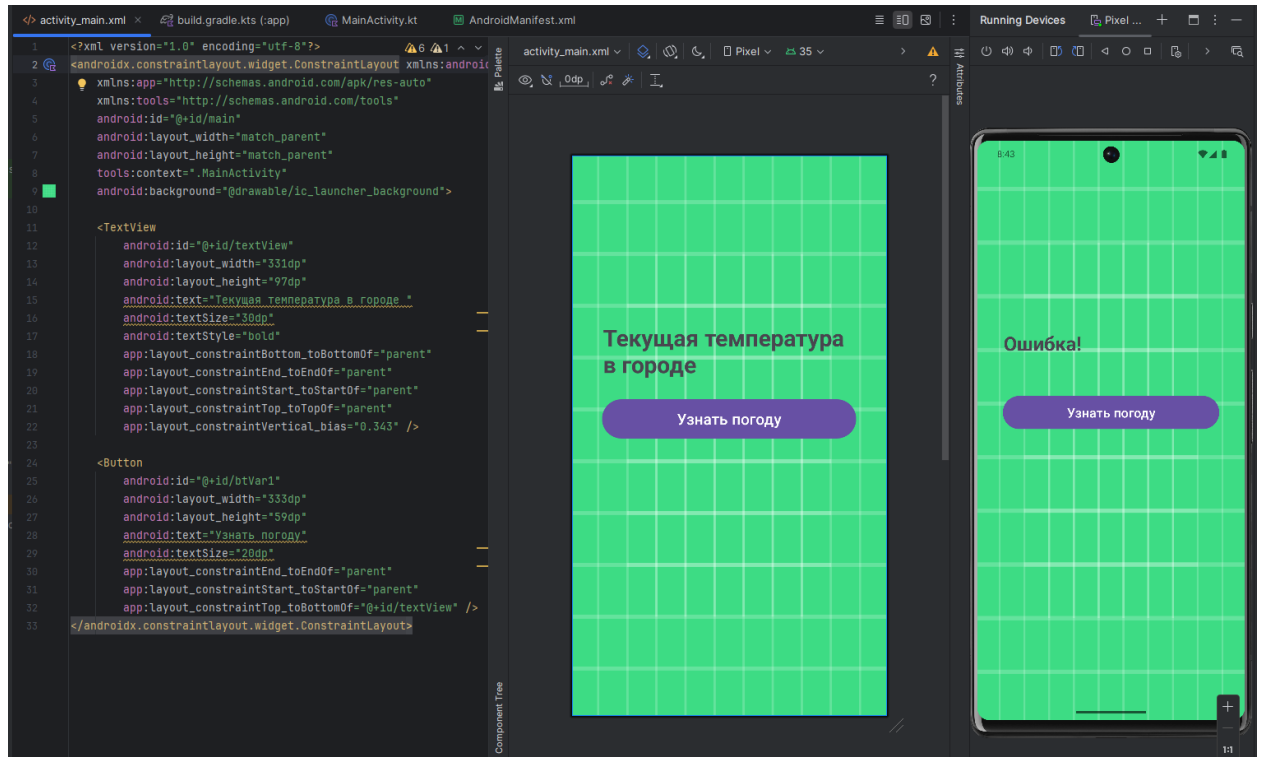


Практика 11. API



```
</> activity_main.xml  build.gradle.kts (:app)  MainActivity.kt  AndroidManifest.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools">
4
5      <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
6      <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
7      <uses-permission android:name="android.permission.INTERNET"/>
8
9      <application
10         android:allowBackup="true"
11         android:dataExtractionRules="@xml/data_extraction_rules"
12         android:fullBackupContent="@xml/backup_rules"
13         android:icon="@mipmap/ic_launcher"
14         android:label="@string/api_weather"
15         android:roundIcon="@mipmap/ic_launcher_round"
16         android:supportsRtl="true"
17         android:theme="@style/Theme.Api_weather"
18         tools:targetApi="31">
19         <activity
20             android:name=".MainActivity"
21             android:exported="true">
22             <intent-filter>
23                 <action android:name="android.intent.action.MAIN" />
24
25                 <category android:name="android.intent.category.LAUNCHER" />
26             </intent-filter>
27         </activity>
28     </application>
29
30 </manifest>
```

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

19

import com.google.android.gms.location.LocationServices

20

import org.json.JSONObject

21

import android.Manifest;

22

23

class MainActivity : AppCompatActivity() {

24

var api_key = "abd5136f877fca4162fd4da56b3b1c16"

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

30

@SuppressLint("MissingInflatedId")

31

override fun onCreate(savedInstanceState: Bundle?) {

32

super.onCreate(savedInstanceState)

33

enableEdgeToEdge()

34

setContentView(R.layout.activity_main)

35

ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) {

36

v, insets ->

37

val systemBars =

38

insets.getInsets(WindowInsetsCompat.Type.systemBars())

39

v.setPadding(systemBars.left, systemBars.top, systemBars.right,

40

systemBars.bottom)

41

insets

42

}

43

44

textView = findViewById(R.id.textView)

45

btVar1 = findViewById(R.id.btVar1)

46

47

48

fusedLocationClient = LocationServices.getFusedLocationProviderClient(activity: this)

49

50

51

// Проверяем разрешения на доступ к местоположению

52

btVar1.setOnClickListener {

53

checkForPermission()

54

}

55

56

}

57

private fun checkForPermission() {

58

if (ActivityCompat.checkSelfPermission(context: this,

59

Manifest.permission.ACCESS_FINE_LOCATION) !=

60

PackageManager.PERMISSION_GRANTED &&