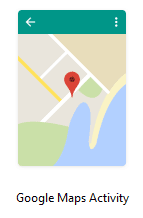
**Практическая работа**

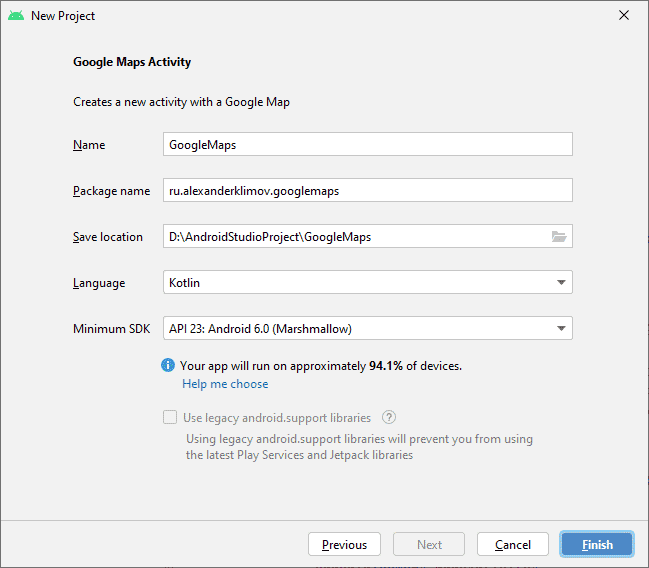
**Тема: Создание андроид-приложения с картами Google Maps с использованием Google Services и Android Studio**

Ход выполнения

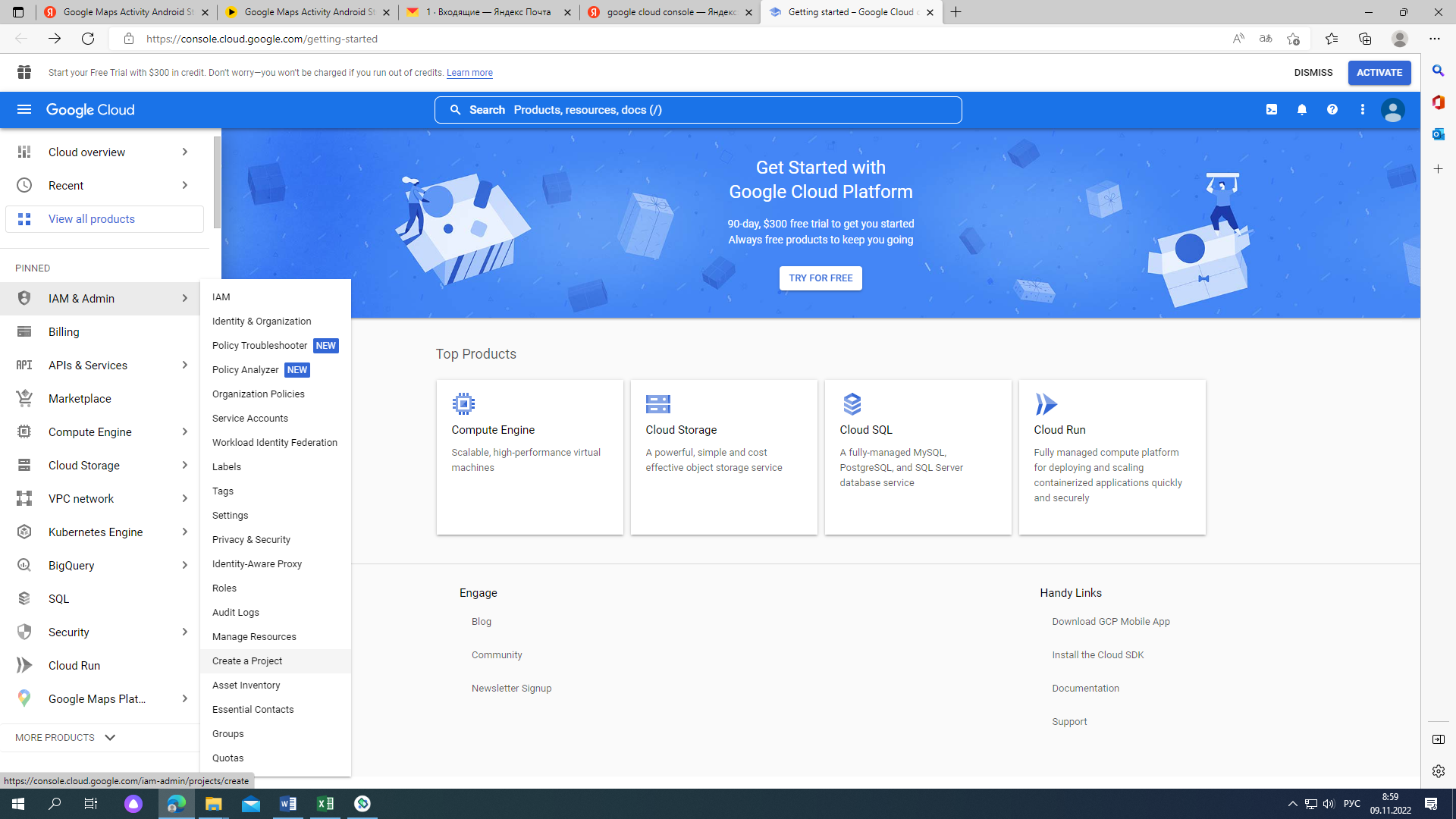
1. Запускаем новый проект и в шаблонах выбираем **Google Maps Activity**.

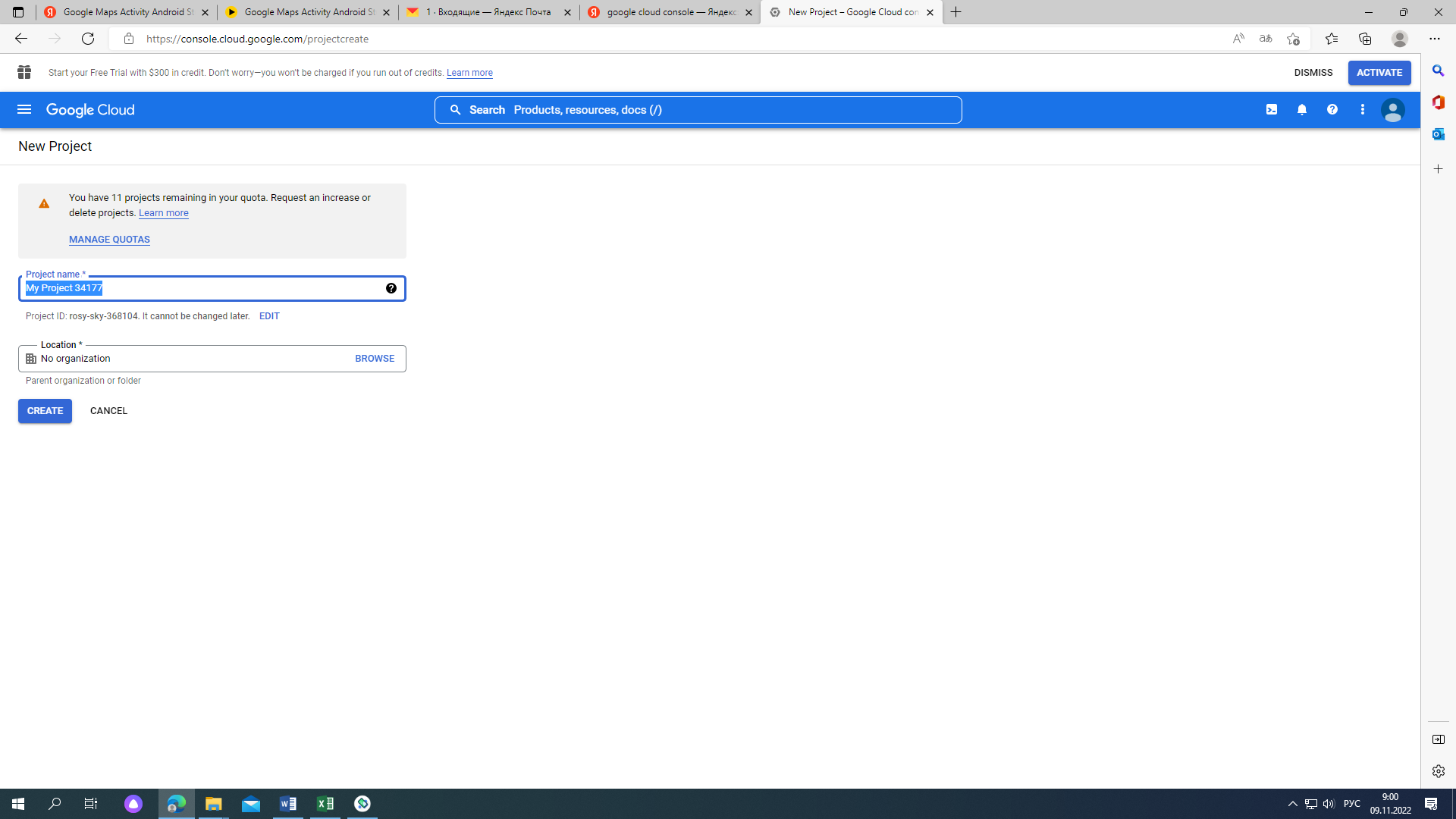


В следующем шаге делаем необходимые настройки.

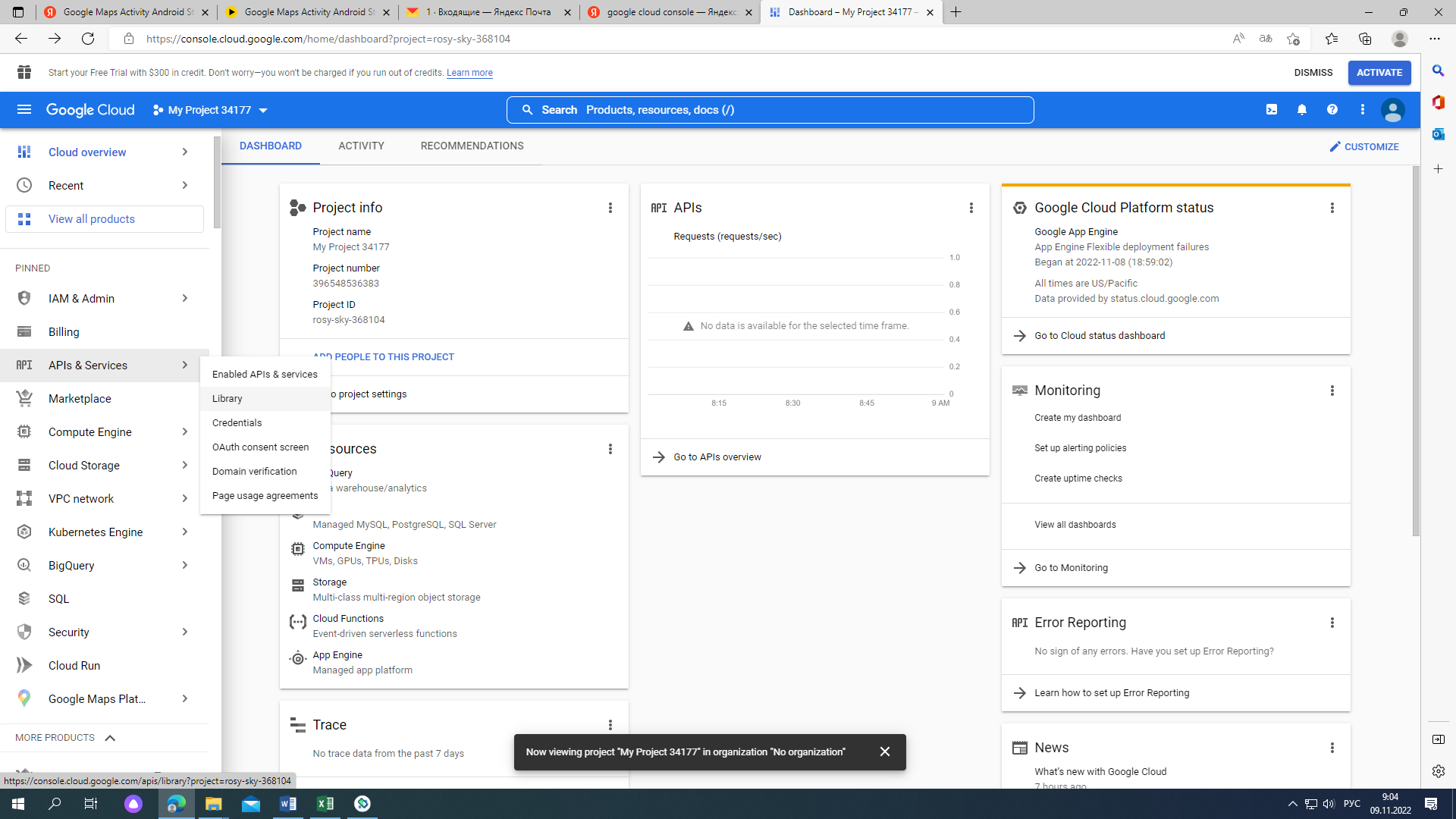


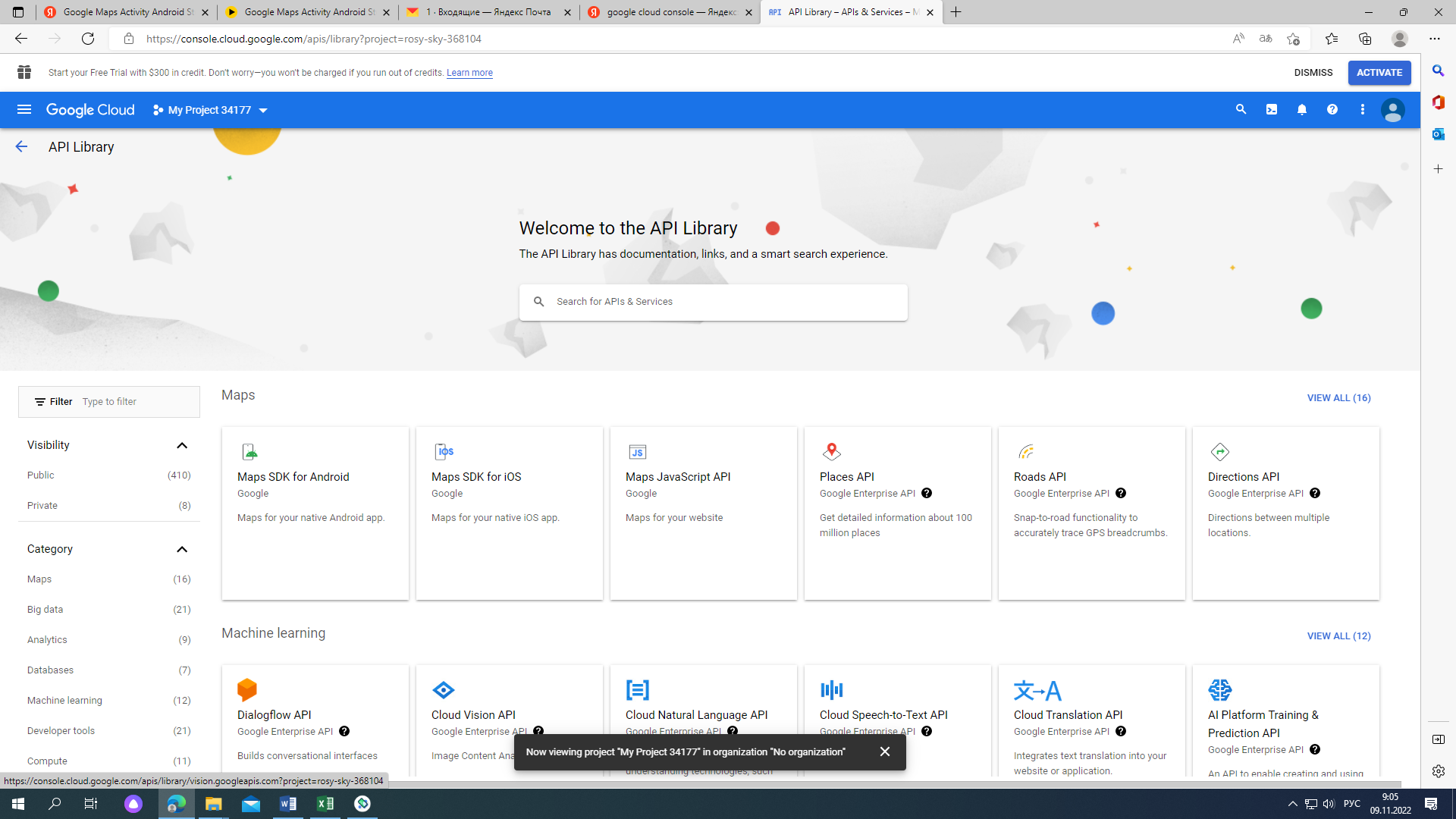
1. Заходим в браузере на [Консоль Google Cloud](https://console.cloud.google.com/getting-started)

и создаем проект 



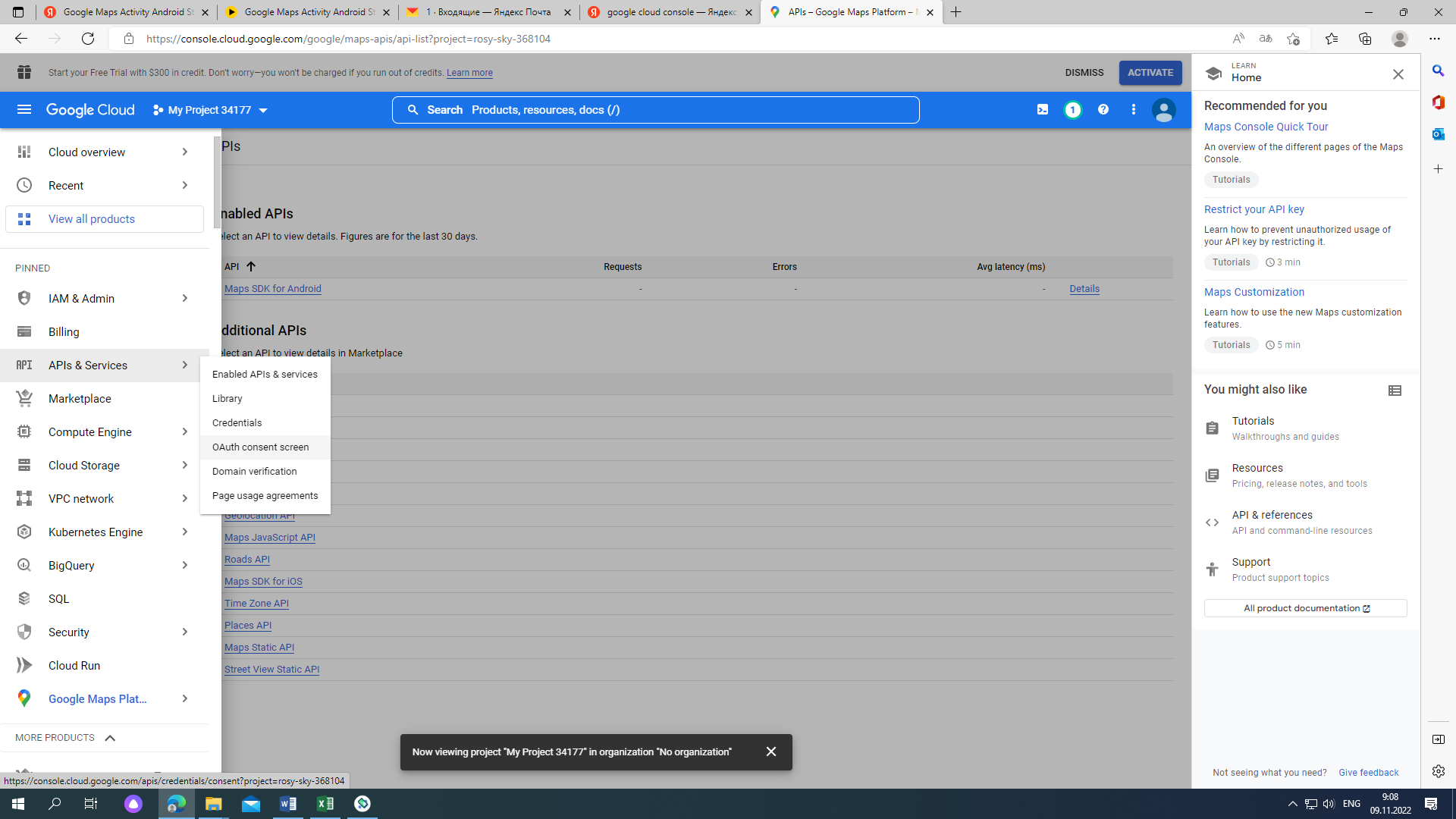
Переходим в библиотеки

Выбираем SDK под Android

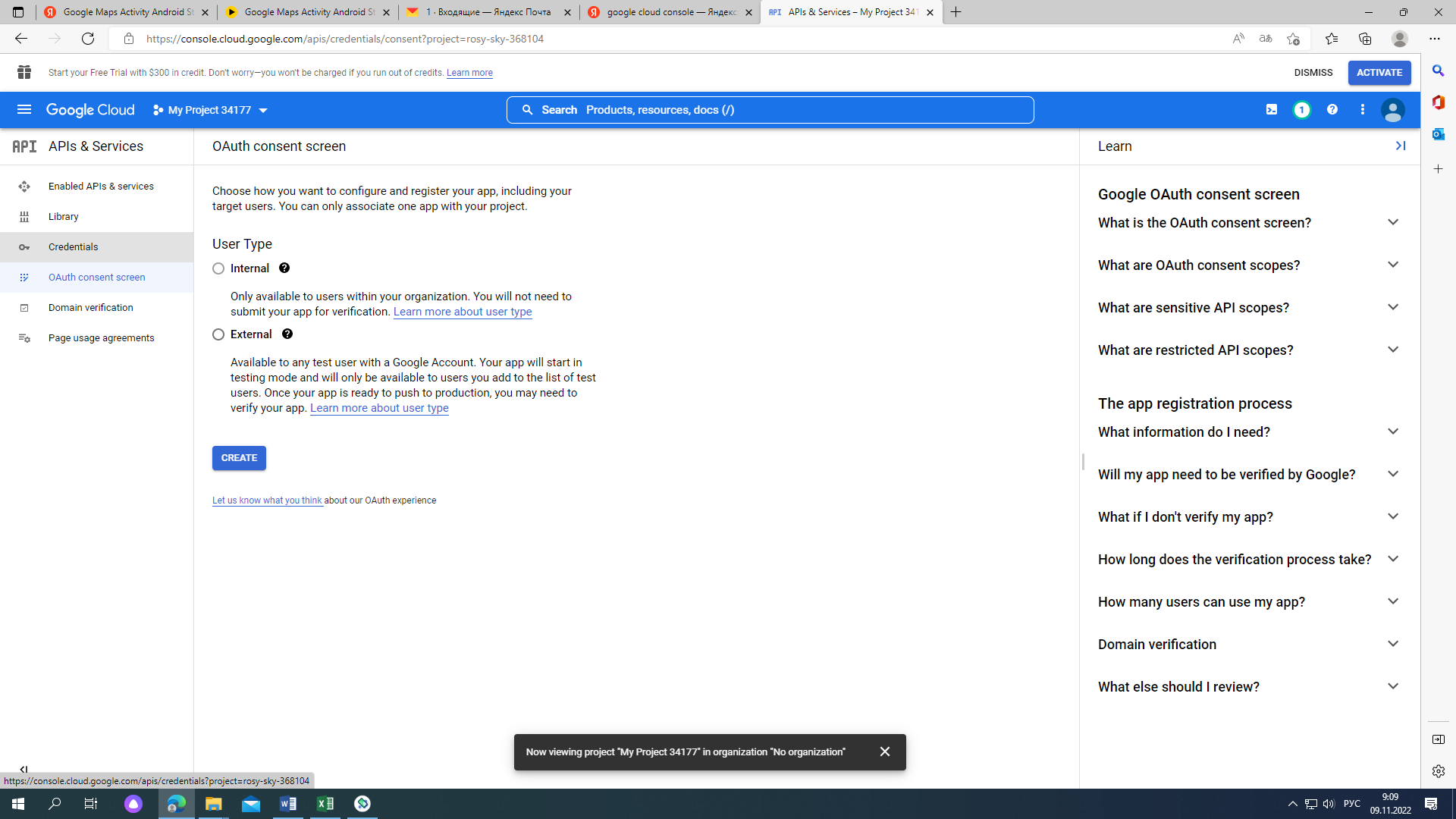


Включаем его при нажатии на кнопку Enable

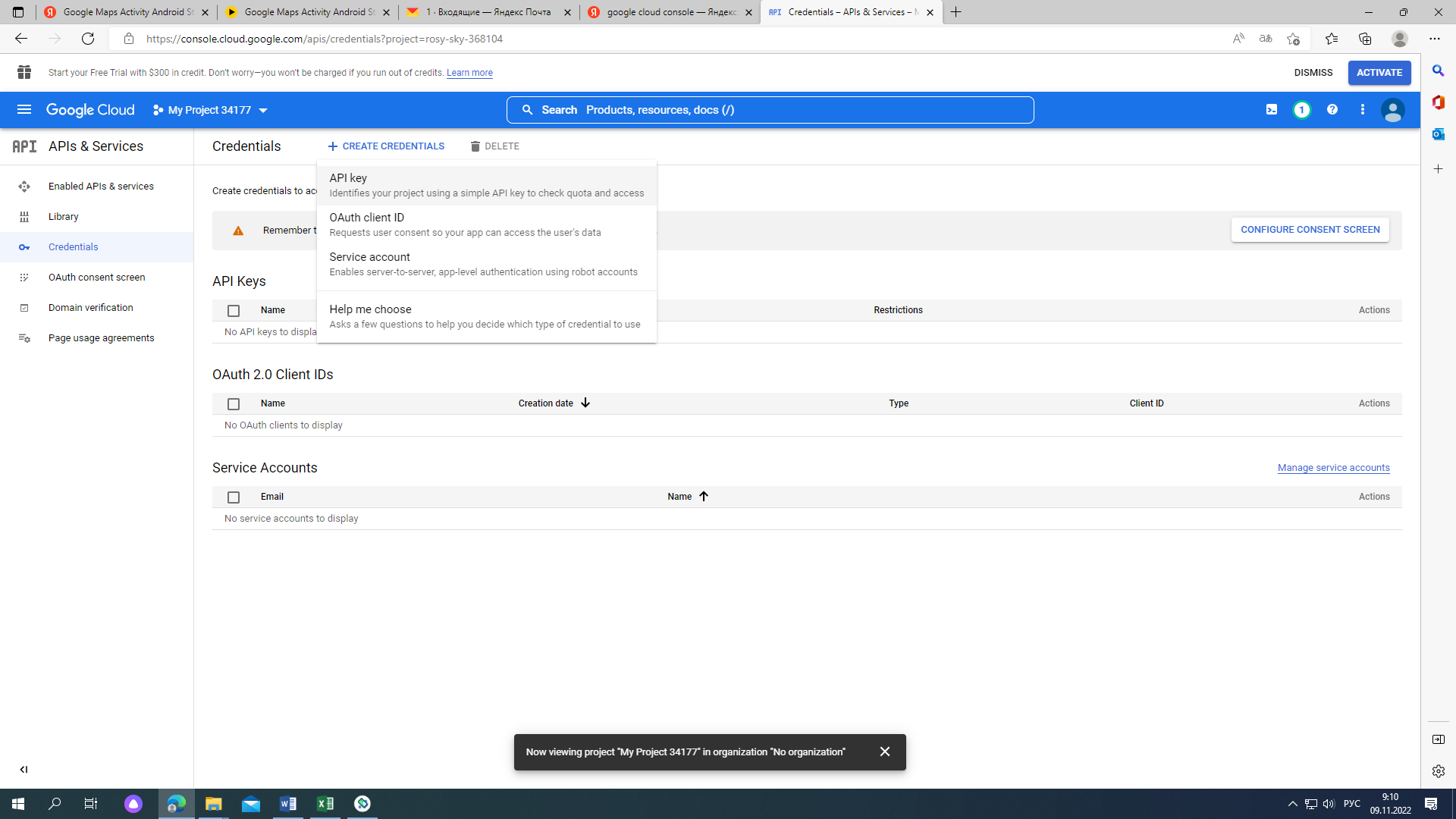
и в новом окне выбираем из меню



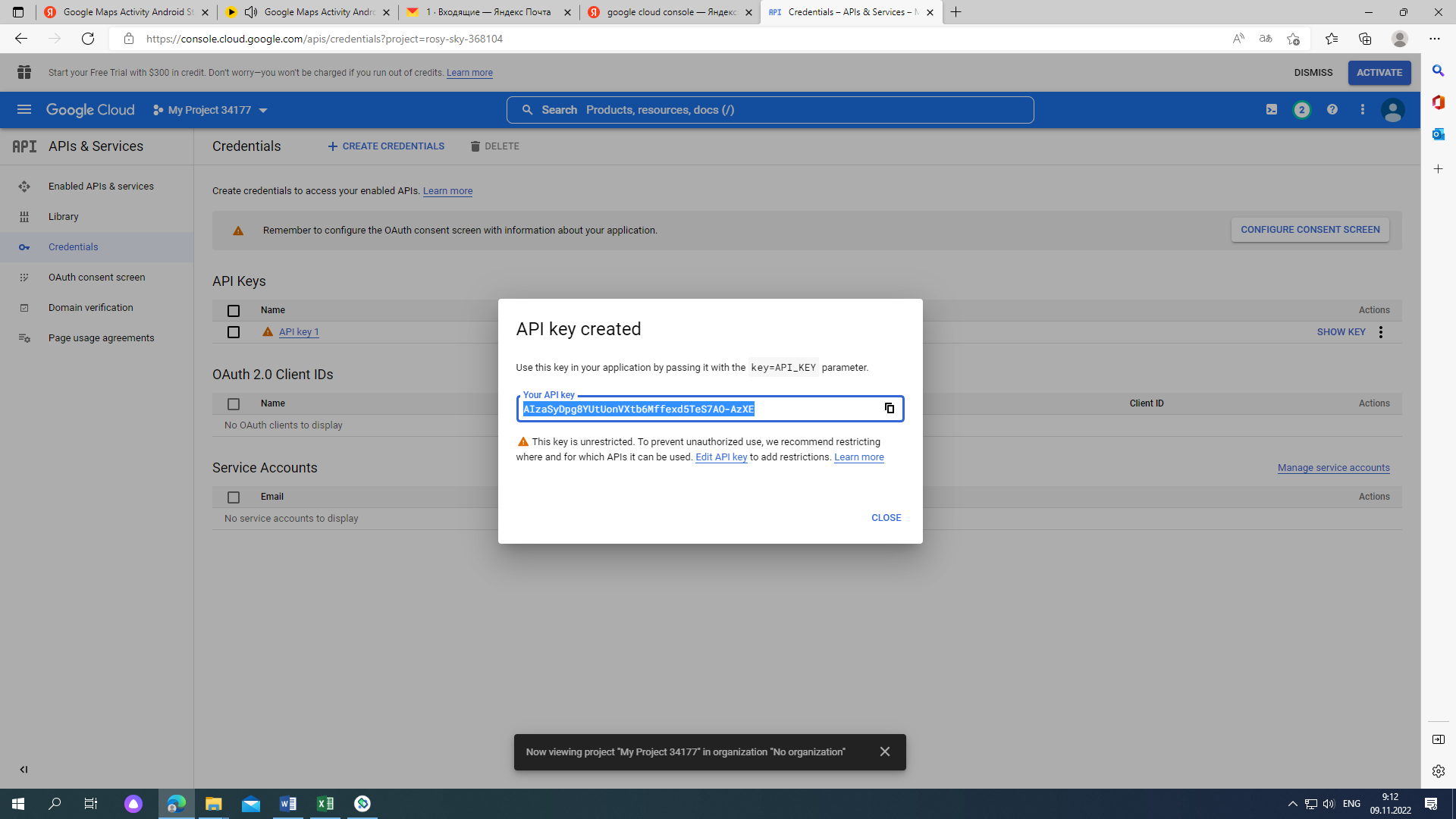
и затем в новом окне выбираем Crenditals



Генерируем ключ нажав на create Crenditels -> API key



Копируем ключ!!

Переходим в приложении Android

Откройте файл манифеста.

Возвращаемся в манифест и редактируем блок .

<meta-data

android:name="com.google.android.geo.API\_KEY"

android:value="ВСТАВЛЯЕМ СВОЙ СКОППИРОВАННЫЙ КЛЮЧ" />

Немного подождать на синхронизацию приложения

На этом предварительная подготовка завершена. Можно запустить проект. Если версия **Сервисы Google Play** устарела, то на экране появится кнопка **Обновить**. Если на устройстве свежая версия, то запустится пример с картой.

В текущей версии шаблона указаны координаты Сиднея, Австралия. Нажав на маркер, вы увидите всплывающий текст **Marker in Sydney**. Карту можно увеличивать двойным щелчком, а также сдвигать/раздвигать пальцами.

Познакомимся поближе с проектом. Для разметки используется фрагмент с классом Для разметки используется фрагмент с классом **com.google.android.gms.maps.SupportMapFragment**.

<?xml version="1.0" encoding="utf-8"?> <fragment

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/map"

android:name="com.google.android.gms.maps.SupportMapFragment"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MapsActivity" />

Код активности.

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import com.google.android.gms.maps.CameraUpdateFactory

import com.google.android.gms.maps.GoogleMap

import com.google.android.gms.maps.OnMapReadyCallback

import com.google.android.gms.maps.SupportMapFragment

import com.google.android.gms.maps.model.LatLng

import com.google.android.gms.maps.model.MarkerOptions

import ru.alexanderklimov.googlemapskot.databinding.ActivityMapsBinding

class MapsActivity : AppCompatActivity(), OnMapReadyCallback {

private lateinit var mMap: GoogleMap

private lateinit var binding: ActivityMapsBinding

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

binding = ActivityMapsBinding.inflate(layoutInflater)

setContentView(binding.root)

// Obtain the SupportMapFragment and get notified when the map is ready to be used.

val mapFragment = supportFragmentManager

.findFragmentById(R.id.map) as SupportMapFragment

mapFragment.getMapAsync(this)

}

/\*\*

\* Manipulates the map once available.

\* This callback is triggered when the map is ready to be used.

\* This is where we can add markers or lines, add listeners or move the camera. In this case,

\* we just add a marker near Sydney, Australia.

\* If Google Play services is not installed on the device, the user will be prompted to install

\* it inside the SupportMapFragment. This method will only be triggered once the user has

\* installed Google Play services and returned to the app.

\*/

override fun onMapReady(googleMap: GoogleMap) {

mMap = googleMap

// Add a marker in Sydney and move the camera

val sydney = LatLng(-34.0, 151.0)

mMap.addMarker(MarkerOptions().position(sydney).title("Marker in Sydney"))

mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney))

}

}

Класс **MapsActivity** наследуется от **AppCompatActivity** и использует интерфейс **OnMapReadyCallback**.

В методе **onCreate()** получаем доступ к фрагменту и загружаем карту в асинхронном режиме через метод **getMapAsync()**.

Метод **onMapReady()** является частью интерфейса **OnMapReadyCallback**. В этом методе можно писать код, когда карта готова к использованию. В примере добавляется маркер (метод **addMarker()**) в указанной точке (через объект **LatLng**) с указанием текста (метод **title()**). Метод **moveCamera()** перемещает карту в указанную позицию и мы можем видеть сразу нужное место.

Изменим координаты на Тюмень и подписи города

val tyumen = LatLng(57.09, 65.32)  
mMap.addMarker(MarkerOptions().position(tyumen).title("Tyumen"))

Сейчас там только пример добавления нового маркера в указанную позицию и заголовка, который появится при нажатии на маркер. Допишем код, добавив подсказку, используя точечный синтаксис.

mMap.addMarker(MarkerOptions().position(tyumen).title("Tyumen**").snippet("Тюмень-лучший город!!"))**

Подсказка появляется под описанием маркера с другим цветом. Если на карте должно быть несколько маркеров, то просто вызываете методы **addMarker()** несколько раз с разными параметрами.



**Часть 2.** Добавим в свои карты на базе карт Google сервис пробки, локация устройства и как достать фрагмент и поместить его в layout.

1. Теперь нам нужно указать разрешения для нашего приложения. Мы должны добавить следующие разрешения в файл манифеста:

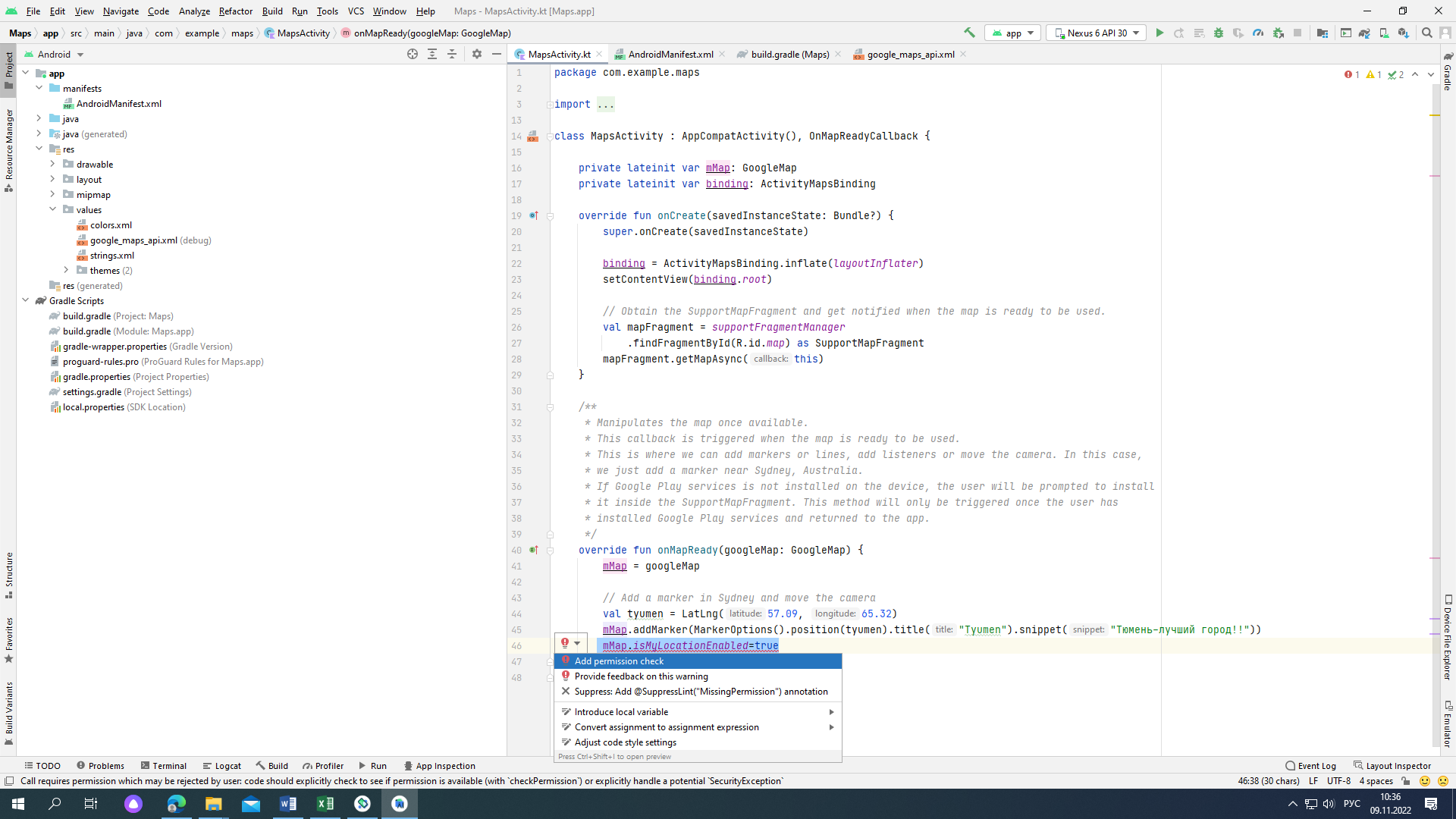


|  |  |
| --- | --- |
| 1  2  3  4  5  6 | <uses-permission android:name="android.permission.INTERNET" />  <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />  <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  C:\Users\t.kosygina\Downloads\IMG_20220324_130151.jpg |

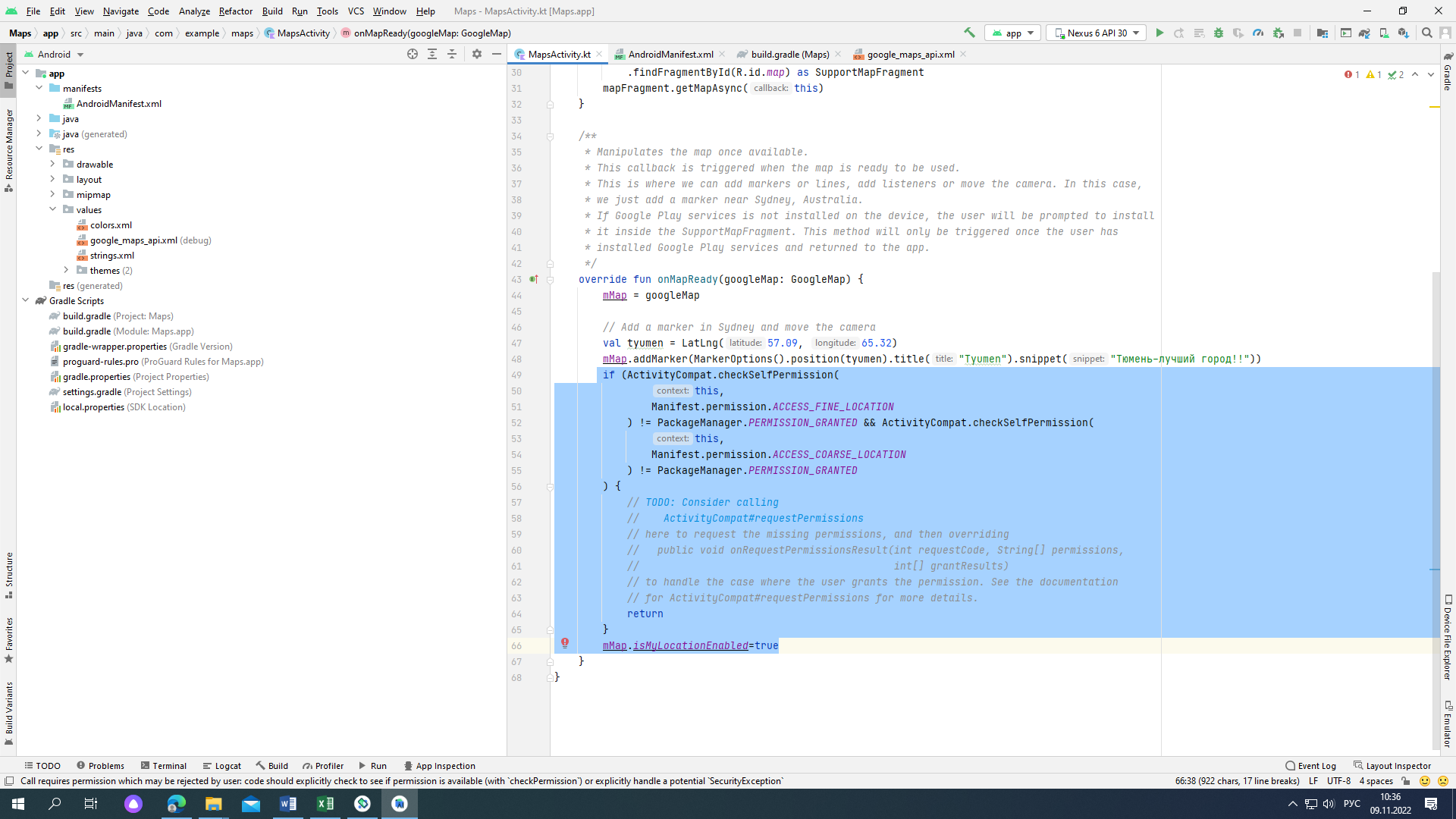
1. в функцию override fun onMapReady(googleMap: GoogleMap) длпишем показ загруженность догог

mMap.*isTrafficEnabled*=true

1. тут же добавляем проверку локации строкой mMap.isMyLocationEnabled=true и добавляем проверку разрешения использовать эту локацию методом

****

**получаем**

****

1. и тут прописываем разрешение проверки return перед

ActivityCompat.requestPermissions(this, *arrayOf*(Manifest.permission.*ACCESS\_COARSE\_LOCATION*), 0)

1. заходим в файл activity\_maps.xml и меняем на такой вид

*<?*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:map="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MapsActivity" >  
 <fragment  
 android:id="@+id/map"  
 android:name="com.google.android.gms.maps.SupportMapFragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
   
 </fragment>  
</androidx.constraintlayout.widget.ConstraintLayout >

1. не забудьте на странице дизайн прикрепить фрагмент к layout!!
2. добавим на фрагмент объект часы следующим кодом

<AnalogClock  
 android:layout\_width="80dp"  
 android:layout\_height="80dp">