

22.1 LocationManager 이용

22.1.1. 위치 정보 제공자

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

- LocationManager 획득

```
manager = (LocationManager) getSystemService(LOCATION_SERVICE);
```

LocationProvider

- GPS Provider: GPS 위성을 이용하여 위치 정보 획득
- Network Provider: 이동통신사 망 정보를 이용하여 위치 정보 획득
- Wifi Provider: 와이파이의 AP 정보를 이용하여 위치 정보 획득
- Passive Provider: 다른 앱에서 이용한 마지막 위치 정보 획득

위치 정보 제공자 정보 획득

- getAllProviders(): 스마트폰에서 제공하는 모든 위치 정보 제공자
- getProviders(boolean enabledOnly): 사용 가능한 위치 정보 제공자

```
List<String> providers=manager.getAllProviders();  
for(String provider:providers){  
    //...}  
  
List<String> enabledProviders=manager.getProviders(true);  
for(String provider:enabledProviders){  
    //...}
```



22.1 LocationManager 이용

Criteria

- setAccuracy(int accuracy): 정확도에 대한 조건
- setAltitudeRequired(boolean altitudeRequired): 고도 제공
- setBearingRequired(boolean bearingRequired): 방향 제공
- setCostAllowed(boolean costAllowed): 비용이 드는 것을 허용할 것인지
- setSpeedRequired(boolean speedRequired): 속도 제공
- setPowerRequirement(int level): 전원 소모량 조건

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

22.1.2. 위치 정보 획득

```
Location location=manager.getLastKnownLocation(provider);
```



22.1 LocationManager 이용

Location

- getAccuracy(): 정확도
- getAltitude(): 고도
- getBearing(): 방위
- getLatitude(): 위도
- getLongitude(): 경도
- getProvider(): 위치 정보 제공자
- getSpeed(): 속도
- getTime(): 획득 시간

LocationListener

```
LocationListener listener=new LocationListener() {  
    public void onStatusChanged(String provider, int status, Bundle extras) {  
    }  
    public void onProviderEnabled(String provider) {  
    }  
    public void onProviderDisabled(String provider) {  
    }  
    public void onLocationChanged(Location location) {  
    }  
};
```



22.1 LocationManager 이용

- LocationListener를 LocationManager에 등록

```
manager.requestLocationUpdates(bestProvider, 10000, 10, listener);
```

- 등록을 해제

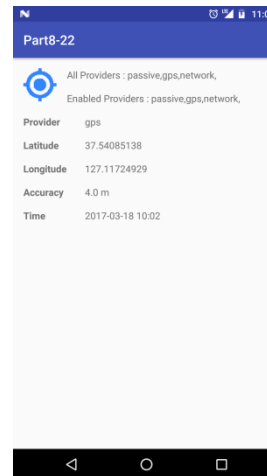
```
manager.removeUpdates(listener);
```



Step by Step 22-1 - LocationManager

LocationManager를 이용한 위치 정보 획득을 테스트

1. Module 생성
2. 파일 복사
3. AndroidManifest.xml 작업
4. MainActivity.java 작성
5. 실행



22.2 Fused Location API 이용

2013년 Google IO 행사에서 발표된 API

위치 정보 제공자를 이용해야 하는지에 대한 최적의 알고리즘을 제공

- 낮은 전력 소모
- 정확도 향상
- 간단한 APIs
- 멋지고 새로운 기능 도입
- 대부분 안드로이드 기기에서 사용 가능

| |  GPS |  WiFi |  Cell |  Sensors |
|----------|--|---|---|--|
| Power |  |  |  |  |
| Accuracy |  |  |  |  |
| Coverage |  |  |  |  |

implementation 'com.google.android.gms:play-services:11.2.0'



22.2 Fused Location API 이용

- FusedLocationProviderApi: 위치 정보 획득
- GoogleApiClient: 위치 정보 제공자 이용 준비, 다양한 콜백 제공
- GoogleApiClient.ConnectionCallbacks: 위치 정보 제공자가 사용 가능 상태가 되었을 때 호출되는 onConnected() 함수와 사용 불가능 상태가 되었을 때 호출되는 onConnectionSuspended() 함수를 가지는 인터페이스
- GoogleApiClient.OnConnectionFailedListener: 위치 정보 제공자를 얻지 못할 때 호출되는 onConnectionFailed() 함수를 가지는 인터페이스

```
public class Lab2Activity extends AppCompatActivity implements GoogleApiClient.ConnectionCallbacks,
GoogleApiClient.OnConnectionFailedListener {
    @Override
    public void onConnected(@Nullable Bundle bundle) {

    }

    @Override
    public void onConnectionSuspended(int i) {

    }

    @Override
    public void onConnectionFailed(@NonNull ConnectionResult connectionResult) {

    }
}
```


22.2 Fused Location API 이용

- GoogleApiClient, FusedLocationProviderApi의 객체를 준비

```
apiClient = new GoogleApiClient.Builder(this)  
    .addApi(LocationServices.API)  
    .addConnectionCallbacks(this)  
    .addOnConnectionFailedListener(this)  
    .build();  
  
providerApi = LocationServices.FusedLocationApi;
```

- 위치 정보 제공자를 준비

```
apiClient.connect();
```

- 위치 값 획득

```
public void onConnected(@Nullable Bundle bundle) {  
    Location location = providerApi.getLastLocation(apiClient);  
}
```



22.2 Fused Location API 이용

- LocationListener

```
import com.google.android.gms.location.LocationListener; //... @Override
public void onConnected(@Nullable Bundle bundle) {
    LocationRequest locationRequest = LocationRequest.create();
    locationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    locationRequest.setInterval(3000);

    providerApi.requestLocationUpdates(apiClient, locationRequest, listener);
}

LocationListener listener=new LocationListener() {
    @Override
    public void onLocationChanged(Location location) {

    }
};
```



Step by Step 22-2 – FusedLocation API

Fused Location API 테스트

Step1 에서 표준 라이브러리의 LocationManager로 획득했던 위치 정보를 Fused Location API로 획득해 정보를 화면 출력

1. Activity 생성
2. 파일 복사
3. play service 라이브러리 등록
4. Lab22_2Activity 작성
5. Lab22_2Activity.java 실행

