



구글맵과 위치추적



Contents title

19.1 구글맵 활용 원리

19.1.1 구글맵 활용의 예

19.1.2 구글맵 활용원리

19.1.3 activity와 fragment

19.1.4 구글맵 개발환경 구축(Google Play Services 추가)

19.2 구글맵 프로젝트의 시작과 구글맵 API 키 받기

19.2.1 프로젝트 개요

19.2.2 프로젝트 개발

19.2.3 구글맵 유형과 UI 컨트롤

19.2.4 특정 위치 중심의 구글맵 출력

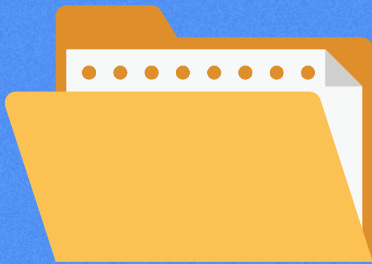
19.2.5 Marker와 Info Window

19.2.6 Custom Icon으로 만든 Marker 표시

19.2.7 다수의 Marker 표시

19.2.8 현재 위치 추적

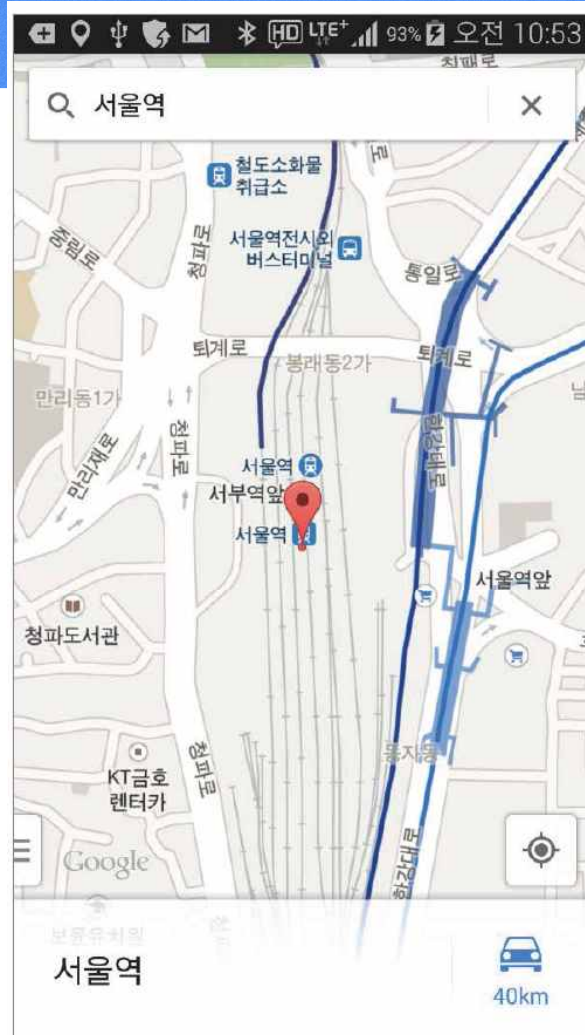




19.1 구글맵 활용 원리



19.1.1 구글맵 활용의 예



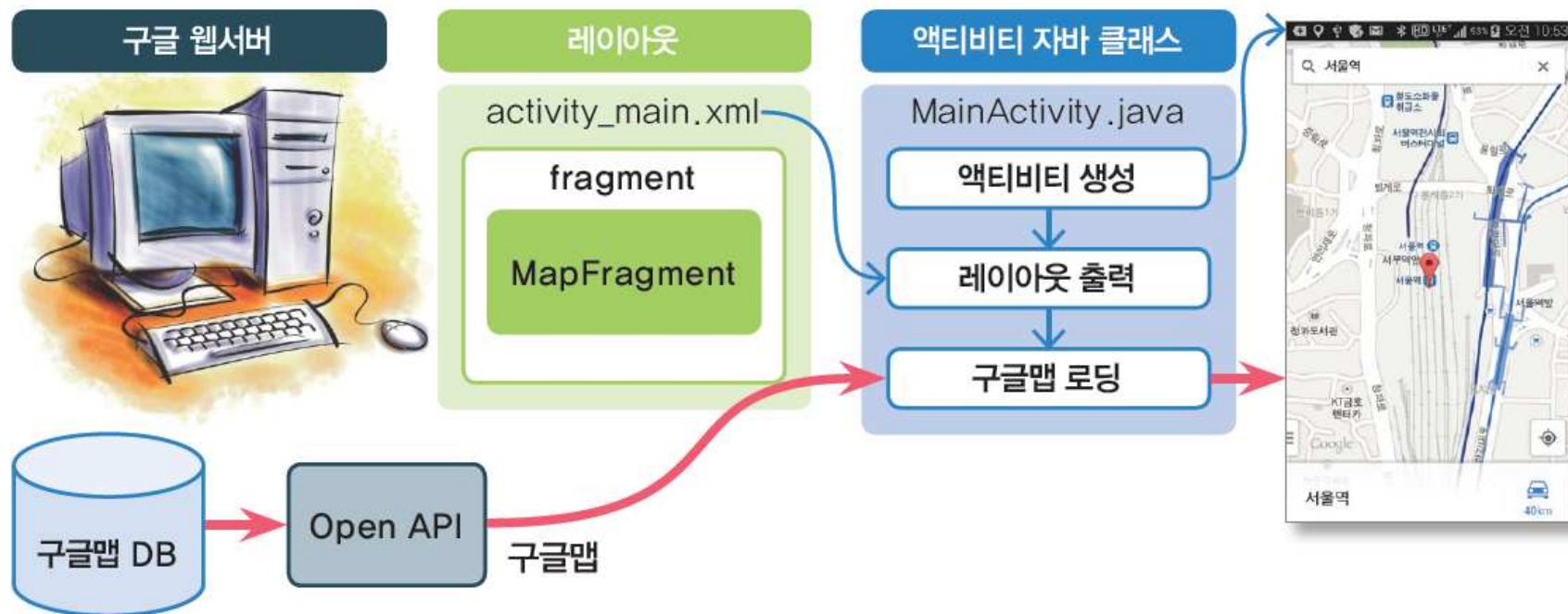
(a) 지도 및 위치 검색



(b) 위성 사진

- 구글맵의 예

19.1.2 구글맵 활용원리

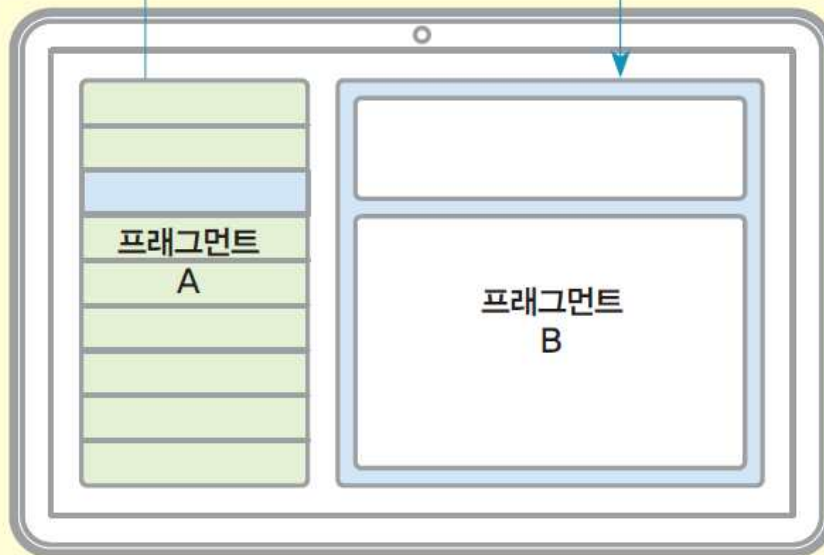


19.1.3

activity와 fragment

태블릿

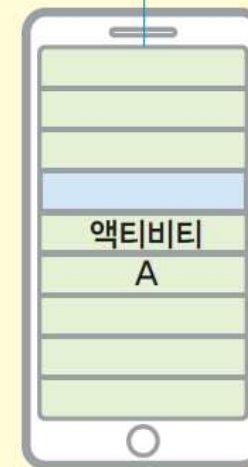
메뉴 아이템을 클릭하면
fragment B가
업데이트됨.



activity A는 fragment A와 B로 구성됨.

핸드폰

메뉴 아이템을 클릭하면
activity B가 시작됨.



activity A는
fragment A로 구성됨.



activity B는
fragment B로 구성됨

- Fragment를 사용하는 태블릿 디자인과 핸드폰 디자인의 비교
(자료원: developer.android.com)

하나의 액티비티에 두 개의 fragment를 배치한 예

```
01 <?xml version="1.0" encoding="utf-8"?>
02 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
03     android:orientation="horizontal"
04     android:layout_width="match_parent"
05     android:layout_height="match_parent">
06     <fragment
07         android:name="com.example.news.ActivityClassA"
08         android:id="@+id/list"
09         android:layout_weight="1"
10         android:layout_width="0dp"
11         android:layout_height="match_parent" />
12     <fragment
13         android:name="com.example.news.ActivityClassB"
14         android:id="@+id/viewer"
15         android:layout_weight="2"
16         android:layout_width="0dp"
17         android:layout_height="match_parent" />
18 </LinearLayout>
```

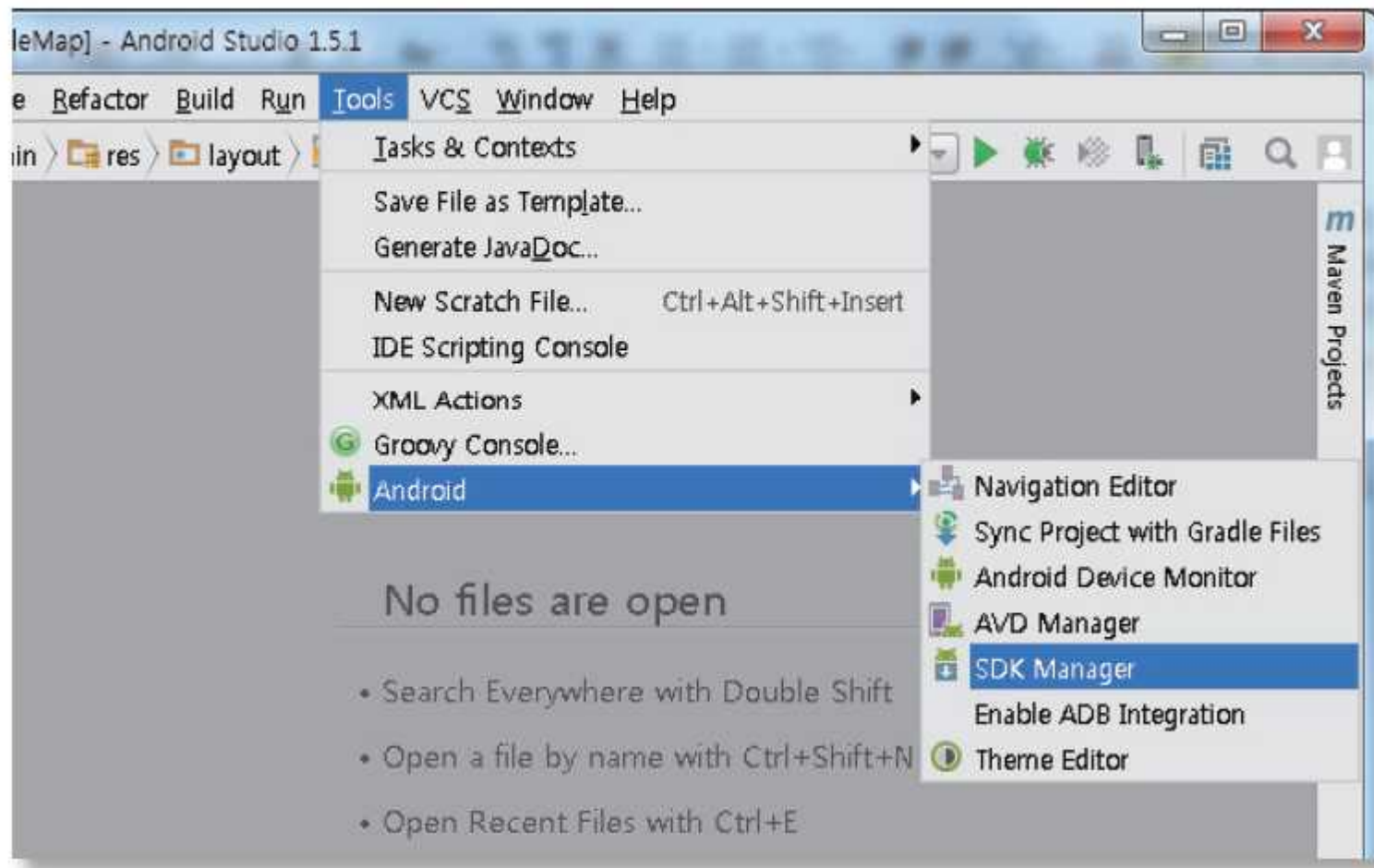
프래그먼트1에서
실행될 자바 클래스

프래그먼트1

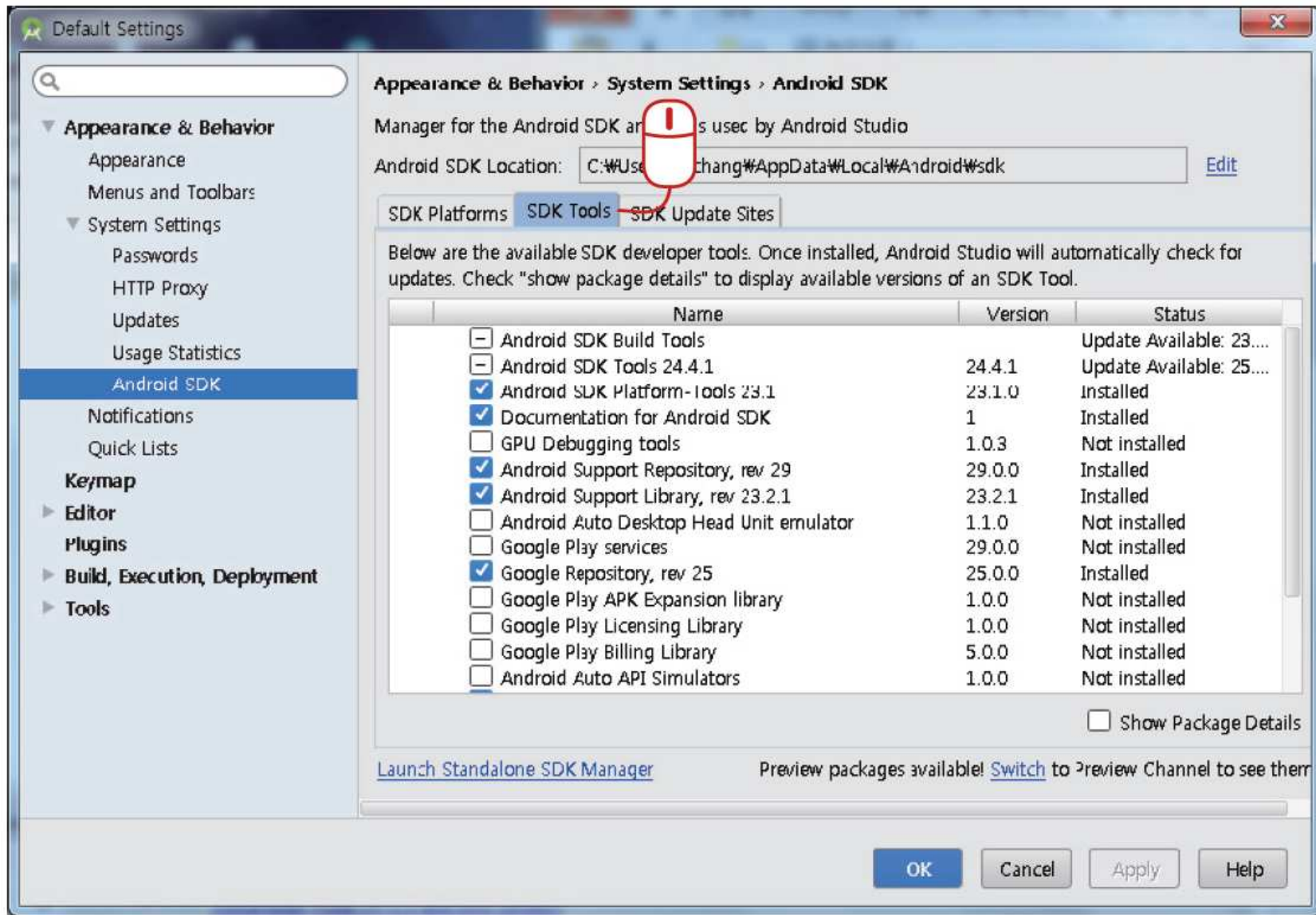
프래그먼트2

19.1.4 구글맵 개발환경 구축(Google Play Services 추가)

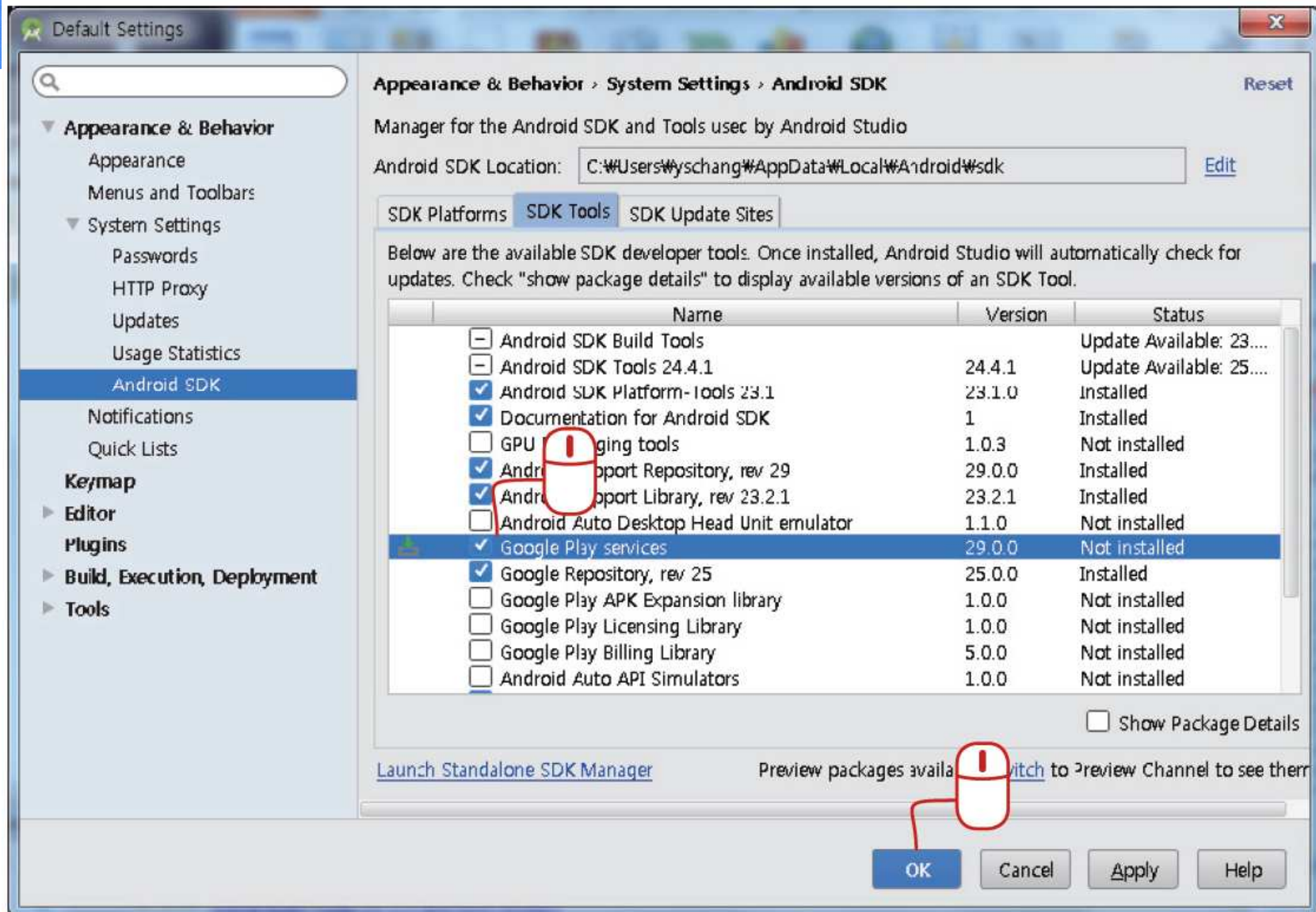
■ 메뉴 Tools → Android → SDK Manager 클릭

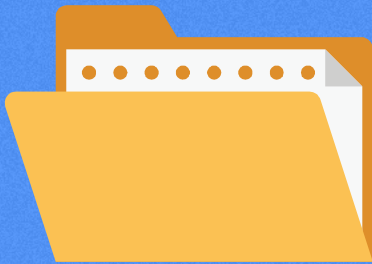


■ SDK Tools 탭 선택



■ Google Play Services 라이브러리 선택 후 'OK' 버튼 클릭





19.2 구글맵 프로젝트의 시작 과 구글맵 API 키 받기: Google Map 프로젝트(Map)



19.2.1 프로젝트 개요

프로젝트 개요: 구글맵 출력

Application Name: Google Map

어플리케이션 라벨: Map

초기 화면(디폴트로 위도 34,
경도 151 위치를 중심으로
지도가 마커가 출력되고 나타남

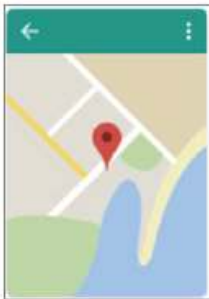


마커를 클릭하면, 마커 위에
정보 창이 나타나고 우측 하단에는
맵 툴바가 나타남



STEP 1

프로젝트 생성

절차	내용
① 프로젝트 시작	메뉴에서 'File → New Project' 클릭
② 프로젝트 구성	Application Name: Google Map , Company Domain: yschang.example.com
③ 제품형태	Phone and Tablet
④ 액티비티 유형	 Google Map Activity
⑤ 액티비티 구성	디폴트 값으로 설정 Activity Name: MapsActivity , Layout Name: activity_maps Title: Map

STEP 2

파일 편집

모듈	폴더	소스 파일	편집 내용
manifests		AndroidManifest.xml	<ul style="list-style-type: none"> 구글맵 출력 환경
java	com.example.yschang.googlemap	MapsActivity.java	<ul style="list-style-type: none"> 구글맵 출력 구글맵 지도유형 변경 특정 위치 중심 지도 출력 커스텀 아이콘 출력 다수 마커 출력 현재 위치 출력
res	drawable	coffee.png	<ul style="list-style-type: none"> 커스텀 마커의 아이콘 이미지
	layout	activity_maps.xml	<ul style="list-style-type: none"> 구글맵 출력화면 배치
	mipmap	ic_launcher.png	
	values	colors.xml	
		google_maps_api.xml	<ul style="list-style-type: none"> 구글맵 API 키 관리
		strings.xml	
		styles.xml	

■ 수정 ■ 추가

● 파일 간의 연관관계

strings.xml에는 초기치로 설정되어 있는 어플리케이션 라벨을 'Map'으로 수정한다.

google_maps_api.xml에는 발급받은 구글맵 API 키 값으로 수정한다.

drawable 폴더에 구글맵 마커 커스텀 아이콘으로 출력할 coffee.png를 저장한다.

activity_main.xml에는 구글맵 출력을 위한 클래스를 배치한다.

MainActivity.java에는 위도와 경도에 대한 위치를 설정하고, 마커를 추가하며, 해당 위치를 중심으로 하는 지도를 출력한다.

AndroidManifest.xml에는 GPS 사용을 허용하도록 설정한다.

이미지 (구글맵 마커 커스텀 아이콘)

추가



coffee.png (drawable)

텍스트 리소스

수정

string

google_maps_key

...구글맵 API 키...

google_maps_api.xml (values)

구글맵
API 키

텍스트 리소스

수정

string

app_name Google Map

strings.xml (values)

앱 라벨

화면 레이아웃

fragment

```
tools:context com.  
example.googlemap.MapsActivity
```

수정

구글맵 출력을
위한 클래스

activity_main.xml (layout)

액티비티 제어

onCreate()

```
super.onCreate()  
setContentView(R.layout.activity_main)
```

수정

시드니 위도 경도

onMapReady()

```
sydney = new LatLng(-34, 151)  
mMap.addMarker(  
    .icon(.fromResource(R.drawable.  
        coffee))  
    mMap.moveCamera()
```

해당 위치에
마커 표시

MainActivity.java (java)

노션ID API의 URL

해당 위치로 지도 중심 이동

어플리케이션 기본 정보

uses-permission

```
name  
android.permission.ACCESS_FINE_LOCATION
```

수정

GPS 사용 허용

application

```
icon @mipmap/ic_launcher  
label @string/app_name  
theme @style/AppTheme
```

meta-data

```
name com.google.android.geo.API_KEY  
value @string/google_maps_key
```

activity

```
name MainActivity
```

AndroidManifest.xml (manifest)



사용자

출력



스마트폰

컴파일/빌더

컴파일/빌더 정보

```
build gradle(Project)  
build gradle(Module app)  
gradle properties  
settings gradle  
local properties
```

(Gradle Scripts)

● 구글맵 API 키 발급 받기

■ google_maps_api.xml 파일 열기(SHA-1 지문을 포함하는 URL 확인)

Google Map - [C:\AndroidProjects\GoogleMap] - [app] - ...Wapp\src\debug\res\values\google_maps_api.xml - Android Studio 1.5.1

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

GoogleMap > app > src > debug > res > values > google_maps_api.xml

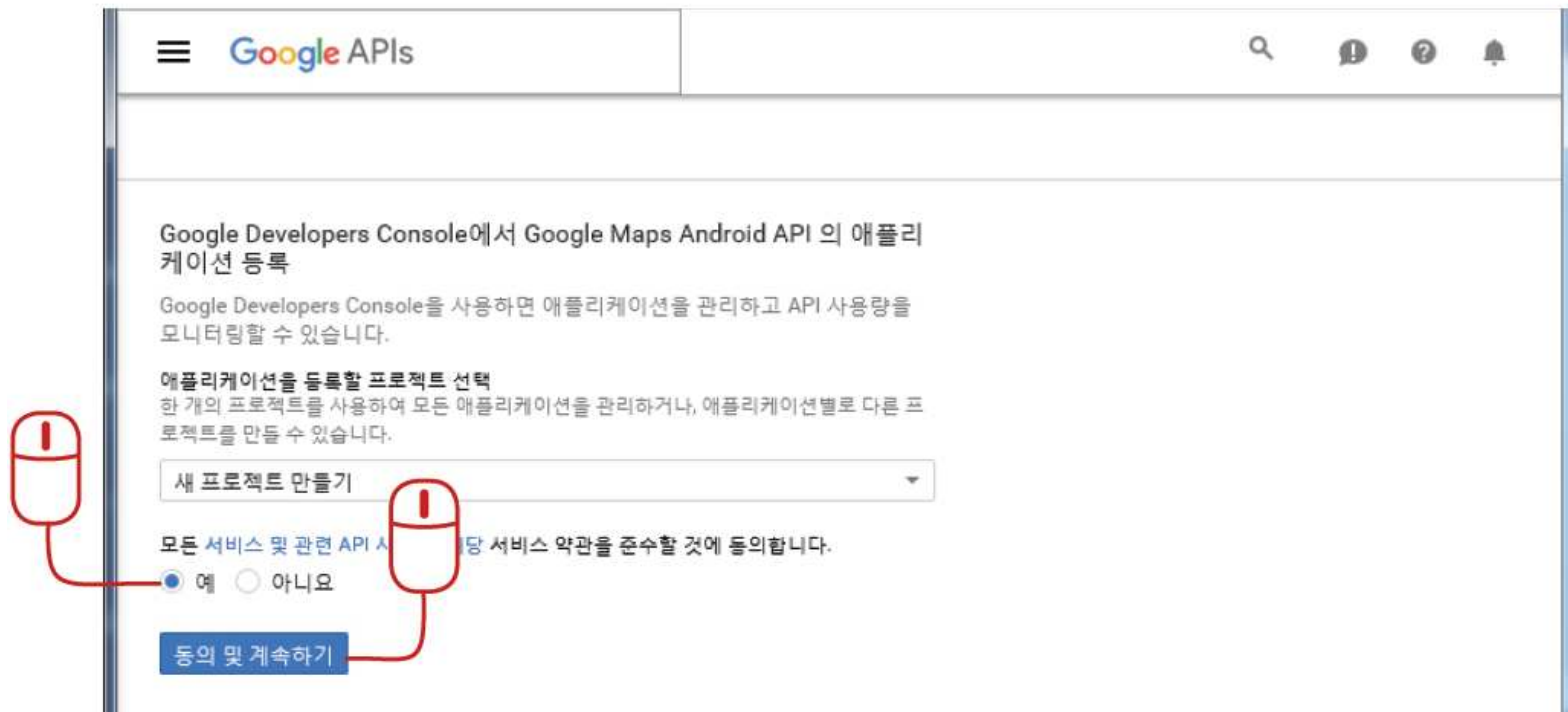
res

```
<!--  
  TODO: Before you run your application, you need a Google Maps API key.  
  
  To get one, follow this link, follow the directions and press "Create" at the end:  
  
  https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID  
  
  You can also add your credentials to an existing key, using this line:  
  1A:00:80:8E:1A:00:80:8E:1A:00:80:8E:1A:00:80:8E:1A:00:80:8E:1A:00:80:8E com.example.googlemap  
  
  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">  
</string>  
</resources>
```

SHA-1 지문

url (SHA-1 지문 포함)
https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID&r=SHA-1지문com.example.googlemap

- 웹브라우저에서 URL로 이동(Google API 사이트) → 약관 동의를 위해 '예'를 체크하고 '동의 및 계속하기' 버튼 클릭



■ API 사용을 위해 '사용자 인증 정보로 이동' 버튼 클릭



- API 키 이름을 디폴트로 지정된 이름으로 사용. '생성' 버튼 클릭

The screenshot shows the '사용자 인증 정보' (Credentials) page in the Google APIs console. The left sidebar has 'API 관리자' (API Manager) and '사용자 인증 정보' (Credentials). The main area is titled '사용자 인증 정보' and contains a section for 'Android API 키 생성' (Generate Android API key). A blue callout points to the '이름' (Name) field, which contains 'Android 키 1', stating 'API 키 이름을 디폴트로 지정된 'Android 키 1' 사용' (Use the default API key name 'Android 키 1'). Below this is a section for 'Android 앱의 사용량 제한 (선택사항)' (Usage limits for Android apps), which includes instructions and a terminal command: `keytool -list -v -keystore <keystore>.keystore`. Two blue callouts point to the '패키지 이름' (Package name) and 'SHA-1 인증서 지문' (SHA-1 certificate fingerprint) fields. The package name field contains 'com.example.googlemap' and is labeled '패키지 이름'. The fingerprint field contains '1A:0D:80:8' and is labeled 'SHA-1 인증서 지문'. Below these fields is a button labeled '+ 패키지 이름 및 지문 추가' (+ Add package name and fingerprint). At the bottom, there is a note: '참고: 설정이 적용되는 데 최대 5분이 걸릴 수 있습니다.' (Note: It may take up to 5 minutes for the settings to be applied). A red mouse cursor icon points to the '생성' (Generate) button, which is next to a '취소' (Cancel) button.

google APIs

My Project

API 관리자

사용자 인증 정보

Android API 키 생성

이름

Android 키 1

API 키 이름을 디폴트로 지정된 'Android 키 1' 사용

Android 앱의 사용량 제한 (선택사항)

Add your package name and SHA-1 signing-certificate fingerprint to restrict usage to your Android apps [Learn more](#)

Get the package name from your AndroidManifest.xml file. Then use the following command to get the fingerprint:

```
keytool -list -v -keystore <keystore>.keystore
```

패키지 이름

SHA-1 인증서 지문

패키지 이름

SHA-1 인증서 지문

com.example.googlemap

1A:0D:80:8

+ 패키지 이름 및 지문 추가

참고: 설정이 적용되는 데 최대 5분이 걸릴 수 있습니다.

생성 취소

■ 생성된 API 키 확인

The screenshot shows the Google APIs console interface. The left sidebar contains a menu with 'API 관리자' (API Manager) and '사용자 인증 정보' (Credentials). The main area is titled '사용자 인증 정보' (Credentials) and has tabs for '사용자 인증 정보', 'OAuth 동의 화면', and '도메인 확인'. Under '사용자 인증 정보', there is a '사용자 인증 정보 만들기' (Create Credentials) button and a '삭제' (Delete) button. Below this, a message states: '사용 설정한 API에 액세스하려면 사용자 인증 정보를 만드세요. 자세한 내용은 API 문서를 참조하세요.' (To access the APIs you've enabled, create credentials. For more details, see the API documentation.)

The 'API 키' (API Keys) section is expanded, showing a table with the following columns: '이름' (Name), '생성일' (Creation Date), '유형' (Type), and '키' (Key). The table contains one entry:

이름	생성일	유형	키
Android 키 1	2016. 4. 12.	Android	AlzaSyA_P_yTB4sA7GzxOpp1MPbsYAbkUAxuFP0

A red rectangular box highlights the API key 'AlzaSyA_P_yTB4sA7GzxOpp1MPbsYAbkUAxuFP0'. A blue callout bubble with the text '생성된 API 키' (Generated API Key) points to this key.

● 발급받은 API 키 복사

'Google_maps_api.xml' 파일의 'YOUR_KEY_HERE' 부분을 발급받은 API 키로 대체

소스 | google_maps_api.xml

```
01 .....  
02     <string name="google_maps_key" templateMergeStrategy=  
        "preserve" translatable="false">YOUR_KEY_HERE</string>.
```

소스 | google_maps_api.xml

```
01 .....  
02     <string name="google_maps_key" templateMergeStrategy="preserve"  
        translatable="false">AIzaSyA.....UAXuFP0</string>  
03 .....
```

● 구글맵 관련 파일의 이해

① 텍스트 리소스

- Strings.xml 수정하지 않는다
 - app_name 속성값에 어플리케이션 라벨로,
 - Title_activity_maps는 액티비티 라벨로 사용

소스 | strings.xml

```
01 <resources>
02     <string name="app_name">Google Map</string>
03     <string name="title_activity_maps">Map</string>
04 </resources>
```


2 구글맵 API 키 관리

소스 | google_maps_api.xml

```
01 <resources>
02     <!--
03     TODO: Before you run your application, you need a Google Maps API key.
04
05     To get one, follow this link, follow the directions and press "Create"
                                at the end:
06
07     https://console.developers.google.com/flows/enableapi?apiid=
    maps\_android\_backend&keyType=CLIENT\_SIDE\_ANDROID&r=1A:0D:80:8E:23:8B:B6:BC:
    4C:FC:A2:7A:18:B9:17:6C:31:54:DA:2E%3Bcom.example.googlemap
08
09     You can also add your credentials to an existing key, using this line:
10     1A:0D:80:8E:23:8B:B6:BC:4C:FC:A2:7A:18:B9:17:6C:31:54:DA:2E;
                                com.example.googlemap
11
12     Alternatively, follow the directions here:
13     https://developers.google.com/maps/documentation/android/start#get-key
14
15     Once you have your key(it starts with "AIza"),
                                replace the "google_maps_key"
16     string in this file.
17     -->
18     <!--
19     <string name="google_maps_key" templateMergeStrategy="preserve"
                                translatable="false">YOUR_KEY_HERE</string>
20     -->
21     <string name="google_maps_key" templateMergeStrategy="preserve"
                                translatable="false">AIzaSyA_P.....UAXuFP0</string>
22 </resources>
```

발급받은 구글맵 API 키를 등록

3 구글맵 화면설계

소스 | activity_maps.xml → A

```
01 <fragment xmlns:android="http://schemas.android.com/apk/res/android"
02     xmlns:map="http://schemas.android.com/apk/res-auto"
03     xmlns:tools="http://schemas.android.com/tools"
04     android:id="@+id/map" → B
05     android:name="com.google.android.gms.maps.SupportMapFragment"
06     android:layout_width="match_parent"
07     android:layout_height="match_parent"
08     tools:context="com.example.googlemap.MapsActivity" />
```

지도를 출력하는 영역

지도를 출력하기 위해 사용하는 클래스

4 액티비티 제어

소스 | MapsActivity.java

```
13 public class MapsActivity extends FragmentActivity
                                implements OnMapReadyCallback {
14
15     private GoogleMap mMap;
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.activity_maps);
21         SupportMapFragment mapFragment = (SupportMapFragment)
22             getSupportFragmentManager().findFragmentById(R.id.map);
23         mapFragment.getMapAsync(this);
24     }
25
26     @Override
27     public void onMapReady(GoogleMap googleMap) {
28         mMap = googleMap;
29
30         // Add a marker in Sydney and move the camera
31         LatLng sydney = new LatLng(-34, 151);
32         mMap.addMarker(new MarkerOptions().position(sydney).title(
33             "Marker in Sydney"));
34         mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
35     }
36 }
```

A id가 map인 프래그먼트를 위한 객체 생성

B 구글맵 객체가 사용 준비될 때 작동할 콜백 객체 생성

앱이 사용 준비가 될 때 호출되는 콜백 메소드

호주 시드니의 위도와 경도

시드니를 중심으로 하는 지도 출력

시드니 위치에 마커 추가.
title은 마우스를 마커에 오버랩시키면 나타남.

Class와 속성/메소드

● 클래스

클래스/인터페이스	설명
CameraUpdateFactory	33행. 지도를 변화하는 CameraUpdate 객체를 생성함
FragmentActivity	13행. 액티비티 내의 fragment와 상호작용을 위한 인터페이스
FragmentManager	액티비티 내에서 fragment 객체 간 상호 작용을 위한 인터페이스
GoogleMap	15행. Google Map Android API의 주 클래스로 지도 관련 모든 메소드를 가지고 있음
LatLng	31행. 위도와 경도 좌표를 표현하는 클래스
OnMapReadyCallback	13행. 맵이 사용될 준비가 될 때 호출되는 콜백 인터페이스
MarkerOptions	32행. 마커를 위한 마커 옵션 설정
SupportMapFragment	21행. 안드로이드 3.0(API 레벨 11)부터 사용 가능한 Fragment 클래스의 서브 클래스로 더 낮은 버전에서도 사용 가능

● 메소드

클래스	메소드	설명
FragmentActivity	FragmentManager <code>getSupportFragmentManager()</code>	22행. 액티비티와 연관된 fragment와 상호작용을 위한 FragmentManager를 반환함
FragmentManager	abstract Fragment <code>findFragmentById(int id)</code>	22행. 주어진 id에 해당하는 fragment를 찾음
OnMapReadyCallback	abstract void <code>onMapReady(GoogleMap googleMap)</code>	27행. 맵이 사용될 준비가 될 때 호출됨
MarkerOptions	MarkerOptions <code>position</code> (LatLng position)	32행. 마커의 위치 설정
	MarkerOptions <code>title</code> (String title)	32행. 마커의 제목 설정
CameraUpdateFactory	static CameraUpdate <code>newLatLng</code> (LatLng latLng)	33행. 위도 경도를 화면의 중심으로 이동하는 CameraUpdate를 반환
GoogleMap	Final Marker <code>addMarker</code> (MarkerOptions options)	32행. 지도에 마커를 추가함
	final void <code>moveCamera</code> (CameraUpdate update)	33행. update 지시대로 카메라를 움직임
LatLng	<code>LatLng</code> (double latitude, double longitude)	31행. 주어진 위도와 경도를 가진 LatLng 생성자
SupportMapFragment	void <code>getMapAsync</code> (OnMapReadyCallback callback)	23행. Googlemap 객체가 사용될 준비가 될 때 작동할 콜백 객체를 설정

5 환경 설정

소스 | AndroidManifest.xml

```
01 <?xml version="1.0" encoding="utf-8"?>
02 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
03     package="com.example.yschang.googlemap">
04
05     <uses-permission android:name="
                                android.permission.ACCESS_FINE_LOCATION" />
06
07     <application
08         android:allowBackup="true"
09         android:icon="@mipmap/ic_launcher"
10         android:label="@string/app_name"
11         android:supportsRtl="true"
12         android:theme="@style/AppTheme">
13
14         <meta-data
15             android:name="com.google.android.geo.API_KEY"
16             android:value="@string/google_maps_key" />
17
18         <activity
19             android:name=".MapsActivity"
20             android:label="@string/title_activity_maps">
21             <intent-filter>
22                 <action android:name="android.intent.action.MAIN" />
```

GPS 사용 허용

STEP 3

프로젝트 실행

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭

19.2.3 구글맵 유형과 UI 컨트롤

일반형 지도



인공위성 지도



하이브리드 지도
(인공위성 지도에 도로와 지형지물 라벨)



지형 지도



STEP 2

파일 편집

● 편집: 액티비티 제어

소스 | MapActivity.java

```
17  @Override
18  protected void onCreate(Bundle savedInstanceState) {
19      super.onCreate(savedInstanceState);
20      setContentView(R.layout.activity_maps);
21      SupportMapFragment mapFragment = (SupportMapFragment)
22                                          getSupportFragmentManager()
23                                          .findFragmentById(R.id.map);
24      mapFragment.getMapAsync(this);
25
26      GoogleMap googleMap = mapFragment.getMap();
27      googleMap.setMapType(GoogleMap.MAP_TYPE_SATELLITE);
28  }
29
30  @Override
31  public void onMapReady(GoogleMap googleMap) {
32      mMap = googleMap;
33
34      LatLng sydney = new LatLng(-34, 151);
35      mMap.addMarker(new MarkerOptions().position(sydney).title(
36                                          "Marker in Sydney"));
37      mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
38  }
```

구글맵 객체 생성

구글맵 유형을 인공위성 지도로 설정

Class와 속성/메소드

● 상수

클래스	메소드	설명
GoogleMap	int <i>MAP_TYPE_HYBRID</i>	하이브리드 지도. 위성사진 데이터와 지형지물 라벨이 같이 표시됨
	int <i>MAP_TYPE_NORMAL</i>	27행. 일반 지도. 전형적인 로드맵. 도로나 강 위에 지형지물 라벨들이 표시됨
	int <i>MAP_TYPE_SATELLITE</i>	위성사진 지도. 위성사진 데이터가 표시되며, 지형지물 라벨들이 보이지 않음
	int <i>MAP_TYPE_TERRAIN</i>	지형 지도. 지형 데이터. 지도에 색, 등고선, 라벨들이 나타남
	int <i>MAP_TYPE_NONE</i>	기본 맵타입이 없어 지도가 나타나지 않음

● 메소드

클래스	메소드	설명
SupportMapFragment	final GoogleMap <i>getMap()</i>	26행. Fragment에 사용될 Googlemap 반환
GoogleMap	final void <i>setMapType</i> (int type)	26행. 출력 될 맵타입의 유형을 설정함

STEP 3**프로젝트 실행**

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭

19.2.4

특정 위치 중심의 구글맵 출력

'카페베네신논현점' 중심의
지도 출력



마커 클릭 시, 정보 창이 나타나고
우측 하단에는 맵툴바가 나타남



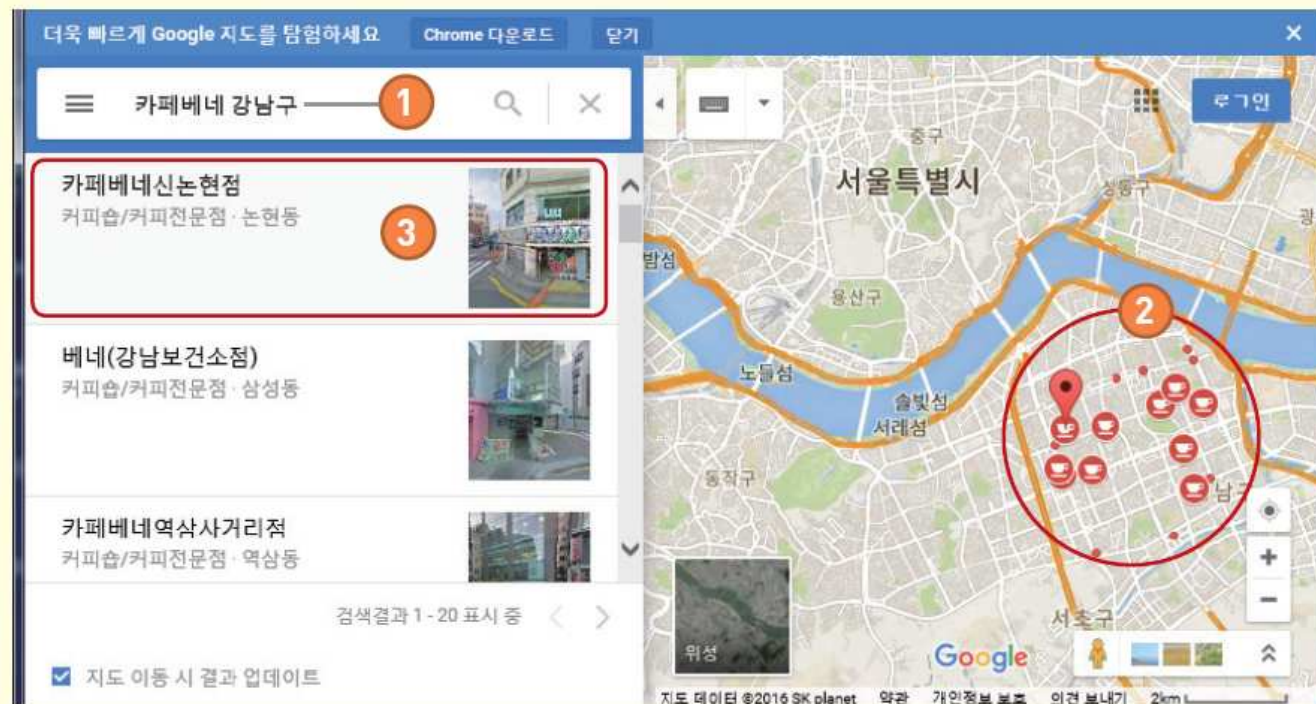
STEP 2

파일 편집

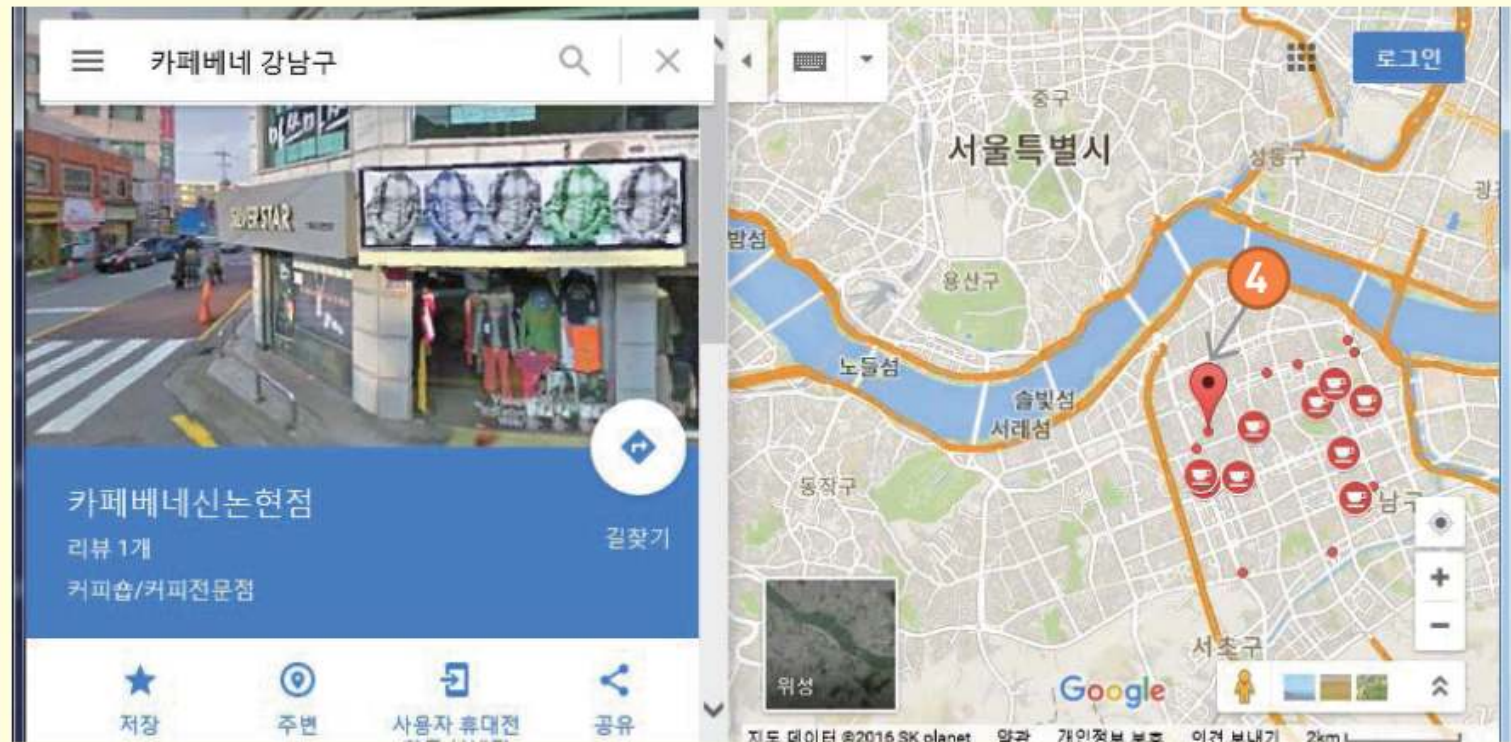
● 편집

① 위도, 경도 좌표 검색

특정지역의 위도와 경도는 구글맵스(maps.google.co.kr)에서 검색할 수 있다. 검색 키워드로 '카페베네 강남구'를 입력하고 검색하면(①), 서울 강남구 지역의 카페베네들의 목록과 지도가 출력된다(②).



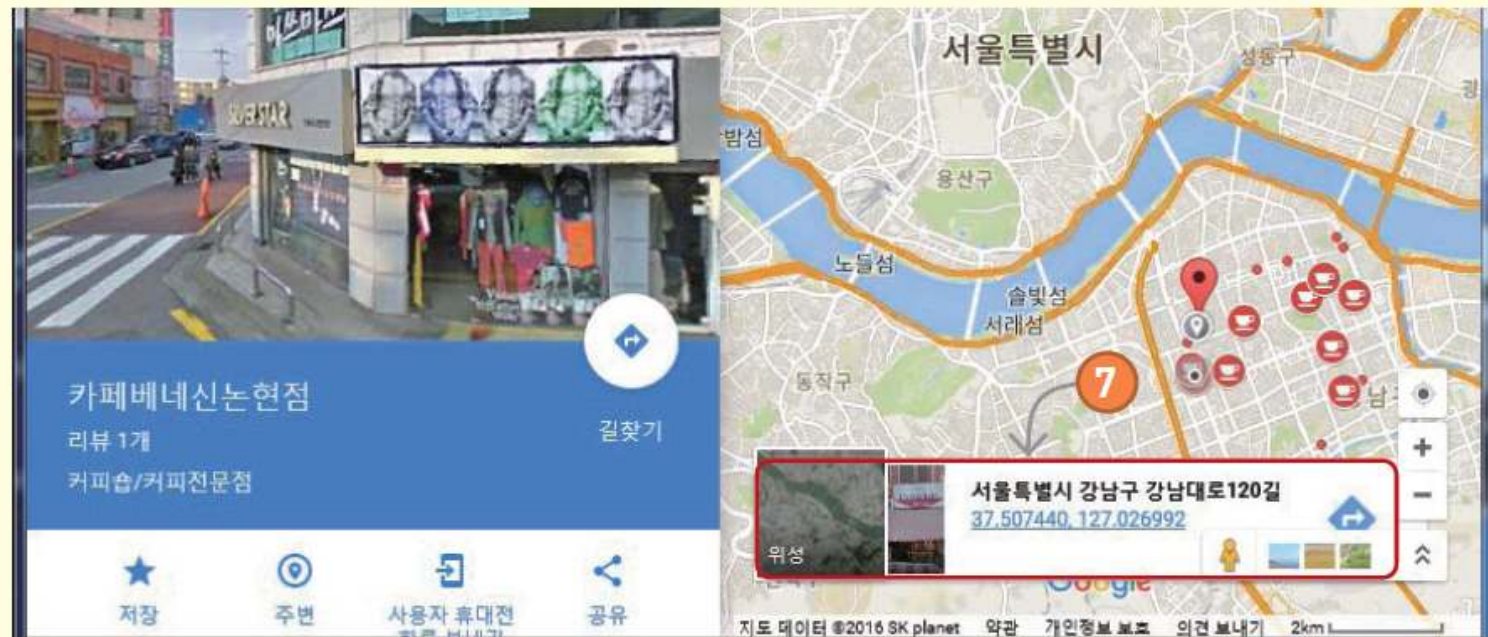
'카페베네신논현점'을 클릭하면(③) 오른쪽에 마커가 표시된다(④).



그 지점에 대해 오른쪽 마우스로 클릭하면 나타나는 팝업메뉴에서 '이곳이 궁금한가?' 메뉴를 클릭한다(⑥).



하단부에 주소와 위도, 경도의 좌표가 나타난다(⑦).



2 액티비티 제어

소스 | MapActivity.java

```
17  @Override
18  protected void onCreate(Bundle savedInstanceState) {
19      super.onCreate(savedInstanceState);
20      setContentView(R.layout.activity_maps);
21      SupportMapFragment mapFragment = (SupportMapFragment)
22                                          getSupportFragmentManager()
23                                          .findFragmentById(R.id.map);
24      mapFragment.getMapAsync(this);
25
26      GoogleMap googleMap = mapFragment.getMap();
27      googleMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);
28  }
29
30  @Override
31  public void onMapReady(GoogleMap googleMap) {
32      mMap = googleMap;
33
34      double latitude = 37.507630;
35      double longitude = 127.026648;
36      LatLng LOC = new LatLng(latitude, longitude);
37
38      mMap.addMarker(new MarkerOptions().position(LOC).title(
39                                          "카페베네신논현점"));
40      mMap.moveCamera(CameraUpdateFactory.newLatLng(LOC));
41      mMap.moveCamera(CameraUpdateFactory.zoomTo(16));
42  }
```

일반형 지도 설정

'카페베네신논현점'의
위도와 경도

위도와 경도에 대한
객체 생성

지도를 16 레벨로 출력(0은 세계지도, 최대 확대는 21)

STEP 3**프로젝트 실행**

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭

19.2.5

Marker와 Info Window

프로젝트

녹색 마커



마커 클릭 시, 제목과 설명의 정보 창 출력



STEP 2

파일 편집

● 편집: 액티비티 제어

소스 | MapsActivity.java

```
30  @Override
31  public void onMapReady(GoogleMap googleMap) {
32      mMap = googleMap;
33
34      double latitude = 37.507630;
35      double longitude = 127.026648;
36      LatLng LOC = new LatLng(latitude, longitude);
37
38      mMap.addMarker(new MarkerOptions()
39          .position(LOC)
40          .title("카페베네신논현점")
41          .snippet("아메리카노 10% 할인")
42          .icon(BitmapDescriptorFactory.defaultMarker(
43              BitmapDescriptorFactory.HUE_GREEN)));
44      mMap.moveCamera(CameraUpdateFactory.newLatLng(LOC));
45      mMap.moveCamera(CameraUpdateFactory.zoomTo(16));
46  }
```

마커 제목 하단부에 스니펫(토막 정보) 추가

마커 색을 녹색으로 변경

Class와 속성/메소드

● 클래스

클래스/인터페이스	설명
BitmapDescriptorFactory	42행. marker 아이콘으로 사용되는 이미지 정의를 위해 사용됨

● 상수

클래스	상수	설명
BitmapDescriptorFactory	public static final float HUE_GREEN	42행. Hue는 색의 특성 중 한 가지로, 상수값 120.0에 해당함. HUE_AZURE, HUE_BLUE, HUE_CYAN, HUE_GREEN, HUE_MAGENTA, HUE_ORANGE, HUE_RED, HUE_ROSE, HUE_VIOLET, HUE_YELLOW가 있음

● 메소드

클래스	메소드	설명
BitmapDescriptorFactory	static BitmapDescriptor defaultMarker (float hue)	42행. 디폴트 marker 이미지의 색을 지정함
MarkerOptions	MarkerOptions snippet(String snippet)	41행. marker에 snippet을 설정함

STEP 3**프로젝트 실행**

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭

19.2.6

Custom Icon으로 만든 Marker 표시

프로젝트

커스텀 아이콘의 마커 표시



마커 클릭 시, 정보 창과 맵툴바 표시



STEP 2

파일 편집

● 편집

1 Custom icon 등록

drawable 폴더	이미지 파일
drawable-xxhdpi	<div></div> <div>coffee.png</div>

2 MainActivity.java 편집

소스 | MapsActivity.java

```
31  @Override
32  public void onMapReady(GoogleMap googleMap) {
33      mMap = googleMap;
34
35      double latitude = 37.507630;
36      double longitude = 127.026648;
37      LatLng LOC = new LatLng(latitude, longitude);
38
39      mMap.addMarker(new MarkerOptions()
40          .position(LOC)
41          .title("카페베네신논현점")
42          .snippet("아메리카노 10% 할인")
43          .icon(BitmapDescriptorFactory.fromResource(
44              R.drawable.coffee)));
45      mMap.moveCamera(CameraUpdateFactory.newLatLng(LOC));
46      mMap.moveCamera(CameraUpdateFactory.zoomTo(16));
47  }
```

마커를 사용자의 커스텀
아이콘(coffee.png)으로 변경



STEP 3**프로젝트 실행**

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭

19.2.7 다수의 Marker 표시

프로젝트

다수 마커와 하나의 정보 창 표시



마커 클릭에 따른 정보 창 표시



● 편집

① Custom icon 등록

Drawable 폴더에서 저장한 coffee.png 파일을 custom icon 으로 사용

2 MainActivity.java 편집

소스 | MapsActivity.java

```
42     LOC = new LatLng(latitude, longitude);
43     Marker mk1 = mMap.addMarker(new MarkerOptions()
44         .position(LOC)
45         .title("카페베네신논현점")
46         .snippet("아메리카노 10% 할인")
47         .icon(BitmapDescriptorFactory.fromResource(
48             R.drawable.coffee)));
49
50     latitude = 37.499485;
51     longitude = 127.026433;
52     LOC = new LatLng(latitude, longitude);
53     Marker mk2 = mMap.addMarker(new MarkerOptions()
54         .position(LOC)
55         .title("카페베네강남대로점")
56         .snippet("대학생 20% 할인")
57         .icon(BitmapDescriptorFactory.fromResource(
58             R.drawable.coffee)));
59
60     mk2.showInfoWindow();
61     mMap.moveCamera(CameraUpdateFactory.newLatLng(LOC));
62     mMap.moveCamera(CameraUpdateFactory.zoomTo(15));
63 }
```

첫 번째 마커 설정

두 번째 마커 설정

두 번째 마커의 정보 창을 출력

Class와 속성/메소드

● 클래스

클래스/인터페이스	설명
Marker	43,52행. 지도 위의 특정 지점에 위치하는 아이콘

● 메소드

클래스	메소드	설명
Marker	void <code>showInfoWindow()</code>	58행. info window를 나타냄. info window는 화면에 하나씩 출력 가능함. 클릭한 marker에 대한 info window가 나타나면, 다른 marker의 info window는 사라짐

STEP 3**프로젝트 실행**

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭

19.2.8 현재 위치 추적

현재 위치 표시



STEP 2

파일 편집

● 편집: 액티비티 제어

소스 | MapsActivity.java

```
38 public void onMapReady(GoogleMap googleMap) {
39     mMap = googleMap;
40
41     LocationManager locationManager = (LocationManager) this.
        getSystemService(Context.LOCATION_SERVICE);
42
43     LocationListener locationListener = new LocationListener() {
44         public void onLocationChanged(Location location) {
45             updateMap(location);
46         }
47
48         public void onStatusChanged(
49             String provider, int status, Bundle extras) {
50             alertStatus(provider);
51         }
52
53         public void onProviderEnabled(String provider) {
54             alertProvider(provider);
55         }
56
57         public void onProviderDisabled(String provider) {
58             checkProvider(provider);
59     };
```

시스템의 위치 서비스에 접근하는 개체 생성

위치가 변할 때 자동으로 호출

provider의 상태가 변할 때 자동으로 호출

사용자에 의해 provider가 사용 가능하게 될 때 자동으로 호출

사용자에 의해 provider가 사용 불가능하게 될 때 자동으로 호출

위치가 변할 때 LocationManager로부터 공지를 받고, 각 메소드를 실행하는 LocationListener객체를 생성

E

D

```
60
61     String locationProvider = LocationManager.NETWORK_PROVIDER;
62
63     locationManager.requestLocationUpdates(
64         locationProvider, 0, 0, locationListener);
65 }
66
67 public void updateMap(Location location) {
68     double latitude = location.getLatitude();
69     double longitude = location.getLongitude();
70     final LatLng LOC = new LatLng(latitude, longitude);
71
72     mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(LOC, 16));
73     Marker mk = mMap.addMarker(new MarkerOptions()
74         .position(LOC)
75         .title("현재 위치"));
76     mk.showInfoWindow();
77 }
78
79 public void checkProvider(String provider) {
80     Toast.makeText(this, provider + "에 의한 위치서비스가 꺼져 있습니다.  
켜주세요...", Toast.LENGTH_SHORT).show();
```

위치 파악을 위한 provider를
NETWORK_PROVIDER로 설정

위치 변경 사항을 요구함


```
81     Intent intent = new Intent(  
            android.provider.Settings.ACTION_LOCATION_SOURCE_SETTINGS);  
82     startActivity(intent);  
83 }  
84  
85 public void alertProvider(String provider) {  
86     Toast.makeText(this, provider + "서비스가 켜졌습니다!",  
            Toast.LENGTH_LONG).show();  
87 }  
88  
89 public void alertStatus(String provider) {  
90     Toast.makeText(this, "위치서비스가 "+provider+"로 변경되었습니다!",  
            Toast.LENGTH_LONG).show();  
91 }  
92 }
```

provider 설정을 위한 인텐트 생성

D provider 설정화면으로 이동

E

Class와 속성/메소드

● 클래스

클래스/인터페이스	설명
Settings	시스템 수준 설정(예: 알림의 활성화 여부 지정 등)을 포함하는 클래스
Location	66,44행. 지리적 위치를 표현하는 데이터 클래스
LocationListener	43행. 위치가 변할 때 LocationManager로부터 공지를 받는데 사용됨
LocationManager	41,61행. 시스템 위치 서비스 접근을 제공

● 상수

클래스	상수	설명
Context	String LOCATION_SERVICE	41행. 위치 업데이트를 제어하기 위한 LocationManager를 추출하기 위해 getSystemService(Class)와 함께 사용
LocationManager	String NETWORK_PROVIDER	61행. Network location provide의 이름. Cell tower(전화 기지국)와 WiFi 액세스 포인트를 결정
Settings	String ACTION_LOCATION_SOURCE_SETTINGS	81행. 현재 위치를 얻기 위한 구성을 허락하는 설정을 보여 줌

● 메소드

클래스	메소드	설명
CameraUpdateFactory	static CameraUpdate newLatLngZoom (LatLng latLng, float zoom)	71행 . 화면 중심의 위치를 LatLng로 주어지는 위 도 경도 지점으로 이동하며 zoom 레벨로 변하는 CameraUpdate를 반환함
GoogleMap	final void animateCamera (CameraUpdate update)	71행 . 현재 위치에서 업데이트될 위치까지 카메라의 이 동을 애니메이션함
LocationListener	abstract void onLocationChanged (Location location)	44행 . 위치가 변할 때 불려짐
	abstract void onStatusChanged (String provider, int status, Bundle extras)	48행 . provider 상태가 변할 때 불려짐
	abstract void onProviderEnabled (String provider)	52행 . 사용자에게 의해 provider가 사용 가능하게 설정 될 때 불려짐
	abstract void onProviderDisabled (String provider)	56행 . 사용자에게 의해 provider가 사용 불가능하게 설 정될 때 불려짐
LocationManager	void requestLocationUpdates (String provider, long minTime, float minDistance, LocationListener listener)	63행 . 주어진 provider를 사용하여 위치 변경 사항을 요구함. minTime은 위치 변경 추적 간격(1/1000초), minDistance는 위치 변경 최소 거리(미터)

STEP 3**프로젝트 실행**

절차	내용
실행메뉴 선택	'Run' 메뉴에서 'Run app' 클릭(또는 'Run app' 아이콘 클릭)
디바이스 선택	스마트폰 디바이스를 선택하고 'OK' 버튼 클릭



Thank you