

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <ImageView
        android:background="@drawable/i"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
    <TextView
        android:id="@+id/textF"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Текущая температура в городе"
        android:textColor="#595881"
        android:textSize="30sp"
        android:textStyle="bold"
        android:textAlignment="center"
        android:layout_marginTop="50dp"
        android:paddingHorizontal="20dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <Button
        android:id="@+id/btVar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:clickable="true"
        android:focusable="true"
        android:text="Взять погоду"
        android:textColor="@color/black"
        android:textSize="19sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
dependencies {
    implementation(libs.play.services.location)
    implementation(libs.volley)
    implementation(libs.androidx.core.ktx)
    implementation(libs.androidx.appcompat)
    implementation(libs.material)
    implementation(libs.androidx.activity)
    implementation(libs.androidx.constraintlayout)
    testImplementation(libs.junit)
    androidTestImplementation(libs.androidx.junit)
    androidTestImplementation(libs.androidx.espresso.core)
}
```

```
ity_main.xml  AndroidManifest.xml  MainActivity.kt  strings.xml  build.gradle.kts (:app)

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.INTERNET"/>
    <permission android:name="android.permission.ACCESS_COARSE_LOCATION"
        tools:ignore="ReservedSystemPermission" />
    <permission android:name="android.permission.ACCESS_FINE_LOCATION"
        tools:ignore="ReservedSystemPermission" />
    <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.Ex11"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

```
activity_main.xml  AndroidManifest.xml  MainActivity.kt  strings.xml  build.gradle.kts (:app)

1  package com.onlylose.ex11
2
3  > import ...
22
23  class MainActivity : AppCompatActivity() {
24      var api_key = "7d5e382de296925794fcc67fb7f86677"
25      private lateinit var btVar1: Button
26      private lateinit var textView: TextView
27      private lateinit var fusedLocationClient: FusedLocationProviderClient
28      private val LOCATION_PERMISSION_REQUEST_CODE = 1
29      @SuppressWarnings("MissingInflatedId")
30      override fun onCreate(savedInstanceState: Bundle?) {
31          super.onCreate(savedInstanceState)
32          enableEdgeToEdge()
33          setContentView(R.layout.activity_main)
34
35          textView = findViewById(R.id.textF)
36          btVar1 = findViewById(R.id.btVar1)
37          fusedLocationClient =
38              LocationServices.getFusedLocationProviderClient(this)
39          btVar1.setOnClickListener {
40              checkForPermission()
41          }
42      }
43
44      private fun checkForPermission() {
45          if (ActivityCompat.checkSelfPermission( context: this,
46              Manifest.permission.ACCESS_FINE_LOCATION) !=
47              PackageManager.PERMISSION_GRANTED &&
48              ActivityCompat.checkSelfPermission( context: this,
49              Manifest.permission.ACCESS_COARSE_LOCATION) !=
```

```
activity_main.xml  AndroidManifest.xml  MainActivity.kt  strings.xml  build.gradle.kts (app)

23  class MainActivity : AppCompatActivity() {
44      private fun checkForPermission() {
47          PackageManager.PERMISSION_GRANTED &&
48          ActivityCompat.checkSelfPermission(context, this,
49              Manifest.permission.ACCESS_COARSE_LOCATION) !=
50              PackageManager.PERMISSION_GRANTED) {
51              ActivityCompat.requestPermissions(activity, this,
52                  arrayOf(Manifest.permission.ACCESS_FINE_LOCATION,
53                      Manifest.permission.ACCESS_COARSE_LOCATION),
54                      LOCATION_PERMISSION_REQUEST_CODE)
55          } else {
56              obtainLocation()
57          }
58      }
59
60      override fun onRequestPermissionsResult(requestCode: Int, permissions:
61          Array<out String>, grantResults: IntArray) {
62          super.onRequestPermissionsResult(requestCode, permissions,
63              grantResults)
64          if (requestCode == LOCATION_PERMISSION_REQUEST_CODE) {
65              if ((grantResults.isNotEmpty() && grantResults[0] ==
66                  PackageManager.PERMISSION_GRANTED)) {
67                  obtainLocation()
68              } else {
69                  Toast.makeText(context, this, text "Разрешения отклонено",
70                      Toast.LENGTH_SHORT).show()
71              }
72          }
73      }
74      @SuppressWarnings("MissingPermission")
75      private fun obtainLocation() {
76          fusedLocationClient.lastLocation
77          .addOnSuccessListener { location: Location? ->
78              if (location != null) {
79                  val weatherUrl =
80                      "https://api.openweathermap.org/data/2.5/weather?lat=${location.latitude}&
81                      getTemp(weatherUrl)
82              } else {
83                  Toast.makeText(context, this, text "Не удалось получить местоположение", Toast.LENGTH_SHORT).show()
84              }
85          }
86          .addOnFailureListener {
87              Toast.makeText(context, this, text "Location Permission not granted",
88                  Toast.LENGTH_SHORT).show()
89          }
90      }
91      private fun getTemp(url: String) {
92          val queue = Volley.newRequestQueue(context)
93          val stringReq = StringRequest(
94              Request.Method.GET, url, { response ->
95                  val obj = JSONObject(response)
96                  val main: JSONObject = obj.getJSONObject(name "main")
97                  val temperature = main.getString(name "temp")
98                  println(temperature)
99                  val city = obj.getString(name "name")
100                  println(city)
101                  textView.text = "${temperature} градусов по цельсиям в ${city}"
102                  System.out.println(obj.toString())
103              },
104              { textView.text = "Ошибка" })
105          queue.add(stringReq)
106      }
107  }
```



Результат с права на лево)3