

10. 위치기반 서비스



시작하면서



◎ 목차

- 위치정보 출력
- 요구 기준에 의한 위치정보 조회
- 지오코딩
- MapView와 MapActivity
- Maps API 키 발급

위치정보 출력

위치정보 출력



◎ 위치 정보 제공

- GPS
- 무선 LAN
- 이동통신망

◎ 위치기반 서비스

- android.location 패키지
 - 휴대 단말의 위치를 얻기 위한 클래스를 포함
 - 위치 정보는 지리적 좌표(위도와 경도) 사용
 - 위치 정보와 주소 정보를 변환하는 Geocoder 클래스 제공
- Google Maps 외부 라이브러리 com.google.android.maps
 - 지도를 표시하거나 접근

◎ LocationManager 클래스

- 위치 기반 서비스에서 가장 중요한 시스템 서비스
- 위치 기반 서비스를 위한 API 제공

◎ AVD를 통한 위치 기반 서비스 개발시

- 실제로 GPS로부터 위도와 경도 좌표를 받지 못함
- 가상적으로 DDMS에서 위도와 경도 좌표 설정

위치정보 출력



- 관련 메소드
 - LocationManager locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
 - List<String> providers = locationManager.getProviders(true);
- 실습 단계
 - 시스템 서비스 설정
 - 위치정보 제공자 얻기
 - 위치 변경 조건이 발생할 때마다 통지될 액티비티로 현 액티비티를 지정
 - 최근 위치정보 가져옴
- 위치 정보 접근 허가 필요
 - <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>

위치정보 출력 실습



◎ <실습 10-1> 좌표 정보 출력

- BasicLocDemo 프로젝트를 생성
 - <코드 10-1>을 main.xml에 복사
 - <코드 10-2>로 AndroidManifest.xml을 수정
 - <코드 10-3>~<코드 10-4>를 BasicLocDemo.java에 복사
 - 좌표값 입력 방법
 - DDMS를 선택 후, Emulator Control-Location Controls로 이동
 - [그림 10-1]: 위도값 37.422006, 경도값(longitude) -122.084095을 입력
 - 안드로이드 2.2로 실행

위치정보 출력 실습



The screenshot shows the DDMS (Dalvik Debug Monitor Service) interface within the Eclipse IDE. The 'Devices' tab is active, displaying a list of processes for the emulator 'emulator-5554'. The 'Emulator Control' panel is also visible, showing 'Location Controls' with 'Manual' selected. The 'Longitude' field is set to -122.084095 and the 'Latitude' field is set to 37.422006. A 'Send' button is located below the fields.

Name	Online
emulator-5554	Online
system_process	51
com.android.settings	95
com.android.phone	99
android.process.acore	102
com.coreia.BitmapDemo	606
com.coreia.ShapePathDen	633
com.coreia.LinearGradient	738
com.coreia.TweenAniDem	818

The screenshot shows the output of an Android application named 'BasicLocDemo'. The interface displays location information for two different location providers: 'passive' and 'gps'. The time shown is 7:47 AM.

```
BasicLocDemo
위치 공급자 :passive
위도:37.422834216666665, 경도:
-122.08536667833332
위치 공급자 :gps
위도:37.422834216666665, 경도:
-122.08536667833332
```

요구 기준에 의한 위치 정보 조회

요구 기준에 의한 좌표 정보 조회



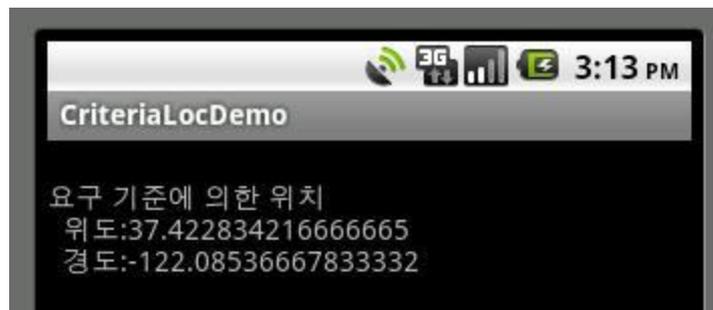
◎ 안드로이드는 다양한 위치 정보 제공자를 제공

- 개발자들은 자신의 요구 기준에 적합한 위치 정보를 제공받을 수 있음
- Criteria 클래스
 - 개발자들이 다양한 위치 정보 제공자 중에서 자신에 적합한 제공자를 선택할 수 있는 기능 제공
 - Criteria 클래스를 통해 위치 정보와 관련된 설정 후
 - 위치 정보의 정밀도 수준, 고도 정보, 방위 정보, 비용 발생 여부, 전력 소모량의 수준 등 관련 기준을 설정
 - 안드로이드는 최적의 위치정보 제공자 반환

요구 기준에 의한 좌표 조회 실습

◎ <실습 10-2> 요구기준에 의한 좌표 정보 출력

- CriteriaLocDemo 프로젝트 생성
 - <코드 10-5>를 main.xml에 복사
 - <코드 10-2>로 AndroidManifest.xml을 수정
 - <코드 10-6>~<코드 10-7>을 CriteriaLocDemo.java에 복사
 - DDMS를 선택 후, Emulator Control-Location Controls로 이동
 - 위도값 37.422006, 경도값(longitude) -122.084095을 입력
 - 안드로이드 2.2로 실행
 - <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>



지오코딩

지오코딩(Geocoding)



- 위도와 경도의 지리적 좌표와 거리 주소^{street address} 또는 우편번호와 같은 문자적 위치 정보로 상호 변환하는 과정
 - 역방향 지오코딩^{Reverse geocoding}
 - 지리적 좌표인 위도 및 경도를 문자적 위치 정보로 변환
 - 순방향 지오코딩^{Forward geocoding}
 - 문자적 위치 정보(거리 주소,우편번호 등)를 위도 및 경도 좌표로 변환
- 관련 메소드
 - `Geocoder gc = new Geocoder(this, Locale.getDefault());`
 - `List<Address> addresses = gc.getFromLocation(lat, lng, 1);`

역방향 지오코딩 실습



◎ <실습 10-3> 역방향 지오코딩 실습

- RevGeoDemo 프로젝트 생성
 - <코드 10-8>을 main.xml에 복사
 - <코드 10-2>로 AndroidManifest.xml을 수정
 - <코드 10-9>~<코드 10-11>을 RevGeoDemo.java에 복사
- DDMS를 선택 후, Emulator Control-Location Controls로 이동
 - [그림 10-1]: 위도값 37.422006, 경도값(longitude) -122.084095을 입력
 - Google API 2.1로 실행
- <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>



순방향 지오코딩(Forward Geocoding)

◎ 개념

- 일상적으로 위치를 표현하는 명칭들
 - 지역명: "대한민국 부산광역시"
 - 주소: "서울시 종로구 청와대로 1"
 - 우편번호: "302-789"
 - 코드: "ICN"(인천국제공항)
- 문자적 위치 정보를 지리적 좌표 정보로 변환
- 관련 메소드
 - `List<Address> geocodeResults =
 coder.getFromLocationName(placeName, 3);`

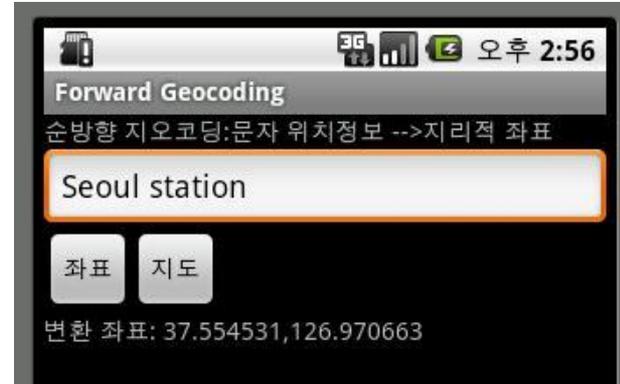
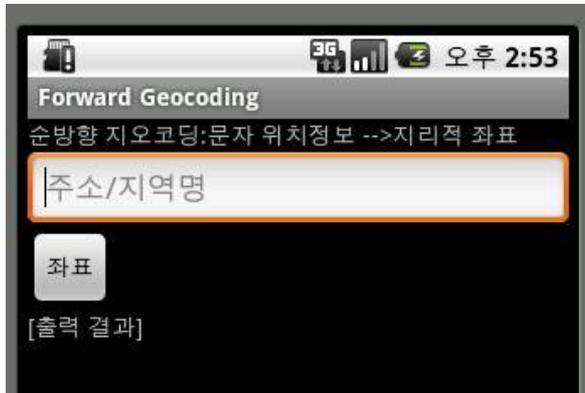
순방향 지오키딩 실습



◎ <실습 10-4> 순방향 지오키딩 실습

- ForGeoLocDemo 프로젝트 생성
 - <코드 10-12>를 geoaddress.xml에 복사
 - <코드 10-2>로 AndroidManifest.xml을 수정
 - <코드 10-13>~<코드 10-14>를 ForGeoLocDemo.java에 복사
 - DDMS를 선택 후, Emulator Control-Location Controls로 이동
 - [그림 10-1]: 위도값 37.422006, 경도값(longitude) -122.084095을 입력
 - Google API 2.1로 실행
 - <uses-permission android:name=
"android.permission.ACCESS_FINE_LOCATION"/>

순방향 지오코딩 실습



MAPVIEW와 MAPACTIVITY

MapView와 MapActivity



◎ 지도 기반의 애플리케이션 작성시

- MapActivity를 사용하는 것이 편리
- 지도가 들어가는 부분에 `com.google.android.MapView` 요소를 추가

- 관련 메소드
 - `MapView myMapView = (MapView)findViewById(R.id.myMapView);`
 - `mapController = myMapView.getController();`

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

 - `Double geoLat = location.getLatitude()*1E6;`
 - `Double geoLng = location.getLongitude()*1E6;`
 - `GeoPoint point = new GeoPoint(geoLat.intValue(), geoLng.intValue());`
 - `mapController.animateTo(point);`

MapView와 MapActivity

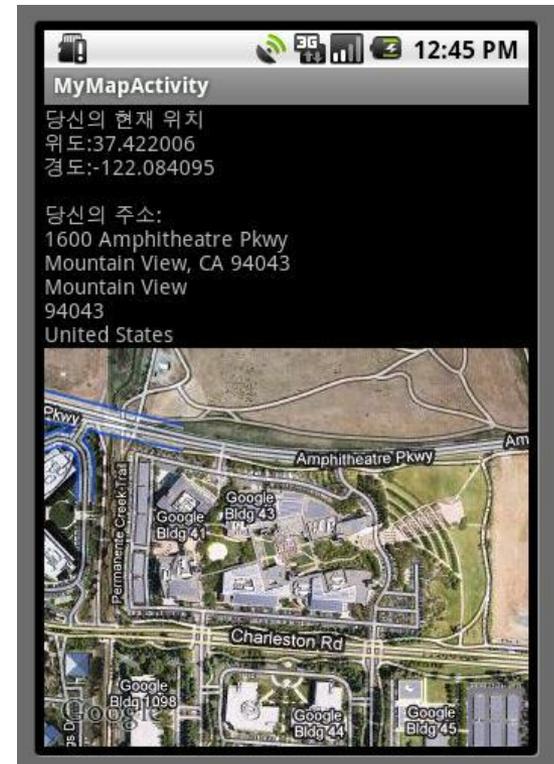


- 지도 라이브러리를 포함
 - `<uses-library android:name="com.google.android.maps" />`
- 인터넷 사용 및 지도 접근을 위한 정밀도 설정
 - `<uses-permission android:name="android.permission.INTERNET" />`
 - `<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />`
- main.xml에 Mapview와 관련된 조건을 지정
 - `android:apiKey` 조건은 구글 지도 사용시 필요한 API 키를 지정
 - `android:enabled="true"`
 - `android:clickable="true"`
 - `android:apiKey="0eT1yZDNW8P9HuV9O_t-YfewvcXDfQVhZsXdQCw"`

MapView와 MapActivity 실습

◎ <실습 10-5> 좌표, 주소, 지도 출력

- MapViewLocDemo 프로젝트 생성
 - <코드 10-15>를 main.xml에 복사
 - <코드 10-16>로 AndroidManifest.xml을 수정
 - <코드 10-17>~<코드 10-19>를 MapViewLocDemo.java에 복사
 - DDMS를 선택 후,
 - Emulator Control-Location Controls로 이동
 - [그림 10-1]: 위도값 37.422006, 경도값(longitude) -122.084095을 입력
 - Google API[안드로이드 2.1]로 실행



Maps API 키 발급



- 구글 지도를 사용하려면 안드로이드 지도 API 키 필요
 - 먼저 MD5 지문 획득
- 진행 절차
 - ① 편의상 C:/Program Files/Java/jre1.60_03/bin/ 밑에 있는 keytool.exe 파일과 jli.dll 파일을 C:/Document and Settings/user/로 복사한다.
 - ② 편의상 C:/Document and Settings/user/.android/ 밑에 있는 debug.keystore 파일을 C:/ 밑으로 복사한다.
 - ③ 다음에 C:/Document and Settings/user/에 있는 keytool.exe 파일을 다음과 같이 실행한다.
 - C:/Document and Settings/user>keytool -list -alias androiddebugkey -keystore C:/debug.keystore -storepass android -keypass android

```
C:#Documents and Settings#user>keytool -list -alias androiddebugkey -keystore C:
#debug.keystore -storepass android -keypass android
androiddebugkey, 2010. 3. 8, PrivateKeyEntry,
인증서 지문(MD5): 7F:8F:7A:04:1E:E7:D3:E9:F7:5A:60:37:FA:1A:7B:BB

C:#Documents and Settings#user>
```

Maps API 키 발급



④ [그림 10-7]과 같이 MD5 지문이 출력. MD5 지문은 두 개의 문자와 : 로 구성된 문자열 패턴

예) "7F:8F:7A:04:1E:E7:D3:E9:F7:5A:60:37:FA:1A:7B:BB"

⑤ 구글사이트(www.google.com)에 가서 자신의 계정을 만들고 로그인한 다음, <http://code.google.com/android/maps-api-signup.html>에 MD5 지문을 입력하면 API Key가 생성됨

⑥ 생성된 API Key를 다음과 같이 main.xml 파일의 android:apiKey 속성에 복사

```

▪ <com.google.android.maps.MapView
  android:id="@+id/myMapView"
  android:enabled="true"
  android:clickable="true"
  android:apiKey="0a1nyi..... B0Dt-Q" <-- 여기에 apiKey 복사 입력
/>

```

- Google 지도 서비스를 이용하면 됨.

Maps API 키 발급



????? ?? API - ?????? - Windows Internet Explorer

http://www.google.com/glm/mmap/a/api?fp=47%3A3C%3A94

Google

파일(E) 편집(E) 보기(V) 즐겨찾기(A) 도구(D) 도움말(H)

????? ?? API - ??????

Google **Google ?? API**

Google ?? ? > Google ?? API > Google ?? API ?????

?????? ?? API ?? ??????? ??????.

??? ?:

00wSfxTEsO_yUUyrJ22YdFiWHy4egZDzELop5Yg

? ??

47:3C:94:49:57:F8:A8:99:FA:CC:99:80:49:55:20:C1

??? ??? ??? ????? ?? ?? ?????????? ??? ? ?????.

??? ?? ??? ????? ??? ?? xml ??? ??????

```
<com.google.android.maps.MapView
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:apiKey="00wSfxTEsO_yUUyrJ22YdFiWHy4egZDzELop5Yg"
/>
```

??? ??? API ????? ??????.

완료 인터넷 100%