



Lab #03 – Google Maps app

Submission Type: **INDIVIDUAL WORK**

Objectives:

The learning objective of this lab is for students to learn how to use the Google maps API and define a custom location.

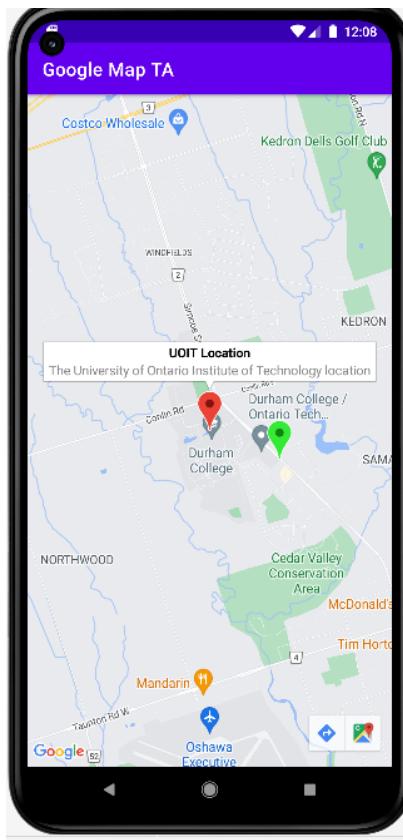
Important Notes:

1. Navigate to: C:\Users\your user\AndroidStudioProjects
2. You'll find a folder named after your project name (e.g.: "Lab03")
3. Create a text file containing your student's name and ID.
4. Create a new folder and name it your student ID
5. Place your project and the text in that folder, and zip them

Evaluation Scheme:

Component	Marks	Description
Interface design	10%	App design including, fonts, colors, sizes, etc.
Basics Functionally	55%	All the bare basic requirements of the simple layout must be satisfied and basic components are aligned perfectly.
Bug-free	35%	App doesn't crash when clicking on buttons
Total	100%	

Your Google Map app will look like this:



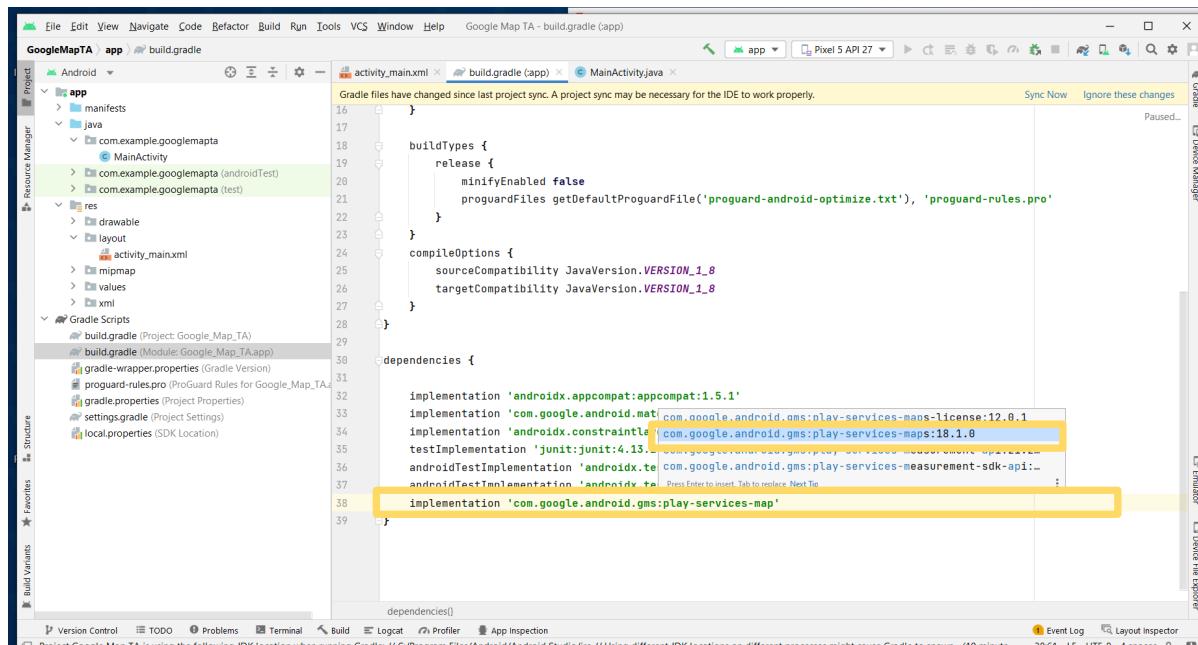
Tasks:

For this lab, you're required to:

1. Use the Relative Layout instead of the Constraint Layout.
2. Use Google Maps API (it's free, but requires setting up a billing account).
3. Define markers on a map that shows your home location, and a second marker that shows the UOIT location.
4. Set the color for your location to be green.

Steps:

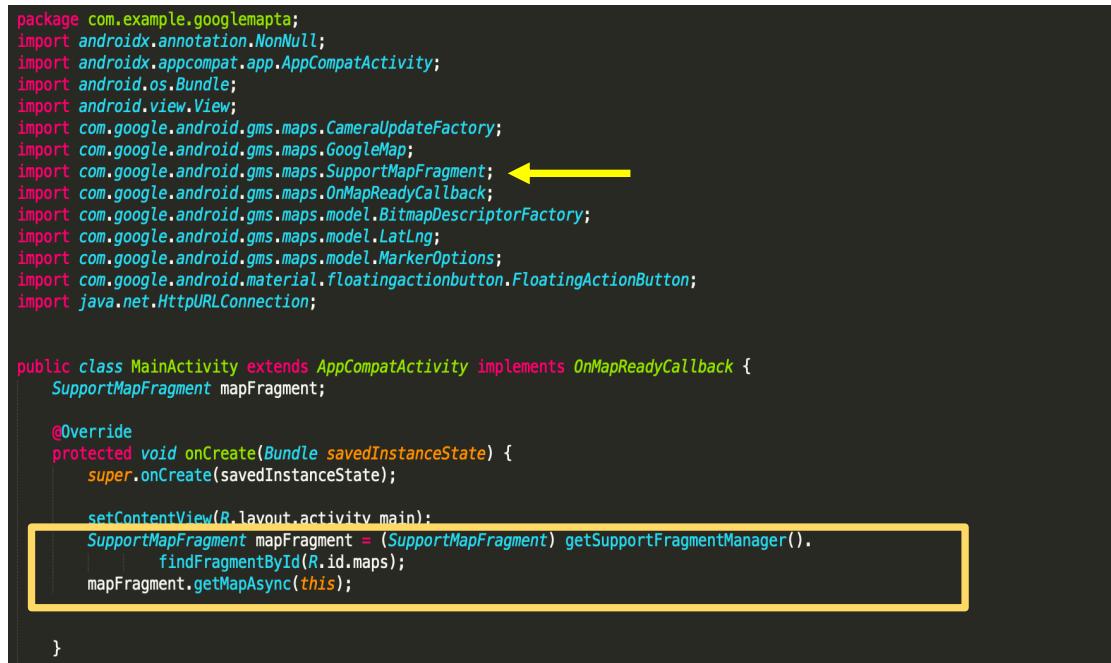
- 1- Create an empty activity Android application, and name your lab “ lab 3 Google Map – YOUR Student ID”
- 2- Implement the Google maps SDK API in the build.gradle file.



- 3- In your XML file, remove the existing View xml code, and add fragment element which used to place a map in an application.
- 4- Set the width, height to match parent, and set the id to “maps”.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <fragment
        android:id="@+id/maps"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center" />
</RelativeLayout>
```

- 5- In your MainActivity file, import the SupportMapFragment class.
- 6- Inside the OnCreate method, get the reference to the google maps Fragment in the layout using the Fragment manager.



```

package com.example.googlemaps;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.SupportMapFragment; ← Yellow arrow here
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import java.net.HttpURLConnection;

```

```

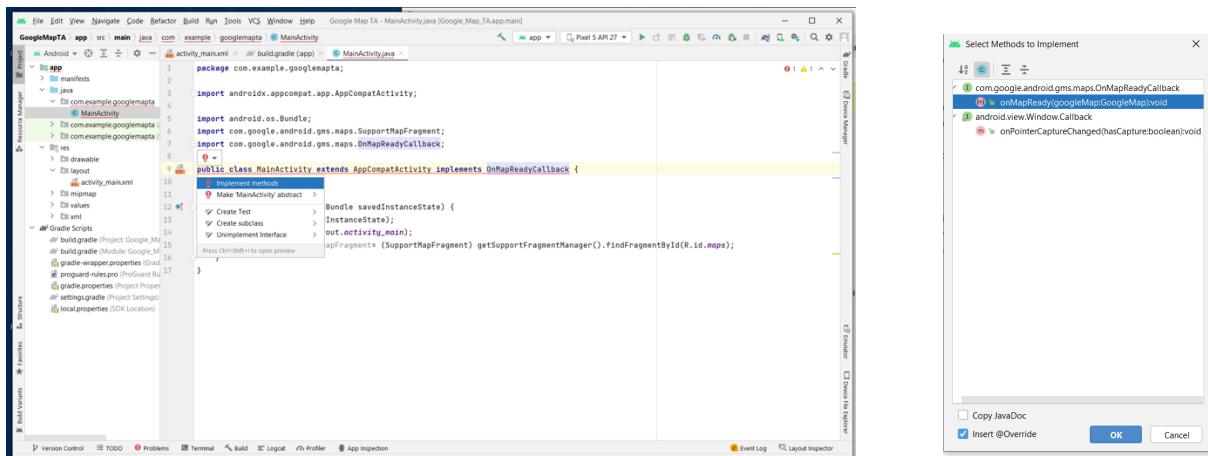
public class MainActivity extends AppCompatActivity implements OnMapReadyCallback {
    SupportMapFragment mapFragment;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager().
            findFragmentById(R.id.maps);
        mapFragment.getMapAsync(this);
    }
}

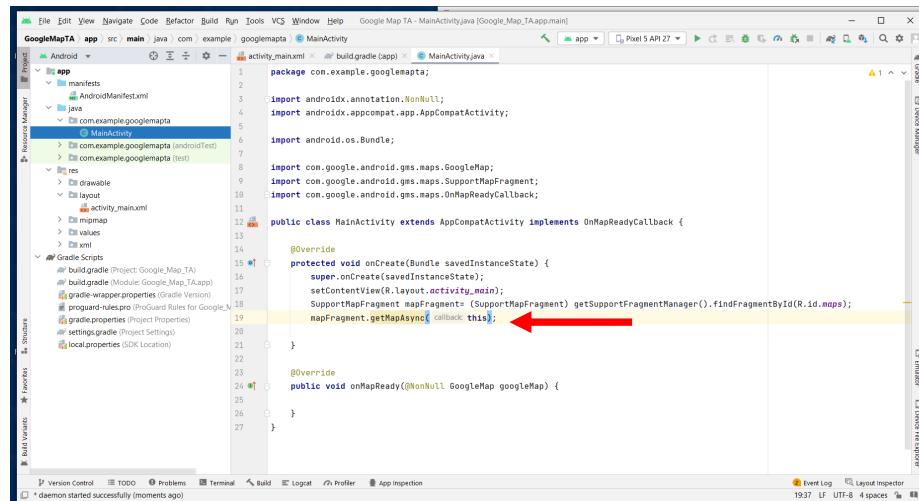
```

Once you have the reference for the map, you need to attach a callback to it so we know when the map is loaded.

- 7- Import the “OnMapReady” callback class.
- 8- In the public class header implement the callback and then press on the red mark and select the implement methods option, and implement the OnMapReady callback method.



- 9- In the OnCreate method, call the google maps fragment, and pass it in “this” for the callback. The map callback is triggered when the map is ready. This is the method to add the map to your app.



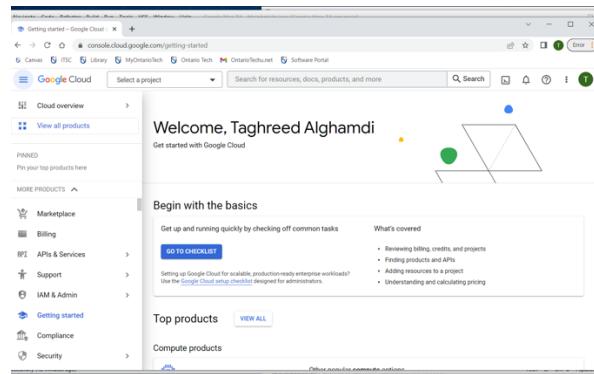
```

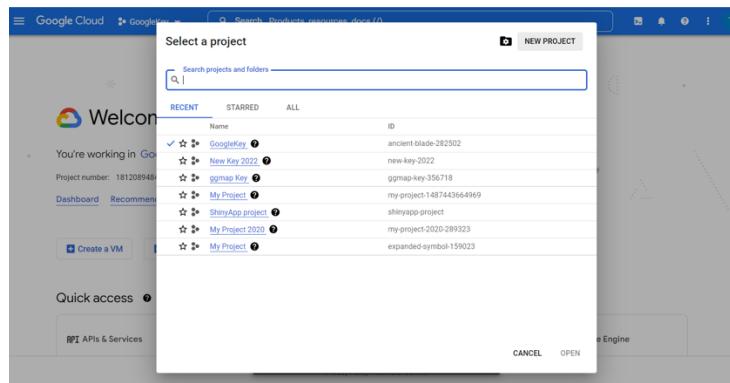
1 package com.example.googlemaps;
2
3 import android.annotation.NonNull;
4 import androidx.appcompat.AppCompatActivity;
5
6 import android.os.Bundle;
7
8 import com.google.android.gms.maps.GoogleMap;
9 import com.google.android.gms.maps.SupportMapFragment;
10 import com.google.android.gms.maps.OnMapReadyCallback;
11
12 public class MainActivity extends AppCompatActivity implements OnMapReadyCallback {
13
14     @Override
15     protected void onCreate(Bundle savedInstanceState) {
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_main);
18         SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.maps);
19         mapFragment.getMapAsync(callback);
20     }
21
22     @Override
23     public void onMapReady(@NonNull GoogleMap googleMap) {
24     }
25
26 }
27

```

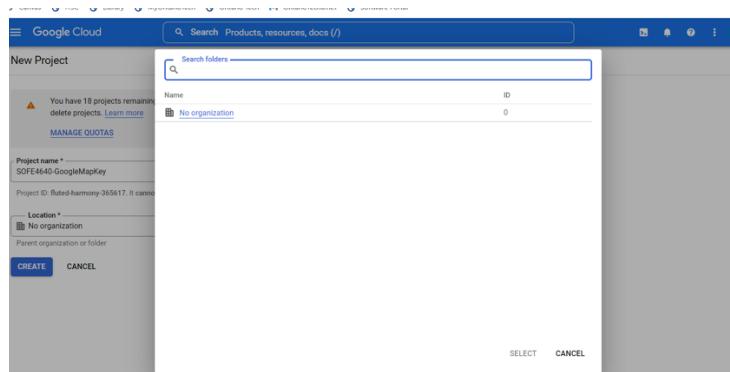
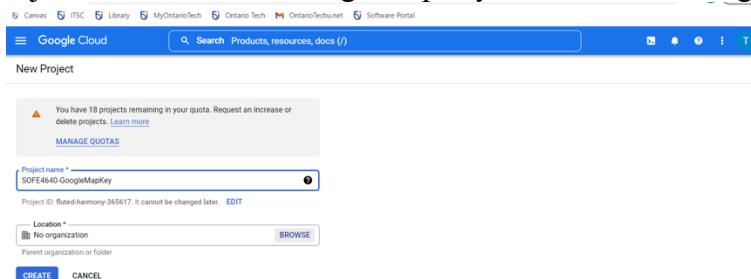
- 10- The next step we need to do is to generate Google Maps API key, so we can use the google maps services.

- Go to console.cloud.google.com.
- Sign in with your google account and create a new project.

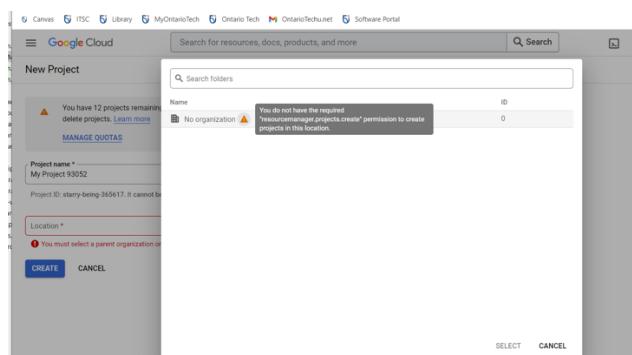




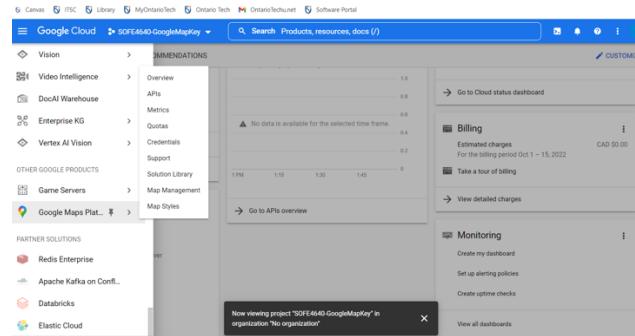
- Name your project as “SOFE4640 GoogleMapKey” and select “No organization”.



- Note that you need to set a billing account or you face an error when you create the project. It's recommended that you use your personal Gmail, not the university domain.



- After creating your project, go to the left menu, and scroll down until you see the google maps platform tab, then click ok.



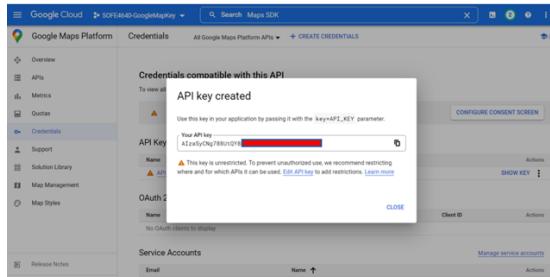
- From the APIs section, select the Maps SDK for Android option and enable the service.

The first screenshot shows the "Additional APIs" section of the Google Maps Platform API page. The "Maps SDK for Android" option is highlighted with a red arrow. The second screenshot shows the "Enabled APIs" section, where "Maps SDK for Android" is listed and also highlighted with a red arrow.

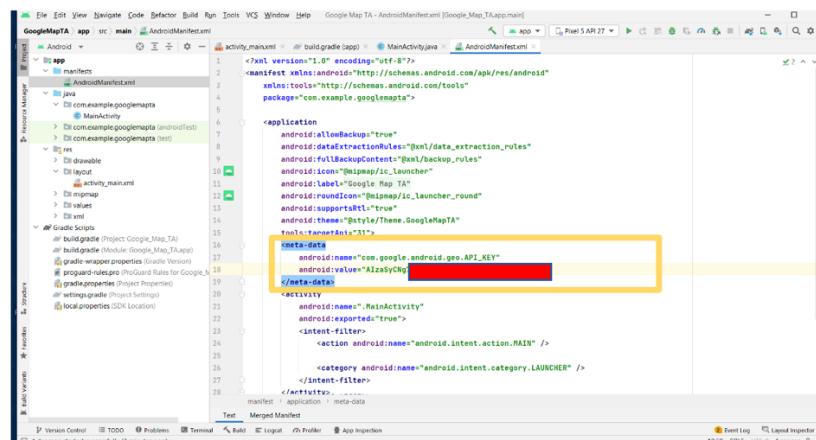
- Once the service is enabled, click on the credentials tab on the left, then click on create credentials and select API key.

The screenshot shows the "Credentials" page for the Google Maps Platform API. At the top right, there is a blue button labeled "+ CREATE CREDENTIALS" with a red arrow pointing to it. The page displays sections for "Credentials compatible with this API", "API Keys", "OAuth 2.0 Client IDs", and "Service Accounts".

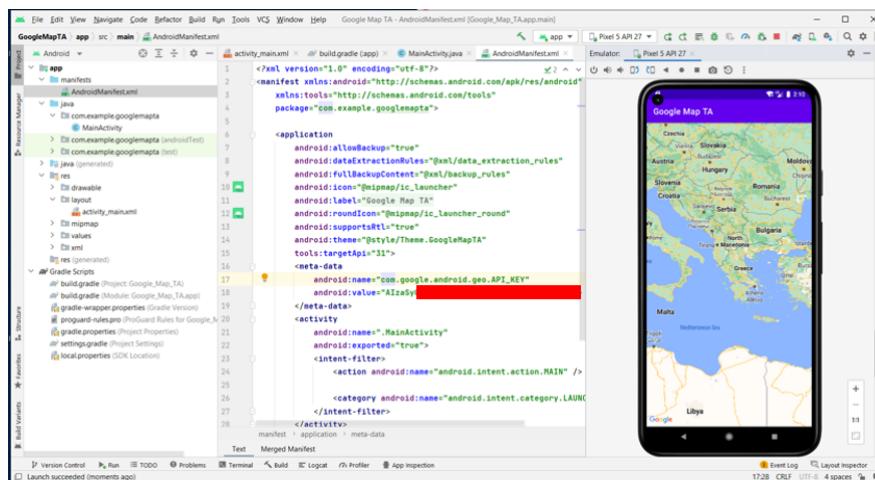
- An API key will be generated. Copy the key and go back to you Android studio to complete your Google map application.



11- Go to the manifest file, and add metadata element inside the application and set the google map key.



12- Run the app, and you should see a map as below:

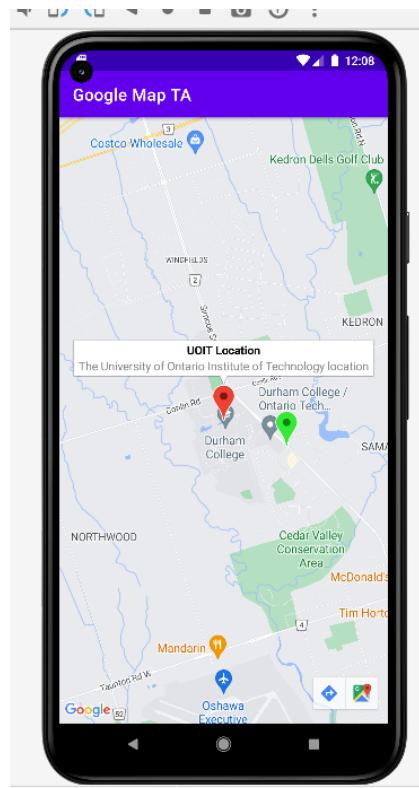


13- In the OnMapReady method, add 2 markers on your google map, the first one shows your home location, and the second one shows the university location.

14- Use “snippet()” to display more text over the UOIT marker when it’s tapped showing the university name.

15- Use the “animateCamera(CameraUpdateFactory)” method to move the map according to the locations with zoom degree 5.

```
    @Override
    public void onMapReady(NonNull GoogleMap googleMap) {
        //Show Marker on a Location
        googleMap.addMarker(new MarkerOptions()
            .position(new LatLng(43.94385315388242, -78.89035755987454))
            .title("My Location")
            .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN)));
        googleMap.addMarker(new MarkerOptions()
            .position(new LatLng(43.94578182437417, -78.89680805987453))
            .title("UOIT Location")
            .snippet("The University of Ontario Institute of Technology location"));
        googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(new LatLng(43.94385315388242, -78.89035755987454), 5));
    }
```



- Save your app and run it.
- When done, show your TA your work running before submitting for your grade.
- You're highly encouraged to use Google's Android Docs and other online resources.

MainActivity Code:

```
package com.example.googlemapta;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import java.net.HttpURLConnection;

public class MainActivity extends AppCompatActivity implements OnMapReadyCallback {
    SupportMapFragment mapFragment;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager().
            findFragmentById(R.id.maps);
        mapFragment.getMapAsync(this);

    }

    @Override
    public void onMapReady(@NonNull GoogleMap googleMap) {
        //Show Marker on a Location
        googleMap.addMarker(new MarkerOptions()
            // and move the map's camera to the same location.
            .position(new LatLng(43.94385315388242, -78.89035755987454))
            .title("My Location")
            .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN)));

        googleMap.addMarker(new MarkerOptions()
            .position(new LatLng(43.94578182437417, -78.89680805987453))
            .title("UOIT Location")
            .snippet("The University of Ontario Institute of Technology location"));

        googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(new LatLng(43.94385315388242, -78.89035755987454), 5));
    }
}
```