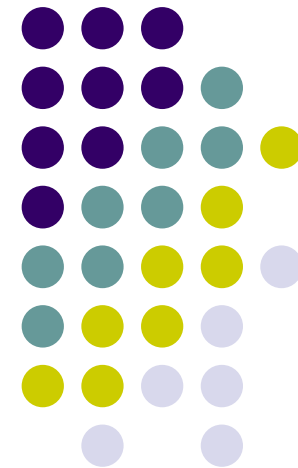


# Τεχνολογία και Προγραμματισμός Κινητών Συσκευών

8



Διαχείριση GPS  
Ιωάννης Έλληνας



# GPS & Υπηρεσίες Προσδιορισμού θέσης



## Location Based Services & Goggle maps

# Google maps



- **Google maps Android API** μέσω των Google Play Services (φροντίζουμε από την επιλογή SDK manager → SDK Tools να τα ενσωματώσουμε).
- Κλάση **MapView** για την εμφάνιση του χάρτη.
- Κλάση **Marker** για τη σημείωση πάνω στον χάρτη με τη μέθοδο **addMarker()**.
- Για την χρήση των χαρτών χρειάζεται ένα **API key** το οποίο ανακτάται με συγκεκριμένη διαδικασία αφού δημιουργήσουμε ένα project με αντικείμενο την χρήση των χαρτών Google (Περιγραφή στις Σημειώσεις Android). Το κλειδί τοποθετείται στο αρχείο μανιφέστου.



- Στο αρχείο μανιφέστου ζητείται η άδεια πρόσβασης με την ιδιότητα “ACCESS\_FINE\_LOCATION” και τοποθετείται το API key. Η τιμή του κλειδιού βρίσκεται στο περιγραφικό αρχείο google\_maps\_api.xml.

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

<meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="@string/google_maps_key" />
```

- Στο αρχείο UI (activity\_maps.xml) ορίζεται το fragment για την παρουσίαση των χαρτών.

```
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    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" />
```

- Στο αρχείο κώδικα (MapsActivity.java) χρησιμοποιείται η κλάση **GoogleMap** και η διασύνδεση “**OnMapReadyCallback**” η οποία παρέχει την μέθοδο επανάκλησης “**onMapReady()**” για τους χειρισμούς όταν οι χάρτες είναι έτοιμοι.



```
public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {  
  
    private GoogleMap mMap;  
  
    .....  
    // 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) {  
    ..... .
```



- Τοποθετούμε στη μέθοδο επανάκλησης εντολές χειρισμού των χαρτών. Για παράδειγμα δίνουμε τις συντεταγμένες της Ακρόπολης για την τοποθέτηση marker και ζουμάρουμε σε αυτήν με ζουμ=10.

```
mMap = googleMap;  
  
// Add a marker in Acropolis and move the camera  
LatLng acrop = new LatLng(37.9715323, 23.7235605);  
mMap.addMarker(new MarkerOptions().position(acrop).title("ACROPOLIS OF ATHENS"));  
mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(acrop, 10));
```

- Για να πάρουμε τις συντεταγμένες του σημείου που θα κάνουμε Touch χρησιμοποιούμε τη διασύνδεση **OnMapClickListener** και τη μέθοδο επανάκλησης **onMapClick()**. Στην εφαρμογή myMap συμπληρώστε αυτή τη δυνατότητα δίνοντας τις συντεταγμένες σε ένα Toast message (πληροφορίες στο βιβλίο).

# Γεωκοδικοποίηση (Geocoding)



- Είναι η μετατροπή της ονομασίας μιας θέσης στο χάρτη σε συντεταγμένες (π.χ. Ακρόπολη → 37.9715323, 23.7235605).
- Η αντίστροφη διαδικασία λέγεται Reverse Geocoding, όπου οι συντεταγμένες μετατρέπονται στην πλησιέστερη υπάρχουσα ονομασία.
- Στην πρώτη περίπτωση χρησιμοποιείται η κλάση **Geocoder** και η μέθοδος **getFromLocationName()**. Δώστε το όνομα “Acropolis Museum” και τοποθετείστε έναν marker στις συντεταγμένες που προκύπτουν και μια σύντομη περιγραφή μετά τον τίτλο με το `snippet()`.
- Στη δεύτερη περίπτωση χρησιμοποιείται η μέθοδος **getFromLocation()**. Δώστε τις συντεταγμένες ενός σημείου στον χάρτη ή πατήστε πάνω σε ένα σημείο και δώστε σε Toast message την περιγραφή.

# Προσδιορισμός θέσης



- Η κλάση για την υπηρεσία LBS είναι η **LocationManager**, η οποία περιέχει περιοδικές ενημερώσεις για τη θέση της συσκευής με τη μέθοδο **requestLocationUpdates()** με διάφορες παραμέτρους (πάροχο, ελάχιστο χρόνο ειδοποίησης, ελάχιστη μεταβολή απόστασης και τον ακροατή **LocationListener**). Στον ακροατή υπάρχουν μέθοδοι επανάκλησης για τη μεταβολή της θέσης (**onLocationChanged**), του παρόχου (**onStatusChanged**).
- Στο αρχείο μανιφέστου πρέπει να υπάρχει η άδεια “ACTION\_FINE\_LOCATION” ή πρέπει να υπάρχει στον κώδικα runtime permission (για συσκευές με API 6.0 και μετά-Marshmallow).



```
public class MainActivity extends AppCompatActivity implements LocationListener{

    LocationManager locationManager;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        .....
        locationManager=(LocationManager) getSystemService(LOCATION_SERVICE);
    }

    @Override
    public void onLocationChanged(Location location) {
        textView1.setText(String.valueOf(location.getLongitude()));
        textView2.setText(String.valueOf(location.getLatitude()));
    }
}
```



- Στον κώδικα τα runtime permissions παρέχονται ως εξής:

```
if (ActivityCompat.checkSelfPermission(this,
    Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
    ActivityCompat.requestPermissions(this, new
        String[] {Manifest.permission.ACCESS_FINE_LOCATION}, 1);
}
else {
    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, this);
}
```

# Εργασία



- Να γίνει εφαρμογή η οποία να σημειώνει στον χάρτη την τρέχουσα θέση της συσκευής μας, με τίτλο το όνομά μας. Η εφαρμογή θα εστιάζει στον χάρτη στην τρέχουσα θέση.
- Εάν απομακρυνθούμε από τη θέση μας κατά μια ορισμένη απόσταση θα δίνει ένα Toast message “ALERT”.