Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

# Отчет по программе «Практическое занятие 19»

Выполнил: Меньшиков Артур Андреевич

Группа: ПР-22

Преподаватель: Мирошниченко Г. В.

2023

**Задание:**

Используя UI-фрагменты и FragmentManager, начать строить приложение

CriminalIntent. Оно предназначено для хранения информации об «офисных

преступлениях»: грязной посуде, оставленной в раковине, или пустом лотке общего

принтера после печати документов.

**Входные и выходные данные**

id – UUID

title – string/строки

isSloved – Boolean/**логический тип данных**

date – LocalDate/неизменяемый объект даты и времени

titleField – Edittext

**Листинг программы (если есть)**

**Activity\_main**

<?xml version="1.0" encoding="utf-8"?>  
<FrameLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:id="@+id/fragment\_container"  
 android:layout\_height="match\_parent">  
  
</FrameLayout >

**Crime**

package com.example.prak\_19  
  
import java.time.LocalDate  
import java.util.UUID  
  
data class crime(  
 var id: UUID = UUID.randomUUID(), var title: String, var isSloved: Boolean,  
 val date: LocalDate ) {  
}

**frahment\_crime**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
  
 android:layout\_margin="16dp">  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 style="?android:listSeparatorTextViewStyle"  
 android:text="@string/crime\_title\_label"/>  
  
 <EditText  
 android:id="@+id/crime\_title"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="@string/crime\_title\_hint"/>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 style="?android:listSeparatorTextViewStyle"  
 android:text="@string/crime\_details\_label"/>  
  
 <Button  
 android:id="@+id/crime\_date"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 tools:text="Wed Nov 14 11:56 EST 2018"/>  
  
 <CheckBox  
 android:id="@+id/crime\_sloved"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/crime\_sloved\_label"/>  
</LinearLayout

**CrimeFragment**

package com.example.prak\_19  
  
import android.os.Bundle  
import android.text.Editable  
import android.text.TextWatcher  
import android.view.LayoutInflater  
import android.view.View  
import android.view.ViewGroup  
import android.widget.Button  
import android.widget.CheckBox  
import android.widget.EditText  
import androidx.fragment.app.Fragment  
import java.time.LocalDate  
import java.util.UUID  
  
class CrimeFragment : Fragment() {  
  
 private lateinit var crime: crime  
 private lateinit var titleField: EditText  
 private lateinit var dateButton: Button  
 private lateinit var solvedCheckBox: CheckBox  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 crime = crime( id = UUID.randomUUID(), title = "Кража", isSloved = true, date = LocalDate.now())  
 }  
  
 override fun onCreateView(  
 inflater: LayoutInflater,  
 container: ViewGroup?,  
 savedInstanceState: Bundle?  
 ): View? {  
 val view = inflater.inflate(R.layout.frahment\_crime, container, false)  
 titleField = view.findViewById(R.id.crime\_title) as EditText  
  
 dateButton = view.findViewById(R.id.crime\_date) as Button  
 dateButton.apply **{** text = crime.date.toString()  
 isEnabled = false  
 **}** solvedCheckBox = view.findViewById(R.id.crime\_sloved) as CheckBox  
 return view  
 }  
  
 override fun onStart() {  
 super.onStart()  
 val titleWatcher = object : TextWatcher  
 {  
 override fun beforeTextChanged(s: CharSequence?, start: Int, count: Int, after: Int) {  
  
 }  
  
 override fun onTextChanged(s: CharSequence?, start: Int, before: Int, count: Int) {  
 crime.title = s.toString()  
 }  
  
 override fun afterTextChanged(s: Editable?) {  
  
 }  
 }  
 titleField.addTextChangedListener(titleWatcher)  
 solvedCheckBox.apply **{** setOnCheckedChangeListener **{** \_, isChecked **->** crime.isSloved = isChecked **}  
 }** solvedCheckBox.setOnCheckedChangeListener **{** \_, isChecked **->** dateButton.isEnabled = isChecked **}** }  
}

**Strings**

<resources>  
 <string name="app\_name">Prak\_19</string>  
 <string name="crime\_title\_hint">Enter a title for the crime.</string>  
 <string name="crime\_title\_label">Title</string>  
 <string name="crime\_details\_label">Details</string>  
 <string name="crime\_sloved\_label">Sloved</string>  
</resources>

**Themes**

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
  
 <style name="Theme.Prak\_19" parent="Theme.AppCompat.Light.DarkActionBar" />  
</resources>

**AndroidManifest**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools">  
  
 <application  
 android:allowBackup="true"  
 android:dataExtractionRules="@xml/data\_extraction\_rules"  
 android:fullBackupContent="@xml/backup\_rules"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.Prak\_19"  
 tools:targetApi="31">  
 <activity  
 android:name=".MainActivity2"  
 android:theme="@style/Theme.Prak\_19"  
 android:exported="true">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

**Тестовые ситуации**

**↓↓↓↓↓**

