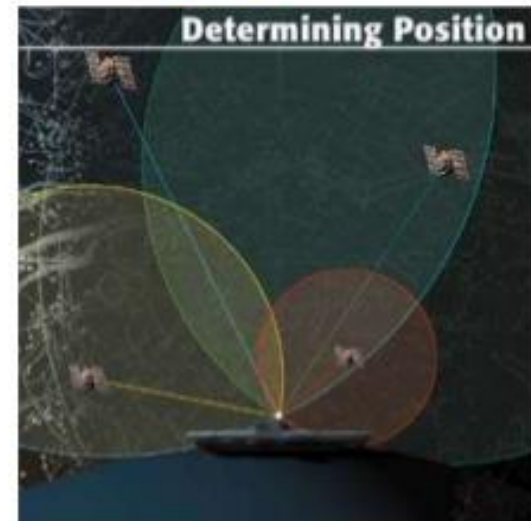
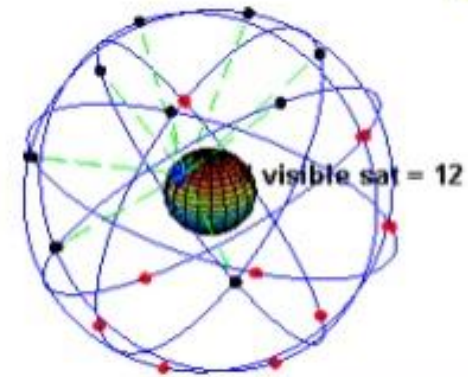
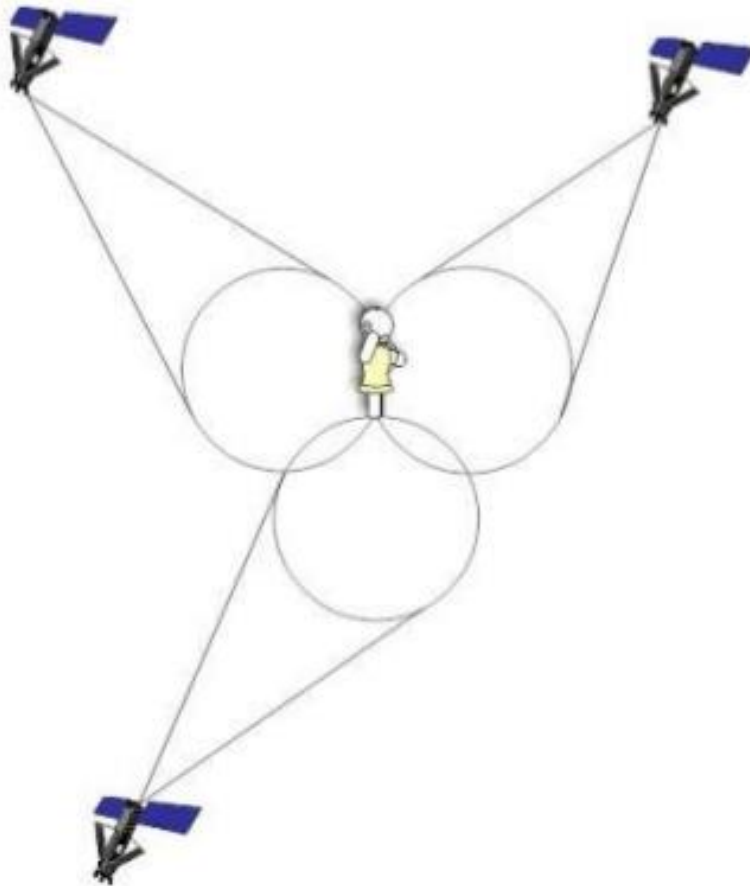


# Android Geolocation



# GPS Technology: Determining Position



# Location Services

- Android gives your applications access to the location services supported by the device through classes in the `android.location` package
- The central component of the location framework is the [LocationManager](#) system service, which provides APIs to determine location and bearing of the underlying device

# LocationManager

- You do not instantiate a [LocationManager](#) directly
- You request an instance from the system by calling:  
*getSystemService(Context.LOCATION\_SERVICE)*
- The method returns a handle to a new LocationManager instance.

# Δυνατότητες χρήσης του LocationManager

- Query for the list of all LocationProviders for the last known user location.
- Register/unregister for periodic updates of the user's current location from a location provider (specified either by criteria or name).
- Register/unregister for a given Intent to be fired if the device comes within a given proximity (specified by radius in meters) of a given lat/long.

# requestLocationUpdates(...)

```
public void requestLocationUpdates (String provider, long minTime, float minDistance, LocationListener listener)
```

Register for location updates using the named provider, and a pending intent.

See `requestLocationUpdates(long, float, Criteria, PendingIntent)` for more detail on how to use this method.

## Parameters

- |                    |   |
|--------------------|---|
| <i>provider</i>    | the name of the provider with which to register   |
| <i>minTime</i>     | minimum time interval between location updates, in milliseconds   |
| <i>minDistance</i> | minimum distance between location updates, in meters  |
| <i>listener</i>    | a <code>LocationListener</code> whose <code>onLocationChanged(Location)</code> method will be called for each location update |

## Throws

- |                                 |   |
|---------------------------------|---|
| <i>IllegalArgumentException</i> | if provider is null or doesn't exist on this device |
| <i>IllegalArgumentException</i> | if listener is null                                 |
| <i>RuntimeException</i>         | if the calling thread has no Looper                 |
| <i>SecurityException</i>        | if no suitable permission is present                |

# removeUpdates(...)

```
public void removeUpdates (LocationListener listener)
```

Removes all location updates for the specified `LocationListener`.

Following this call, updates will no longer occur for this listener.

## Parameters

*listener* listener object that no longer needs location updates

## Throws

[IllegalArgumentException](#) if listener is null

# getLastKnownLocation(...)

```
public Location getLastKnownLocation (String provider)
```

Added in [API level 1](#)

Returns a [Location](#) indicating the data from the last known location fix obtained from the given provider.

This can be done without starting the provider. Note that this location could be out-of-date, for example if the device was turned off and moved to another location.

If the provider is currently disabled, null is returned.

## Parameters

*provider* the name of the provider

## Returns

the last known location for the provider, or null

## Throws

[SecurityException](#) if no suitable permission is present

[IllegalArgumentException](#) if provider is null or doesn't exist



# 1<sup>st</sup> Step

## Add permissions in Manifest

```
<uses-sdk  
    android:minSdkVersion="16"  
    android:targetSdkVersion="19" />  
<uses-permission android:name="android.permission.INTERNET"/>  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

## 2<sup>nd</sup> Step

# Request Location Updates, Implementing “LocationListener”

```
public class MainActivity extends Activity implements LocationListener {  
    private LocationManager locationManager;
```

```
    locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
```

```
    locationManager.requestLocationUpdates( LocationManager.GPS_PROVIDER, 3000, 10, this);
```

# Public Interface LocationListener

Public Methods	
abstract void	<code>onLocationChanged(Location location)</code> Called when the location has changed.
abstract void	<code>onProviderDisabled(String provider)</code> Called when the provider is disabled by the user.
abstract void	<code>onProviderEnabled(String provider)</code> Called when the provider is enabled by the user.
abstract void	<code>onStatusChanged(String provider, int status, Bundle extras)</code> Called when the provider status changes.

- **void android.location.LocationManager.requestLocationUpdates(String provider, long minTime, float minDistance, LocationListener listener)**

***public void requestLocationUpdates ([String](#) provider, long minTime, float minDistance, [LocationListener](#) listener)***

Added in [API level 1](#)

Register for location updates using the named provider, and a pending intent.

See [requestLocationUpdates\(long, float, Criteria, PendingIntent\)](#) for more detail on how to use this method.

# 3<sup>rd</sup> Step

## Override “onLocationChanged” function

```
@Override
    public void onLocationChanged(Location location) {
        double lat = (double) (location.getLatitude());
        double lng = (double) (location.getLongitude());
        latitudeField.setText(String.valueOf(lat));
        longitudeField.setText(String.valueOf(lng));
    }
```

# How to stop location tracking?

```
locationManager.removeUpdates(this);
```

