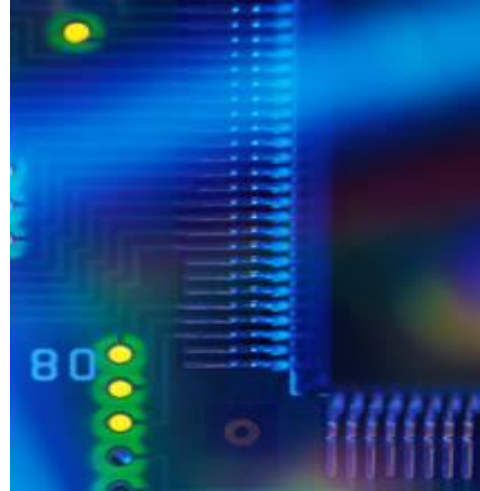




KEMENTERIAN KOMUNIKASI DAN INFORMATIKA  
REPUBLIK INDONESIA

*Menuju Masyarakat Informasi Indonesia*



## JUNIOR MOBILE PROGRAMMER

# Menyusun Mobile Location Based Service, GPS dan Mobile Navigation

# Deskripsi Singkat

## Deskripsi Singkat mengenai Topik

Pada pelatihan ini akan memberikan kompetensi untuk menyusun mobile location based service, GPS dan mobile navigation

## Tujuan Pelatihan

1. Mampu menentukan lokasi dengan menggunakan perangkat mobile computing
2. Mampu menentukan lokasi pengguna dengan Location Based Service
3. Mampu menentukan lokasi pengguna dengan GPS
4. Mampu menerapkan Mobile Navigation pada perangkat mobile computing

# Deskripsi Singkat

## Materi Yang akan disampaikan:

1. Location Base Service
2. Global Positioning System (A-GPS)
3. Get The Last Know Location
4. Change Location Setting
5. Receive Location Update
6. Display a Location Address
7. Add Maps
8. Algoritma Mobile Navigation

## Tugas :

Membuat aplikasi navigasi dengan menggunakan Google Maps API

## Outcome/Capaian Pelatihan

Menyusun mobile location based service, GPS dan mobile navigation

# Location Based Service

# Pelatihan



Location Based Service menentukan lokasi pengguna berdasarkan GSM Network, Cell Tower dan Wi-Fi signals.

## Location Based Service

## Pelatihan

LBS (Location Based Service) adalah sebuah layanan informasi yang dapat diakses dengan perangkat bergerak melalui jaringan dan mampu menampilkan posisi informasi geografis keberadaan perangkat bergerak tersebut

## Location Based Service

## Pelatihan

### Kelebihan

- Mampu memberikan informasi lokasi pengguna baik di indoor dan outdoor
- Respons lebih cepat
- Lebih hemat baterai

## Location Based Service

## Pelatihan

### Kelemahan

- Informasi lokasi pengguna tidak se-akurat dibandingkan dengan GPS,
- Karena berdasarkan lokasi tower dan wifi
- Hasil akurasi pergeseran pembacaan yang diperoleh adalah sebesar 10.9489 meter

## **Location Based Service**

## **Pelatihan**

### **Metode Location Based Service**

**Enhanced Observed Time Difference (E-OTD)**

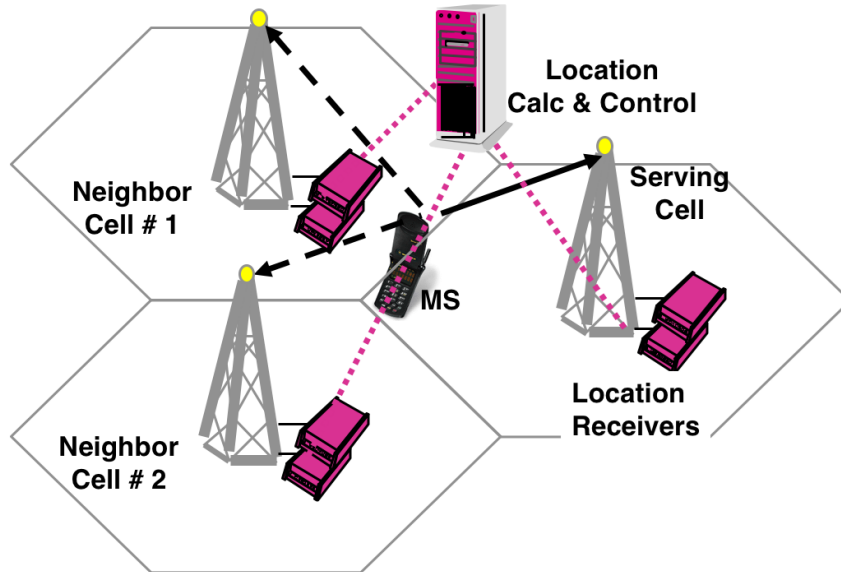
**Time Difference of Arrival (TDOA)**



# Location Based Service

## Pelatihan

### Time Difference of Arrival (TDOA)

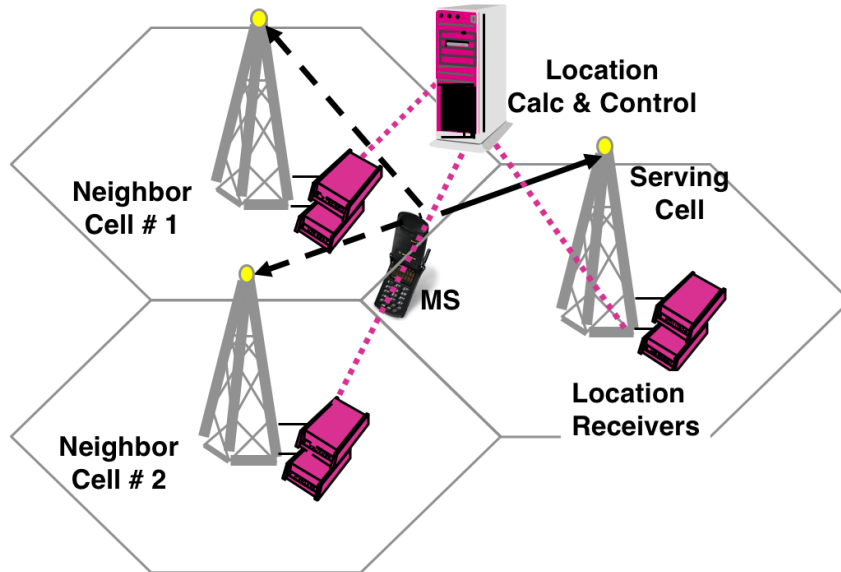


- Berdasarkan pada perbedaan waktu kedatangan
- Menggunakan perhitungan perbedaan waktu dari minimal tiga BTS yang terdekat

# Location Based Service

# Pelatihan

## Time Difference of Arrival (TDOA)

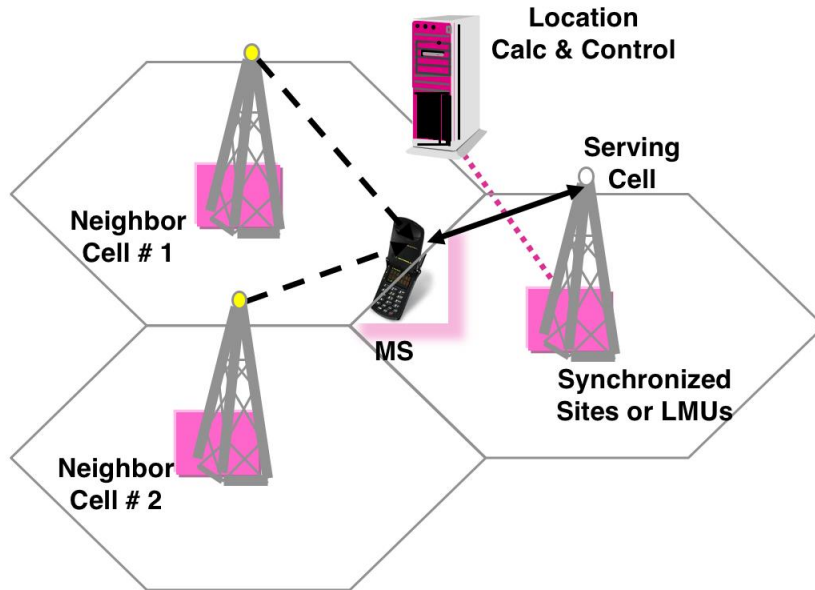


- Minimum of 3 sites measure arrival time of MS transmission
- Location solution based on apparent arrival time differences between pairs of sites

# Location Based Service

# Pelatihan

## Enhanced Observed Time Difference (E-OTD)

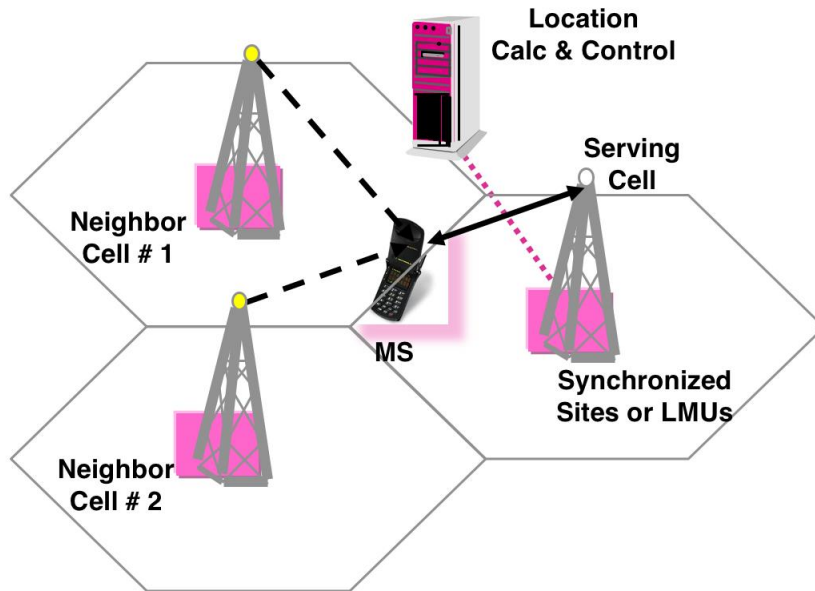


- Menggunakan perhitungan yang hampir sama dengan metode TDoA
- Kedua sisi yaitu BTS dan perangkat pengguna (ditambahkan aplikasi tertentu) digunakan untuk menghitung perbedaan waktu antara BTS

# Location Based Service

# Pelatihan

## Enhanced Observed Time Difference (E-OTD)

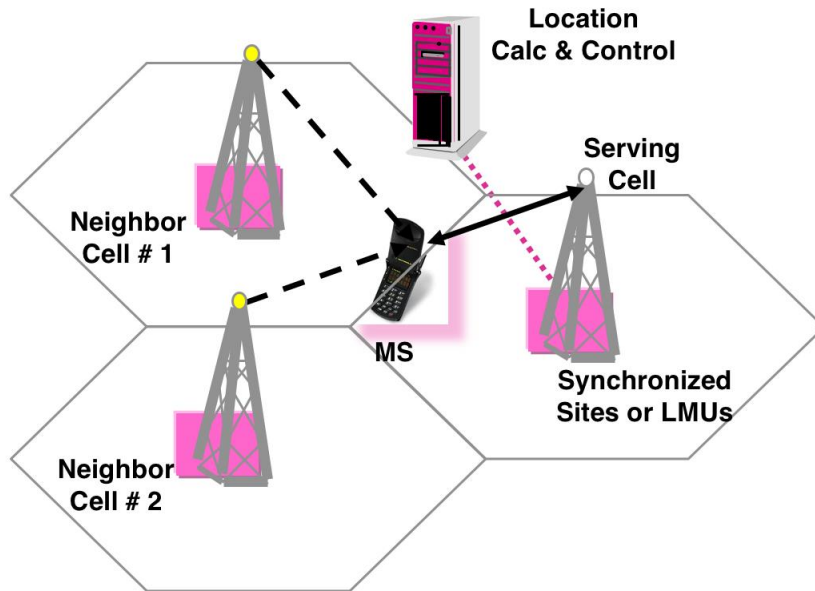


- Estimasi posisi dihitung berdasarkan perbedaan waktu dari minimal tiga BTS yang terdekat dengan menggunakan teknik posisi hiperbola.

# Location Based Service

# Pelatihan

## Enhanced Observed Time Difference (E-OTD)



- TDOA solution in handset
- Arrival time of transmissions from a minimum of 3 sites
- Location solution based on apparent arrival time differences between pairs of sites

# Global Positioning System (GPS - AGPS)

## Location Based Service

## Pelatihan

### GPS (Global Positioning System)

- Sistem yang berguna untuk menentukan letak di permukaan bumi dengan bantuan penyelarasan (Synchronization) dengan sinyal satelit
- Menggunakan 24 satelit yang mengirimkan sinyal mikro ke bumi
- Sinyal mikro ditangkap dan diterima oleh perangkat GPS yang ada di bumi dan data yang ditangkap tersebut diolah untuk mendapatkan informasi lokasi, kecepatan, arah dan waktu.

## Location Based Service

## Pelatihan

### GPS (Global Positioning System)

- Perangkat ponsel yang memiliki fitur GPS harus memiliki minimal 3 perangkat satelit yang tepat berada di atas lokasi Anda berdiri
- 3 perangkat tersebut akan membuat lokasi Anda menjadi semakin tepat sehingga tidak ada miscalculasi lokasi dari sinyal yang diberikan oleh satelit tersebut
- Setelah ponsel Anda menangkap sinyal GPS yang diberikan oleh 3 satelit tersebut, perangkat lunak yang ada di aplikasi GPS tersebut akan mengolah data yang diterima dan kemudian ditampilkan dalam bentuk yang dimengerti oleh manusia seperti dalam bentuk koordinat bumi



## Location Based Service

## Pelatihan

### A-GPS (Assisted Global Positioning System)

- Kekurangan dari fitur GPS yang disematkan di smartphone adalah Chip atau perangkat yang bekerja menangkap sinyal GPS dari satelit tidak terlalu peka, sehingga untuk menangkap sinyal informasi yang dikirimkan oleh satelit membutuhkan waktu yang agak sedikit lama.
- Di dalam fitur A-GPS, operator yang Anda gunakan membantu menambahkan informasi yang kurang sehingga waktu untuk menentukan lokasi Anda berada menjadi lebih cepat
- Menara BTS yang mengeluarkan sinyal memiliki fitur GPS juga yang dapat menentukan lokasi Anda dengan tepat

# Get The Last Know Location

# Setup Google Play Service

# Pelatihan

## Add Google Play Services ke Project

Untuk membuat API **Google Play Services** tersedia langkah-langkahnya sebagai berikut :

1. Buka **build.gradle**, terdapat di application module directory
1. Tambahkan rule baru di dalam dependencies versi terbaru dari play-services. ([Lihat API Description](#))
1. Pastikan top-level di build.grandle, menggunakan reference to the google() repo or to maven { url "https://maven.google.com" }.
1. Simpan perubahan, dan click **Sync Project with Gradle Files**

# Setup Google Play Service

## Pelatihan

### API Description di build.gradle

API	Description in build.gradle
Google+	<code>com.google.android.gms:play-services-plus:16.0.0</code>
Google Account Login	<code>com.google.android.gms:play-services-auth:16.0.1</code>
Google Actions, Base Client Library	<code>com.google.android.gms:play-services-base:16.1.0</code>
Google Sign In	<code>com.google.android.gms:play-services-identity:16.0.0</code>
Google Analytics	<code>com.google.android.gms:play-services-analytics:16.0.8</code>
Google Awareness	<code>com.google.android.gms:play-services-awareness:16.0.0</code>
Google Cast	<code>com.google.android.gms:play-services-cast:16.1.2</code>
Google Cloud Messaging	<code>com.google.android.gms:play-services-gcm:16.1.0</code>
Google Drive	<code>com.google.android.gms:play-services-drive:16.1.0</code>
Google Fit	<code>com.google.android.gms:play-services-fitness:16.0.1</code>
Google Location and Activity Recognition	<code>com.google.android.gms:play-services-location:16.0.0</code>
Google Mobile Ads	<code>com.google.android.gms:play-services-ads:17.2.0</code>
Mobile Vision	<code>com.google.android.gms:play-services-vision:17.0.2</code>

# Setup Google Play Service

## Pelatihan

### Contoh Implementasi penggunaan API Description Dependencies

```
apply plugin: 'com.android.application'  
...  
  
dependencies {  
    implementation 'com.google...'  
}
```

## Specify app permissions

## Pelatihan

Aplikasi yang menggunakan location service harus request location permissions.

Android menyediakan 2 location permissions :

1. [ACCESS\\_COARSE\\_LOCATION](#)

Lebih akurat dengan akurasi kira-kira setara dengan blok kota.

2. [ACCESS\\_FINE\\_LOCATION](#)

Lebih cepat tetapi kurang akurat

## Specify app permissions

## Pelatihan

Cara Penggunaannya request permission dengan menambahkan code berikut pada app manifest.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.google.android.gms.location.sample.basiclocationsample" >

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

## Get The Last Known Location

## Pelatihan

- Dengan menggunakan Google Play Service Location API, aplikasi diperbolehkan request lokasi terakhir pengguna (last known location) dari Device (Handphone) pengguna.
- Menggunakan **fused location provider** untuk mendapatkan device's last known location.
- The fused location provider adalah salah satu dari location API di Google Play Service, dengan akurasi yang tinggi
- serta dapat mengoptimalkan penggunaan daya baterai



## Get The Last Known Location

## Pelatihan

### Create location services client

Dalam activity `onCreate()` method, membuat instance dari Fused Location Provider Client, seperti code berikut :

```
private FusedLocationProviderClient fusedLocationClient;

// ..

@Override
protected void onCreate(Bundle savedInstanceState) {
    // ...

    fusedLocationClient = LocationServices.getFusedLocationProviderClient(this);
}
```

# Get The Last Known Location

## Pelatihan

### Get the last known location

Untuk me-request lokasi terakhir pengguna, dengan memanggil `getLastLocation()` method. Berikut ini adalah contoh code untuk merequest dan contoh handling response.

```
fusedLocationClient.getLastLocation()
    .addOnSuccessListener(this, new OnSuccessListener<Location>() {
        @Override
        public void onSuccess(Location location) {
            // Got last known location. In some rare situations this can be null.
            if (location != null) {
                // Logic to handle location object
            }
        }
    });
```

## Get The Last Known Location

## Pelatihan

### Get the last known location

Metode `getLastLocation()` return location object dengan koordinat latitude dan longitude dari lokasi geografi. Location object return null dalam beberapa kondisi berikut :

- Akses Location dimatikan pada device settings
- Device tidak pernah mencatat lokasi -> device baru / device setelah di factory reset
- Google Play Service pada device melakukan restrart sehingga tidak terdapat Fused Location Provider Client.

# Change Location Setting

## Change location settings

## Pelatihan

- Diperlu **me-request** lokasi atau menerima pembaruan lokasi, perangkat harus mengaktifkan pengaturan sistem yang sesuai, seperti pemindaian GPS atau Wi-Fi.
- Daripada mengaktifkan langsung layanan seperti GPS perangkat, aplikasi Anda menentukan tingkat akurasi / konsumsi daya yang diperlukan dan interval pembaruan yang diinginkan,
- perangkat secara otomatis membuat perubahan yang sesuai dengan pengaturan sistem. Pengaturan ini ditentukan oleh objek data `LocationRequest`.

## Setup Location Request

## Pelatihan

### Location Request

Untuk menyimpan parameter ketika melakukan request ke Fused Location Provider, dengan create Location Request.

Menggunakan parameter untuk menentukan tingkat akurasi untuk permintaan lokasi.

int	<code>PRIORITY_BALANCED_POWER_ACCURACY</code>	Used with <code>setPriority(int)</code> to request "block" level accuracy.
int	<code>PRIORITY_HIGH_ACCURACY</code>	Used with <code>setPriority(int)</code> to request the most accurate locations available.
int	<code>PRIORITY_LOW_POWER</code>	Used with <code>setPriority(int)</code> to request "city" level accuracy.
int	<code>PRIORITY_NO_POWER</code>	Used with <code>setPriority(int)</code> to request the best accuracy possible with zero additional power consumption.

## Setup Location Request

## Pelatihan

### Update Interval - setInterval()

Untuk menentukan kecepatan dalam satuan milliseconds untuk menerima perubahan lokasi pada aplikasi.

## Setup Location Request

## Pelatihan

### Fastest update interval - `setFastestInterval()`

Metode ini menetapkan tingkat tercepat dalam milliseconds di mana aplikasi dapat menangani pembaruan lokasi.

Hal ini dilakukan karena Jika kecepatan ini lebih cepat daripada yang bisa ditangani aplikasi, Kemungkinan mengalami masalah dengan UI flicker atau data overflow. Untuk mencegah hal ini menggunakan `setFastestInterval()`.



## Setup Location Request

## Pelatihan

### Priority - `setPriority()`

Metode ini untuk menetapkan prioritas permintaan, yang memberikan layanan Google Play Location memberikan petunjuk tentang sumber lokasi mana yang akan digunakan.

Nilai yang tersedia :

- `PRIORITY_BALANCED_POWER_ACCURACY`
- `PRIORITY_HIGH_ACCURACY`
- `PRIORITY_LOW_POWER`
- `PRIORITY_NO_POWER`

## Contoh Source Code

```
protected void createLocationRequest() {  
    LocationRequest locationRequest = LocationRequest.create();  
    locationRequest.setInterval(10000);  
    locationRequest.setFastestInterval(5000);  
    locationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);  
}
```

## Get Current Location Setting

## Pelatihan

- Untuk mendapatkan current location setting pada device pengguna

```
LocationSettingsRequest.Builder builder = new LocationSettingsRequest.Builder()  
    .addLocationRequest(locationRequest);
```

- Periksa pengaturan lokasi, apakah sudah sesuai

```
LocationSettingsRequest.Builder builder = new LocationSettingsRequest.Builder();  
  
// ...  
  
SettingsClient client = LocationServices.getSettingsClient(this);  
Task<LocationSettingsResponse> task = client.checkLocationSettings(builder.build());
```

## Prompt user untuk mengubah location settings

## Pelatihan

- Hal ini diperlukan untuk memberikan anjuran user telah mengatur *location settings* dengan benar.
- `OnSuccessListener` untuk validasi *locations setting* yang telah sesuai.
- `OnFailureListener` adalah *event* ketika *location settings* yang belum sesuai.

# Prompt user untuk mengubah location settings

## Pelatihan

### Contoh Source Code

```
task.addOnSuccessListener(this, new OnSuccessListener<LocationSettingsResponse>() {  
    @Override  
    public void onSuccess(LocationSettingsResponse locationSettingsResponse) {  
        // All location settings are satisfied. The client can initialize  
        // location requests here.  
        // ...  
    }  
});  
  
task.addOnFailureListener(this, new OnFailureListener() {  
    @Override  
    public void onFailure(@NonNull Exception e) {  
        if (e instanceof ResolvableApiException) {  
            // Location settings are not satisfied, but this can be fixed  
            // by showing the user a dialog.  
            try {  
                // Show the dialog by calling startResolutionForResult(),  
                // and check the result in onActivityResult().  
                ResolvableApiException resolvable = (ResolvableApiException) e;  
                resolvable.startResolutionForResult(MainActivity.this,  
                    REQUEST_CHECK_SETTINGS);  
            } catch (IntentSender.SendIntentException sendEx) {  
                // Ignore the error.  
            }  
        }  
    }  
});
```

# Receive Location Update

## Receive location updates

## Pelatihan

- Digunakan untuk mendapatkan perubahan lokasi device dari pengguna
- *Return* berupa lokasi koordinat dalam latitude dan longitude
- `getLastLocation()` untuk **mendapatkan** lokasi terakhir
- metode ini untuk **me-request** pembaharuan lokasi agar sesuai dengan lokasi terbaru.
- Keakuratan bergantung provider yang tersedia

## Request location updates

## Pelatihan

- Sebelum melakukan *request*, aplikasi harus terkoneksi dengan location service terlebih dahulu
- Call `requestLocationUpdates()` untuk request lokasi terbaru.
- Contoh method

```
private void startLocationUpdates() {  
    fusedLocationClient.requestLocationUpdates(locationRequest,  
        locationCallback,  
        null /* Looper */);  
}
```



## Request location updates

## Pelatihan

- Contoh penggunaan request location update

```
@Override
protected void onResume() {
    super.onResume();
    if (requestingLocationUpdates) {
        startLocationUpdates();
    }
}

private void startLocationUpdates() {
    fusedLocationClient.requestLocationUpdates(locationRequest,
        locationCallback,
        null /* Looper */);
}
```

## Define the location update callback

## Pelatihan

- Digunakan untuk melakukan aksi ketika mendapatkan perubahan lokasi terbaru
- Contoh implementasi Source code

```
private LocationCallback locationCallback;  
  
// ...  
  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    // ...  
  
    locationCallback = new LocationCallback() {  
        @Override  
        public void onLocationResult(LocationResult locationResult) {  
            if (locationResult == null) {  
                return;  
            }  
            for (Location location : locationResult.getLocations()) {  
                // Update UI with location data  
                // ...  
            }  
        }  
    };  
};  
}
```

## Stop Location Update

## Pelatihan

- Digunakan ketika tidak lagi fokus pada activity yang menggunakan location service
- Untuk meminimalkan penggunaan daya baterai

```
private void stopLocationUpdates() {  
    fusedLocationClient.removeLocationUpdates(locationCallback);  
}
```

# Stop Location Update

## Pelatihan

Contoh implementasi penggunaan

```
@Override
protected void onPause() {
    super.onPause();
    stopLocationUpdates();
}

private void stopLocationUpdates() {
    fusedLocationClient.removeLocationUpdates(locationCallback);
}
```

# Display a Location Address

## Display a Location Address

## Pelatihan

- Lokasi koordinat didapatkan dari Get Last Know Location dan Receive Location update.
- Tapi dibeberapa kasus, menampilkan alamat lokasi device lebih bermanfaat daripada lokasi koordinat.
- Contoh : Untuk mengetahui nama jalan sekitar dari device pengguna.

# Add Maps

## Choose your project

Phone and Tablet

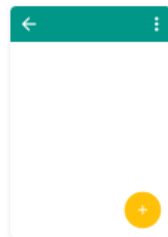
Wear OS

TV

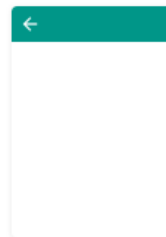
Android Auto

Android Things

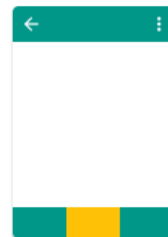
Add No Activity



Basic Activity



Empty Activity



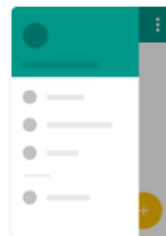
Bottom Navigation Activity



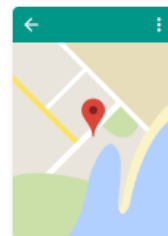
Fullscreen Activity



Master/Detail Flow



Navigation Drawer Activity



Google Maps Activity

### Google Maps Activity

Creates a new activity with a Google Map

Previous

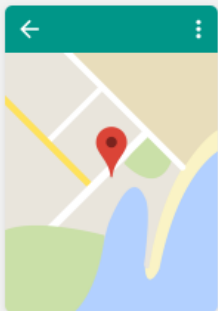
Next

Cancel

Finish



## Configure your project



Google Maps Activity

Creates a new activity with a Google Map

Name

Aplikasi Peta

Package name

com.example.aplikasipeta

Save location

C:\Users\LPA\AndroidStudioProjects\AplikasiPeta

Language

Java

Minimum API level API 15: Android 4.0.3 (IceCreamSandwich)



Your app will run on approximately **100%** of devices.

[Help me choose](#)



This project will support instant apps



Use androidx.\* artifacts

Previous

Next

Cancel

Finish

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

AplikasiPeta > app > src > debug > res > values > google\_maps\_api.xml

app

1: Project

Android

app

manifests

java

generatedJava

res

drawable

layout

mipmap

values

colors.xml

google\_maps\_api.xml (debug)

strings.xml

styles.xml

Gradle Scripts

Build Variants

Structure

```
1 <resources>
```

```
2 <!--
```

```
3  TODO: Before you run your application, you need a Google Maps API key.
```

```
4
```

```
5  To get one, follow this link, follow the directions and press "Create" at the end:
```

```
6
```

```
7  https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID
```

```
8
```

```
9  You can also add your credentials to an existing key, using these values:
```

```
10
```

```
11  Package name:
```

```
12  62:09:A2:0C:7B:E1:F8:C4:4B:F5:D5:E1:C8:26:EB:30:48:52:15:7B
```

```
13
```

```
14  SHA-1 certificate fingerprint:
```

```
15  62:09:A2:0C:7B:E1:F8:C4:4B:F5:D5:E1:C8:26:EB:30:48:52:15:7B
```

```
16
```

```
17  Alternatively, follow the directions here:
```

```
18  https://developers.google.com/maps/documentation/android/start#get-key
```

```
19
```

```
20  Once you have your key (it starts with "AIza"), replace the "google_maps_key"
```

resources



Your free trial is waiting: activate now to get \$300 credit to explore Google Cloud products. [Learn more](#)



Google APIs Select a project ▼



## Register your application for Maps SDK for Android in Google API Console

Google API Console allows you to manage your application and monitor API usage.

You have no existing projects. A new project named "My Project" will be created.

[Continue](#)



Your free trial is waiting: activate now to get \$300 credit to explore Goc



Google APIs

Select a project ▼

## The API is enabled

The project has been created and Maps SDK for Android has been enabled.

Next, you'll need to create an API key in order to call the API.

Create API key

# API key created

Use this key in your application by passing it with the `key=API_KEY` parameter.

Your API key

AIzaSyAGwWMALHsuc4bZ6aUgWKYgW3b0peSq\_GE



 Restrict your key to prevent unauthorized use in production.

[CLOSE](#)

[RESTRICT KEY](#)

```
Activity.java x google_maps_api.xml x  
package name:  
62:09:A2:0C:7B:E1:F8:C4:4B:F5:D5:E1:C8:26:EB:30:48:52:15:7B  
  
SHA-1 certificate fingerprint:  
62:09:A2:0C:7B:E1:F8:C4:4B:F5:D5:E1:C8:26:EB:30:48:52:15:7B  
  
Alternatively, follow the directions here:  
https://developers.google.com/maps/documentation/android/start#get-key  
  
Once you have your key (it starts with "AIza"), replace the "google_maps_key"  
string in this file.  
-->  
<string name="google_maps_key" templateMergeStrategy="preserve" translatable="false">YOUR_KEY_HERE</string>  
</resources>
```

Activity.java x

google\_maps\_api.xml x

```
:F8:C4:4B:F5:D5:E1:C8:26:EB:30:48:52:15:7B
```

```
fingerprint:
```

```
:F8:C4:4B:F5:D5:E1:C8:26:EB:30:48:52:15:7B
```

```
Follow the directions here:
```

```
s.google.com/maps/documentation/android/start#get-key
```

```
For key (it starts with "AIza"), replace the "google_maps_key"  
value.
```

```
google_maps_key" templateMergeStrategy="preserve" translatable="false">AIzaSyAGwWMALHsuc4bZ6aUgWKYgW3b0peSq_GE</s
```



```
31      * This callback is triggered when the map is ready to be used.
32      * This is where we can add markers or lines, add listeners or move the camera. In t
33      * we just add a marker near Sydney, Australia.
34      * If Google Play services is not installed on the device, the user will be prompted
35      * it inside the SupportMapFragment. This method will only be triggered once the use
36      * installed Google Play services and returned to the app.
37      */
38      @Override
39      public void onMapReady(GoogleMap googleMap) {
40          mMap = googleMap;
41
42          // Add a marker in Sydney and move the camera
43          LatLng sydney = new LatLng(V, -34, V1: 151);
44          mMap.addMarker(new MarkerOptions().position(sydney).title("Marker in Sydney"));
45          mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
46      }
47  }
48
```



```
MapsActivity.java x google_maps_api.xml x
32  * This is where we can add markers or lines, add listeners or move the camera. In this
33  * we just add a marker near Sydney, Australia.
34  * If Google Play services is not installed on the device, the user will be prompted to
35  * it inside the SupportMapFragment. This method will only be triggered once the user has
36  * installed Google Play services and returned to the app.
37  */
38  @Override
39  public void onMapReady(GoogleMap googleMap) {
40      mMap = googleMap;
41
42      // Add a marker in Sydney and move the camera
43      LatLng sydney = new LatLng(-34, 151);
44      mMap.addMarker(new MarkerOptions().position(sydney).title("Marker in Sydney"));
45      mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
46  }
47  }
```

C MapsActivity.java x

google\_maps\_api.xml x

```
31      * This callback is triggered when the map is ready to
32      * This is where we can add markers or lines, add lis
33      * we just add a marker near Sydney, Australia.
34      * If Google Play services is not installed on the de
35      * it inside the SupportMapFragment. This method will
36      * installed Google Play services and returned to the
37      */
38      @Override
39      public void onMapReady(GoogleMap googleMap) {
40          mMap = googleMap;
41
42
43      }
44  }
45
```

30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45

```
* Manipulates the map once available.  
* This callback is triggered when the map is ready to be used.  
* This is where we can add markers or lines, add listeners or mo  
* we just add a marker near Sydney, Australia.  
* If Google Play services is not installed on the device, the us  
* it inside the SupportMapFragment. This method will only be tri  
* installed Google Play services and returned to the app.  
*/
```

```
@Override
```

```
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    mMap.setMyLocationEnabled(true);  
}
```

! Add permission check

! Provide feedback on this warning

X Suppress: Add @SuppressWarnings("MissingPermission") annotation

✎ Add method contract to 'setMyLocationEnabled'

✎ Annotate method 'setMyLocationEnabled'...

```
*/  
  
@Override  
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    if (checkSelfPermission(Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED &&  
        // TODO: Consider calling  
        //     Activity#requestPermissions  
        // here to request the missing permissions, and then overriding  
        //   public void onRequestPermissionsResult(int requestCode, String[] permissions,  
        //                                           int[] grantResults)  
        // to handle the case where the user grants the permission. See the documentation  
        // for Activity#requestPermissions for more details.  
        return;  
    }  
    mMap.setMyLocationEnabled(true);  
}
```

@Override

```
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    if (ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_FINE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED  
        && ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_COARSE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED) {  
        // TODO: Consider calling  
        //     Activity#requestPermissions  
        // here to request the missing permissions, and then overriding  
        //     public void onRequestPermissionsResult(int requestCode, String[] permissions,  
        //                                             int[] grantResults)  
        // to handle the case where the user grants the permission. See the documentation  
        // for Activity#requestPermissions for more details.  
        return;  
    }  
    mMap.setMyLocationEnabled(true);  
}
```

@Override

```
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    if (ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_FINE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED  
        && ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_COARSE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED) {  
        // TODO: Consider calling  
        //     Activity#requestPermissions  
        // here to request the missing permissions, and then overriding  
        //     public void onRequestPermissionsResult(int requestCode, String[] permissions,  
        //                                             int[] grantResults)  
        // to handle the case where the user grants the permission. See the documentation  
        // for Activity#requestPermissions for more details.  
        return;  
    }  
    mMap.setMyLocationEnabled(true);  
}
```

```
@Override
```

```
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    if (ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_FINE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED  
        && ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_COARSE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED) {  
  
        return;  
    }  
    mMap.setMyLocationEnabled(true);  
}
```

MapsActivity.java x

google\_maps\_api.xml x

```
1 package com.example.aplikasipeta;
2
3 import ...
15
16 public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
17
18     private GoogleMap mMap;
19     private static final int LOCATION_REQUEST = 500;
20
```



@Override

```
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    if (ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_FINE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED  
        && ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_COARSE_LOCATION)  
        != PackageManager.PERMISSION_GRANTED) {  
        ActivityCompat.requestPermissions(activity: this, new String[]{Manifest.permission.ACCESS_FINE_LOCATION},  
            LOCATION_REQUEST);  
    }  
    return;  
    mMap.setMyLocationEnabled(true);  
}
```

## @Override

```
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
    mMap.getUiSettings().setZoomControlsEnabled(true);  
    if (ActivityCompat.checkSelfPermission(context, this,  
        != PackageManager.PERMISSION_GRANTED  
        && ActivityCompat.checkSelfPermission(context,  
        != PackageManager.PERMISSION_GRANTED) {  
        ActivityCompat.requestPermissions(activity, this, n  
            LOCATION_REQUEST);  
    }  
    return;  
}  
mMap.setMyLocationEnabled(true);  
}
```

## onReq

```
m ○↑ public void onRequestPermissionsResult Frag..  
m ○↑ public boolean onSearchRequested()... Activity  
m ○↑ public boolean onSearchRequested(S... Activity  
J public int getLocationRequest() {...}
```

iva x google\_maps\_api.xml x

```
        != PackageManager.PERMISSION_GRANTED
        && ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_COARSE_LOCATION)
        != PackageManager.PERMISSION_GRANTED) {
    ActivityCompat.requestPermissions(activity: this, new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
        LOCATION_REQUEST);
    return;
}
mMap.setMyLocationEnabled(true);
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
}
```

```
        != PackageManager.PERMISSION_GRANTED
        && ActivityCompat.checkSelfPermission(context: this, Manifest.permission.ACCESS_COARSE_LOCATION)
        != PackageManager.PERMISSION_GRANTED) {
    ActivityCompat.requestPermissions(activity: this, new String[] {Manifest.permission.ACCESS_FINE_LOCATION},
        LOCATION_REQUEST);
    return;
}
mMap.setMyLocationEnabled(true);
}

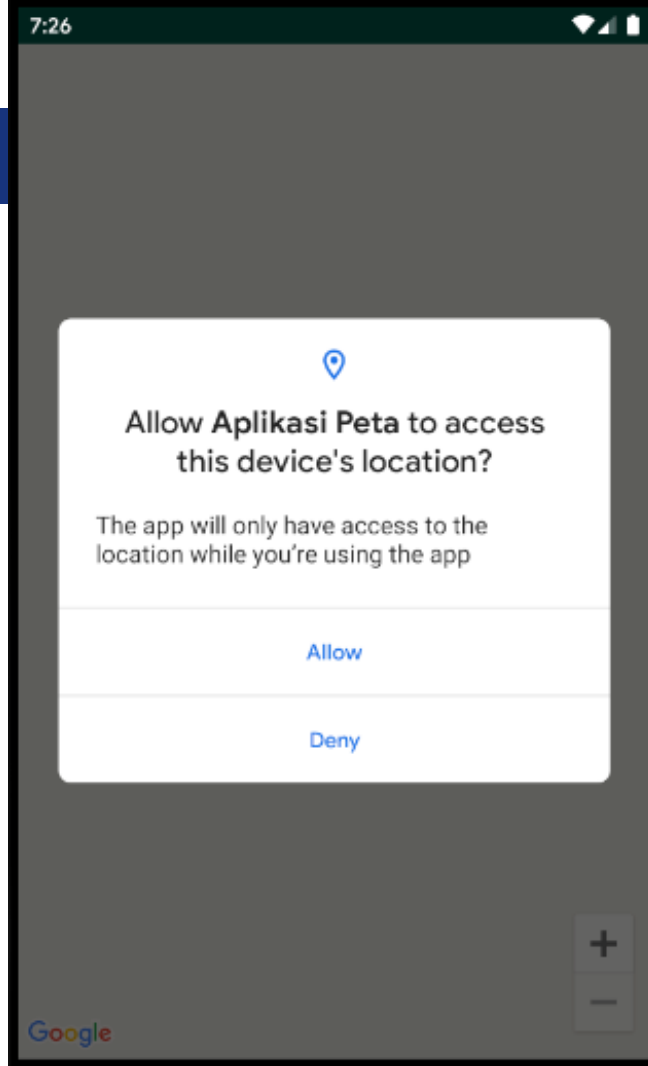
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
}
}
```

```
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    switch (requestCode) {
        case LOCATION_REQUEST:
            if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                mMap.setMyLocationEnabled(true);
            }
        }
    }
}
```

- ! Add permission check
- ! Provide feedback on this warning
- X Suppress: Add @SuppressWarnings("MissingPermission") annotation
- Remove braces from 'if' statement

```
@SuppressWarnings("MissingPermission")
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    switch (requestCode) {
        case LOCATION_REQUEST:
            if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                mMap.setMyLocationEnabled(true);
            }
        }
    }
}
```

```
@SuppressWarnings("MissingPermission")
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    switch (requestCode) {
        case LOCATION_REQUEST:
            if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                mMap.setMyLocationEnabled(true);
            }
            break;
    }
}
```





```
public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {  
  
    private GoogleMap mMap;  
    private static final int LOCATION_REQUEST = 500;  
    ArrayList<LatLng> listPoints;
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_maps);  
    // Obtain the SupportMapFragment and get notified when the map is ready to be used.  
    SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()  
        .findFragmentById(R.id.map);  
    mapFragment.getMapAsync(new OnMapReadyCallback() {  
        @Override  
        public void onMapReady() {  
            listPoints = new ArrayList<>();  
        }  
    });  
}
```

```
mMap.setMyLocationEnabled(true);  
mMap.setOnMapLongClickListener(new GoogleMap.OnMapLongClickListener()  
{  
    @Override  
    public void onMapLongClick(LatLng latLng) {  
        //Reset marker kalau sudah 2  
        if (listPoints.size() == 2){  
            listPoints.clear();  
            mMap.clear();  
        }  
        //Simpan titik pertama yang dipilih  
        listPoints.add(latLng);  
        //buat marker  
        MarkerOptions markerOptions = new MarkerOptions();  
        markerOptions.position(latLng);  
        if (listPoints.size() == 1){  
            //Tambahkan marker pertama pada map  
            markerOptions.icon(BitmapDescriptorFactory.defaultMarker(  
                BitmapDescriptorFactory.HUE_GREEN));  
        }  
    }  
});
```

```
if (listPoints.size() == 1){  
    //Tambahkan marker pertama pada map  
    markerOptions.icon(BitmapDescriptorFactory.defaultMarker(  
        BitmapDescriptorFactory.HUE_GREEN));  
} else{  
    //Tambahkan marker kedua pada map  
    markerOptions.icon(BitmapDescriptorFactory.defaultMarker(  
        BitmapDescriptorFactory.HUE_RED));  
}  
mMap.addMarker(markerOptions);
```

```
mMap.addMarker(markerOptions);
```

```
if (listPoints.size() == 2){
```

```
    //Buat URL untuk mendapat request dari marker pertama ke kedua
```

```
    String url = getUrl(listPoints.get(0), listPoints.get(1));
```

```
}
```

❗ Create method 'getUrl'

❗ Rename reference

☰ Split into declaration and assignment ▶

```
mMap.addMarker(markerOptions);
```

```
if (listPoints.size() == 2){
```

```
    //Buat URL untuk mendapat request dari marker pertama ke kedua
```

```
    String url = getUrl(listPoints.get(0), listPoints.get(1));
```

```
}
```

Choose Target Class

Anonymous in onMapReady() in MapsActivity (com.example.aplikasipeta) app

MapsActivity (com.example.aplikasipeta) app

```
private String getUrl(LatLng latLng, LatLng latLng1) {
```

```
}
```

```
private String getRequestId(LatLng asal, LatLng tujuan) {  
}
```



```
private String getRequestUrl(LatLng asal, LatLng tujuan) {  
    //nilai asal  
    String str_asal = "asal=" + asal.latitude + "," + asal.longitude;  
    //nilai tujuan  
    String str_tujuan = "tujuan=" + tujuan.latitude + "," + tujuan.longitude;  
    //set nilai sensor  
    String sensor = "sensor=false";  
    //mode untuk menemukan arah  
    String mode = "mode=driving";  
    //membentuk parameter  
    String param = str_asal + "&" + str_tujuan + "&" + sensor + "&" + mode;  
    //output format  
    String output = "json";  
    //buat url untuk request  
    String url = "https://maps.googleapis.com/maps/api/directions/" + output  
        + "?" + param;  
    return url;  
}
```

```
private String requestDirection(String reqUrl){
    String responseString = "";
    InputStream inputStream = null;
    HttpURLConnection httpURLConnection = null;
    try{
        URL url = new URL(reqUrl);
        httpURLConnection = (HttpURLConnection) url.openConnection();
        httpURLConnection.connect();

        //dapatkan hasil response
        inputStream = httpURLConnection.getInputStream();
        InputStreamReader inputStreamReader = new InputStreamReader(inputStream);
        BufferedReader bufferedReader = new BufferedReader(inputStreamReader);

        StringBuffer stringBuffer = new StringBuffer();
        String line = "";
        while ((line = bufferedReader.readLine()) != null ){
            stringBuffer.append(line);
        }
    }
```

```
while ((line = bufferedReader.readLine()) != null ) {  
    stringBuffer.append(line);  
}  
  
    responseString = stringBuffer.toString();  
    bufferedReader.close();  
    inputStreamReader.close();  
} catch (Exception e) {  
    e.printStackTrace();  
} finally {  
    if (inputStream != null) {  
        try {  
            inputStream.close();  
        } catch (IOException e) {  
            e.printStackTrace();  
        }  
    }  
    httpURLConnection.disconnect();  
}  
return responseString;  
}
```

```
public class TaskRequestDirections extends AsyncTask<String, Void, String>{  
  
}  
}
```

Implement methods

Make 'TaskRequestDirections' abstract

Create subclass

Flip ','



android.os.AsyncTask



doInBackground(params:Params...):Result



Copy Javadoc



Insert @Override

OK

Cancel

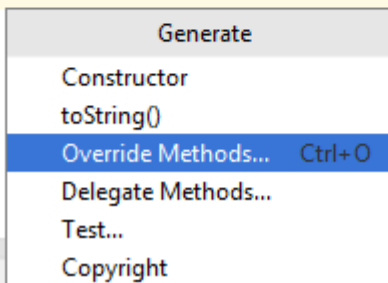
```
public class TaskRequestDirections extends AsyncTask<String, Void, String>{  
  
    @Override  
    protected String doInBackground(String... strings) {  
        return null;  
    }  
}
```

```
public class TaskRequestDirections extends AsyncTask<String, Void, String>{  
  
    @Override  
    protected String doInBackground(String... strings) {  
  
    }  
}
```

```
public class TaskRequestDirections extends AsyncTask<String, Void, String>{  
  
    @Override  
    protected String doInBackground(String... strings) {  
        String responseString = "";  
        responseString = requestDirection(strings[0]);  
        return responseString;  
    }  
}
```



```
public class TaskRequestDirections extends AsyncTask<String, Void, String>{  
  
    @Override  
    protected String doInBackground(String... strings) {  
        String responseString = "";  
        responseString = requestDirection(strings[0]);  
        return responseString;  
    }  
}
```



Alt + Insert

↓% ↓a C I

- ▼ android.os.AsyncTask
  - m AsyncTask()
  - m onPreExecute():void
  - m onPostExecute(result:Result):void
  - m onProgressUpdate(values:Progress...):void
  - m onCancelled(result:Result):void
  - m onCancelled():void
- ▼ java.lang.Object
  - m hashCode():int
  - m equals(obj:Object):boolean
  - m clone():Object
  - m toString():String
  - m finalize():void

☐ Copy Javadoc  
☒ Insert @Override

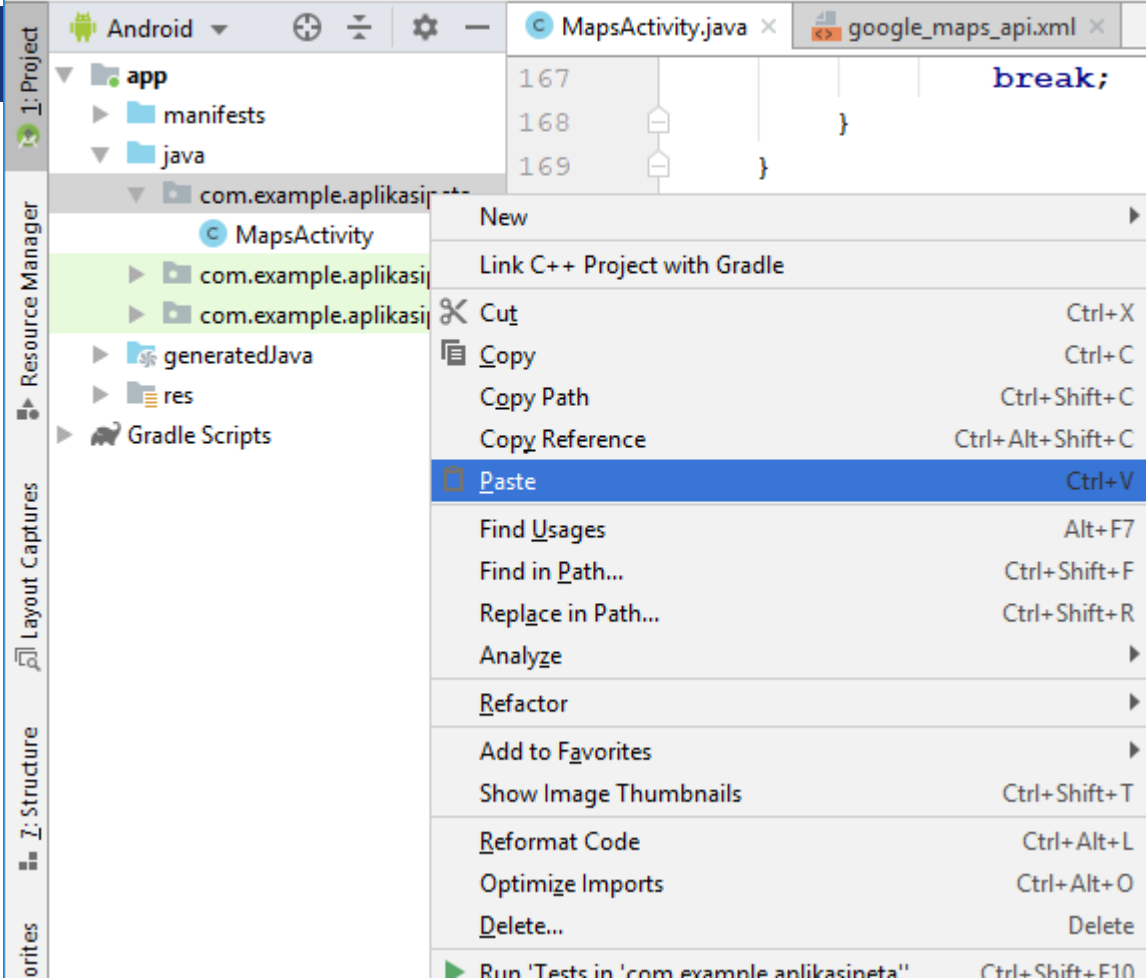
OK Cancel

```
@Override
```

```
protected void onPostExecute(String s) {
```

```
    super.onPostExecute(s);
```

```
}
```



Copy dan paste file  
DirectionParser.java

 Copy



Copy file C:\Users\LPA\Nextc...\ngoctri\mapdirectionsample\DirectionsParser.java

New name:

To directory:  ▼ ...

Use Ctrl+Space for path completion

☒ Open copy in editor

OK

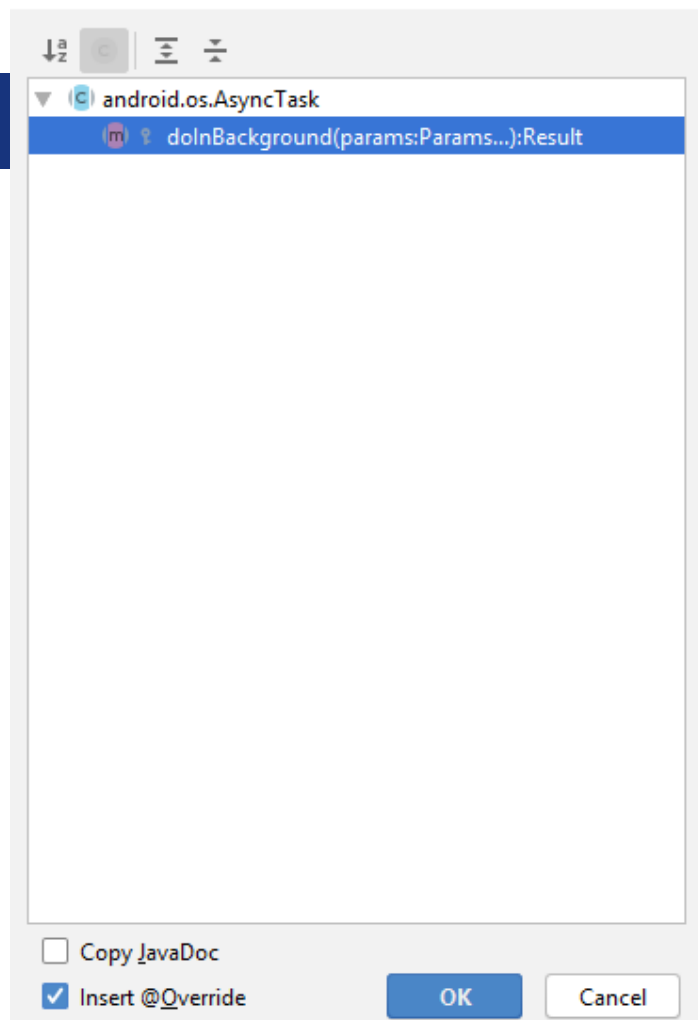
Cancel

Help

```
! public class TaskParser extends AsyncTask<String, Void, List<List<HashMap<String, String>>>> >{  
  
}
```

MapsActivity > TaskParser

- ! Implement methods
- ! Make 'TaskParser' abstract
- 🔧 Create subclass ▶
- 🔧 Unimplement Class ▶



```
public class TaskParser extends AsyncTask<String, Void, List<List<HashMap<String, String>>>> >{  
  
    @Override  
    protected List<List<HashMap<String, String>>> doInBackground(String... strings) {  
        return null;  
    }  
}
```



```
public class TaskParser extends AsyncTask<String, Void, List<List<HashMap<String, String>>>> >{  
  
    @Override  
    protected List<List<HashMap<String, String>>> doInBackground(String... strings) {  
        |  
    }  
}
```

```
public class TaskParser extends AsyncTask<String, Void, List<List<HashMap<String, String>>>> >{

    @Override
    protected List<List<HashMap<String, String>>> doInBackground(String... strings) {
        JSONObject jsonObject = null;
        List<List<HashMap<String, String>>> routes = null;
        try{
            jsonObject = new JSONObject(strings[0]);
            DirectionsParser directionsParser = new DirectionsParser();
            routes = directionsParser.parse(jsonObject);
        } catch (JSONException e){
            e.printStackTrace();
        }
        return routes;
    }
}
```

on

protected void onCancelled() {...} AsyncTask

protected void onCancelled(List r... AsyncTask

protected void onPostExecute(List result) {...} AsyncTask

protected void onPreExecute() {...} AsyncTask

protected void onProgressUpdate( ... AsyncTask

protected Object clone() {...} Object

```
@Override
```

```
protected void onPostExecute(List<List<HashMap<String, String>>> lists) {
```

```
|
```

```
}
```

@Override

```
protected void onPostExecute(List<List<HashMap<String, String>>> lists) {  
    //ambil rute dan tampilkan pada map  
    ArrayList points = null;  
    PolylineOptions polylineOptions = null;  
    for (List<HashMap<String, String>> path: lists){  
        points = new ArrayList();  
        polylineOptions = new PolylineOptions();  
  
        for (HashMap<String, String> point : path){  
            double lat = Double.parseDouble(point.get("lat"));  
            double lon = Double.parseDouble(point.get("lon"));  
            points.add(new LatLng(lat,lon));  
        }  
  
        polylineOptions.addAll(points);  
        polylineOptions.width(15);  
        polylineOptions.color(Color.BLUE);  
        polylineOptions.geodesic(true);  
    }  
    if (polylineOptions != null){  
        mMap.addPolyline(polylineOptions);  
    } else{  
        Toast.makeText(getApplicationContext(), text: "Arah tidak ketemu!", Toast.LENGTH_SHORT).show();  
    }  
}
```

```
public class TaskRequestDirections extends AsyncTask<String, Void, String>{

    @Override
    protected String doInBackground(String... strings) {
        String responseString = "";
        responseString = requestDirection(strings[0]);
        return responseString;
    }

    @Override
    protected void onPostExecute(String s) {
        super.onPostExecute(s);
        TaskParser taskParser = new TaskParser();
        taskParser.execute(s);
    }
}
```

```
if (listPoints.size() == 2){  
    //Buat URL untuk mendapat request dari marker pertama ke kedua  
    String url = getUrl(listPoints.get(0), listPoints.get(1));  
    TaskRequestDirections taskRequestDirections = new TaskRequestDirections();  
    taskRequestDirections.execute(url);  
}
```

# Algorithm Mobile Navigation



# Algoritma Shortest Path Google Maps

## Pelatihan

- Algoritma yang digunakan sangat complex untuk mencari rute terpendek pada google maps.
- Google maps melihat suatu peta sebagai grafik yang terdiri dari nodus menghubungkan lokasi yang berbeda-beda
- Ketika kita ingin pergi dari tempat A ke tempat Z google akan membacanya sebagai titik A ke titik Z, algoritma dijkstra bekerja dengan menghubungkan simpul simpul yang ada di setiap tepi dan mencari jarak terpendek untuk sampai ditempat tujuan

# Algoritma Dijkstra

## Pelatihan

- Algoritma yang dipakai untuk memecahkan permasalahan jarak terpendek untuk graf berarah
- Memeriksa dan membandingkan setiap jalur yang ada
- Menentukan rute paling efisien

# Algoritma Dijkstra

## Pelatihan

Dijkstra juga menggunakan parameter seperti :

- Panjang jalan,
- Jumlah jalur,
- Lampu lalu lintas,
- Kepadatan Jalan
- Data lalu lintas (real time / perkiraan)

Untuk menentukan rute paling efisien.

# Algoritma Dijkstra

## Pelatihan

Perhitungan algoritma djikstra :

$U_i$  jarak terpendek dari titik 1 ke titik i.  $d_{ij}$  ( $\geq 0$ ) panjang dari (i,j).

Label untuk titik j didefinisikan sebagai :

$$[u_{i,j}] = (u_i + d_{ij}, i) , d_{ij} \geq 0$$

# Algoritma Dijkstra

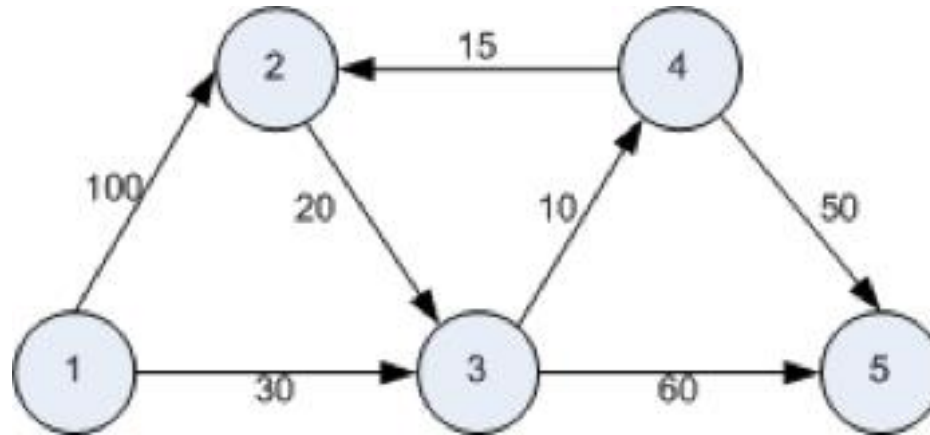
## Pelatihan

Terdapat 2 Label

- Sementara
  - Permanen
1. Label Sementara diganti dengan label lain jika ditemukan rute lain yang lebih pendek.
  2. Jika tak ada rute lain yang lebih baik, status tetap (permanen).

# Algoritma Dijkstra

## Pelatihan



Contoh menghitung jarak terpendek dari 1 ke 2

# Algoritma Dijkstra

## Pelatihan

Pelabelan [ , ]  
Permanen & Sementara

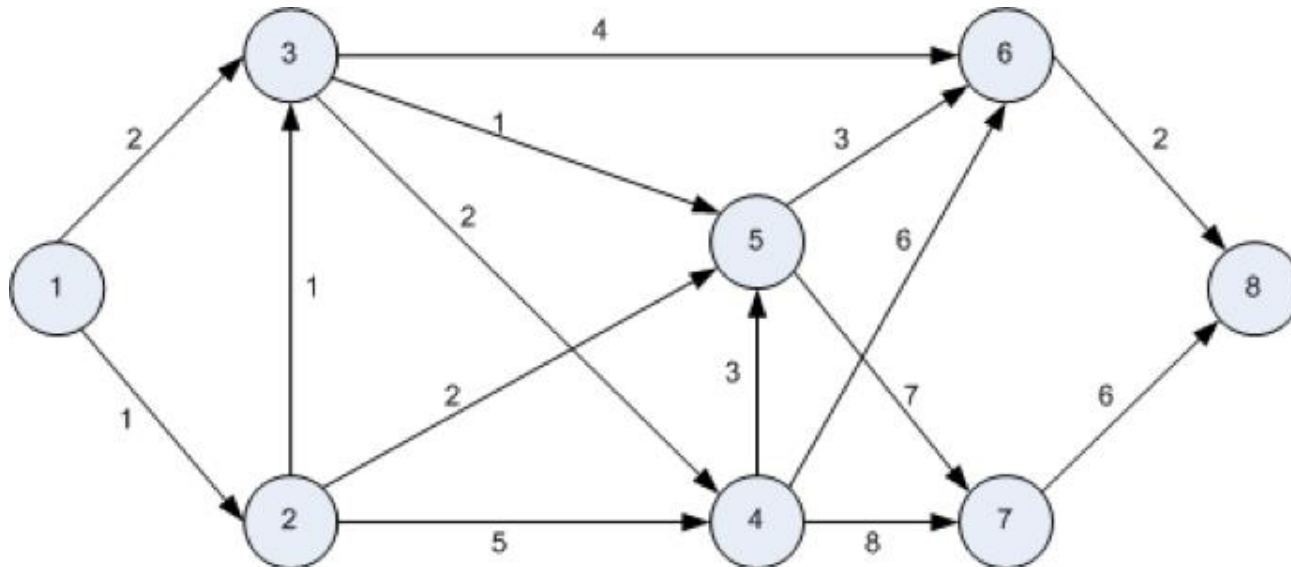
Iterasi	Titik	Label	Status
0	1	[0,-]	Permanen
1	1 2 3	[0,-] [0+100,1] [0+30,1]	Permanen Sementara Sementara
2	1 2 3 4 5	[0,-] [100,1] [30,1] [40,3] [90,3]	Permanen Sementara Permanen Permanen Sementara
3	1 2 3 4 5	[0,-] [55,4] [30,1] [40,3] [90,3]	Permanen Permanen Permanen Permanen Sementara

## Latihan Algoritma Dijkstra

## Pelatihan

Cari jarak terpendek dan rutenya dari :

1 ke 8 & 1 ke 6





## Latihan Algoritma Djikstra

## Pelatihan

Jarak 1 ke 8

Iterasi	Titik	Label	Status
0	1	[0,-]	Permanen
1	2	[1,1]	Permanen
	3	[2,1]	Sementara
2	3	[2,2]	Permanen
	4	[6,2]	Sementara
	5	[3,2]	Sementara
3	4	[4,3]	Sementara
	5	[3,3]	Permanen
	6	[6,3]	Sementara
4	6	[6,5]	Permanen
	7	[10,5]	Sementara
5	8	[8,6]	Permanen

## Latihan Algoritma Djikstra

## Pelatihan

Jarak 1 ke 6

Iterasi	Titik	Label	Status
0	1	[0,-]	Permanen
1	<b>2</b>	[1,1]	Permanen
	3	[2,1]	Sementara
2	3	[2,2]	Permanen
	4	[6,2]	Sementara
	5	[3,2]	Sementara
3	4	[4,3]	Sementara
	5	[3,3]	Permanen
	6	[6,3]	Sementara
4	6	[6,5]	Permanen

# Kesimpulan Pertemuan

1. Location Based Service menentukan lokasi pengguna berdasarkan GSM Network, Cell Tower dan Wi-Fi signals
2. Menentukan lokasi pengguna dengan GPS lebih akurat hasilnya daripada menggunakan Location Based Service tetapi memiliki kekurangan (Lebih boros penggunaan baterai)
3. Google maps API memungkinkan pihak developer untuk menampilkan peta / maps, fitur navigasi, dll ke dalam di aplikasi

# Referensi:

1. <https://developer.android.com>
2. <https://www.quora.com/How-does-the-algorithm-of-Google-Maps-work>

# Tim Penyusun:

- Alif Akbar Fitrawan, S.Pd, M. Kom (Politeknik Negeri Banyuwangi);
- Anwar, S.Si, MCs. (Politeknik Negeri Lhokseumawe);
- Eddo Fajar Nugroho (BPPTIK Cikarang);
- Eddy Tungadi, S.T., M.T. (Politeknik Negeri Ujung Pandang);
- Fitri Wibowo (Politeknik Negeri Pontianak);
- Ghifari Munawar (Politeknik Negeri Bandung);
- Hetty Meileni, S.Kom., M.T. (Politeknik Negeri Sriwijaya) ;
- I Wayan Candra Winetra, S.Kom., M.Kom (Politeknik Negeri Bali) ;
- Irkham Huda (Vokasi UGM) ;
- Josseano Amakora Koli Parera, S.Kom., M.T. (Politeknik Negeri Ambon) ;
- I Komang Sugiarta, S.Kom., MMSI (Universitas Gunadarma) ;
- Lucia Sri Istiyowati, M.Kom (Institut Perbanas) ;
- Maksy Sendiang, ST, MIT (Politeknik Negeri Manado) ;
- Medi Noviana (Universitas Gunadarma) ;
- Muhammad Nashrullah (Politeknik Negeri Batam) ;
- Nat. I Made Wiryana, S.Si., S.Kom., M.Sc. (Universitas Gunadarma) ;
- Rika Idmayanti, ST, M.Kom (Politeknik Negeri Padang) ;
- Rizky Yuniar Hakkun (Politeknik Elektronik Negeri Surabaya) ;
- Robinson A.Wadu, ST., MT (Politeknik Negeri Kupang) ;
- Roslina. M.IT (Politeknik Negeri Medan) ;
- Sukamto, SKom., MT. (Politeknik Negeri Semarang) ;
- Syamsi Dwi Cahya, M.Kom. (Politeknik Negeri Jakarta) ;
- Syamsul Arifin, S.Kom, M.Cs (Politeknik Negeri Jember) ;
- Usmanudin (Universitas Gunadarma) ;
- Wandy Alifha Saputra (Politeknik Negeri Banjarmasin) ;