

08

08-1. 네트워크 정보 활용

네트워크 프로그래밍

18-1 스마트폰 정보 구하기

전화 상태 변화 감지하기 – TelephonyCallback

- 안드로이드 12 버전부터는 스마트폰의 각종 상태 변화를 감지할 때 TelephonyCallback 이용

• 스마트폰 상태 퍼미션

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

• TelephonyCallback을 이용한 전화 상태 변화 감지

```
telephonyManager = getSystemService(Context.TELEPHONY_SERVICE) as TelephonyManager
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.S) {
    telephonyManager.registerTelephonyCallback(
        mainExecutor,
        object : TelephonyCallback(), TelephonyCallback.CallStateListener {
            override fun onCallStateChanged(state: Int) {
                when (state) {
                    TelephonyManager.CALL_STATE_IDLE -> {
                        Log.d("kkang", "IDLE")
                    }
                    TelephonyManager.CALL_STATE_OFFHOOK -> {
                        Log.d("kkang", "OFFHOOK")
                    }
                    TelephonyManager.CALL_STATE_RINGING -> {
                        Log.d("kkang", "RINGING")
                    }
                }
            }
        })
}
```

18-1 스마트폰 정보 구하기

- 각종 상태 변화를 감지하는 인터페이스
 - TelephonyCallback.CellLocationListener: 셀(cell) 위치 변화 감지
 - TelephonyCallback.ServiceStateListener: 서비스 상태 변화 감지
 - TelephonyCallback.SignalStrengthsListener: 신호 세기 변화 감지
 - TelephonyCallback.DataActivityListener: 데이터 송수신 상태 변화 감지
 - TelephonyCallback.DataConnectionStateListener: 데이터 접속 상태 변화 감지
 - TelephonyCallback.CallStateListener: 전화 상태 변화 감지

18-1 스마트폰 정보 구하기

네트워크 제공 국가, 사업자, 전화번호 얻기 - TelephonyManager

- TelephonyManager는 네트워크 제공 국가, 사업자, 전화번호 등을 반환하는 다음 함수도 제공
 - `getNetworkCountryIso()`: 네트워크 제공 국가
 - `getNetworkOperatorName()`: 네트워크 제공 사업자

18-1 스마트폰 정보 구하기

- 사용자 스마트폰의 전화번호를 추출
 - API Level 33 이전까지는 TelephonyManager의 getLine1Number() 함수를 사용
 - 33 버전부터는 SubscriptionManager의 getPhoneNumber()를 권장

• getPhoneNumber() 함수로 전화번호 추출하기

```
var phoneNumber: String = "unknown"

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.TIRAMISU) {
    val subscriptionManager =
        getSystemService(Context.TELEPHONY_SUBSCRIPTION_SERVICE) as SubscriptionManager

    for (subscriptionInfo: SubscriptionInfo in subscriptionManager.activeSubscription
        InfoList !!) {
        val activeSubscriptionId: Int = subscriptionInfo.subscriptionId
        phoneNumber = subscriptionManager.getPhoneNumber(activeSubscriptionId)
    }
} else {
    //deprecated
    phoneNumber = telephonyManager.line1Number
}
```

• 사용자 전화번호를 읽는 퍼미션

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

18-1 스마트폰 정보 구하기

네트워크 접속 정보 - ConnectivityManager

- 네트워크 상태 접근 퍼미션

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

- `getActiveNetwork()` 함수를 이용하는 방법
 - 스마트폰에서 네트워크가 가능한지, 가능하다면 어떤 네트워크에 접속되었는지 알아볼 때는 `ConnectivityManager`의 `getActiveNetwork()` 함수로 `Network` 객체를 얻어서 이용

18-1 스마트폰 정보 구하기

• 네트워크 접속 가능 여부

```
private fun isNetworkAvailable(): Boolean {  
    val connectivityManager = getSystemService(Context.CONNECTIVITY_SERVICE)  
        as ConnectivityManager  
    val nw = connectivityManager.activeNetwork ?: return false  
    val actNw = connectivityManager.getNetworkCapabilities(nw) ?: return false  
    return when {  
        actNw.hasTransport(NetworkCapabilities.TRANSPORT_WIFI) -> {  
            Log.d("kkang", "wifi available")  
            true  
        }  
        actNw.hasTransport(NetworkCapabilities.TRANSPORT_CELLULAR) -> {  
            Log.d("kkang", "cellular available")  
            true  
        }  
        else -> false  
    }  
}
```

18-1 스마트폰 정보 구하기

- requestNetwork() 함수를 이용하는 방법

• CHANGE_NETWORK_STATE 퍼미션

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

- NetworkRequest의 addCapability()와 addTransportType() 함수를 이용하면 네트워크 타입을 지정

• 네트워크 타입 지정

```
val networkReq: NetworkRequest = NetworkRequest.Builder()
    .addCapability(NetworkCapabilities.NET_CAPABILITY_INTERNET)
    .addTransportType(NetworkCapabilities.TRANSPORT_CELLULAR)
    .addTransportType(NetworkCapabilities.TRANSPORT_WIFI)
    .build()
```


18-1 스마트폰 정보 구하기

- NetworkRequest 객체에 네트워크 타입을 설정한 후 객체를 requestNetwork() 함수의 두 번째 매개변수로 지정

• 네트워크 접속 가능 여부

```
conManager.requestNetwork(networkReq, object : ConnectivityManager.NetworkCallback() {  
    override fun onAvailable(network: Network) {  
        super.onAvailable(network)  
        Log.d("kkang", "NetworkCallback...onAvailable....")  
    }  
    override fun onUnavailable() {  
        super.onUnavailable()  
        Log.d("kkang", "NetworkCallback...onUnavailable....")  
    }  
})
```



감사합니다

단단히 마음먹고 떠난 사람은
산꼭대기에 도착할 수 있다.
산은 올라가는 사람에게만 정복된다.



윌리엄 셰익스피어
William Shakespeare