**IT5512- WEB TECHNOLOGY LAB-SESSION-6**

**DATE: 11/10/2021**

**NAME: A.S. PRUTHIEV**

**REG NO.2019506067**

**ANDROID LOCATION BASED APPLICATION USING JAVA ANDROID STUDIO**

**1)AIM:**

To write a Java Program to display the longitude and latitude of the selected place

**PROGRAM CODE:**

**MainAcitivity.java:**

package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.app.ActivityCompat;

import android.Manifest;

import android.content.pm.PackageManager;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

Button btnShowLocation;

private static final int REQUEST\_CODE\_PERMISSION = 2;

String mPermission = Manifest.permission.ACCESS\_FINE\_LOCATION;

// GPSTracker class

GPSTracker gps;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

try {

if (ActivityCompat.checkSelfPermission(this, mPermission)

!= PackageManager.PERMISSION\_GRANTED) {

ActivityCompat.requestPermissions(this, new String[]{mPermission},

REQUEST\_CODE\_PERMISSION);

// If any permission above not allowed by user, this condition will

//execute every time, else your else part will work

}

} catch (Exception e) {

e.printStackTrace();

}

btnShowLocation = (Button) findViewById(R.id.button);

// show location button click event

btnShowLocation.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View arg0) {

// create class object

gps = new GPSTracker(MainActivity.this);

// check if GPS enabled

if(gps.canGetLocation()){

double latitude = gps.getLatitude();

double longitude = gps.getLongitude();

// \n is for new line

Toast.makeText(getApplicationContext(), "Your Location is - \nLat: "

+ latitude + "\nLong: " + longitude, Toast.LENGTH\_LONG).show();

}else{

// can't get location

// GPS or Network is not enabled

// Ask user to enable GPS/network in settings

gps.showSettingsAlert();

}

}

});

}

}

**GPSTracker.java:**

package com.example.myapplication;

import android.annotation.SuppressLint;

import android.app.AlertDialog;

import android.app.Service;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.os.Bundle;

import android.os.IBinder;

import android.provider.Settings;

import android.util.Log;

public class GPSTracker extends Service implements LocationListener {

private final Context mContext;

// flag for GPS status

boolean isGPSEnabled = false;

// flag for network status

boolean isNetworkEnabled = false;

// flag for GPS status

boolean canGetLocation = false;

Location location; // location

double latitude; // latitude

double longitude; // longitude

// The minimum distance to change Updates in meters

private static final long MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES = 10; // 10 meters

// The minimum time between updates in milliseconds

private static final long MIN\_TIME\_BW\_UPDATES = 1000 \* 60 \* 1; // 1 minute

// Declaring a Location Manager

protected LocationManager locationManager;

public GPSTracker(Context context) {

this.mContext = context;

getLocation();

}

@SuppressLint("MissingPermission")

public Location getLocation() {

try {

locationManager = (LocationManager) mContext.getSystemService(LOCATION\_SERVICE);

// getting GPS status

isGPSEnabled = locationManager.isProviderEnabled(LocationManager.GPS\_PROVIDER);

// getting network status

isNetworkEnabled = locationManager

.isProviderEnabled(LocationManager.NETWORK\_PROVIDER);

if (!isGPSEnabled && !isNetworkEnabled) {

// no network provider is enabled

} else {

this.canGetLocation = true;

// First get location from Network Provider

if (isNetworkEnabled) {

locationManager.requestLocationUpdates(

LocationManager.NETWORK\_PROVIDER,

MIN\_TIME\_BW\_UPDATES,

MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES, this);

Log.d("Network", "Network");

if (locationManager != null) {

location = locationManager

.getLastKnownLocation(LocationManager.NETWORK\_PROVIDER);

if (location != null) {

latitude = location.getLatitude();

longitude = location.getLongitude();

}

}

}

// if GPS Enabled get lat/long using GPS Services

if (isGPSEnabled) {

if (location == null) {

locationManager.requestLocationUpdates(

LocationManager.GPS\_PROVIDER,

MIN\_TIME\_BW\_UPDATES,

MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES, this);

Log.d("GPS Enabled", "GPS Enabled");

if (locationManager != null) {

location = locationManager

.getLastKnownLocation(LocationManager.GPS\_PROVIDER);

if (location != null) {

latitude = location.getLatitude();

longitude = location.getLongitude();

}

}

}

}

}

} catch (Exception e) {

e.printStackTrace();

}

return location;

}

/\*\*

\* Stop using GPS listener

\* Calling this function will stop using GPS in your app

\* \*/

public void stopUsingGPS(){

if(locationManager != null){

locationManager.removeUpdates(GPSTracker.this);

}

}

/\*\*

\* Function to get latitude

\* \*/

public double getLatitude(){

if(location != null){

latitude = location.getLatitude();

}

// return latitude

return latitude;

}

/\*\*

\* Function to get longitude

\* \*/

public double getLongitude(){

if(location != null){

longitude = location.getLongitude();

}

// return longitude

return longitude;

}

/\*\*

\* Function to check GPS/wifi enabled

\* @return boolean

\* \*/

public boolean canGetLocation() {

return this.canGetLocation;

}

/\*\*

\* Function to show settings alert dialog

\* On pressing Settings button will lauch Settings Options

\* \*/

public void showSettingsAlert(){

AlertDialog.Builder alertDialog = new AlertDialog.Builder(mContext);

// Setting Dialog Title

alertDialog.setTitle("GPS is settings");

// Setting Dialog Message

alertDialog.setMessage("GPS is not enabled. Do you want to go to settings menu?");

// On pressing Settings button

alertDialog.setPositiveButton("Settings", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog,int which) {

Intent intent = new Intent(Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS);

mContext.startActivity(intent);

}

});

// on pressing cancel button

alertDialog.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

dialog.cancel();

}

});

// Showing Alert Message

alertDialog.show();

}

@Override

public void onLocationChanged(Location location) {

}

@Override

public void onProviderDisabled(String provider) {

}

@Override

public void onProviderEnabled(String provider) {

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

}

@Override

public IBinder onBind(Intent arg0) {

return null;

}

}

**Activity\_main.xml**

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

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

android:layout\_width = "fill\_parent"

android:layout\_height = "fill\_parent"

android:orientation = "vertical" >

<Button

android:id = "@+id/button"

android:layout\_width = "fill\_parent"

android:layout\_height = "wrap\_content"

android:text = "getlocation"/>

</LinearLayout>

**Strings.xml**

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

<resources>

<string name = "app\_name">Pruthiev Location App</string>

</resources>

**AndroidManifest.xml:**

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

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

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

package="com.example.myapplication">

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

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

<uses-permission android:name="android.permission.INTERNET" />

<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.MyApplication">

<activity

android:name=".MainActivity"

android:exported="true">

<intent-filter>

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

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

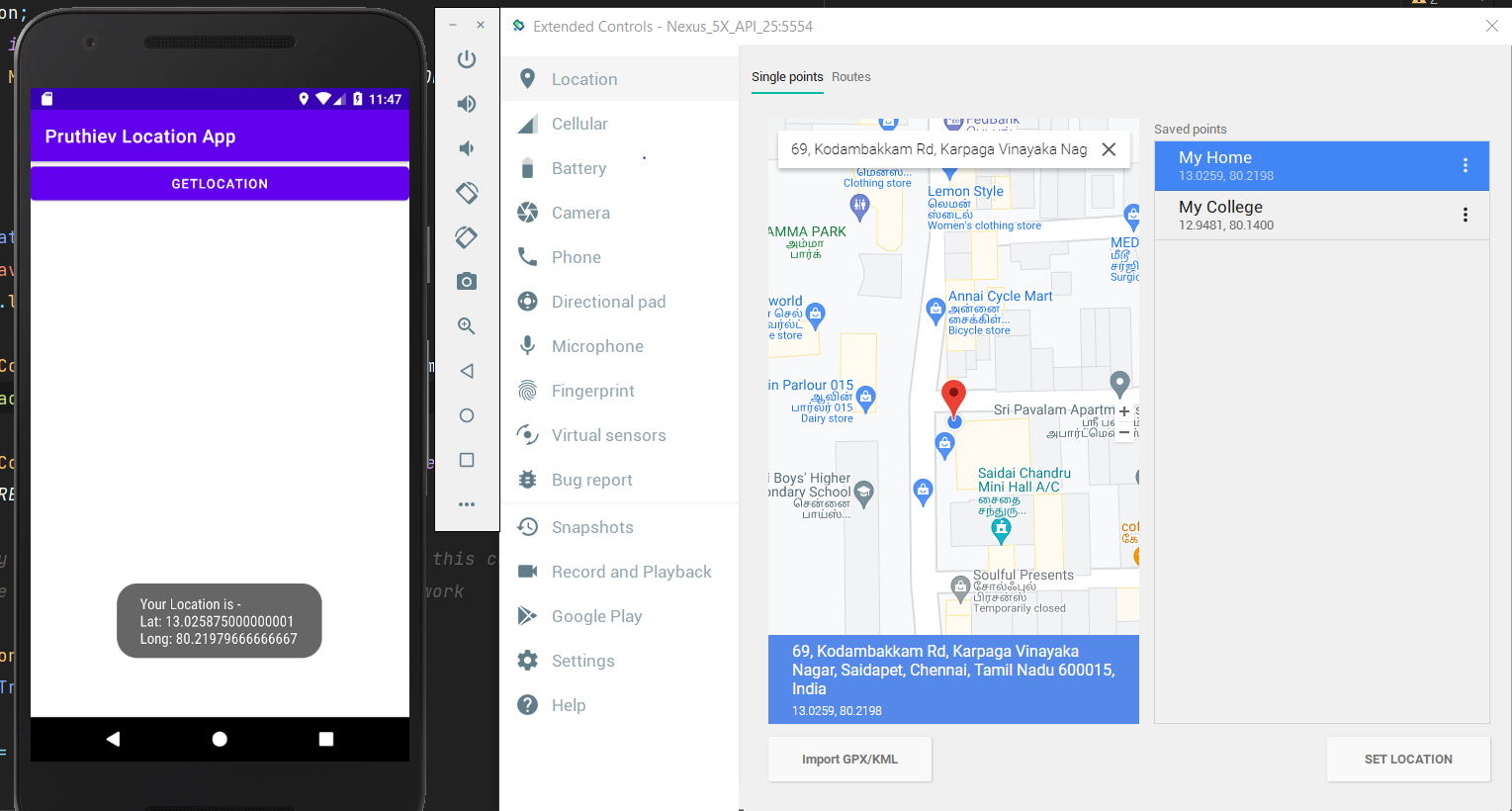
</intent-filter>

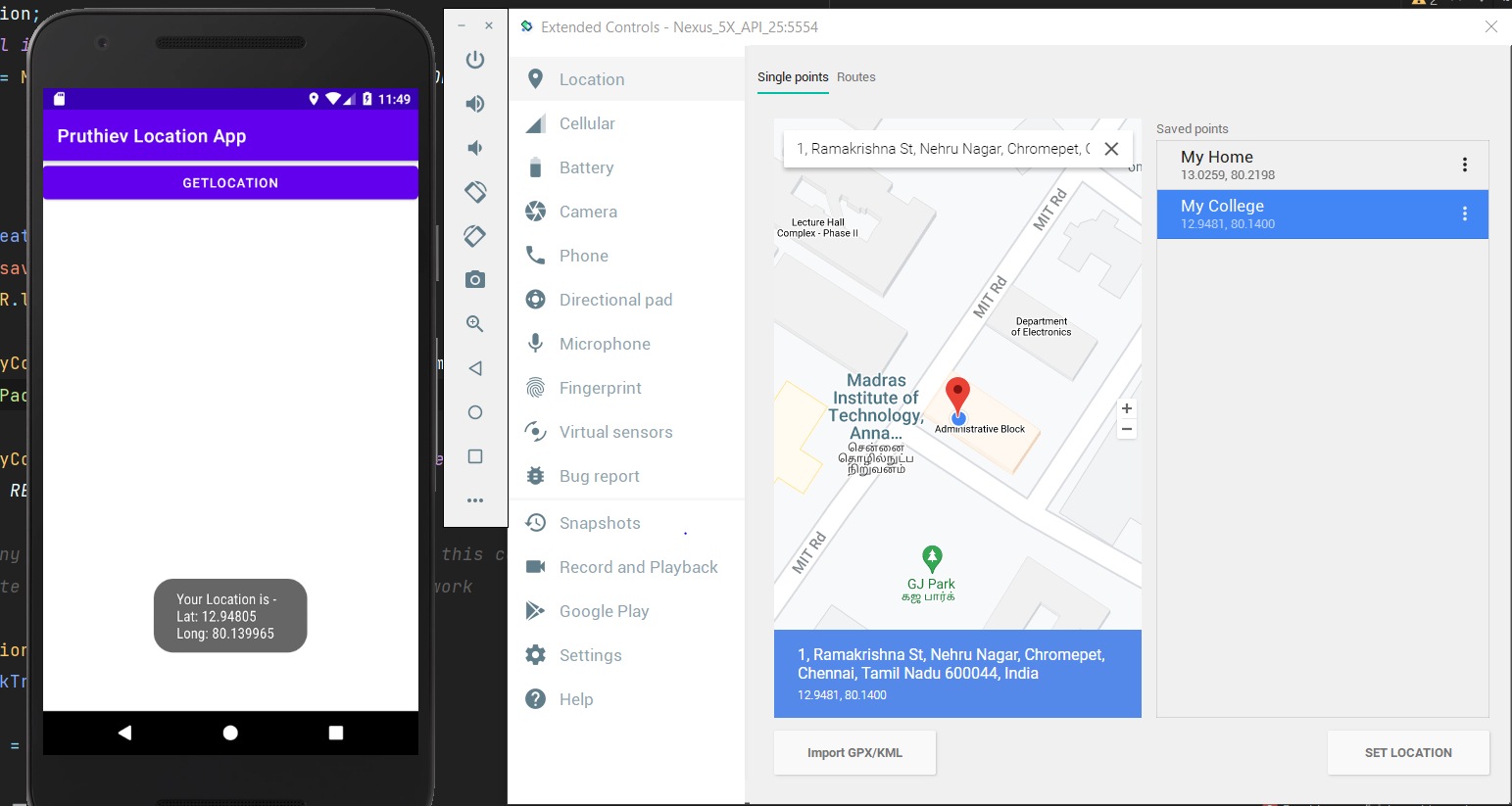
</activity>

</application>

</manifest>

**OUTPUT:**





**RESULT :**

Thus a Andoid location based Application is created using java(Android Studio) and executed successfully.