

# Localizzazione

Lezione 12

# Localizzazione

---

- Una delle caratteristiche uniche delle applicazioni mobili è la consapevolezza della posizione (location awareness)
- Gli utenti di dispositivi mobili portano con sé i propri dispositivi ovunque e l'aggiunta della consapevolezza della posizione alla app offre agli utenti un'esperienza più contestuale
- Le API di location disponibili nei servizi di Google Play facilitano l'aggiunta di informazioni sulla posizione con rilevamento automatico della posizione, geofencing (la latitudine, la longitudine e il raggio definiscono un geofence, creando un'area circolare intorno alla posizione di interesse) e riconoscimento delle attività (come camminare, guidare, ecc -- Activity Recognition Transition API)

# Richiedere location permissions

---

- Per proteggere la privacy degli utenti, le app che utilizzano i servizi di localizzazione devono richiedere le autorizzazioni di localizzazione
- Quando richiedi le autorizzazioni di localizzazione, segui le stesse best practice che faresti per qualsiasi altra autorizzazione di runtime
- Una differenza importante quando si tratta di autorizzazioni di posizione è che il sistema include più autorizzazioni relative alla posizione
  - Le autorizzazioni che richiedi e il modo in cui le richiedi dipendono dai requisiti di posizione per il caso d'uso della tua app

# Tipi di location access

---

- Ogni autorizzazione ha una combinazione delle seguenti caratteristiche:
  - **Category:** Foreground location oppure Background location
  - **Accuracy:** Posizione precisa o posizione approssimativa

# Foreground location

- Se l'app contiene una funzionalità che condivide o riceve informazioni sulla posizione solo una volta o per un periodo di tempo definito, tale funzionalità richiede l'accesso alla posizione in foreground
  - Alcuni esempi includono quanto segue:
    - All'interno di un'app di navigazione, una funzione consente agli utenti di ottenere indicazioni stradali dettagliate
    - All'interno di un'app di messaggistica, una funzionalità consente agli utenti di condividere la loro posizione corrente con un altro utente
- Foreground location - se una funzionalità dell'app accede alla posizione corrente del dispositivo in una delle seguenti situazioni:
  - È visibile un'activity che appartiene alla tua app. La tua app esegue un servizio in primo piano. Quando un servizio in primo piano è in esecuzione, il sistema aumenta la consapevolezza dell'utente mostrando una notifica persistente
  - La tua app mantiene l'accesso quando viene posizionata in background, ad esempio quando l'utente preme il pulsante Home sul proprio dispositivo o spegne il display del proprio dispositivo

# Foreground location

- Occorre dichiarare un servizio di tipo foreground

<!-- Recommended for Android 9 (API level 28) and lower. -->

<!-- Required for Android 10 (API level 29) and higher. -->

<service

    android:name="MyNavigationService"

**android:foregroundServiceType="location" ... >**

    <!-- Any inner elements would go here. -->

</service>

# Foreground location

- Dichiarare il bisogno di un foreground location quando si usi ACCESS\_COARSE\_LOCATION oppure ACCESS\_FINE\_LOCATION :

```
<manifest ... >
```

```
<!-- Always include this permission -->
```

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

```
<!-- Include only if your app benefits from precise location access. -->
```

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

# Background location

---

- Un'app richiede l'accesso alla posizione in background se una funzionalità all'interno dell'app condivide costantemente la posizione con altri utenti o utilizza l'API Geofencing
- Diversi esempi includono quanto segue:
  - All'interno di un'app di condivisione della posizione della famiglia, una funzione consente agli utenti di condividere continuamente la posizione con i membri della famiglia
  - All'interno di un'app IoT, una funzionalità consente agli utenti di configurare i propri dispositivi domestici in modo che si spengano quando l'utente esce di casa e si riaccendano quando l'utente torna a casa. Il sistema considera l'app che utilizza la posizione in background se accede alla posizione corrente del dispositivo in una situazione diversa da quelle descritte nella sezione posizione in primo piano



# Background location

- Su Android 10 (livello API 29) e versioni successive, devi dichiarare l'autorizzazione `ACCESS_BACKGROUND_LOCATION` nel manifest della tua app per richiedere l'accesso alla posizione in background in fase di esecuzione. Nelle versioni precedenti di Android, quando l'app riceve l'accesso alla posizione in primo piano, riceve automaticamente anche l'accesso alla posizione in background

```
<manifest ... >
```

```
  <!-- Required only when requesting background location access on  
    Android 10 (API level 29) and higher. -->
```

```
  <uses-permission
```

```
    android:name="android.permission.ACCESS_BACKGROUND_LOCATION" />
```

```
</manifest>
```

# Accuracy

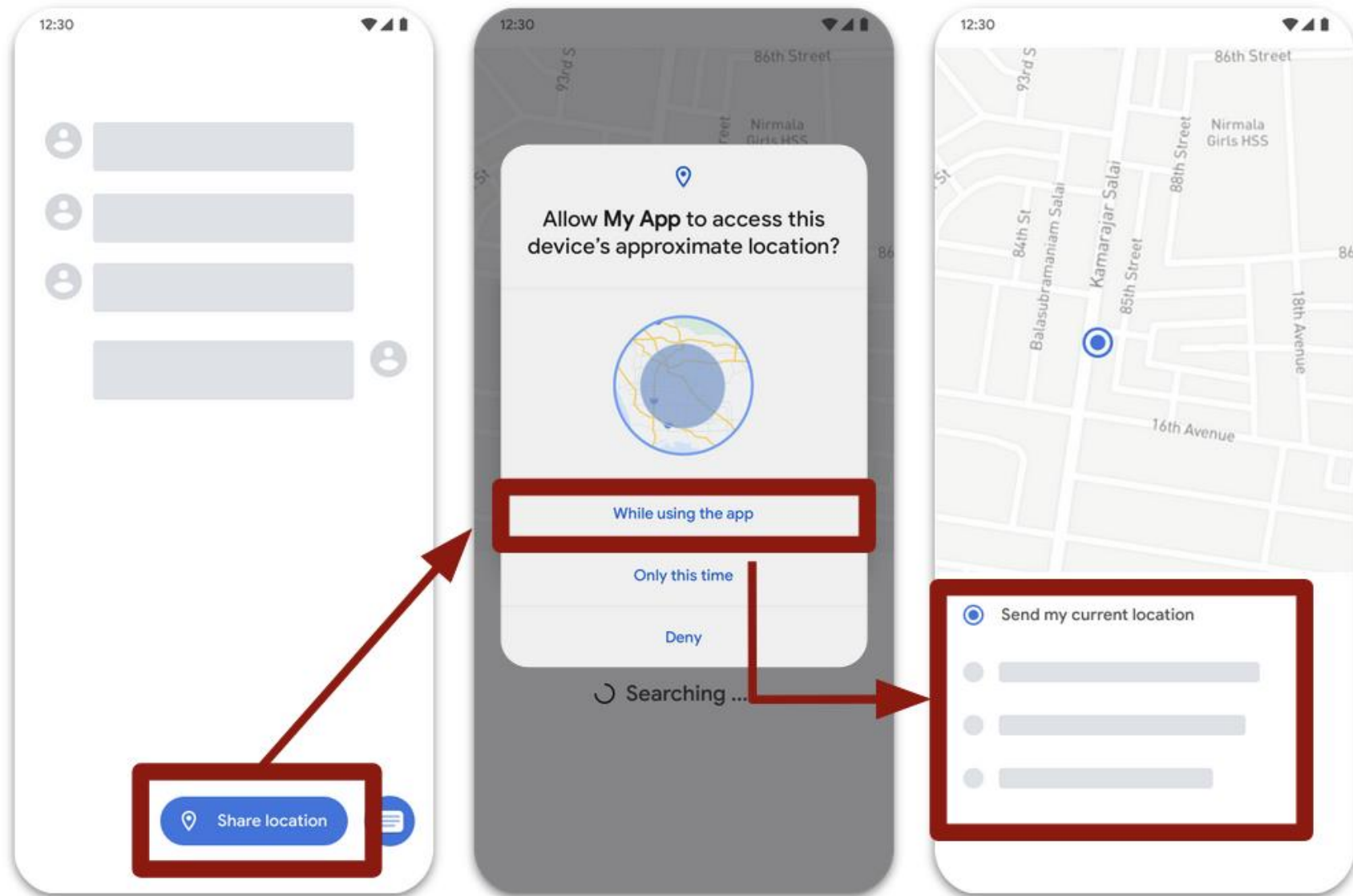
---

- Android supporta due livelli di accuracy
  - **Approximate**
    - Fornisce una stima della posizione del dispositivo. Se questa stima della posizione proviene da LocationManagerService o FusedLocationProvider, questa stima è accurata entro circa 3 chilometri quadrati (circa 1,2 miglia quadrate). La tua app può ricevere posizioni con questo livello di precisione quando dichiari l'autorizzazione ACCESS\_COARSE\_LOCATION ma non l'autorizzazione ACCESS\_FINE\_LOCATION
  - **Precise**
    - Fornisce una stima della posizione del dispositivo quanto più precisa possibile. Se la stima della posizione proviene da LocationManagerService o FusedLocationProvider, questa stima è in genere entro circa 50 metri (160 piedi) e talvolta è precisa entro pochi metri (10 piedi) o superiore. La tua app può ricevere posizioni con questo livello di precisione se dichiari l'autorizzazione ACCESS\_FINE\_LOCATION

# Richiedere permessi a runtime

---

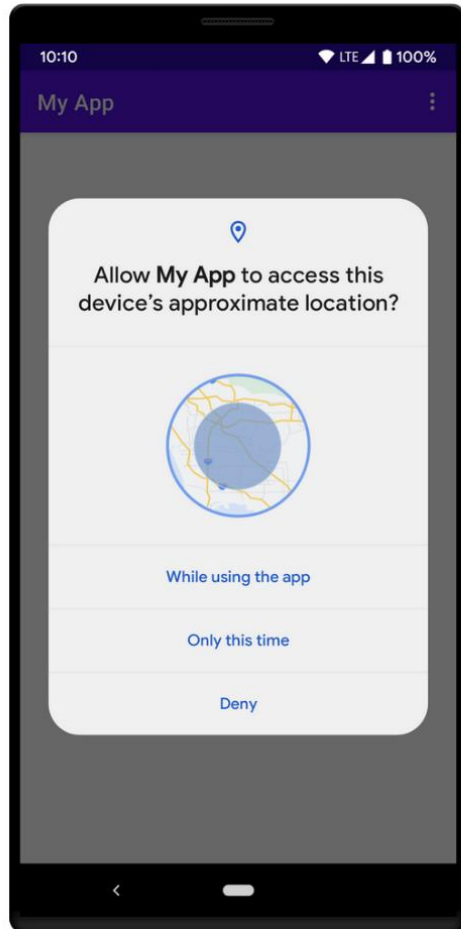
- Quando una funzionalità nella tua app richiede l'accesso alla posizione, attendi che l'utente interagisca con la funzionalità prima di effettuare la richiesta di autorizzazione
- Questo flusso di lavoro segue la best practice per richiedere le autorizzazioni di runtime nel contesto (lezione Permessi!)



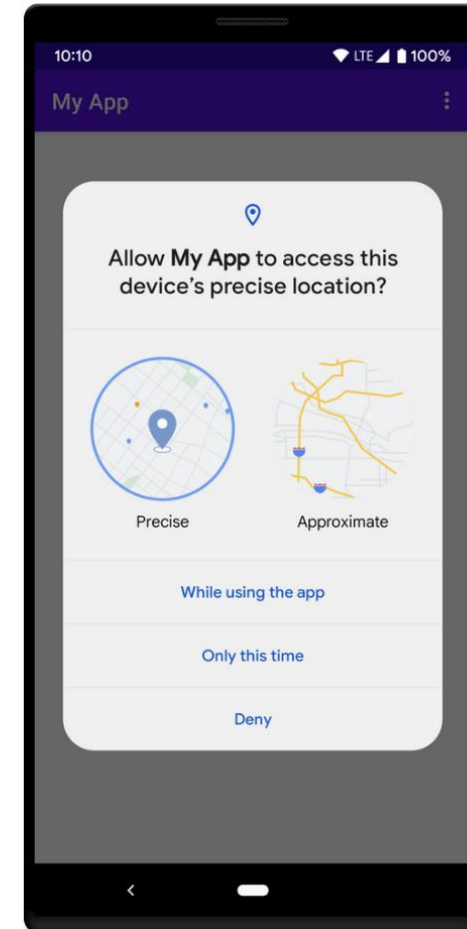
**Figure 1.** Location-sharing feature that requires foreground location access. The feature is enabled if the user selects **Allow only while using the app**.

# L'utente può concedere l'autorizzazione solo alla posizione approssimativa

- Su Android 12 (livello API 31) o versioni successive, gli utenti possono richiedere che la tua app recuperi solo informazioni sulla posizione approssimativa, anche quando la tua app richiede l'autorizzazione di runtime `ACCESS_FINE_LOCATION`
- Per gestire questo potenziale comportamento dell'utente, non richiedere l'autorizzazione `ACCESS_FINE_LOCATION` da sola. Richiedi invece sia l'autorizzazione `ACCESS_FINE_LOCATION` che l'autorizzazione `ACCESS_COARSE_LOCATION` in un'unica richiesta di runtime.
- Quando la tua app richiede sia `ACCESS_FINE_LOCATION` che `ACCESS_COARSE_LOCATION`, la finestra di dialogo delle autorizzazioni di sistema include le seguenti opzioni per l'utente:
  - Precise: consente all'app di ottenere informazioni precise sulla posizione.
  - Approximate: consente all'app di ottenere solo informazioni sulla posizione approssimativa



**Figure 2.** System permissions dialog that appears when your app requests `ACCESS_COARSE_LOCATION` only.



**Figure 3.** System permissions dialog that appears when your app requests both `ACCESS_FINE_LOCATION` and `ACCESS_COARSE_LOCATION` in a single runtime request.

# La scelta dell'utente influisce sulle autorizzazioni

- La tabella seguente mostra le autorizzazioni che il sistema concede all'app, in base alle opzioni scelte dall'utente nella finestra di dialogo delle autorizzazioni di runtime

	Precise	Approximate
While using the app	ACCESS_FINE_LOCATION and ACCESS_COARSE_LOCATION	ACCESS_COARSE_LOCATION
Only this time	ACCESS_FINE_LOCATION and ACCESS_COARSE_LOCATION	ACCESS_COARSE_LOCATION
Deny	No location permissions	No location permissions

- Per determinare quali autorizzazioni l'utente ha concesso alla tua app, controlla il valore restituito dalla tua richiesta di autorizzazione

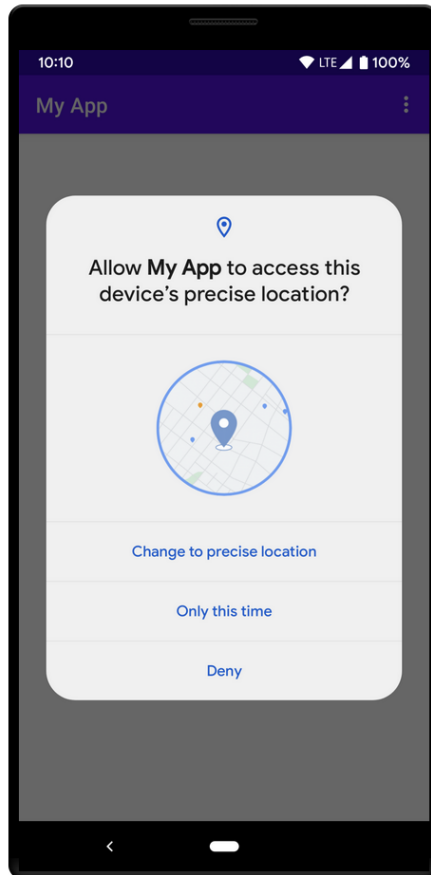
```
val locationPermissionRequest = registerForActivityResult(  
    ActivityResultContracts.RequestMultiplePermissions()  
) { permissions ->  
    when {  
        permissions.getDefault(Manifest.permission.ACCESS_FINE_LOCATION, false) -> {  
            // Precise location access granted.  
        }  
        permissions.getDefault(Manifest.permission.ACCESS_COARSE_LOCATION, false) -> {  
            // Only approximate location access granted.  
        } else -> {  
            // No location access granted.  
        }  
    }  
}
```

// ...

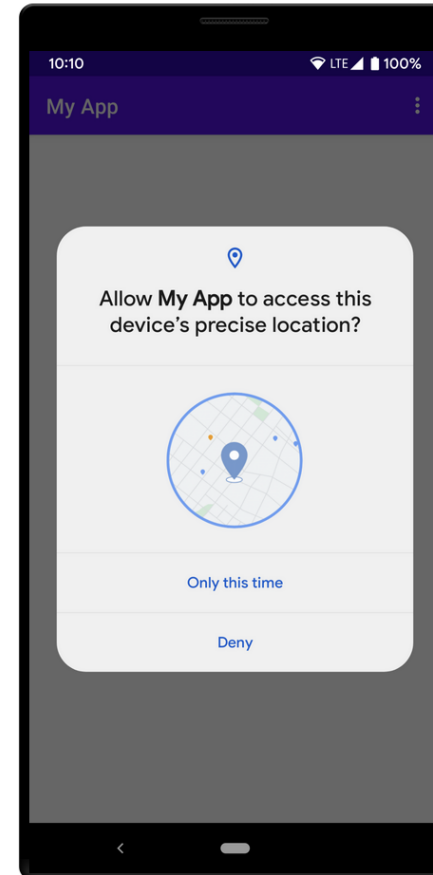
// Before you perform the actual permission request, check whether your app  
// already has the permissions, and whether your app needs to show a permission  
// rationale dialog. For more details, see [Request permissions](#).  
locationPermissionRequest.launch(arrayOf(  
 Manifest.permission.ACCESS\_FINE\_LOCATION,  
 Manifest.permission.ACCESS\_COARSE\_LOCATION))



# Richiedi un upgrade a una posizione precisa



**Figure 4.** The user previously selected **Approximate** and **While using the app** (in the dialog from figure 3).



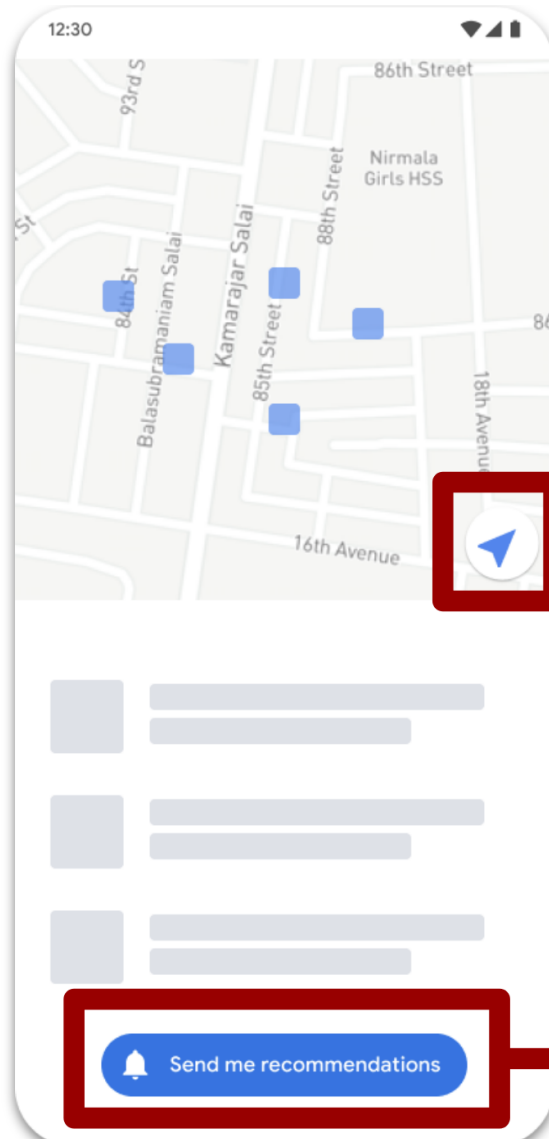
**Figure 5.** The user previously selected **Approximate** and **Only this time** (in the dialog from figure 3).

# Nota

---

- Anche se diverse funzionalità della tua app richiedono l'accesso alla posizione, è probabile che solo alcune di esse richiedano l'accesso alla posizione in background
- Pertanto, è consigliabile che l'app esegua richieste incrementalmente per le autorizzazioni di posizione, chiedendo l'accesso alla posizione in primo piano e successivamente l'accesso alla posizione in background
  - Eseguendo richieste incrementalmente, offri agli utenti maggiore controllo e trasparenza perché possono capire meglio quali funzionalità della tua app richiedono l'accesso alla posizione in background

# Esempio: Richieste incrementali



“Show current location”  
feature

Requires foreground  
location access

“Recommend nearby places”  
feature

Requires background  
location access

# Usare la localizzazione

---

- Controlla se l'autorizzazione alla posizione è stata concessa
- Chiedi il permesso se necessario
- Richiedi la posizione più recente
- Richiedi aggiornamenti sulla posizione

# Ottenere la posizione del device

---

- Usa *FusedLocationProviderClient*
- Invia richieste di posizione combinando GPS, Wi-Fi e rete cellulare
- Bilancia risultati rapidi e precisi con un consumo minimo della batteria
- Restituisce l'oggetto *Location* con latitudine e longitudine
  - Si può utilizzare Geocoder per ottenere l'indirizzo fisico date lat e long

# Ottenere l'ultima posizione

---

- Utilizzando le API di localizzazione di Google Play Services, la tua app può richiedere l'ultima posizione nota del dispositivo dell'utente
  - Nella maggior parte dei casi, sei interessato alla posizione corrente dell'utente, che di solito equivale all'ultima posizione nota del dispositivo
- In particolare, puoi utilizzare il fused location provider per ottenere l'ultima posizione nota del dispositivo
  - Il fused location provider è una delle API di location nei servizi di Google Play
  - Gestisce la tecnologia di localizzazione sottostante e fornisce una semplice API in modo da poter specificare i requisiti ad alto livello, come alta precisione o bassa potenza
  - Ottimizza inoltre l'utilizzo della carica della batteria da parte del dispositivo

# Steps

---

## 1. Set up Google Play services

- Per accedere al fused location provider, il tuo progetto deve includere Google Play Services
  - Scarica e installa il componente Google Play Services tramite SDK Manager e aggiungi la libreria al tuo progetto ([dettagli](#))

# Setting up Google Play services

---

- Location services sono forniti da Google Play Services
- Install Google Repository in Android Studio
  1. Select **Tools > SDK Manager > Android SDK**
  2. Select the **SDK Tools** tab
  3. Select **Google Play Services** and click **OK**



# Aggiungere Google Play al progetto

---

- Aggiungere in *dependencies* in build.gradle (Module: app), l'ultima versione di play-services

```
apply plugin: 'com.android.application'
```

```
...
```

```
dependencies {  
    implementation 'com.google.android.gms:play-services-location:21.3.0'  
}
```

# Steps

---

## 1. Set up Google Play services

- Per accedere al fused location provider, il tuo progetto deve includere Google Play Services
  - Scarica e installa il componente Google Play Services tramite SDK Manager e aggiungi la libreria al tuo progetto ([dettagli](#))

## 2. Specifica app permissions

## 3. Crea location services client

1. Nel metodo onCreate() della tua attività, crea un'istanza del client Fused Location Provider

### 3. Crea location services client

---

```
private lateinit var fusedLocationClient:  
FusedLocationProviderClient  
  
override fun onCreate(savedInstanceState: Bundle?) {  
    // ...  
  
    fusedLocationClient =  
    LocationServices.getFusedLocationProviderClient(this)  
}
```

## 4. Ottenere l'ultima posizione conosciuta

---

- Dopo aver creato il Fused Location Provider Client, è possibile ottenere l'ultima posizione nota del dispositivo di un utente attraverso il metodo *getLastLocation()* del provider di posizione per recuperare la posizione del dispositivo
  - Nota: solitamente l'ultima posizione è quella corrente
  - La precisione della posizione restituita da questa chiamata è determinata dalle impostazioni di autorizzazione inserite nel manifest dell'app

## 4. Ottieni l'ultima posizione nota

---

```
fusedLocationClient.lastLocation
    .addOnSuccessListener { location : Location? ->
        // Got last known location. In some rare situations this can
        be null.
    }
```

# getLastLocation()

- Il metodo `getLastLocation()` restituisce il Task che è possibile utilizzare per ottenere un oggetto Location con le coordinate di latitudine e longitudine di una posizione geografica
  - Tramite l'oggetto Location si può ricavare latitudine e longitudine
- L'oggetto location potrebbe essere null nelle seguenti situazioni:
  - La posizione è disattivata nelle impostazioni del dispositivo. Il risultato potrebbe essere nullo anche se l'ultima posizione è stata precedentemente recuperata poiché la disabilitazione della posizione cancella anche la cache
  - Il dispositivo non ha mai registrato la sua posizione, che potrebbe essere il caso di un nuovo dispositivo o di un dispositivo che è stato ripristinato alle impostazioni di fabbrica
  - I servizi di Google Play sul dispositivo sono stati riavviati e non esiste alcun Fused Location Provider Client attivo che abbia richiesto la posizione dopo il riavvio dei servizi. Per evitare questa situazione, puoi creare un nuovo client e richiedere tu stesso gli aggiornamenti della posizione

# Richiedi aggiornamenti sulla posizione

---

- La tua app può ottenere l'ultima posizione nota
- Può anche richiedere aggiornamenti regolari per monitorare la posizione
- Utilizzare `LocationRequest` e `LocationCallback` per impostare i parametri per le richieste di aggiornamento della posizione
  - Maggiori info:  
<https://developer.android.com/training/location/request-updates>

# Request location updates

---

```
override fun onResume() {  
    super.onResume()  
    if (requestingLocationUpdates) startLocationUpdates()  
}  
  
private fun startLocationUpdates() {  
    fusedLocationClient.requestLocationUpdates(locationRequest,  
        locationCallback,  
        Looper.getMainLooper())  
}
```

- Fai una location request



# Request location updates

```
private lateinit var locationCallback: LocationCallback
```

```
// ...
```

```
override fun onCreate(savedInstanceState: Bundle?) {
```

```
    // ...
```

```
    locationCallback = object : LocationCallback() {
```

```
        override fun onLocationResult(locationResult: LocationResult?) {
```

```
            locationResult ?: return
```

```
            for (location in locationResult.locations){
```

```
                // Update UI with location data
```

```
                // ...
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

- Definire la callback posizione aggiornata

# Request location updates

---

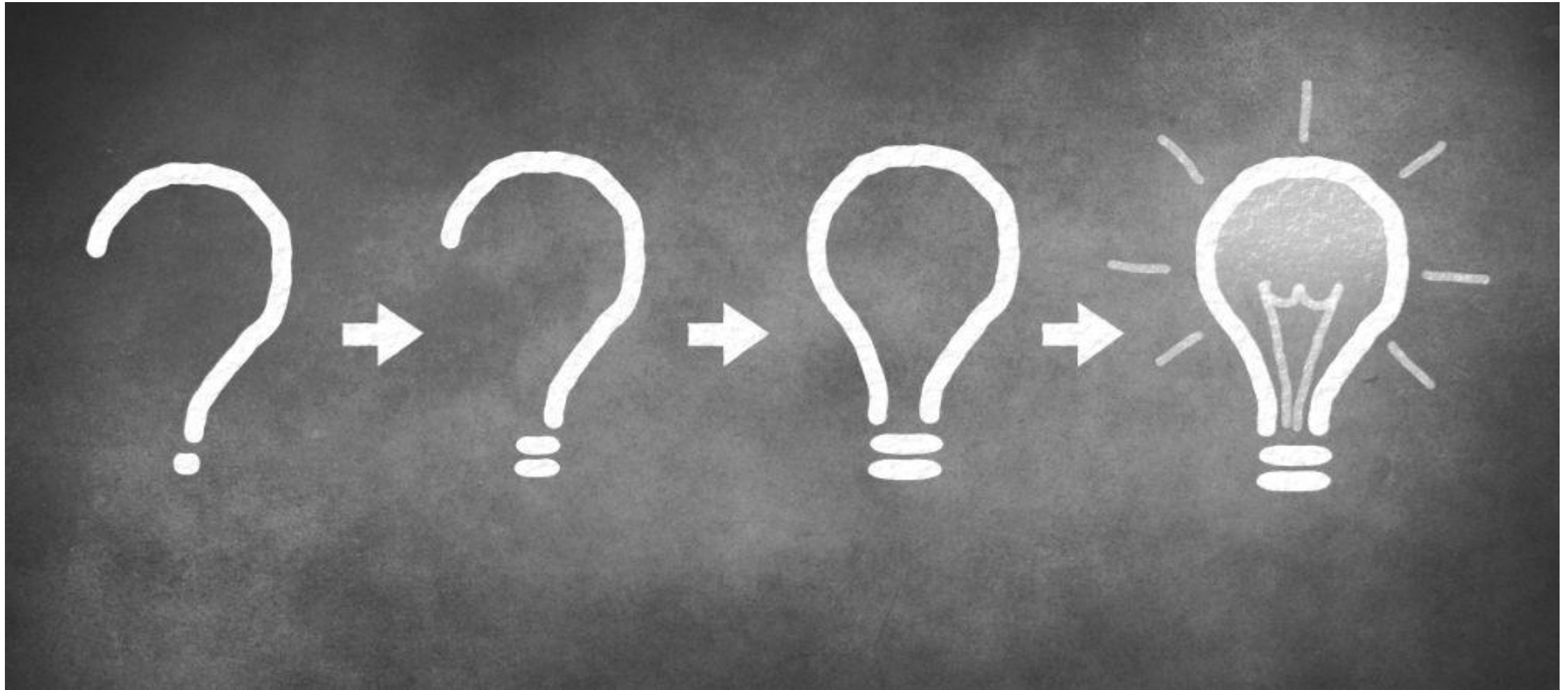
```
override fun onPause() {  
    super.onPause()  
    stopLocationUpdates()  
}
```

```
private fun stopLocationUpdates() {  
    fusedLocationClient.removeLocationUpdates(locationCallback)  
}
```

```
override fun onResume() {  
    super.onResume()  
    if (requestingLocationUpdates) startLocationUpdates()  
}
```

- Stoppare location updates e restart

# Domande???



# Riferimenti

---

- <https://developer.android.com/training/location>
- <https://developer.android.com/training/location/retrieve-current>
- <https://developer.android.com/training/location/permissions>
- <https://developers.google.com/location-context/>
- Approfondimenti
  - <https://developer.android.com/guide/topics/location/transitions>
  - <https://developer.android.com/training/location/geofencing>