



군산대학교  
KUNSAN NATIONAL UNIVERSITY

---

# Location과 Map

모바일 응용

남 광 우

---

# Location and MAP

---

## ❖ 위치 획득 방법

### ✓ GPS

- Global Positioning System (GPS)
  - 미국의 위성기반 측위 시스템(러시아는 GLONAS, 유럽은 GALIEO 등등)
  - 2009년 현재 기본적으로 10~25m 정도의 오차를 가짐
  - 내비게이션은 자체 알고리즘으로 이를 보정함
- 2000년 5월 이전에는 미국의 군사적 보안을 이유로, 민간용은 일부러 100m 오차를 가지게 했었음 (Selective Availability)

### ✓ Cell ID : 인근 휴대폰 기지국

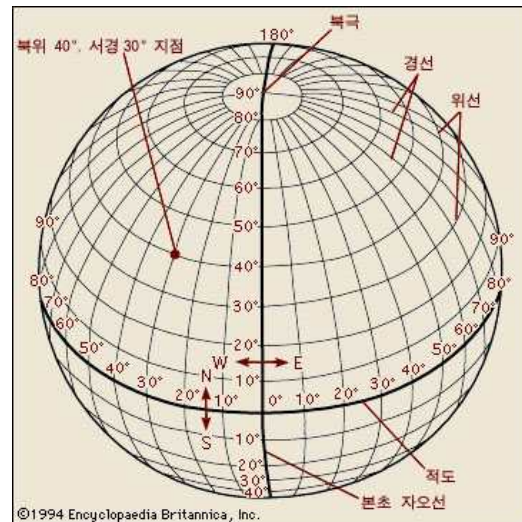
### ✓ WIFI

### ✓ IP Addresses

# Location and MAP

## ❖ Location 표현 방법

- ✓ 위경도 : 위도(Latitude), 경도(longitude)



- ✓ KML : Keyhole Markup Language
  - 구글 맵과 구글 어스의 콘텐츠 언어
- ✓ GPX
  - GPS Tracking 데이터 XML 표현 방법

# Location and MAP

---

## ❖ Location Service 관련 패키지

### ✓ **android.location** Package

- GPS나 무선랜 등의 정보를 이용하여 휴대전화의 현재 위치 정보(위도, 경도)를 얻기 위한 기능을 제공하는 패키지
- 구성 클래스
  - 시스템의 위치 서비스(Location Service)의 접근을 제공하는 LocationManager 클래스
  - 위치정보와 주소정보를 변환하는 Geocoder 클래스
  - GPS엔진 상태를 표현하는 GpsStatus 클래스 등으로 구성

### ✓ **com.google.android.maps** Package

- Google Maps Service에 접근하는 인터페이스를 제공하는 패키지
- 구성 클래스
  - 맵을 표시하는 MapView 클래스
  - MapView를 Activity를 관리하는 MapActivity 클래스 등으로 구성

# Location and MAP

---

## ❖ Location 프로그래밍 절차

- ✓ Manifest.xml에 Location 서비스 Permission 설정
- ✓ LocationManager 획득하기
  - LocationManager locationManager = (LocationManager) getSystemService(Context.LOCATION\_SERVICE);
- ✓ LocationProvider 설정하기
  - 장치의 현재 위치를 결정하는데 사용되는 위치검색 기술을 선택
  - 선택가능한 프로바이더
    - LocationManager.GPS\_PROVIDER
    - LocationManager.NETWORK\_PROVIDER
- ✓ Location 사용하기
  - 현재 위치 가져오기, LocationTracking, Location Triggering

# Location and MAP

---

## ❖ Location 사용 방법

- ✓ 마지막으로 알고 있는 위치에 대하여, LocationManager가 알고 있는 모든 또는 특정 LocationProviders 리스트에게 쿼리를 하는 것
  - `Location location = locationManager.getLastKnownLocation(provider);`
- ✓ (규칙 또는 이름으로 지정되는) 로케이션 프로바이더 LocationProvider에 현재 위치에 대한 주기적인 업데이트를 등록하거나 등록 해제하는 것.
  - LocationListener를 이용하는 방법
- ✓ 만약 디바이스가 주어진 위도/경도의 주어진 근접 범위(미터로 된 반경으로 지정됨)내에 들어오면, 주어진 인텐트가 시작되게 하는 것을 등록하거나 등록 해제하는 것.

# Location and MAP

## ❖ Permission 설정

### ✓ androidmanifest.xml

- <uses-permission  
android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>
  - Network Provider 등의 사용 허용
- <uses-permission  
android:name="android.permission.ACCESS\_FINE\_LOCATION" />
  - GPS 정도의 정확도 사용 허용

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



```
LocationManager m_location_manager = (LocationManager) getSystemService(LOCATION_SERVICE);  
Location lm = m_location_manager.getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
```

# Location and MAP

## ❖ Permission 설정

✓ androidmanifest.xml 의 예

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="xxx.yyy"
    android:versionCode="1"
    android:versionName="1.0">

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

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".hello" android:label="@string/app_name">

            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>

        </activity>
    </application>
    <uses-sdk android:minSdkVersion="3" />
</manifest>
```

# Location and MAP

## ❖ LocationProvider 선택하기

- ✓ Android Default Provider 2가지
  - LocationManager.GPS\_PROVIDER;
  - LocationManager.Network\_PROVIDER;

- ✓ getProvider를 호출 및 선택하기

```
String providerName = LocationManager.GPS_PROVIDER;  
LocationProvider gpsProvider;  
gpsProvider = locationManager.getProvider(providerName);
```

- ✓ 모든 사용가능한 LocationProvider 얻기

```
List<String> providers = locationManager.getProviders(True);
```

# Location and MAP

## ❖ 요구기준에 적합한 LocationProvider 찾기

- ✓ Criteria Class : 사용자의 요구사항에 적합한 위치공급자의 기준을 명시
  - setAccuracy(Criteria.ACCURACY\_COARSE / Criteria.ACCURACY\_FINE) : 정확도
  - setPowerRequirement(Criteria.POWER\_LOW / ) : 전력소비량
  - setAltitudeRequired(true / false) : 고도데이터 사용 유무
  - setBearingRequired(true / false) : 방위데이터 사용 유무 \*\*\*확인
  - setSpeedRequired(true / false) : 속도데이터 사용 유무
  - setCostAllowed(true / false) : 비용 들지 여부
  
- ✓ LocationManager.getBestProvider(criteria, {true|false})
  - 요구사항에 가장 부합하는 위치공급자를 리턴.
  - boolean 현재 사용 가능한 공급자로 결과를 제한
  
- ✓ LocationManager.getProvider(criteria, {true|false})
  - 요구사항에 부합하는 위치공급자를 리스트형태로 리턴.

# Location and MAP

---

## ❖ 예 : 요구기준에 적합한 LocationProvider 찾기

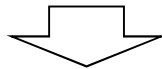
```
Criteria criteria = new Criteria();  
criteria.setAccuracy(Criteria.ACCURACY_FINE);  
criteria.setAltitudeRequired(false);  
criteria.setBearingRequired(false);  
criteria.setCostAllowed(true);  
criteria.setPowerRequirement(Criteria.POWER_LOW);  
  
String provider = locationManager.getBestProvider(criteria, true);
```

# Location and MAP

## ❖ 가장 최근에 찾았던 위치 보기

- ✓ getLastKnownLocation을 이용한 가장 최근 위치 찾기
  - location provider에 현재위치를 업데이트하도록 요청하지는 않음.
  - 리턴받은 location객체는 위도, 경도, 방위, 고도, 속도, 위치 수정이 이뤄진 시간 등을 포함할 수 있고 get메소드를 사용해 얻을 수 있음

```
String provider = LocationManager.GPS_PROVIDER;  
Location location = locationManager.getLastKnownLocation(provider);
```



에뮬레이터에서 사용하거나 전에 위치를 찾았던 적이 없으면 null을 리턴

# Location and MAP

---

## ❖ Location 사용하기

```
double lat = location.getLatitude();  
double lng = location.getLongitude();  
  
float speed = location.getSpeed();  
long time = location.getTime();
```

# Location and MAP

## ❖ LocationListener 를 이용한 위치 획득

- ✓ LocationListener 인터페이스 구현
- ✓ requestLocationUpdate를 통해 등록

```
LocationListener myLocationListener = new LocationListener() {  
    public void onLocationChanged(Location location) {  
        // Update application based on new location.  
    }  
    public void onProviderDisabled(String provider){  
        // Update application if provider disabled.  
    }  
    public void onProviderEnabled(String provider){  
        // Update application if provider enabled.  
    }  
    public void onStatusChanged(String provider, int status, Bundle extras){  
        // Update application if provider hardware status changed.  
    }  
};  
  
locationManager.requestLocationUpdates(provider, t, distance, myLocationListener);
```

# Location and MAP

## ❖ LocationListener 를 이용한 위치 획득

### ✓ requestLocationUpdate

- 위치가 바뀔 때 마다 업데이트된 위치 정보를 얻기 위해 LocationListener 등록
  - 위치 업데이트 시간과 거리 등을 설정 가능
  - locationManager.requestLocationUpdates(provider, 밀리초, 거리, LocationListener)
    - ◆ requestLocationUpdate()함수 인자는 다음과 같습니다.
    - ◆ "network" ==> Provider (gps, network, 3개의기지국? 간의 거리차)
    - ◆ 1000 ==> 업데이트 최소 시간
    - ◆ 1 ==> 업데이트 최소 거리
    - ◆ this ==> LocationListener

### ✓ LocationListener 제거

- 위치정보 업데이트를 중단함.
  - locationManager.removeUpdates(LocationListener)

# Location and MAP

## ❖ 예 : LocationListener를 이용한 위치보기

```
public class testSurfaceview extends Activity implements LocationListener {

    LocationManager mGPSMGR;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);

        mGPSMGR = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
        mGPSMGR.requestLocationUpdates("network", 1000, 1, this);
    }

    @Override
    public void onLocationChanged(Location location) {
        ((TextView)findViewById(R.id.print)).setText(location.toString());
    }

    @Override
    public void onProviderDisabled(String provider) {
    }

    @Override
    public void onProviderEnabled(String provider) {
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {
    }
}
```

# Location and MAP

---

## ❖ 근접 정보 사용하기

- ✓ 사용자가 특정 위치로 이동하거나 그 위치에서 벗어날 때 어플리케이션이 반응하도록하는 기능.
  - locationManager.addProximityAlert
- ✓ 근접 정보 발생시 intent를 생성. 보통은 broadcast intent를 생성.
- ✓ 발생시킬 intent를 지정하려면

```
Intent intent = new Intent(MY_ACTIVITY);  
PendingIntent pendingIntent = PendingIntent.getBroadcast(this, -1, intent, 0);
```

# Location and MAP

## ❖ 예 : 근접 정보 사용하기

```
private static String TREASURE_PROXIMITY_ALERT = "com.paad.treasurealert";

private void setProximityAlert() {
    String locService = Context.LOCATION_SERVICE;
    LocationManager locationManager;
    locationManager = (LocationManager) getSystemService(locService);

    double lat = 73.147536;
    double lng = 0.510638;
    float radius = 100f; // meters
    long expiration = -1; // do not expire

    Intent intent = new Intent(TREASURE_PROXIMITY_ALERT);
    PendingIntent proximityIntent = PendingIntent.getBroadcast(this, -1, intent, 0);

    locationManager.addProximityAlert(lat, lng, radius, expiration, proximityIntent);
}
```

# Location and MAP

---

```
public class ProximityIntentReceiver extends BroadcastReceiver {  
    @Override  
    public void onReceive (Context context, Intent intent) {  
        String key = LocationManager.KEY_PROXIMITY_ENTERING;  
        Boolean entering = intent.getBooleanExtra(key, false);  
        [ ... perform proximity alert actions ... ]  
    }  
}
```

To start listening for them, register your receiver, as shown in the snippet below:

```
IntentFilter filter = new IntentFilter(TREASURE_PROXIMITY_ALERT);  
registerReceiver(new ProximityIntentReceiver(), filter);
```

# Location and MAP

## ❖ 지오코더 사용하기

- ✓ 주소와 경도/위도 맵 좌표간의 정보를 서로 변환.
- ✓ Geocoder class는 두가지 지오코딩 기능을 제공
  - Forward Geocoding : 특정 주소의 위도와 경도를 찾는다
  - Reverse Geocoding : 주어진 위도와 경도에 대한 주소를 찾는다

```
Geocoder goecoder = new Geocoder(getApplicationContext(), Locale.getDefault());
```

- ✓ 역방향 지오코딩
  - 대상 위도와 경도를 지오코더의 getFromLocation메소드에 전달,
  - 일치하는 주소 리스트 리턴하거나 없을 경우 null 리턴.
- ✓ 순방향 지오코딩
  - getFromLocationName메소드 호출.
  - List<Address> 형태의 결과 값을 리턴.

# Location and MAP

---

## ❖ ReverseGeocoding의 예

```
Geocoder gc = new Geocoder(this, Locale.getDefault());  
List<Address> addresses = null;  
try {  
    addresses = gc.getFromLocation(latitude, longitude, 10);  
} catch (IOException e) {}
```

//10 is the # of max. results

# Location and MAP

---

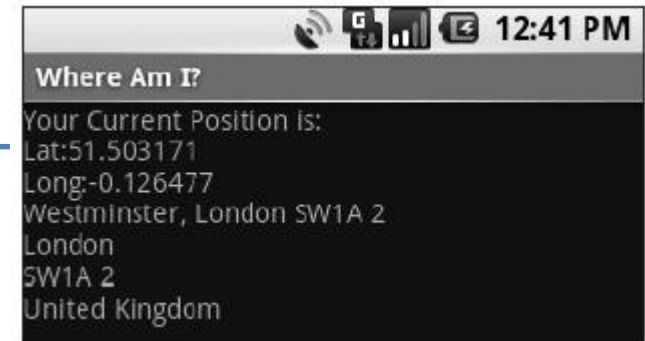
## ❖ Geocoding의 예

```
Geocoder fwdGeocoder = new Geocoder(this, Locale.US);
String streetAddress = "160 Riverside Drive, New York, New York";
List<Address> locations = null;
try {
    locations = fwdGeocoder.getFromLocationName(streetAddress, 10);
} catch (IOException e) {}
```

# Location and MAP

## ❖ GeoCoding Where am I

```
private void updateWithNewLocation(Location location) {  
    String latLongString;  
    TextView myLocationText;  
    myLocationText = (TextView)findViewById(R.id.myLocationText);  
    String addressString = "No address found";  
    if (location != null) {  
        double lat = location.getLatitude();  
        double lng = location.getLongitude();  
        latLongString = "Lat:" + lat + "\nLong:" + lng;  
        double latitude = location.getLatitude();  
        double longitude = location.getLongitude();  
        Geocoder gc = new Geocoder(this, Locale.getDefault());  
        try {  
            List<Address> addresses = gc.getFromLocation(latitude, longitude, 1);  
            StringBuilder sb = new StringBuilder();  
            if (addresses.size() > 0) {  
                Address address = addresses.get(0);
```



# Location and MAP

## ❖ GeoCoding Where am I

```
        for (int i = 0; i < address.getMaxAddressLineIndex(); i++)
            sb.append(address.getAddressLine(i)).append("\n");
        sb.append(address.getLocality()).append("\n");
        sb.append(address.getPostalCode()).append("\n");
        sb.append(address.getCountryName());
    }
    addressString = sb.toString();
} catch (IOException e) {}
} else {
    latLongString = "No location found";
}
myLocationText.setText("Your Current Position is:\n" +
    latLongString + "\n" + addressString);
}
```

# MapView 사용하기

---

## ❖ MapView와 MapActivity소개

- ✓ MapView : 실제 맵 뷰
- ✓ MapActivity : 맵 뷰를 포함할 수 있는 새로운 activity를 만들기 위해 상속받아야 할 class.
- ✓ Overlay : 맵에 주석을 다는데 사용되는 class
- ✓ MapController : 맵을 제어. 중심 위치와 확대 단계를 설정.
- ✓ MyLocationOverlay : 현재 위치와 장치의 방향을 표시하는데 사용
- ✓ ItemizedOverlay, OverlayItem : 둘이 함께 사용되어 drawable 및 그와 연관된 텍스트를 이용해 표시되는 map marker를 만듦

# MapView 사용하기

---

## ❖ MapActivity 구현 순서

- ✓ MapActivity를 상속받은 새로운 액티비티 생성.
- ✓ 액티비티에서 보여줄 레이아웃에는 MapView 추가. (apiKey를 포함)
- ✓ 도로정보를 화면 출력 시 isRouteDisplayed를 재정의.
- ✓ Maps library를 사용 설정 및 인터넷 접속 허용 설정하기
  - Manifest 파일
  - `<use-library android:name="com.google.android.maps"/>` 추가
  - `<use-permission android:name="android.permission.INTERNET"/>` 추가

참고자료 : <http://john.tobe30.com/tc/archive/201003>

# MapView 사용하기

## ❖ Google Map API Key 할당 받기

- ✓ Google Maps 데이터에 접근하는 Android에 애플리케이션을 제작하기 위해서 Google Maps service에 등록 및 약관 동의 절차(Maps API Key Signup)를 거쳐야 합니다. 참고로 Maps API Key 발급은 무료이며, 로그인 가능한 구글 계정이 필요합니다
- ✓ 1) SDK 디버그 서명 증명서의 MD5 핑거프린트(Fingerprint:지문) 확인하기
  - 디버그 서명 증명서(debug.keystore 파일)는 Android SDK가 자동으로 생성하며, 개발환경의OS마다 다른 경로가 생성됩니다

Win Vista/7: C:\Users\%username%\android\debug.keystore  
Win XP: C:\Documents and Settings\%username%\android\debug.keystore  
OS X and Linux: ~/.android/debug.keystore

- keytool를 이용하여 핑거프린트 확인할 수 있다

```
C:\Users\test\android>"%java_home%\bin\keytool.exe" -list -keystore debug.keystore -  
keypass  
s android -storepass android -alias androiddebugkey  
androiddebugkey, 2010. 2. 6, PrivateKeyEntry,  
인증서 지문(MD5): AA:22:33:F7:32:12:22:EC:68:93:8B:55:44:69:ED:47
```

# MapView 사용하기

## ✓ 2) 구글 지도 API 가입하기(Maps API Key Signup)

- 다음 URL 이동하여 위에서 확인한 인증서 지문(MD5) 값을 복사하여 [My certificate's MD5 fingerprint] 입력란에 넣고 Generate API Key를 클릭합니다. 여기서 구글 로그인 이 필요합니다.

e.g. "ajax apis" or "open source"

### Google Projects for Android: Google APIs

Home Docs Blog Forum

Download  
Terms

#### Getting Started

[What is Google APIs Add-on?](#)  
[Installing the Add-on](#)

#### Maps

[Overview](#)  
[Obtaining a Maps API Key](#)  
[Maps API Key Signup](#)  
[API Reference](#)

#### Resources

[Android SDK](#)  
[Android Developer's Guide](#)

### Maps API Key Signup

Use the form on this page to register with the Google Maps service and obtain a Maps API Key. Registration is free.

Before you register, read [Obtaining a Maps API Key](#) to understand how your Maps API Key is used in your Android application, why it's needed, and how to generate an MD5 fingerprint based on your developer certificate.

A single Maps API key is valid for all applications signed by the corresponding developer certificate. For more information about signing, see [Signing Your Applications](#) on the Android Developers site.

To register for a Key, you also need a [Google Account](#). Once you register, your Key will be associated with your Google Account.

iv. YOUR FAILURE TO PROVIDE GOOGLE WITH ACCURATE ACCOUNT INFORMATION; OR  
v. YOUR FAILURE TO KEEP YOUR PASSWORD OR ACCOUNT DETAILS SECURE AND CONFIDENTIAL.

13.2. THE LIMITATIONS ON GOOGLE'S LIABILITY TO YOU IN SECTION 13.1 ABOVE WILL APPLY WHETHER OR NOT GOOGLE, ITS AFFILIATES, LICENSORS OR THEIR SUPPLIERS HAVE BEEN ADVISED OF OR SHOULD HAVE BEEN AWARE OF THE POSSIBILITY OF ANY SUCH LOSSES OR DAMAGES.

14. Indemnity  
14.1. You hereby agree to indemnify, defend and hold Google, its strategic partners, officers, directors, agents, affiliates, licensors and their suppliers ("the Indemnified Parties") harmless from and against any claim or liability arising out of: (a) your use of the

☒ I have read and agree with the terms and conditions ([printable version](#))

My certificate's MD5 fingerprint: AA:22:33:F7:32:12:22:EC:68:93:8B:55:44:69:ED:47

[Generate API Key](#)

출처 : john.tobe30.com

©2010 Google - [Code Home](#) - [Terms of Service](#) - [Privacy Policy](#) - [Site Directory](#)

Google Code offered in: [English](#) - [Español](#) - [日本語](#) - [한국어](#) - [Português](#) - [Pycckий](#) - [中文\(简体\)](#) - [中文\(繁體\)](#)



### Google 지도 API

[Google 코드 홈](#) > [Google 지도 API](#) > [Google 지도 API 가입하기](#)

안드로이드 지도 API 키에 가입해주셔서 감사합니다.

사용자 키:

043KehnZyxJKBT00AvqTqJKDVjxYH-UOpXRgtew

이 키는

38:52:BD:57:32:02:22:EC:68:93:8B:55:05:69:ED:47

지문이 등록된 사용자 인증서가 있는 모든 응용프로그램에서 사용할 수 있습니다.

다음은 지도 기능의 사용법을 설명해 주는 xml 형태의 예제입니다.

```
<com.google.android.maps.MapView
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:apiKey="043KehnZyxJKBT00AvqTqJKDVjxYH-UOpXRgtew"
/>
```

자세한 정보는 [API 가이드](#)를 참조하세요.

출처 : john.tobe30.com

©2008 Google - [개인정보 보호정책](#) - [채용정보](#) - [이용약관](#) - [Google 정보](#)

# MapView 사용하기

## ❖ Maps library를 사용 설정 및 인터넷 접속 허용 설정하기

- ✓ Standard Android library에는 Maps library가 포함되어 있지 않으므로 사용하려면 별도의 설정이 필요. 또한 인터넷에 접속하여 Maps data를 조회하려면 INTERNET Permission을 지정(AndroidManifest.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="tobe30.sample.googlemaphello"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".GoogleMapHello"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <!--map library 사용 설정-->
        <uses-library android:name="com.google.android.maps" />
    </application>
    <uses-sdk android:minSdkVersion="7" />
    <!--INTERNET Permission 지정-->
    <uses-permission android:name="android.permission.INTERNET" />
</manifest>
```

# MapView 사용하기

- ❖ MapView 애플리케이션의 레이아웃 작성하기 (res/layout/main.xml)
  - ✓ com.google.android.maps.MapView 클래스를 레이아웃에 추가한다.
  - ✓ GoogleMapHello의 res/layout/main.xml
  - ✓ Map APIKey 설정

```
<?xml version="1.0" encoding="utf-8"?>
<com.google.android.maps.MapView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/mapview"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:clickable="true"
    android:apiKey="043KehnZyxJK7Ny0hTKJPBcp_rhr6VLCMoDQeA"
/>
```

# MapView 사용하기

## ❖ 애플리케이션의 Activity를 구현하기 (GoogleMapHello.java)

- ✓ MapActivity 클래스는 MapView에 지도 표시를 관리하는 역할을 합니다.  
그래서 GoogleMapHello Activity는 MapActivity 클래스를 상속받아 구현해야 합니다

```
public class GoogleMapHello extends MapActivity {  
    /** Called when the activity is first created. */  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
  
        MapView mapView = (MapView) findViewById(R.id.mapview);  
        mapView.setBuiltInZoomControls(true); //줌컨트롤을 활성화  
    }  
  
    @Override  
    protected boolean isRouteDisplayed() {  
        return false;  
    }  
}
```

isRouteDisplayed() 메소드는 MapActivity의 추상 메소드이므로 반드시 구현해야 합니다. isRouteDisplayed() 메소드는 애플리케이션에 라우트 정보를 표시할 경우 true를 반환하고, 그렇지 않은 경우 false를 반환합니다.

# MapView 사용하기

---

## ❖ 맵뷰 구성하고 사용하기

- ✓ mapView.setSatellite(true) : 위성 뷰
- ✓ mapView.setStreetView(true) : street뷰
- ✓ mapView.setTraffic(true) : 예상 교통량
  
- ✓ map Zoom Controls 사용을 위해
  - View zoomControls = mapView.getZoomControls();
  - mapView.addView(zoomControls, MapView.LayoutParams);
  - mapView.displayZoomControls(true);

참고 : <http://gtkko.springnote.com/pages/5409245>

# MapView 사용하기

## ❖ 맵 컨트롤러 사용하기

- ✓ MapView를 이동/확대 시.
  - getController를 이용해 MapView Controller 레퍼런스를 얻는다.
    - MapViewController mc = MapView.getController();
  - Location객체에 저장된 위도 경도 값을 이용하려면 GeoPoint에 백만단위로 변환한 뒤 저장
    - Double lat = 37.123 \* 1E6;
    - Double lng = 123.123 \* 1E6;
    - GeoPoint point = new GeoPoint(lat.intValue(), lng.intValue());
- ✓ MapView 중심을 다시 잡거나 확대하려면
  - MapController.setCenter(point); //point위치가 화면 중앙에 오도록 설정.
  - MapController.setZoom(1); : 1~21
- ✓ 새로운 위치로 이동시 MapController.animateTo(point)를 이용해 부드러운 화면 이동 처리

# MapView 사용하기

---

## ❖ 오버레이 만들고 사용하기

### ✓ 새 오버레이 만들기

- Overlay를 상속받는 새로운 클래스 생성.
- draw메소드를 재정의해 화면에 보여질 데이터 표기.
- onTap을 재정의해 사용자 클릭에 반응한다.

### ✓ 프로젝션 소개

- Projection 클래스는 위도/경도 좌표와 x/y화면 픽셀 좌표간을 바꿀 수 있도록 해준다.
- Projection projection = mapView.getProjection();
- GeoPoint를 Point로 혹은 그 반대로 바꾸려면 fromPixel, toPixel메소드 이용

# MapView 사용하기

---

## ❖ 오버레이 만들고 사용하기

### ✓ 오버레이 추가하고 제거하기

- MapView는 현재 표시되는 오버레이들의 리스트를 갖고 있다.
- getOverlays를 호출해 이에 대한 레퍼런스를 얻는다
- `List<Overlay> overlays = mapView.getOverlays();`

### ✓ 맵뷰에 오버레이를 추가하려면

- `overlays.add(새로추가할Overlay);`
- `mapView.postInvalidate();` //맵 위의 변경사항을 업데이트

# MapView 사용하기

---

## ❖ MyLocationOverlay소개

- ✓ 현재 위치와 방향을 MapView위에 보여주도록 설계된 오버레이
- ✓ MyLocationOverlay를 사용하려면
  - `List<Overlay> overlays = mapView.getOverlays();`
  - `MyLocationOverlay mylocationoverlay = new MyLocationOverlay(this, mapView);`
  - `overlays.add(mylocationoverlay);`
- ✓ 나침반과 마커 모두를 활성화 하기위해서는
  - `mylocationoverlay.enableCompass();`
  - `mylocationoverlay.enableMyLocation(mapView.getMapController());`

# MapView 사용하기

## ❖ ItemizedOverlay와 OverlayItem소개

- ✓ MapView에 간단한 마커기능을 제공
- ✓ 맵에 ItemizedOverlay 마커레이어 추가하려면
  - ItemizedOverlay<OverlayItem>을 상속받는 클래스 만든다.
  - size 메소드를 재정의해 표시할 마커 수 리턴.
  - createItem 메소드 재정의해 각 마커의 인덱스에 기반한 새로운 OverlayItem생성.
  - 클래스 생성자에서 populate()를 호출해 각 OverlayItem이 생성되도록 만든다.
- ✓ 맵에 ItemizedOverlay를 추가하려면
  - List<Overlay> overlays = mapView.getOverlay();
  - MyItemizedOverlay markers = new MyItemizedOverlay(이미지);
  - overlays.add(markers);

# MapView 사용하기

## ❖ 예제 : MyMap

- ✓ 구글 맵 띄우고 현재 위치 표시하기



# MapView 사용하기

## ❖ 예제 : MyMap

```
package org.example.mymap;
import android.os.Bundle;
import com.google.android.maps.MapActivity;
import com.google.android.maps.MapController;
import com.google.android.maps.MapView;
import com.google.android.maps.MyLocationOverlay;
public class MyMap extends MapActivity {
    private MapView map;
    private MapController controller;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        initMapView();
        initMyLocation();
    }
    @Override
    protected boolean isRouteDisplayed() {
        // Required by MapActivity
        return false;
    }
}
```

# MapView 사용하기

## ❖ 예제 : MyMap

```
/** Find and initialize the map view. */
private void initMapView() {
    map = (MapView) findViewById(R.id.map);
    controller = map.getController();
    map.setSatellite(true);
    map.setBuiltInZoomControls(true);
}

/** Start tracking the position on the map. */
private void initMyLocation() {
    final MyLocationOverlay overlay = new MyLocationOverlay(this, map);
    overlay.enableMyLocation();
    //overlay.enableCompass(); // does not work in emulator
    overlay.runOnFirstFix(new Runnable() {
        public void run() {
            // Zoom in to current location
            controller.setZoom(8);
            controller.animateTo(overlay.getMyLocation());
        }
    });
    map.getOverlays().add(overlay);
}
}
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”



Professional Android

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
import android.graphics.Canvas;
import android.graphics.Paint;
import android.graphics.Point;
import android.graphics.RectF;
import android.location.Location;
import com.google.android.maps.GeoPoint;
import com.google.android.maps.MapView;
import com.google.android.maps.Overlay;
import com.google.android.maps.Projection;

public class MyPositionOverlay extends Overlay {

    @Override
    public void draw(Canvas canvas, MapView mapView, boolean shadow) {
    }

    @Override
    public boolean onTap(GeoPoint point, MapView mapView) {
        return false;
    }
}
```

# MapView 사용하기

---

## ❖ 예제 : Annotating “Where am I?”

```
Location location;

public Location getLocation() {
    return location;
}

public void setLocation(Location location) {
    this.location = location;
}
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
private final int mRadius = 5;

@Override
public void draw(Canvas canvas, MapView mapView, boolean shadow) {
    Projection projection = mapView.getProjection();

    if (shadow == false) {
        // Get the current location
        Double latitude = location.getLatitude()*1E6;
        Double longitude = location.getLongitude()*1E6;
        GeoPoint geoPoint;
        geoPoint = new GeoPoint(latitude.intValue(),longitude.intValue());

        // Convert the location to screen pixels
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
Point point = new Point();
projection.toPixels(geoPoint, point);

RectF oval = new RectF(point.x - mRadius, point.y - mRadius,
                        point.x + mRadius, point.y + mRadius);

// Setup the paint
Paint paint = new Paint();
paint.setARGB(250, 255, 0, 0);
paint.setAntiAlias(true);
paint.setFakeBoldText(true);

Paint backPaint = new Paint();
backPaint.setARGB(175, 50, 50, 50);
backPaint.setAntiAlias(true);

RectF backRect = new RectF(point.x + 2 * mRadius,
                           point.y - 3 * mRadius,
                           point.x + 65, point.y + mRadius);

// Draw the marker
canvas.drawOval(oval, paint);
canvas.drawRoundRect(backRect, 5, 5, backPaint);

canvas.drawText("Here I Am", point.x + 2 * mRadius, point.y, paint);
}
super.draw(canvas, mapView, shadow);
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
MyPositionOverlay positionOverlay;

@Override
public void onCreate(Bundle icle) {
    super.onCreate(icle);
    setContentView(R.layout.main);

    MapView myMapView = (MapView)findViewById(R.id.myMapView);
    mapController = myMapView.getController();

    myMapView.setSatellite(true);
    myMapView.setStreetView(true);
    myMapView.displayZoomControls(false);

    mapController.setZoom(17);

    // Add the MyPositionOverlay
    positionOverlay = new MyPositionOverlay();
    List<Overlay> overlays = myMapView.getOverlays();
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
overlays.add(positionOverlay);

LocationManager locationManager;
String context = Context.LOCATION_SERVICE;
locationManager = (LocationManager) getSystemService(context);

Criteria criteria = new Criteria();
criteria.setAccuracy(Criteria.ACCURACY_FINE);
criteria.setAltitudeRequired(false);
criteria.setBearingRequired(false);
criteria.setCostAllowed(true);
criteria.setPowerRequirement(Criteria.POWER_LOW);
String provider = locationManager.getBestProvider(criteria, true);

Location location = locationManager.getLastKnownLocation(provider);

updateWithNewLocation(location);

locationManager.requestLocationUpdates(provider, 2000, 10,
                                       locationManager);
}
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
private void updateWithNewLocation(Location location) {
    String latLongString;
    TextView myLocationText;
    myLocationText = (TextView)findViewById(R.id.myLocationText);
    String addressString = "No address found";

    if (location != null) {
        // Update my location marker
        positionOverlay.setLocation(location);

        // Update the map location.
        Double geoLat = location.getLatitude()*1E6;
        Double geoLng = location.getLongitude()*1E6;
        GeoPoint point = new GeoPoint(geoLat.intValue(),
                                      geoLng.intValue());

        mapController.animateTo(point);

        double lat = location.getLatitude();
        double lng = location.getLongitude();
        latLongString = "Lat:" + lat + "\nLong:" + lng;

        double latitude = location.getLatitude();
        double longitude = location.getLongitude();

        Geocoder gc = new Geocoder(this, Locale.getDefault());
        try {
            List<Address> addresses = gc.getFromLocation(latitude, longitude, 1);
            StringBuilder sb = new StringBuilder();
```

# MapView 사용하기

## ❖ 예제 : Annotating “Where am I?”

```
if (addresses.size() > 0) {
    Address address = addresses.get(0);

    for (int i = 0; i < address.getMaxAddressLineIndex(); i++)
        sb.append(address.getAddressLine(i)).append("\n");

    sb.append(address.getLocality()).append("\n");
    sb.append(address.getPostalCode()).append("\n");
    sb.append(address.getCountryName());
}
addressString = sb.toString();
} catch (IOException e) {}
} else {
    latLongString = "No location found";
}
myLocationText.setText("Your Current Position is:\n" +
                        latLongString + "\n" + addressString);
}
```