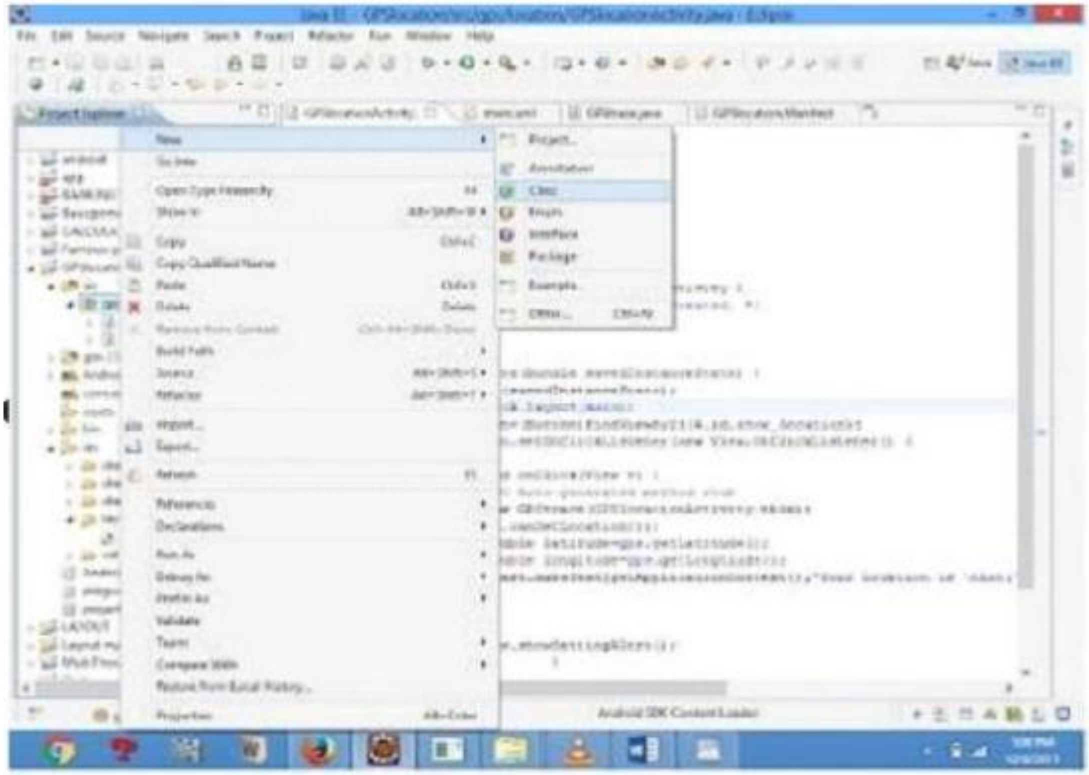
Experiment: 03

Aim: Develop an application that uses GPS location information.

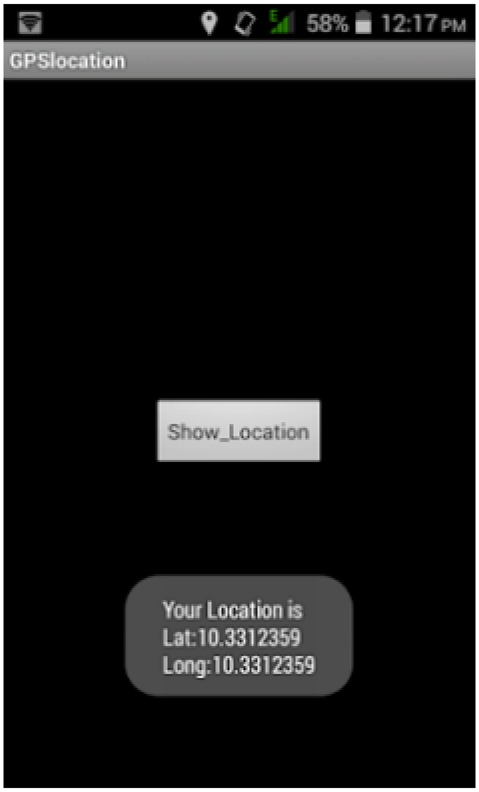
Theory:

1) Open android studio and select new android project
2) Give project name and select next
3) Choose the android version. Choose the lowest android version and select next
4) Enter the package name. package name must be two word separated by comma and click finish
5) Go to package explorer in the left hand side. select our project.
6) Go to res folder and select layout.
7)Now select main acyivity.java and type the coding.
8) Go to src folder and Right Click on your package folder and choose new class and give the class names as GPS trace.



9) Now go to main.xml and right click select run as option and select run configuration

10) Android output is present in the android emulator as shown in below.



Your Location is
Lat:10.3312359
Long:10.3312359

Implementation:

```
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 android.view.View;
import android.view.ViewGroup;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.widget.Toast;
```

```
public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
    private LocationListener locationListener;
    private LocationManager locationManager;
//Requesting Permissions from the User
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION,
Manifest.permission.ACCESS_COARSE_LOCATION},
PackageManager.PERMISSION_GRANTED);
    private GoogleMap mMap;
        @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        locationListener = new LocationListener() {
            @Override
            public void onLocationChanged(@NonNull Location location) {
                try {
                    LatLng latLng = new LatLng(location.getLatitude(),location.getLongitude());
                    mMap.clear();
                    mMap.addMarker(new MarkerOptions().position(latLng).title("Current Position"));
                    mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
                    editTextLatitude.setText(Double.toString(location.getLatitude()));
                    editTextLongitude.setText(Double.toString(location.getLongitude()));
                } catch (Exception e) {
                    e.printStackTrace();}
            }
            @Override
            public void onStatusChanged(String s, int i, Bundle bundle) {}
            @Override
            public void onProviderEnabled(String s) {
            }

            @Override

            public void onProviderDisabled(String s) {


            }

        };

        locationManager = (LocationManager) getSystemService(LOCATION_SERVICE);

        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {

            return; }

        try {

            locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER,
MIN_TIME, MIN_DIST, locationListener);

            locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER ,
MIN_TIME, MIN_DIST, locationListener);

        }

        catch (Exception e){

            e.printStackTrace();

        }}

public void readdbButtonI(View view){

        Log.d("updatedb",main_lat+"\n"+main_long);

        // Create a Toast

        Toast mToast = Toast.makeText(getApplicationContext(),"Latitude :
"+main_lat+"\nLongitude : "+main_long,Toast.LENGTH_LONG);

        mToast.setGravity(Gravity.CENTER_HORIZONTAL,0,0);

        View vieww=mToast.getView();

        TextView  view1 = vieww.findViewById(android.R.id.message);

        view1.setTextColor(Color.WHITE);
```

```
    vieww.setBackgroundResource(R.color.colorPrimaryDark);
    mToast.show();
}}
```

Output:

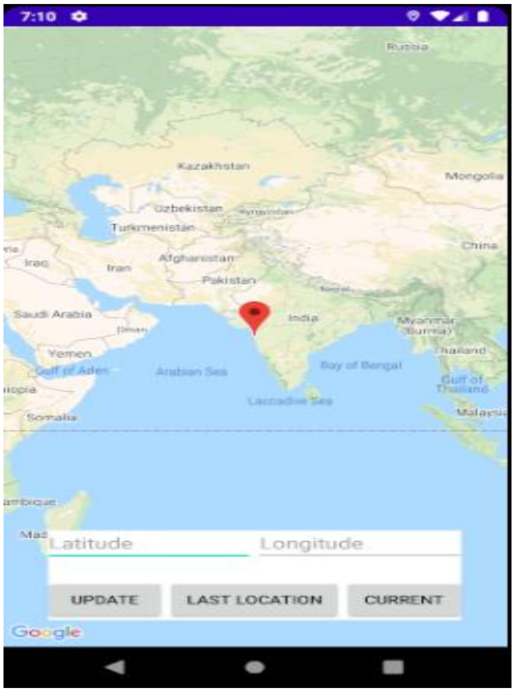
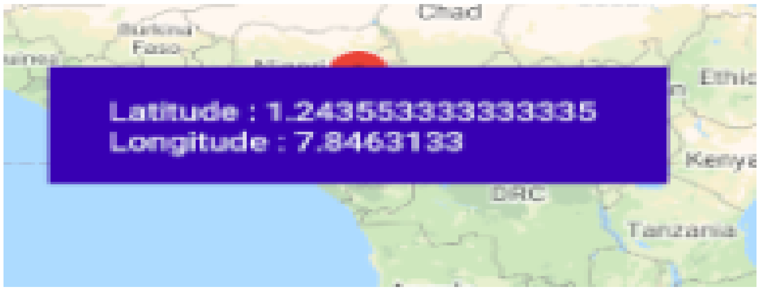


Figure 1: Maps are active with GPS



```
Latitude : 1.2435533333333335
Longitude : 7.8463133
```

Figure 2: The Latitude and Longitude

Conclusion:

From this experiment, we learnt about, how to integrate, GPS with an android application. We create a Location Manager and A location listener. To activate GPS we create a permission that user has to accept. Once the user accepts the permission, Using location listener, we used methods of getLatitude and get Longitude to get the current location of the user.