МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования  
 «Кемеровский государственный университет»**

**Институт фундаментальных наук**

**Кафедра ЮНЕСКО по ИВТ**

**ЛАБОРАТОРНАЯ РАБОТА №4**

**ПО ДИСЦИПЛИНЕ “РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ”**

студента 2 курса

**Крючкова Андрея Олеговича**

Направление 09.03.03 – Прикладная информатика в экономике

Преподаватель:

доцент

С.Ю. Завозкин

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Работа защищена:

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_202\_г.

с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_

Кемерово 202\_

СОДЕРЖАНИЕ

[1. Постановка задач 2](#_Toc85363462)

[2. Реализация задач 3](#_Toc85363463)

[Заключение 7](#_Toc85363464)

# 1. Постановка задач

1. Познакомиться с особенностями работы RecyclerView и обработчиками событий для его содержимого;
2. Познакомиться с основными сетевыми взаимодействиями в рамках разрабатываемого приложения на языке Kotlin.

# 2. Реализация задач

**1 Задание**

**Задачи:**

Создать приложение, содержащее в себе динамически наполняемый RecyclerView;

Реализовать обработчик для содержимого RecyclerView;

1. Добавлен в Activity\_main.xml объект recyclerView с id “rView”

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout 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"  
 android:orientation="vertical"  
 tools:context=".MainActivity">  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/rView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" />  
</LinearLayout>

Код Activity\_main.xml

1. Создана новая схема компановки для объектов recyclerView

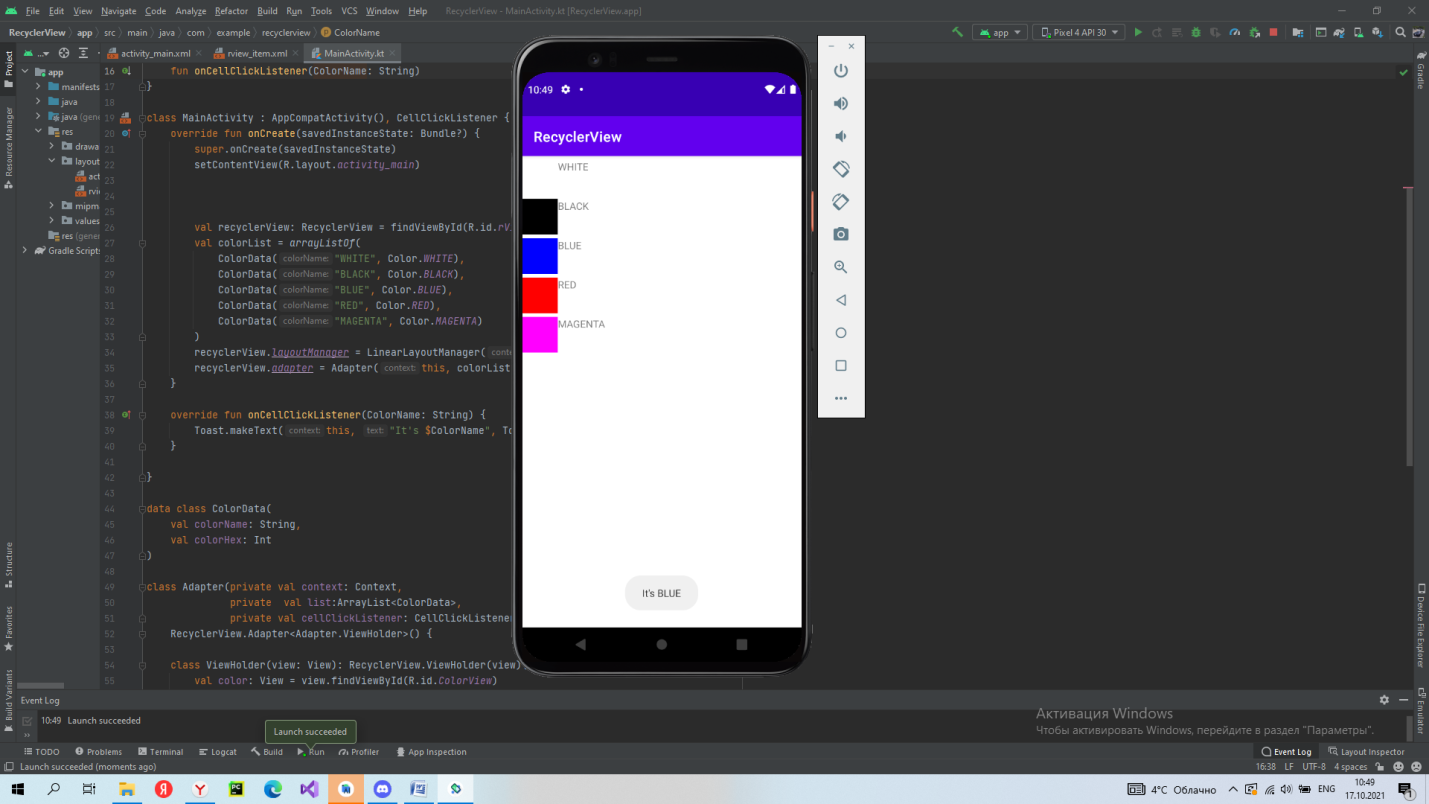
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:visibility="visible">  
  
 <View  
 android:id="@+id/ColorView"  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="5dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.0" />  
  
 <TextView  
 android:id="@+id/TextViewNameColor"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="5dp"  
 app:layout\_constraintStart\_toEndOf="@+id/ColorView"  
 app:layout\_constraintTop\_toTopOf="parent" />  
</LinearLayout>

Код rview\_item.xml

1. Реализованы: класс Адаптер, наполнение recyclerView объектами, появление тоста при нажатии на элемент recyclerView.

Package com.example.recyclerview  
  
import android.content.Context  
import android.graphics.Color   
import ndroid.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.LayoutInflater  
import android.view.View  
import android.view.ViewGroup  
import android.widget.TextView  
import android.widget.Toast  
import ndroid.recyclerview.widget.LinearLayoutManager  
import ndroid.recyclerview.widget.RecyclerView  
  
interface CellClickListener{  
 fun onCellClickListener(ColorName: String)  
}  
  
class MainActivity : AppCompatActivity(), CellClickListener {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
  
  
 val recyclerView: RecyclerView = findViewById(R.id.*rView*)  
 val colorList = *arrayListOf*(  
 ColorData(“WHITE”, Color.*WHITE*),  
 ColorData(“BLACK”, Color.*BLACK*),  
 ColorData(“BLUE”, Color.*BLUE*),  
 ColorData(“RED”, Color.*RED*),  
 ColorData(“MAGENTA”, Color.*MAGENTA*)  
 )  
 recyclerView.*layoutManager* = LinearLayoutManager(this)  
 recyclerView.*adapter* = Adapter(this, colorList, this)  
 }  
  
 override fun onCellClickListener(ColorName: String) {  
 Toast.makeText(this, “It’s $ColorName”, Toast.*LENGTH\_SHORT*).show()  
 }  
  
}  
  
data class ColorData(  
 val colorName: String,  
 val colorHex: Int  
)  
  
class Adapter(private val context: Context,  
 private val list:ArrayList<ColorData>,  
 private val cellClickListener: CellClickListener):  
 RecyclerView.Adapter<Adapter.ViewHolder>() {  
  
 class ViewHolder(view: View): RecyclerView.ViewHolder(view){  
 val color: View = view.findViewById(R.id.*ColorView*)  
 val name: TextView = view.findViewById(R.id.*TextViewNameColor*)  
 }  
  
  
 override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder {  
 val view = LayoutInflater.from(context).inflate(R.layout.*rview\_item*, parent, false)  
 return ViewHolder(view)  
 }  
  
 override fun getItemCount(): Int {  
 return list.*count*()  
 }  
  
 override fun onBindViewHolder(holder: ViewHolder, position: Int) {  
 val data = list[position]  
  
 holder.color.setBackgroundColor(data.colorHex)  
 holder.name.*text* = data.colorName  
 holder.itemView.setOnClickListener**{** cellClickListener.onCellClickListener(data.colorName)  
 **}** }  
  
  
}

Код MainActivity.kt

4) 

Итоговое приложение

**2 Задание**

**Задачи:**

Реализовать приложение, взаимодействующее со статическими ресурсами в сети Интернет.

1. Добавлены 2 кнопки на линейную схему компановки с отцентрированием элементов главной схемы.

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout 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:orientation="vertical"  
 android:gravity="center"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <Button  
 android:id="@+id/btnHTTP"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Get via HTTP" />  
  
 <Button  
 android:id="@+id/btnOkHTTP"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Get via OkHTTP" />  
</LinearLayout>

Код Activity\_main.xml

1. Созданы обработчики кнопок

package com.example.internettest  
  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.util.Log  
import android.widget.Button  
import okhttp3.\*  
import java.net.HttpURLConnection  
import java.net.URL  
import java.io.IOException  
  
class MainActivity : AppCompatActivity() {  
  
 val URL = "https://api.flickr.com/services/rest/?method=flickr.photos.search&api\_key=ff49fcd4d4a08aa6aafb6ea3de826464&tags=cat&format=json&nojsoncallback=1"  
 var okHttpClient: OkHttpClient = OkHttpClient()  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 val buttonHTTP: Button = findViewById(R.id.*btnHTTP*)  
 val buttonOkHTTP: Button = findViewById(R.id.*btnOkHTTP*)  
  
  
  
 buttonHTTP.setOnClickListener **{** val connection =  
 URL("https://api.flickr.com/services/rest/?method=flickr.photos.search&api\_key=ff49fcd4d4a08aa6aafb6ea3de826464&tags=cat&format=json&nojsoncallback=1").openConnection() as HttpURLConnection  
 try {  
  
 Thread(*Runnable* **{** val data = connection.*inputStream*.*bufferedReader*().*readText*()  
 Log.d("Flickr cats", data)  
 **}**).start()  
  
 } finally {  
 connection.disconnect()  
 }  
  
 **}** buttonOkHTTP.setOnClickListener **{** loadReq()  
 **}** }  
  
  
 private fun loadReq(){  
  
  
 val request: Request = Request.Builder().url(URL).build()  
 okHttpClient.newCall(request).enqueue(object : Callback {  
 override fun onFailure(call: Call, e: IOException) {  
 }  
  
 override fun onResponse(call: Call, response: Response) {  
 val json = response.body?.string()  
  
 runOnUiThread **{** Log.i("Flickr OkCats", json.*toString*())  
 **}** }  
 })  
 }  
  
  
}

Код MainActivity.kt

# Заключение

* Я ознакомился с особенностями работы RecyclerView и обработчиками событий для его содержимого.
* Познакомился с основными сетевыми взаимодействиями в рамках разрабатываемого приложения на языке Kotlin.

**Литература**

1. Aayush Puranik. Recycler View using Kotlin and xml - <https://aayushpuranik.medium.com/recycler-view-using-kotlin-and-xml-afc8dc465bf9>
2. Aayush Puranik. Recycler View Using kotlin with cell click listener - <https://aayushpuranik.medium.com/recycler-view-using-kotlin-with-click-listener-46e7884eaf59>
3. HttpURLConnection - <https://developer.android.com/reference/kotlin/java/net/HttpURLConnection>
4. How to add manifest permission to an application - <https://stackoverflow.com/questions/2169294/how-to-add-manifest-permission-to-an-application>
5. NetworkOnMainThreadException - <https://stackoverflow.com/questions/29802323/android-with-kotlin-how-to-use-httpurlconnection>
6. OkHTTP - <https://square.github.io/okhttp/>
7. RecyclerView - <http://developer.alexanderklimov.ru/android/views/recyclerview-kot.php>
8. Android-kotlin-samples. HttpUrlConnectionAsyncActivity -  <https://github.com/irontec/android-kotlin-samples/blob/master/KotlinTest/app/src/main/java/com/irontec/examples/kotlintest/HttpUrlConnectionAsyncActivity.kt>

Android-приложение с фактами о Чаке Норрисе на Kotlin - <https://habr.com/ru/post/450488>