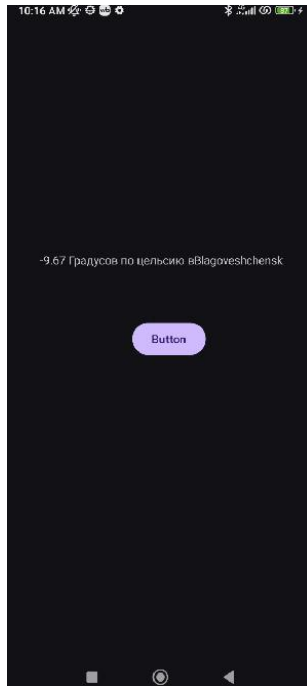


Работа приложения



Main класс

```
MainActivity.kt x  AndroidManifest.xml  libs.versions.toml  gradle-wrapper.properties  local.properties  v  ⋮
22  import org.json.JSONObject
23
24  <> class MainActivity : AppCompatActivity() {
25      // Ваш ключ API для доступа к OpenWeatherMap
26      var api_key = "db435b3925bddf0584929aa6c7c666a3"
27
28      // Определяем переменные для интерфейса пользователя и клиента местоположения
29      private lateinit var btVar1: Button
30      private lateinit var textView: TextView
31      //FusedLocationProviderClient – это высокоуровневый API для отслеживания местоположения, предоставляемый
32      private lateinit var fusedLocationClient: FusedLocationProviderClient
33
34      // Код запроса разрешения на доступ к местоположению
35      private val LOCATION_PERMISSION_REQUEST_CODE = 1
36
37      @ override fun onCreate(savedInstanceState: Bundle?) {
38          super.onCreate(savedInstanceState)
39          enableEdgeToEdge()
40          setContentView(R.layout.activity_main)
41          ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) {
42              v, insets ->
43              val systemBars =
44              insets.getInsets(WindowInsetsCompat.Type.systemBars())
45              v.setPadding(systemBars.left, systemBars.top, systemBars.right,
46              systemBars.bottom)
47              insets
48          }
49
50          textView = findViewById(R.id.textView)
51          btVar1 = findViewById(R.id.btVar1)
52
53
54          // Создаем экземпляр клиента для получения местоположения
55          fusedLocationClient =
56              LocationServices.getFusedLocationProviderClient(activity: this)
57
58  }
```

```

63     }
64
65     //Метод для проверки наличия разрешений
66     private fun checkForPermission() {
67         if (ActivityCompat.checkSelfPermission( context: this,
68             Manifest.permission.ACCESS_FINE_LOCATION) !=
69             PackageManager.PERMISSION_GRANTED &&
70             ActivityCompat.checkSelfPermission( context: this,
71                 Manifest.permission.ACCESS_COARSE_LOCATION) !=
72                 PackageManager.PERMISSION_GRANTED) {
73
74             // Запрашиваем разрешения, если они не предоставлены
75             ActivityCompat.requestPermissions( activity: this,
76                 arrayOf(Manifest.permission.ACCESS_FINE_LOCATION,
77                     Manifest.permission.ACCESS_COARSE_LOCATION),
78                     LOCATION_PERMISSION_REQUEST_CODE)
79         } else {
80             // Если разрешения уже предоставлены, получаем местоположение
81             obtainLocation()
82         }
83     }
84
85     // Метод, который обрабатывает результат запроса разрешений
86     @Override fun onRequestPermissionsResult(requestCode: Int, permissions:
87     Array<out String>, grantResults: IntArray) {
88         super.onRequestPermissionsResult(requestCode, permissions,
89             grantResults)
90         // Проверяем код запроса разрешений
91         if (requestCode == LOCATION_PERMISSION_REQUEST_CODE) {
92             // Если разрешение было предоставлено
93             if ((grantResults.isNotEmpty() && grantResults[0] ==
94                 PackageManager.PERMISSION_GRANTED)) {
95                 // Получаем местоположение
96                 obtainLocation()
97             } else {

```

```

24 class MainActivity : AppCompatActivity() {
128     }
129
130     private fun getTemp(url: String) {
131         // Создаем очередь для запросов
132         //Volley - это HTTP-библиотека, которая используется для
133         // кэширования и выполнения сетевого запроса в приложениях Android.
134         val queue = Volley.newRequestQueue( context: this)
135
136         // Выполняем запрос к API погоды
137         val stringReq = StringRequest(
138             Request.Method.GET, url, { response ->
139                 // Получаем JSON-объект из ответа
140                 val obj = JSONObject(response)
141
142                 // Получаем данные о температуре из объекта
143                 val main: JSONObject = obj.getJSONObject( name: "main")
144                 val temperature = main.getString( name: "temp")
145                 println(temperature)
146
147                 // Получаем название города из объекта
148                 val city = obj.getString( name: "name")
149                 println(city)
150
151                 // Устанавливаем текст в textView, показывая температуру и название города
152                 textView.text = "${temperature} градусов по цельсию в${city}"
153                 System.out.println(obj.toString())
154             },
155             // В случае ошибки показываем сообщение
156             { textView.text = "Ошибка!" })
157
158         // Добавляем запрос в очередь
159         queue.add(stringReq)
160     }
161 }

```

Main файл

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:id="@+id/main"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     tools:context=".MainActivity">
9
10    <TextView
11        android:id="@+id/textView"
12        android:layout_width="wrap_content"
13        android:layout_height="wrap_content"
14        android:text="Hello World!"
15        app:layout_constraintBottom_toBottomOf="parent"
16        app:layout_constraintEnd_toEndOf="parent"
17        app:layout_constraintStart_toStartOf="parent"
18        app:layout_constraintTop_toTopOf="parent"
19        app:layout_constraintVertical_bias="0.373" />
20
21    <Button
22        android:id="@+id/btVar1"
23        android:layout_width="wrap_content"
24        android:layout_height="wrap_content"
25        android:layout_marginStart="160dp"
26        android:layout_marginBottom="372dp"
27        android:text="Button"
28        app:layout_constraintBottom_toBottomOf="parent"
29        app:layout_constraintStart_toStartOf="parent" />
30
31</androidx.constraintlayout.widget.ConstraintLayout>
```