**ZOOM ROTATE CAR**

**Update activity\_main.xml**

Remove the buttons that are not related to zoom and rotate animations. Here's how your activity\_main.xml should look:

xml

Copy code

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

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<ImageView

android:id="@+id/imageview"

android:layout\_width="200dp"

android:layout\_height="200dp"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="40dp"

android:contentDescription="@string/lab6"

android:src="@drawable/car" />

<LinearLayout

android:id="@+id/linear1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/imageview"

android:layout\_marginTop="30dp"

android:orientation="horizontal">

<Button

android:id="@+id/BTNrotate"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:layout\_margin="10dp"

android:text="@string/clockwise" />

<Button

android:id="@+id/BTNzoom"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:layout\_margin="10dp"

android:text="@string/zoom" />

</LinearLayout>

<Button

android:id="@+id/BTNstop"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/linear1"

android:layout\_marginTop="30dp"

android:layout\_marginLeft="30dp"

android:layout\_marginRight="30dp"

android:text="@string/stop\_animation" />

</RelativeLayout>

**2. Update res/values/strings.xml**

Remove the strings related to the other buttons. Your strings.xml should look like this:

xml

Copy code

<resources>

<string name="lab6">YourAppName</string>

<string name="clockwise">Clockwise</string>

<string name="zoom">Zoom</string>

<string name="stop\_animation">Stop Animation</string>

</resources>

**3. Update MainActivity.java**

package com.example.lab6;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.view.animation.Animation;

import android.view.animation.AnimationUtils;

import android.widget.Button;

import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

ImageView imageView;

Button rotateBTN, zoomBTN, stopBTN;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

imageView = findViewById(R.id.imageview);

rotateBTN = findViewById(R.id.BTNrotate);

zoomBTN = findViewById(R.id.BTNzoom);

stopBTN = findViewById(R.id.BTNstop);

rotateBTN.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Animation animation = AnimationUtils.loadAnimation(getApplicationContext(), R.anim.rotateanimation);

imageView.startAnimation(animation);

}

});

zoomBTN.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Animation animation = AnimationUtils.loadAnimation(getApplicationContext(), R.anim.zoomanimation);

imageView.startAnimation(animation);

}

});

stopBTN.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

imageView.clearAnimation();

}

});

}

}

**1. Set up Google Maps API**

1. **Enable Google Maps API:**
   * Go to Google Cloud Console, create a new project, and enable **Google Maps SDK for Android**.
   * Generate an API key and restrict it to Android apps if needed.
2. **Add the API key to your project:**
   * In res/values/strings.xml, add the following string for your API key:

xml

Copy code

<resources>

<string name="google\_maps\_key" templateMergeStrategy="preserve" translatable="false">YOUR\_API\_KEY</string>

</resources>

**2. Add dependencies in build.gradle**

Add the following dependencies in your app/build.gradle file:

gradle

Copy code

dependencies {

implementation 'com.google.android.gms:play-services-maps:18.1.0'

}

Make sure to sync the project after adding dependencies.

**3. Modify AndroidManifest.xml**

Add the required permissions and the metadata for Google Maps API in the AndroidManifest.xml file:

xml

Copy code

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

package="com.example.mapdemo">

<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>

<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:supportsRtl="true"

android:theme="@style/Theme.MapDemo">

<!-- Add your Google Maps API key -->

<meta-data

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

android:value="@string/google\_maps\_key"/>

<activity android:name=".MapsActivity">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>

**4. Create a MapsActivity**

1. In Android Studio, create a new **Google Maps Activity**:
   * **File** > **New** > **Google** > **Google Maps Activity**.
2. This will automatically create a MapsActivity.java file and the activity\_maps.xml layout file.

**5. Customize MapsActivity.java**

The MapsActivity.java file will contain a GoogleMap object. You can customize it as needed:

java

Copy code

package com.example.mapdemo;

import androidx.fragment.app.FragmentActivity;

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 com.example.mapdemo.databinding.ActivityMapsBinding;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

private GoogleMap mMap;

private ActivityMapsBinding binding;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivityMapsBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

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

SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()

.findFragmentById(R.id.map);

mapFragment.getMapAsync(this);

}

@Override

public void onMapReady(GoogleMap googleMap) {

mMap = googleMap;

// Add a marker in a location (e.g., Sydney, Australia)

LatLng sydney = new LatLng(-34, 151);

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

mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));

// Optionally zoom in on the map

mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(sydney, 10));

}

}

**6. Modify activity\_maps.xml**

The layout file (activity\_maps.xml) already contains a SupportMapFragment by default:

xml

Copy code

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

android:id="@+id/map"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"/>