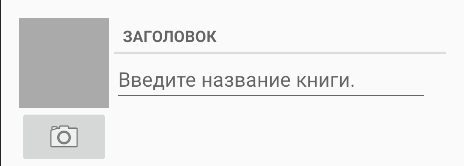


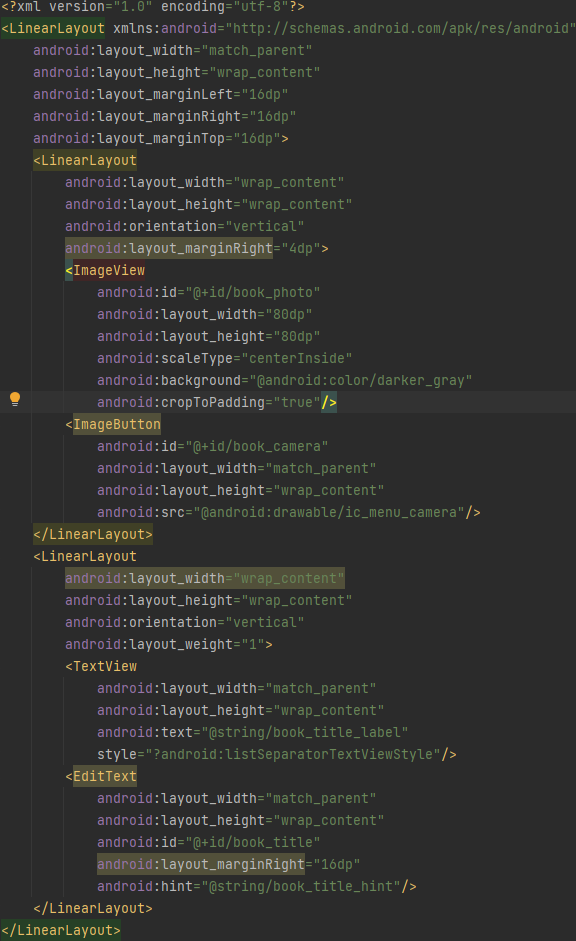
1. Появился список активностей

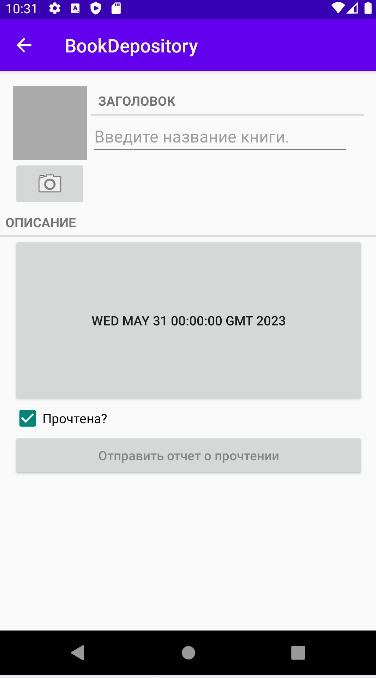
5. Приложение готово

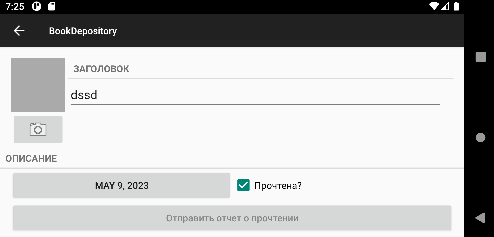
# **15. ЗАЩИТА ПРИЛОЖЕНИЯ. ОЦЕНКА КАЧЕСТВА И НАДЕЖНОСТИ ПРИЛОЖЕНИЯ**



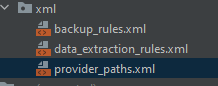
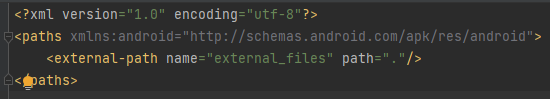


1. Создал новый макет view\_camera\_and\_title.xml

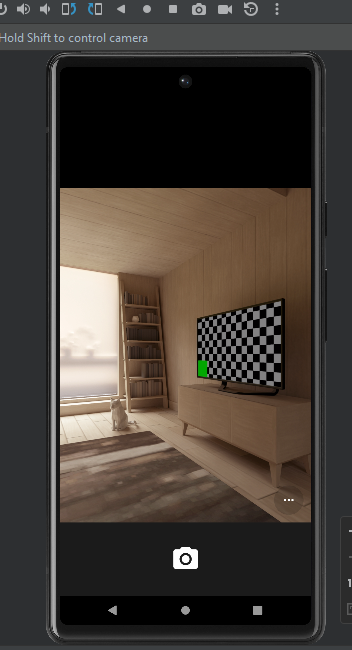




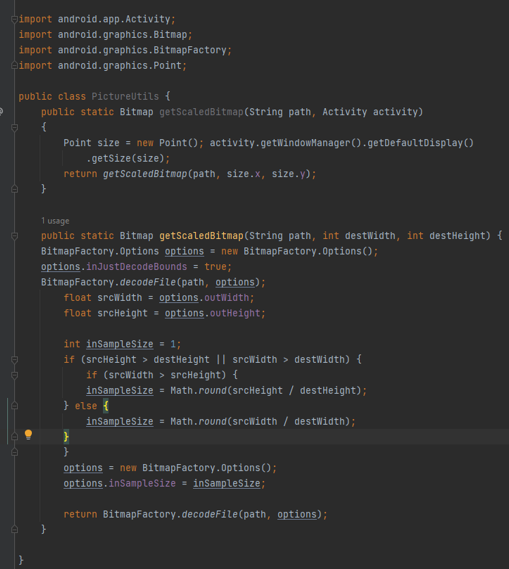
2. Включил макет камеры



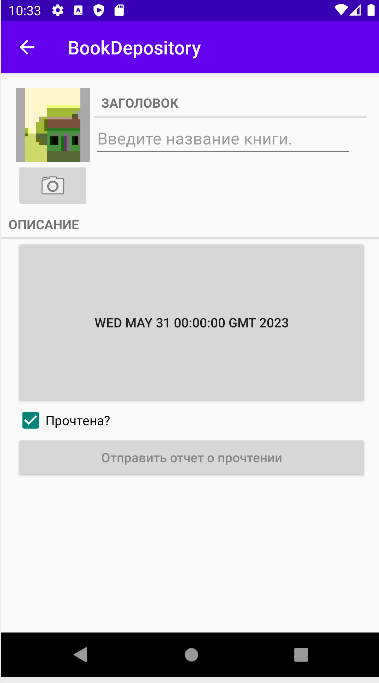
3. Создал новый макет



4.Запустил приложение и открыл камеру



5.Создал новый класс PictureUtils



6.Приложение готово, фотография сохраняется

# **ПРИЛОЖЕНИЯ**

**https://github.com/**