Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «Практическое занятие 23»

Выполнил: Меньшиков Артур Андреевич

Группа: ПР-32

Преподаватель: Мирошниченко Г. В.

2024

**Задание:**

Создание приложения PhotoGallery

**Входные и выходные данные**

url: "services/rest/?method=flickr.interestingness.getList" +  
 "&api\_key=7d0fca767060c3f44bea0c36309e4cd3" +  
 "&format=json" +  
 "&nojsoncallback=1" +  
 "&extras=url\_s"

**Листинг программы (если есть)**

XML:

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

<?xml version="1.0" encoding="utf-8"?>  
 <androidx.recyclerview.widget.RecyclerView  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/photo\_recycler\_view"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:padding="8dp">  
  
 <ImageView  
 android:id="@+id/photo\_image"  
 android:layout\_width="match\_parent"  
 android:layout\_height="200dp"  
 android:scaleType="centerCrop" />  
  
</LinearLayout>

Class:

package com.example.prak23  
  
import android.util.Log  
import androidx.lifecycle.LiveData  
import androidx.lifecycle.MutableLiveData  
import com.example.prak23.api.FlickrApi  
import retrofit2.Call  
import retrofit2.Callback  
import retrofit2.Response  
import retrofit2.Retrofit  
import retrofit2.converter.gson.GsonConverterFactory  
  
private const val *TAG* = "FlickrFetchr"  
  
class FlickrFetchr {  
  
 private val flickrApi: FlickrApi  
  
 init {  
 val retrofit: Retrofit =  
 Retrofit.Builder()  
 .baseUrl("https://api.flickr.com")  
 .addConverterFactory(GsonConverterFactory.create())  
 .build()  
  
 flickrApi = retrofit.create(FlickrApi::class.*java*)  
 }  
  
 fun fetchPhotos(): LiveData<List<GalleryItem>> {  
 val responseLiveData: MutableLiveData<List<GalleryItem>> = MutableLiveData()  
 val flickrRequest: Call<FlickrResponse> = flickrApi.fetchPhotos()  
  
 flickrRequest.enqueue(object : Callback<FlickrResponse> {  
  
 override fun onFailure(call: Call<FlickrResponse>, t: Throwable) {  
 Log.e(*TAG*, "Failed to fetch photos", t)  
 }  
  
 override fun onResponse(call: Call<FlickrResponse>, response: Response<FlickrResponse>) {  
 Log.d(*TAG*, "Response received")  
  
 if (response.*isSuccessful*) {  
 val flickrResponse: FlickrResponse? = response.body()  
 val photoResponse: PhotoResponse? = flickrResponse?.photos  
 var galleryItems: List<GalleryItem> = photoResponse?.galleryItems ?: *mutableListOf*()  
 galleryItems = galleryItems.*filterNot* **{  
 it**.url!!.*isBlank*()  
 **}** responseLiveData.*value* = galleryItems  
 } else {  
 Log.e(*TAG*, "Failed to fetch photos: ${response.code()}")  
 }  
 }  
 })  
  
 return responseLiveData  
 }  
}

package com.example.prak23  
  
class FlickrResponse {  
 lateinit var photos: PhotoResponse  
}

package com.example.prak23  
  
import com.google.gson.annotations.SerializedName  
  
data class GalleryItem(  
 var title: String = "",  
 var id: String = "",  
 @SerializedName("url\_s") var url: String? = null  
)

package com.example.prak23  
  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
  
class PhotoGalleryActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_photo\_gallery*)  
  
 if (savedInstanceState == null) {  
 *supportFragmentManager* .beginTransaction()  
 .add(R.id.*fragmentContainer*, PhotoGalleryFragment.newInstance())  
 .commit()  
 }  
 }  
}

package com.example.prak23  
  
import android.os.Bundle  
import android.util.Log  
import android.view.LayoutInflater  
import android.view.View  
import android.view.ViewGroup  
import android.widget.ImageView  
import android.widget.TextView  
import androidx.fragment.app.Fragment  
import androidx.lifecycle.ViewModelProvider  
import androidx.recyclerview.widget.GridLayoutManager  
import androidx.recyclerview.widget.RecyclerView  
import com.squareup.picasso.Picasso  
  
private const val *TAG* = "PhotoGalleryFragment"  
  
class PhotoGalleryFragment : Fragment() {  
  
 private lateinit var photoGalleryViewModel: PhotoGalleryViewModel  
 private lateinit var photoRecyclerView: RecyclerView  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 photoGalleryViewModel = ViewModelProvider(this).get(PhotoGalleryViewModel::class.*java*)  
 }  
  
 override fun onCreateView(  
 inflater: LayoutInflater,  
 container: ViewGroup?,  
 savedInstanceState: Bundle?  
 ): View {  
 val view = inflater.inflate(R.layout.*fragment\_photo\_gallery*, container, false)  
 photoRecyclerView = view.findViewById(R.id.*photo\_recycler\_view*)  
 photoRecyclerView.*layoutManager* = GridLayoutManager(*context*, 3)  
 return view  
 }  
  
 override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
 super.onViewCreated(view, savedInstanceState)  
 photoGalleryViewModel.galleryItemLiveData.observe(  
 *viewLifecycleOwner* ) **{** galleryItems **->** photoRecyclerView.*adapter* = PhotoAdapter(galleryItems)  
 **}** }  
  
 private class PhotoHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {  
 private val imageView: ImageView = itemView.findViewById(R.id.*photo\_image*)  
  
 fun bindGalleryItem(galleryItem: GalleryItem) {  
 Picasso.get()  
 .load(galleryItem.url)  
 .placeholder(R.drawable.*load*)  
 .into(imageView)  
 }  
 }  
  
 private class PhotoAdapter(private val galleryItems: List<GalleryItem>) : RecyclerView.Adapter<PhotoHolder>() {  
 override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): PhotoHolder {  
 val view = LayoutInflater.from(parent.*context*).inflate(R.layout.*list\_item\_gallery*, parent, false)  
 return PhotoHolder(view)  
 }  
  
 override fun onBindViewHolder(holder: PhotoHolder, position: Int) {  
 val galleryItem = galleryItems[position]  
 holder.bindGalleryItem(galleryItem)  
 }  
  
 override fun getItemCount(): Int = galleryItems.size  
 }  
  
 companion object {  
 fun newInstance() = PhotoGalleryFragment()  
 }  
}

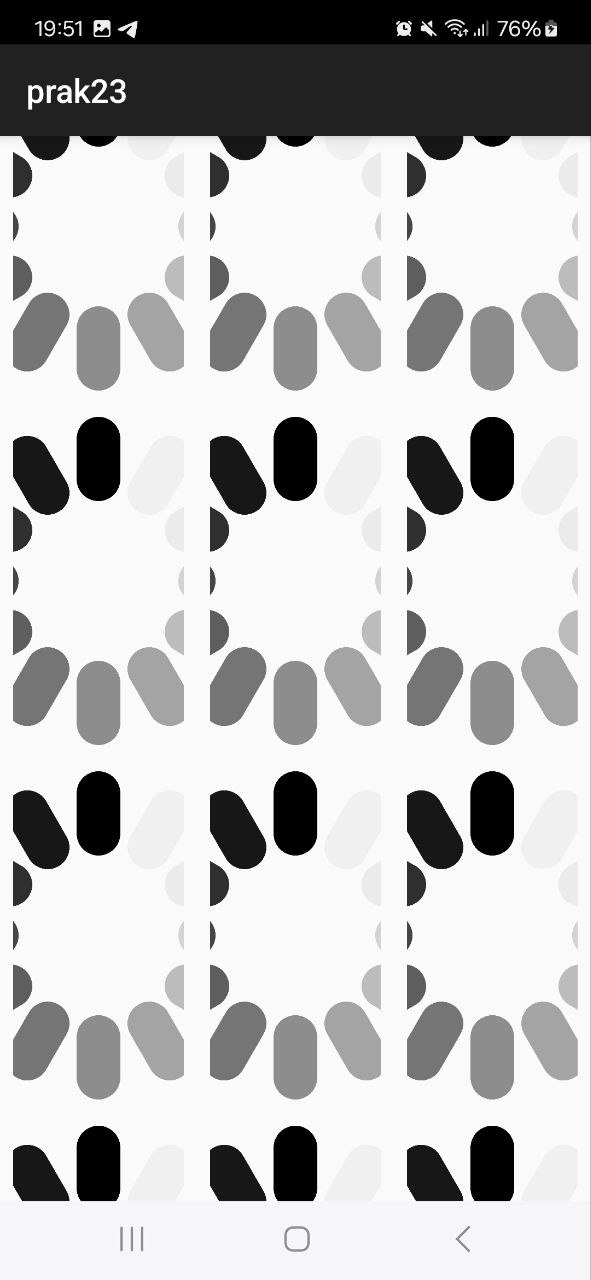
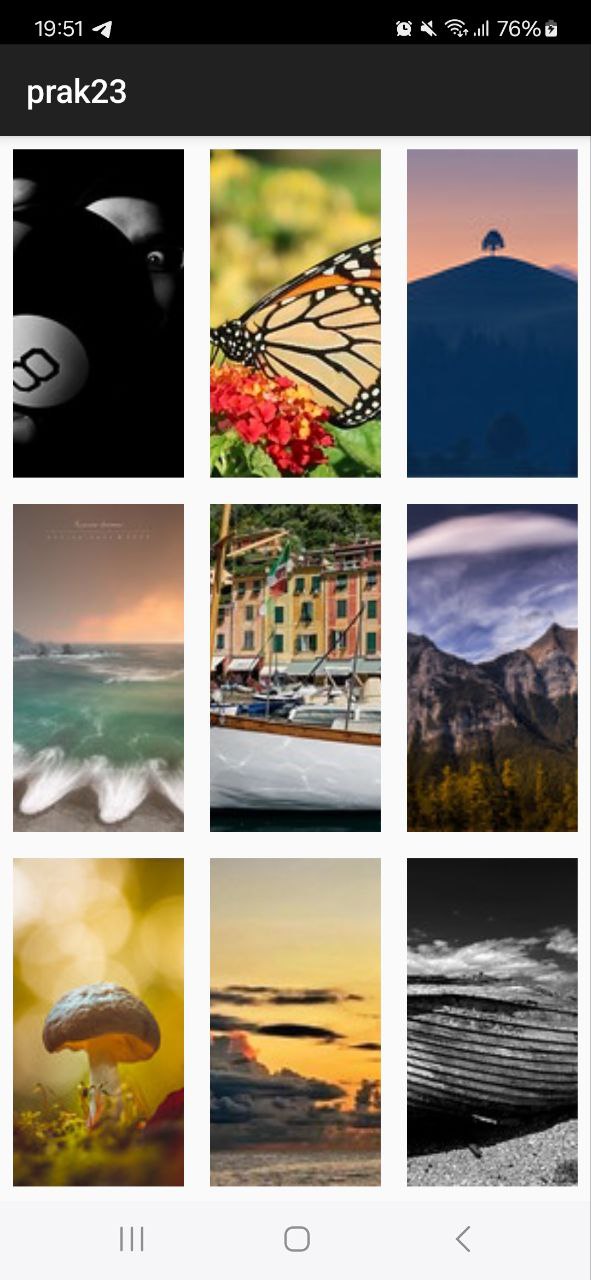
package com.example.prak23  
  
import androidx.lifecycle.LiveData  
import androidx.lifecycle.MutableLiveData  
import androidx.lifecycle.ViewModel  
  
class PhotoGalleryViewModel : ViewModel() {  
 private val \_galleryItemLiveData: MutableLiveData<List<GalleryItem>> = MutableLiveData()  
 val galleryItemLiveData: LiveData<List<GalleryItem>>  
 get() = \_galleryItemLiveData  
  
 init {  
 fetchGalleryItems()  
 }  
 private fun fetchGalleryItems() {  
 val flickrFetchr = FlickrFetchr()  
 flickrFetchr.fetchPhotos().observeForever **{** galleryItems **->** \_galleryItemLiveData.*value* = galleryItems  
 **}** }  
}

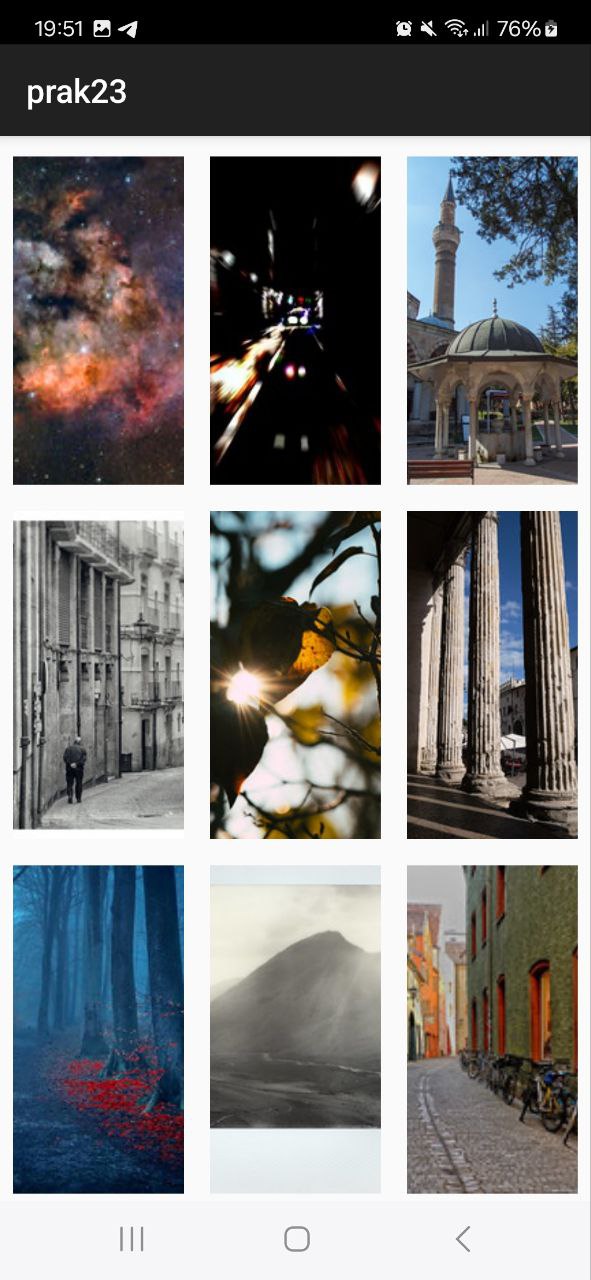
package com.example.prak23  
  
import com.google.gson.annotations.SerializedName  
  
class PhotoResponse {  
 @SerializedName("photo")  
 var galleryItems: List<GalleryItem> = *emptyList*()  
}

Interface:

package com.example.prak23.api  
  
import com.example.prak23.FlickrResponse  
import retrofit2.Call  
import retrofit2.http.GET  
  
interface FlickrApi {  
  
 @GET(  
 "services/rest/?method=flickr.interestingness.getList" +  
 "&api\_key=7d0fca767060c3f44bea0c36309e4cd3" +  
 "&format=json" +  
 "&nojsoncallback=1" +  
 "&extras=url\_s"  
 )  
 fun fetchPhotos(): Call<FlickrResponse>  
}

**Тестовые ситуации**

****

****