

모바일 시스템 프로그래밍

05 Wifi Manager 1

2017 1학기

강승우

WifiManager 클래스

- Wi-Fi connectivity와 관련한 모든 사항을 관리하는 주요 API 제공
 - 주요 기능
 - 설정된 Wi-Fi 네트워크의 목록을 보여줌
 - 현재 active Wi-Fi 네트워크에 대한 정보 제공, 연결을 맺거나 끊을 수 있고, 네트워크의 상태에 대한 동적인 정보를 제공
 - Access Point를 스캔하고 그 결과 정보를 제공
 - Wi-Fi 상태 변화에 따라 전송되는 broadcast에 대한 intent action 이름을 정의
 - SensorManager, LocationManager와 마찬가지로 getSystemService() 메소드를 이용하여 액세스
 - WifiManager wm = (WifiManager) getSystemService(Context.WIFI_SERVICE)

<https://developer.android.com/reference/android/net/wifi/WifiManager.html?hl=ko>

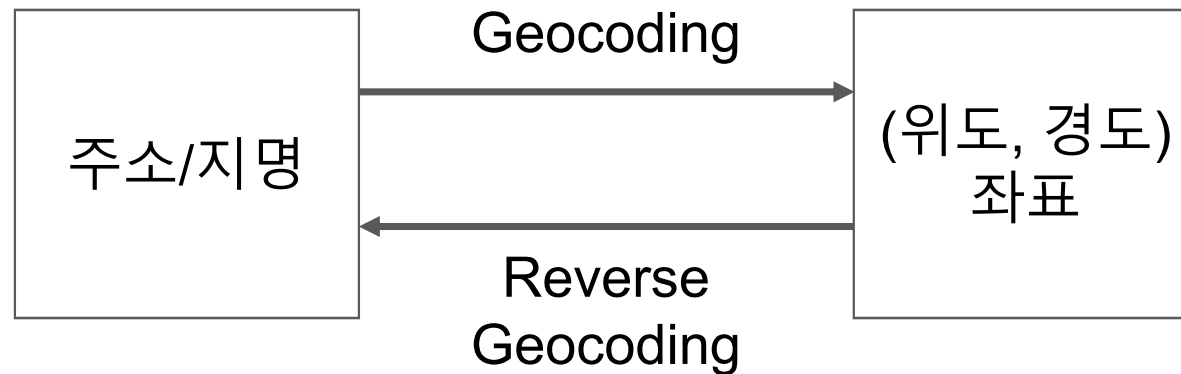
본 수업에서 WifiManager의 이용

- WiFi AP 정보를 이용한 실내 위치 인식
 - 건물 안에서 내 주변의 WiFi AP를 스캔하여 건물 내의 나의 위치를 알아낸다
- 실내 위치 정보의 활용
 - 대형 실내 쇼핑몰, 실내 테마파크 등의 장소에서
 - 실내 네비게이션, 위치 기반 광고, 위치 기반 검색 등의 서비스 제공
- LocationManager로 제공되는 위치 정보를 사용하면 안 되나?

LocationManager를 이용한 위치 정보

- Location 클래스로 제공되는 위치 관련 데이터
 - latitude, longitude, timestamp
 - altitude, velocity 등의 부가 정보
- 위도, 경도 데이터는 사람이 이해할 수 있는 위치/장소 정보가 아님

지오코딩 (Geocoding) / 역지오코딩 (Reverse Geocoding)



1600 Pennsylvania Ave NW Washington DC 20502
↔ (38.898748, -77.037684)

구글 - Google 지도

← → ↺ 안전함 | <https://www.google.co.kr/maps/place/구글/@37.42237,-122.0865785,17z/data=!4r> ☆

앱 ★ Bookmarks Daum 사전 Dkt 네이버 사전 Dictionary.com Corpus of Contempo Daum 지도 Google 지도 >> 기타 북마크

구글

구글
Googleplex

4.4 ★★★★★ · 리뷰 4,451개

기업 단지

저장 주변 사용자 휴대전화로 보내기 공유

1600 Amphitheatre Pkwy, Mountain View, CA 94043 미국

google.com

+1 650-253-0000

수정 제안하기

라벨 추가

구글 사커 필드
Google Soccer Field

Amphitheatre Pkwy

구글 메인 샌드 코트

구글

Google Bldg 40

Google Bldg 41

Google Bldg 42

Google Bldg 43

Google Bldg 1900

Google Bldg 1950

Google Bldg 1098

Google Bldg 44

찰스턴 공원
Charleston Park

찰스턴 로드

위성

지도 데이터 ©2017 Google 약관 의견 보내기 100m

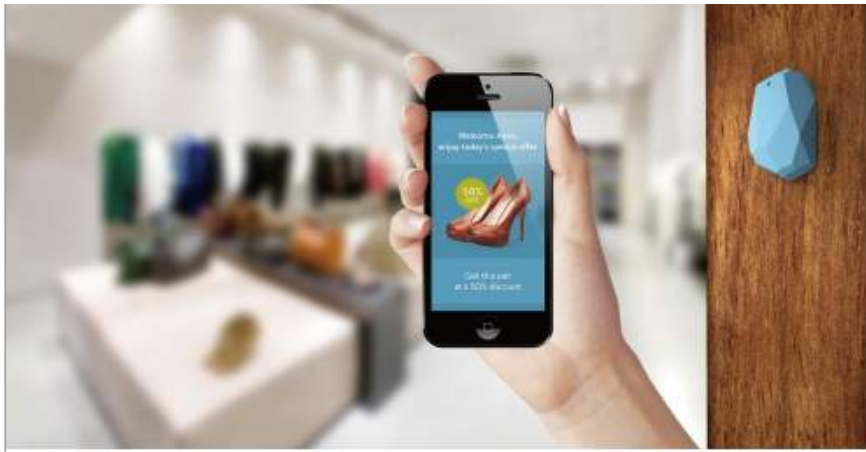
LocationManager를 이용한 위치 정보

- 실내의 경우 GPS 위성 신호를 수신할 수 없기 때문에 정확한 위치(위도, 경도) 정보를 알 수 없음
- 위도, 경도 정보를 안다고 하더라도 건물 내의 위치, 그 장소의 이름(A312호, 학부사무실) 등 자세한 정보는 알기 어려움

실내 위치 인식 방법

- 비콘 이용

- BLE (bluetooth low energy) 기반의 비콘 (예: iBeacon)을 매장 같은 장소에 설치해 두고 비콘이 송신하는 메시지를 모바일 앱에서 수신하여 감지



http://www.hellot.net/new_hellot/magazine/magazine_read.html?code=202&sub=003&idx=21375



<http://blog.lgcns.com/565>

실내 위치 인식 방법

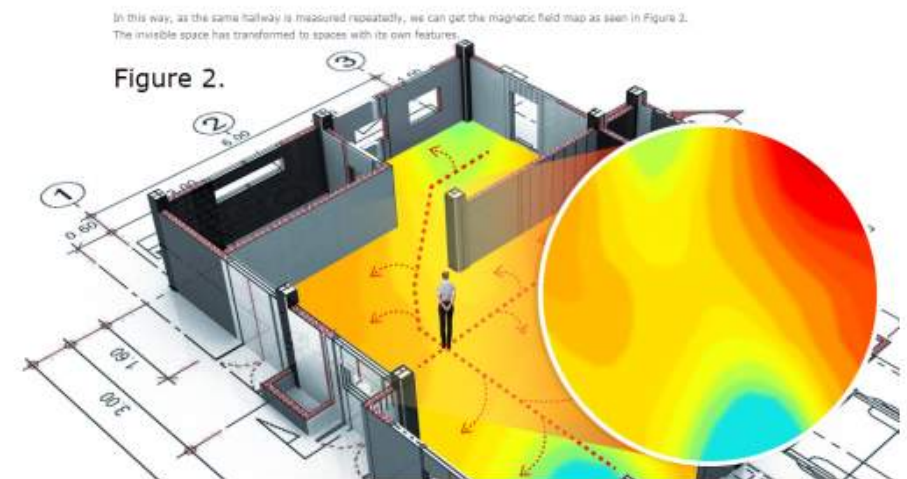
- 소리 이용
 - 사람의 귀로 들을 수 없는 20khz 초음파를 이용
 - 예: 스타벅스 사이렌오더
 - 사용자의 모바일 앱에서 마이크를 이용하여 초음파를 감지하여 매장에 들어왔는지 알아냄



<http://www.bloter.net/archives/226643>

실내 위치 인식 방법

- 건물 내부 자기장의 왜곡 현상 이용
 - 자기장, 가속도, 자이로스코프 등의 센서 사용
 - 건물 내부의 자기장의 왜곡 패턴을 사전에 수집
 - 사용자의 모바일에서 이를 센싱하여 사전에 수집된 데이터와 비교하여 알아냄



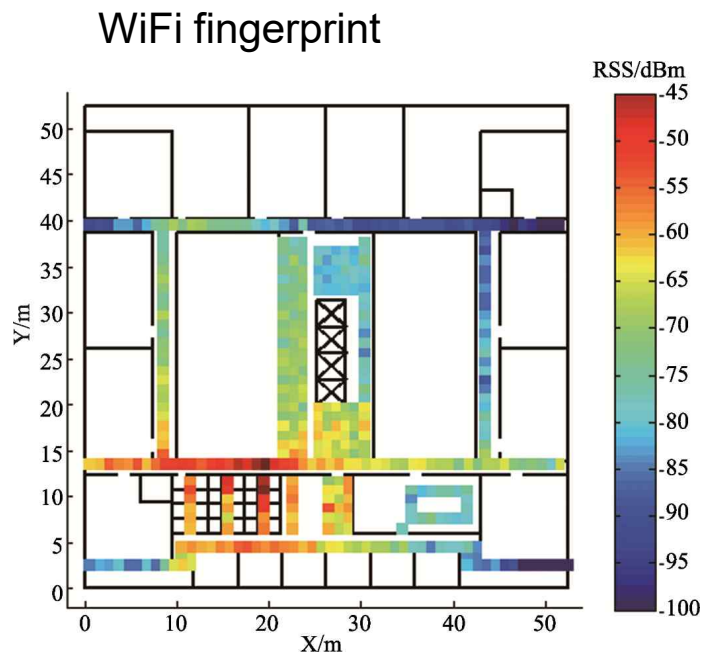
(사진=아이데카)

실내 위치 인식 방법

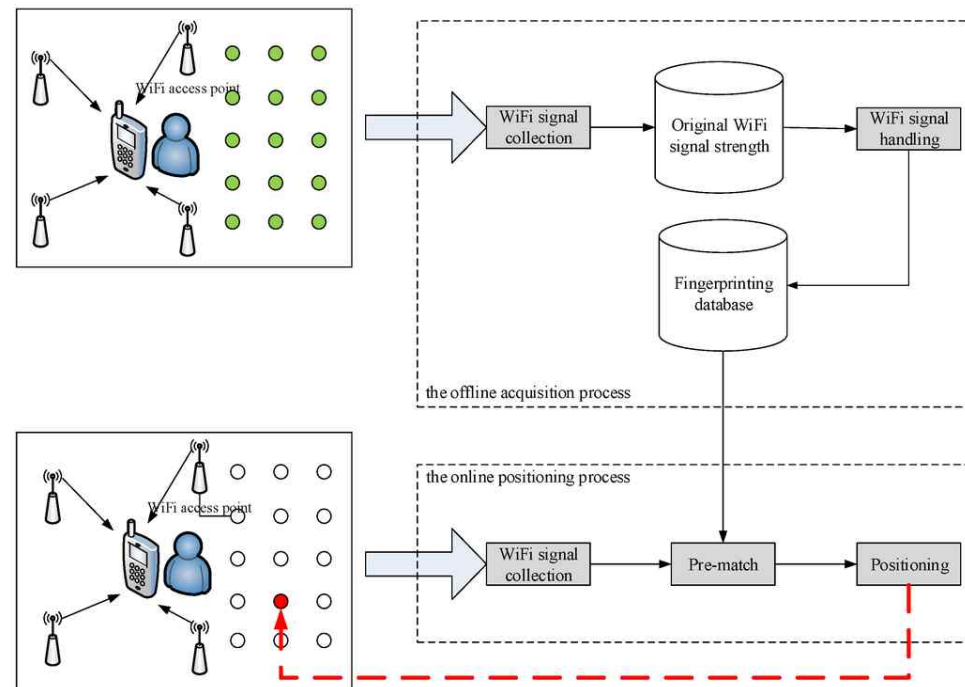
- WiFi AP 정보 이용

- AP에서 전송하는 신호의 세기, 실내 위치 정보 등을 사전에 수집

- 일종의 맵 데이터베이스를 구축하고 현재 수신된 신호 정보와 수집된 정보를 비교하여 인식



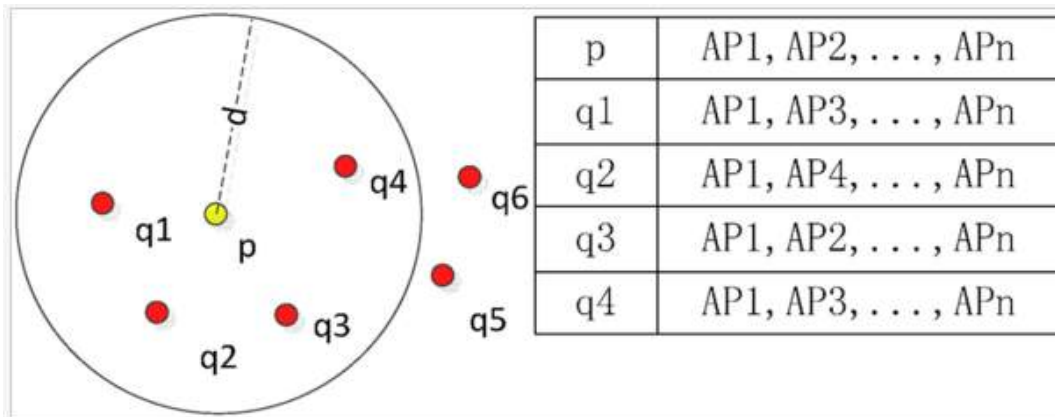
<http://www.progressingeography.com/article/2015/1007-6301/36387>



<http://www.mdpi.com/1424-8220/15/9/21824/htm>

WiFi fingerprint

- 위치에 따른 AP와 신호 강도에 대한 정보



SSID	BSSID	RSSI	Channel	Time
AP1	"C4:01:7C:52:6C:C8"	"-72"	"2"	"2:51:07 PM"
AP2	"C4:01:7C:52:6C:CC"	"-74"	"100"	"2:51:07 PM"
AP3	"C4:01:7C:53:9D:68"	"-73"	"3"	"2:51:14 PM"
AP4	"E0:10:7F:11:EE:58"	"-91"	"7"	"2:51:07 PM"
KUTAP	"F0:B0:52:52:6B:18"	"-77"	"2"	"2:51:14 PM"
mywifi	"F0:B0:52:59:EA:28"	"-84"	"10"	"2:51:17 PM"

WifiManager를 이용한 WiFi AP 스캔

- WifiManager 객체 얻기

```
WifiManager wm = (WifiManager) getSystemService(WIFI_SERVICE);
```

- 스캔 결과를 받기 위한 BroadcastReceiver 선언 및 등록

```
BroadcastReceiver wifiReceiver = new BroadcastReceiver() {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        if (intent.getAction().equals(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION)) {  
            getWifiInfo();  
        }  
    }  
};  
IntentFilter filter = new IntentFilter();  
filter.addAction(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION);  
registerReceiver(wifiReceiver, filter);
```

- 스캔 시작

```
wm.startScan();
```

WifiManager를 이용한 WiFi AP 스캔

- 브로드캐스트로 받은 스캔 결과 처리

```
public void getWifiInfo() {  
    apList = wm.getScanResults();  
    if (wm.getScanResults() != null) {  
        int size = apList.size();  
        for (int i = 0; i < size; i++) {  
            scanResult = (ScanResult) apList.get(i);  
            // scanResult.SSID  
            // scanResult.BSSID  
            // scanResult.level  
        }  
    }  
}
```

- permission

- <uses-permission android:name="android.permission.ACCESS_WIFI_STATE">
- <uses-permission android:name="android.permission.CHANGE_WIFI_STATE">

WifiManager를 이용한 WiFi AP 스캔

- 스캔 결과 얻기

`public List<ScanResult> getScanResults ()`

- ScanResult 클래스
 - public fields: BSSID, SSID, frequency, level, timestamp,
- 유효한 결과를 얻기 위해서는 ACCESS_COARSE_LOCATION or ACCESS_FINE_LOCATION permission 필요
- SSID
 - 무선 네트워크 식별명: KUTAP, ollehWifi, iptime, ...
- BSSID
 - MAC 주소: 0a:11:e3:98:f1:bb
- RSSI (Received Signal Strength Indication)
 - 신호 수신 강도: -39 dBm
- timestamp
 - 스캔 결과를 얻은 시점

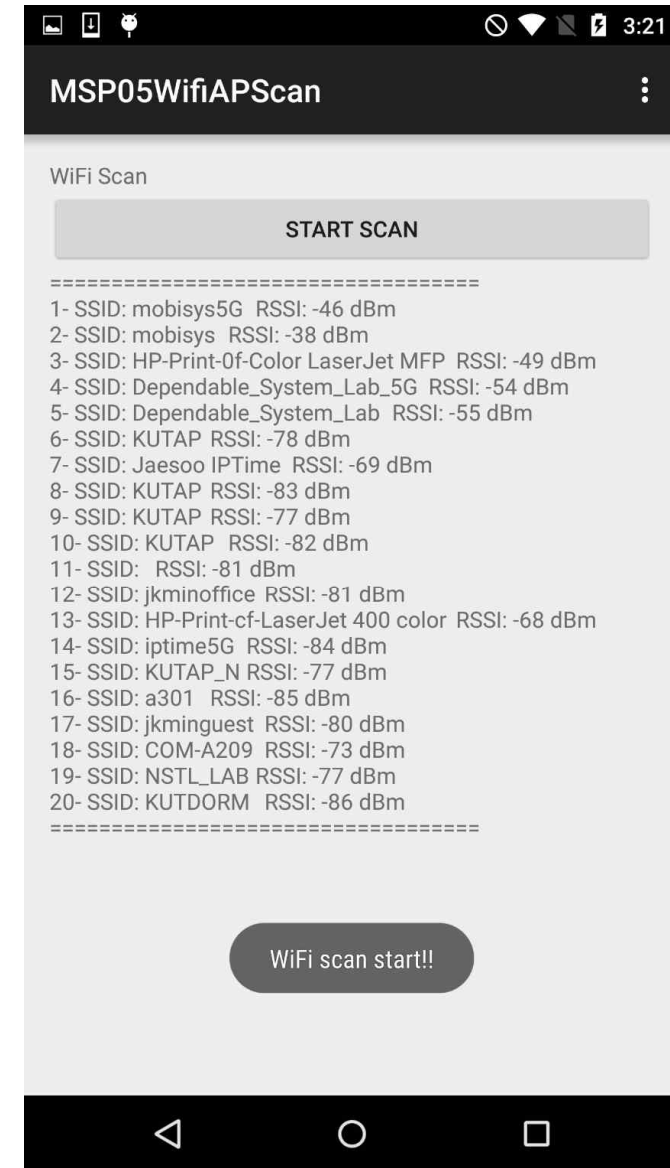
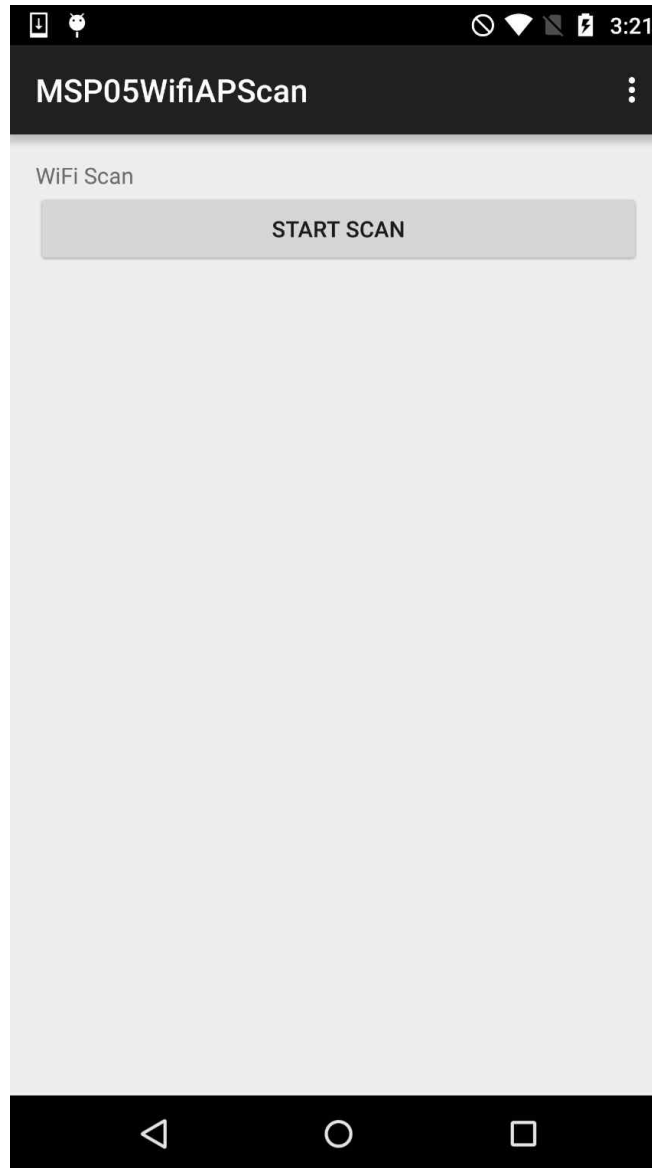
<https://developer.android.com/reference/android/net/wifi/ScanResult.html>

WifiManager를 이용한 WiFi AP 스캔

- 그 외
 - boolean isWifiEnabled()
 - boolean isScanAlwaysAvailable()
 - boolean setWifiEnabled(boolean enabled)

스캔 예제

- 예제 프로젝트 이름
 - MSP05WifiAPScan



MainActivity.java code snippet

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
    requestRuntimePermission();  
    scanResultText = (TextView)findViewById(R.id.result);  
    wifiManager = (WifiManager)getApplicationContext().getSystemService(WIFI_SERVICE);  
  
    // wifi가 활성화되어있는지 확인 후 꺼져 있으면 켜다  
    if(wifiManager.isWifiEnabled() == false)  
        wifiManager.setWifiEnabled(true);  
}
```

@Override

```
protected void onResume() {  
    super.onResume();  
    // wifi scan 결과가 나왔을 때 전송되는 broadcast를 받기 위해  
    // IntentFilter 객체를 생성하고 이를 이용하여 BroadcastReceiver 객체를 등록한다  
    IntentFilter filter = new IntentFilter(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION);  
    registerReceiver(mReceiver, filter);  
}
```

@Override

```
protected void onPause() {  
    super.onPause();  
    unregisterReceiver(mReceiver);  
}
```

MainActivity.java code snippet

```
public void onClick(View view) {  
    if(view.getId() == R.id.start) {  
        Toast.makeText(this, "WiFi scan start!!", Toast.LENGTH_LONG).show();  
  
        if(isPermitted) {  
            // wifi 스캔 시작  
            wifiManager.startScan();  
  
            scanResultText.setText("");  
        } else {  
            Toast.makeText(getApplicationContext(),  
                "Location access 권한이 없습니다..", Toast.LENGTH_LONG).show();  
        }  
    }  
}
```

MainActivity.java code snippet

```
BroadcastReceiver mReceiver = new BroadcastReceiver() {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        String action = intent.getAction();  
        if (action.equals(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION))  
            getWifiInfo();  
    }  
};  
  
private void getWifiInfo() {  
    scanResultList = wifiManager.getScanResults();  
    scanResultText.setText("=====\n");  
    for(int i = 0; i < scanResultList.size(); i++) {  
        ScanResult result = scanResultList.get(i);  
        scanResultText.append((i+1) + "- SSID: " + result.SSID + "\t RSSI: " + result.level + " dBm\n");  
    }  
    scanResultText.append("=====");  
}
```

실습

- 실내의 여러 위치에서 AP 스캔을 해서 AP와 RSSI 값을 관찰해 본다.
- 예
 - 강의실 내의 9 곳
 - 위치에 따라 어떻게 변하는지
 - 혹은 같은 위치라도 시간에 따라 어떻게 변하는지

