Министерство науки и высшего образования Российской Федерации

Федерального государственного бюджетное образовательное учреждение высшего образования

«Российской экономический университет имени Г. В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ.**

ПРАКТИЧЕСКИЕ РАБОТЫ № 5

ДИСЦИПЛИНА: «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Тема: Работа с библиотекой Retrofit.

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

Листов:

|  |  |  |
| --- | --- | --- |
| Выполнил студент |  | Проверил преподаватель |
| Группа П50-3-18 |  | \_\_\_\_\_\_\_\_\_\_\_А.О.Лясников |
| Пахомов Даниил Александрович |  | «\_\_\_» \_\_\_\_\_\_\_\_2020 года |

Москва 2020

ПРАКТИЧЕСКАЯ РАБОТА №5

Работа с библиотекой Retrofit.

Цель работы: научится создавать приложения, которые могут взаимодействовать с библиотекой Retrofit.

1. MainActivity

|  |
| --- |
| **package** com.example.retrofit  **import** android.os.Bundle **import** android.util.Log **import** androidx.appcompat.app.AppCompatActivity **import** com.example.retrofit.Model.Example **import** com.example.retrofit.Model.ViewModel **import** kotlinx.android.synthetic.main.activity\_main.\* **import** retrofit2.Call **import** retrofit2.Callback **import** retrofit2.Response **import** retrofit2.Retrofit **import** retrofit2.converter.gson.GsonConverterFactory  **class** MainActivity : AppCompatActivity() {  **override fun** onCreate(savedInstanceState: Bundle?) {  **super**.onCreate(savedInstanceState)  setContentView(R.layout.*activity\_main*)  *sendNetworkRequest*()  post()  }    **fun** post() {  **val** buld = Retrofit.Builder().baseUrl(**"http://jsonplaceholder.typicode.com/"**)  .addConverterFactory(GsonConverterFactory.create())  **val** retrofit = buld.build()  **var** interf = retrofit.create<Interface>(Interface::**class**.*java*)  **val** call: retrofit2.Call<Example> = interf.getPost(**"test"**,**"test1"**,100)  call.enqueue(**object** : retrofit2.Callback<Example> {  **override fun** onFailure(call: Call<Example>, t: Throwable) {  *TODO*(**"Not yet implemented"**)  }   **override fun** onResponse(call: Call<Example>, response: Response<Example>) {  showResponse(response.body().*toString*());  Log.i(**"TAG"**, **"post submitted to API."** + response.body().*toString*());  }  })  }   **private fun** showResponse(toString: String) {  test.*text* = toString  } }   **fun** sendNetworkRequest() {  **val** builder = Retrofit.Builder()  .baseUrl(**"http://newsapi.org/v2/"**)  .addConverterFactory(GsonConverterFactory.create())  **var** retrofit = builder.build()  **var** interf = retrofit.create<Interface>(Interface::**class**.*java*)  **val** call: retrofit2.Call<ViewModel> = interf.getFile()  call.enqueue(**object** : retrofit2.Callback<ViewModel> {  **override fun** onFailure(call: retrofit2.Call<ViewModel>, t: Throwable) {  *TODO*(**"Not yet implemented"**)  }   **override fun** onResponse(  call: retrofit2.Call<ViewModel>,  response: Response<ViewModel>  ) {  Log.i(**"Test"**, response.body()!!.**file**!!.get(0).**file**!!)  }  })  } |

1. Build.gradle(app)

|  |
| --- |
| apply **plugin**: **'com.android.application'** apply **plugin**: **'kotlin-android'** apply **plugin**: **'kotlin-android-extensions'** android **{** compileSdkVersion 30  buildToolsVersion **"30.0.2"** defaultConfig **{** applicationId **"com.example.retrofit"** minSdkVersion 16  targetSdkVersion 30  versionCode 1  versionName **"1.0"** testInstrumentationRunner **"androidx.test.runner.AndroidJUnitRunner"  }** buildTypes **{** release **{** minifyEnabled **false** proguardFiles getDefaultProguardFile(**'proguard-android-optimize.txt'**), **'proguard-rules.pro'  }  } }** dependencies **{** implementation fileTree(**dir**: **"libs"**, **include**: [**"\*.jar"**])  implementation **"org.jetbrains.kotlin:kotlin-stdlib:**$kotlin\_version**"** implementation **'androidx.core:core-ktx:1.3.2'** implementation **'androidx.appcompat:appcompat:1.2.0'** implementation **'androidx.constraintlayout:constraintlayout:2.0.2'** testImplementation **'junit:junit:4.12'** androidTestImplementation **'androidx.test.ext:junit:1.1.2'** androidTestImplementation **'androidx.test.espresso:espresso-core:3.3.0'** compile **'com.squareup.retrofit2:retrofit:2.5.0'** compile **'com.squareup.retrofit2:converter-gson:2.3.0'** compile **'com.google.code.gson:gson:2.6.1' }** |

1. AndroidManifest

|  |
| --- |
| apply **plugin**: **'com.android.application'** apply **plugin**: **'kotlin-android'** apply **plugin**: **'kotlin-android-extensions'** android **{** compileSdkVersion 30  buildToolsVersion **"30.0.2"** defaultConfig **{** applicationId **"com.example.retrofit"** minSdkVersion 16  targetSdkVersion 30  versionCode 1  versionName **"1.0"** testInstrumentationRunner **"androidx.test.runner.AndroidJUnitRunner"  }** buildTypes **{** release **{** minifyEnabled **false** proguardFiles getDefaultProguardFile(**'proguard-android-optimize.txt'**), **'proguard-rules.pro'  }  } }** dependencies **{** implementation fileTree(**dir**: **"libs"**, **include**: [**"\*.jar"**])  implementation **"org.jetbrains.kotlin:kotlin-stdlib:**$kotlin\_version**"** implementation **'androidx.core:core-ktx:1.3.2'** implementation **'androidx.appcompat:appcompat:1.2.0'** implementation **'androidx.constraintlayout:constraintlayout:2.0.2'** testImplementation **'junit:junit:4.12'** androidTestImplementation **'androidx.test.ext:junit:1.1.2'** androidTestImplementation **'androidx.test.espresso:espresso-core:3.3.0'** compile **'com.squareup.retrofit2:retrofit:2.5.0'** compile **'com.squareup.retrofit2:converter-gson:2.3.0'** compile **'com.google.code.gson:gson:2.6.1' }** |

1. Interface

|  |
| --- |
| **package** com.example.retrofit  **import** com.example.retrofit.Model.Example **import** com.example.retrofit.Model.ViewModel **import** retrofit2.Call **import** retrofit2.http.\*  **interface** Interface {   @GET(**"top-headlines?country=us&category=business&apiKey=8568c9b49aea4bce9e74b754d4493957"**)  **fun** getFile(): Call<ViewModel>   @POST(**"/posts"**)  @FormUrlEncoded  **fun** getPost(  @Field(**"title"**) title: String,  @Field(**"body"**) body: String,  @Field(**"userId"**) userId: Int  ): Call<Example> } |

1. ViewModel

|  |
| --- |
| **package** com.example.retrofit.Model  **import** com.google.gson.annotations.Expose **import** com.google.gson.annotations.SerializedName  **class** ViewModel {  @SerializedName(**"articles"**)  @Expose  **var file**: Array<Articles>? = **null** } **class** Articles {  @SerializedName(**"title"**)  @Expose  **var file**:String? = **null** } |

1. Example

|  |
| --- |
| **package** com.example.retrofit.Model  **import** com.google.gson.annotations.Expose **import** com.google.gson.annotations.SerializedName  **class** Example {   @SerializedName(**"title"**)  @Expose  **private var title**: String = **""** @SerializedName(**"body"**)  @Expose  **private var body**: String? = **null** @SerializedName(**"userId"**)  @Expose  **private var userId**: Int? = **null** @SerializedName(**"id"**)  @Expose  **private var id**: Int? = **null    fun** getTitle(): String {  **return title** }   **fun** setTitle(title: String) {  **this**.**title** = title  }   **fun** getBody(): String? {  **return body** }   **fun** setBody(body: String?) {  **this**.**body** = body  }   **fun** getUserId(): Int? {  **return userId** }   **fun** setUserId(userId: Int?) {  **this**.**userId** = userId  }   **fun** getId(): Int? {  **return id** }   **fun** setId(id: Int?) {  **this**.**id** = id  }    **override fun** toString(): String {  **return "Post{"** +  **"title=’"** + **title** + **"/"** +  **", body=’"** + **body** + **"/"** +  **", userId="** + **userId** +  **", id=»"** + **id** +  **"}"** }  } |

1. Activity\_main

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout  xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  xmlns:app="http://schemas.android.com/apk/res-auto"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity"**>  <**TextView  android:id="@+id/test"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Hello World!"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   </**androidx.constraintlayout.widget.ConstraintLayout**> |

Вывод: научился работать с библиотекой Retrofit. Обрабатывать GET POST запросы.