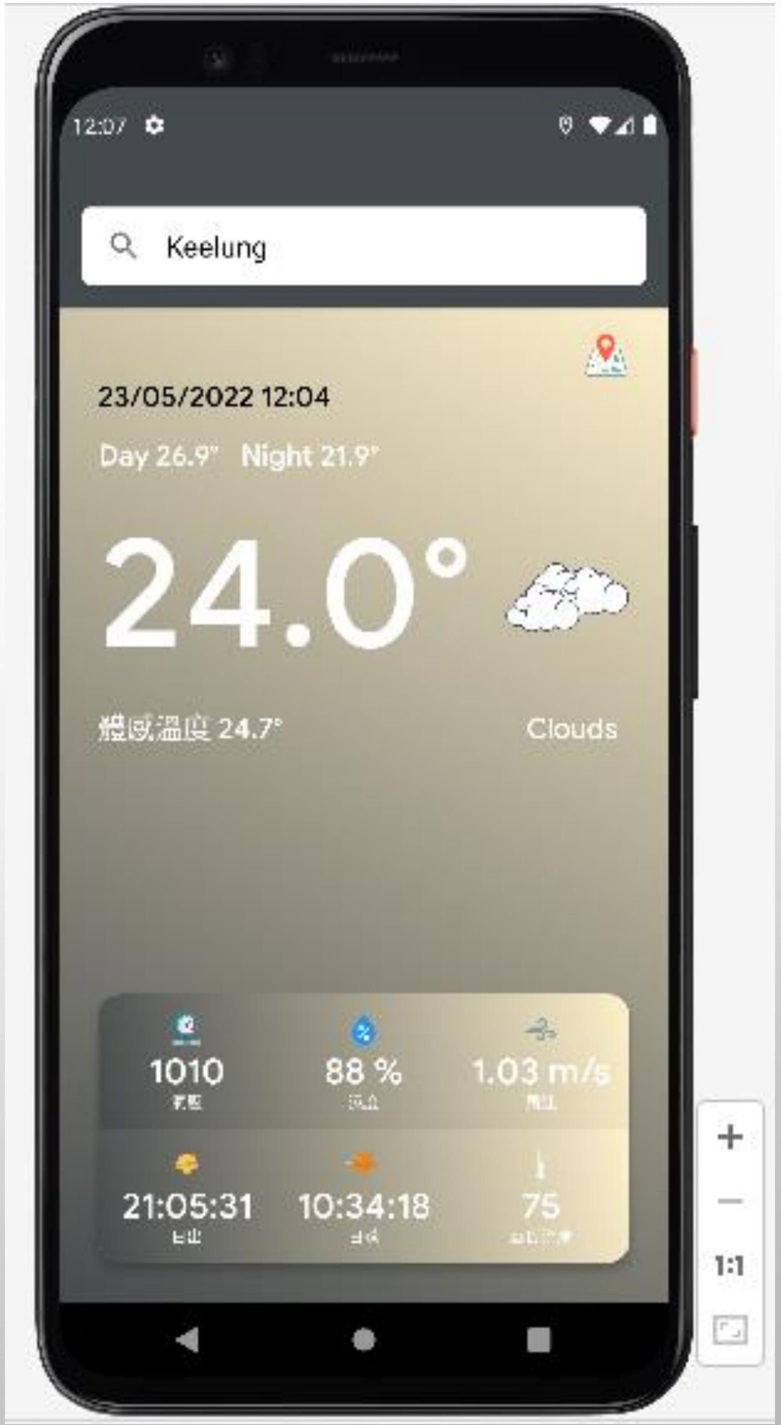
天氣APP（具GPS定位）



12:07

Keelung

23/05/2022 12:04

Day 26.9° Night 21.9°

24.0°

體感溫度 24.7°

Clouds

1010
氣壓

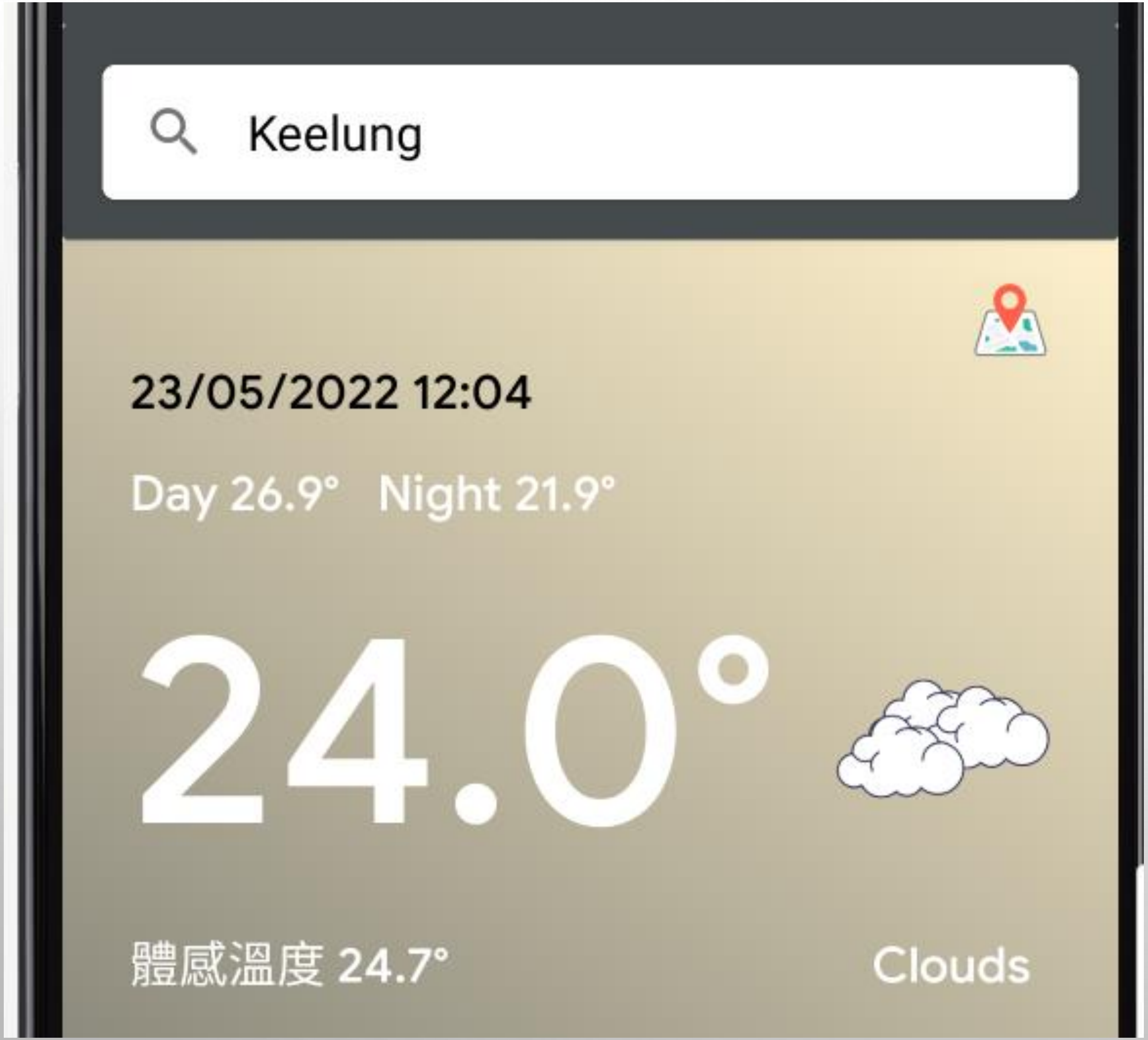
88 %
濕度

1.03 m/s
風速

21:05:31
日出

10:34:18
日落

75
[illegible]



Keelung

23/05/2022 12:04

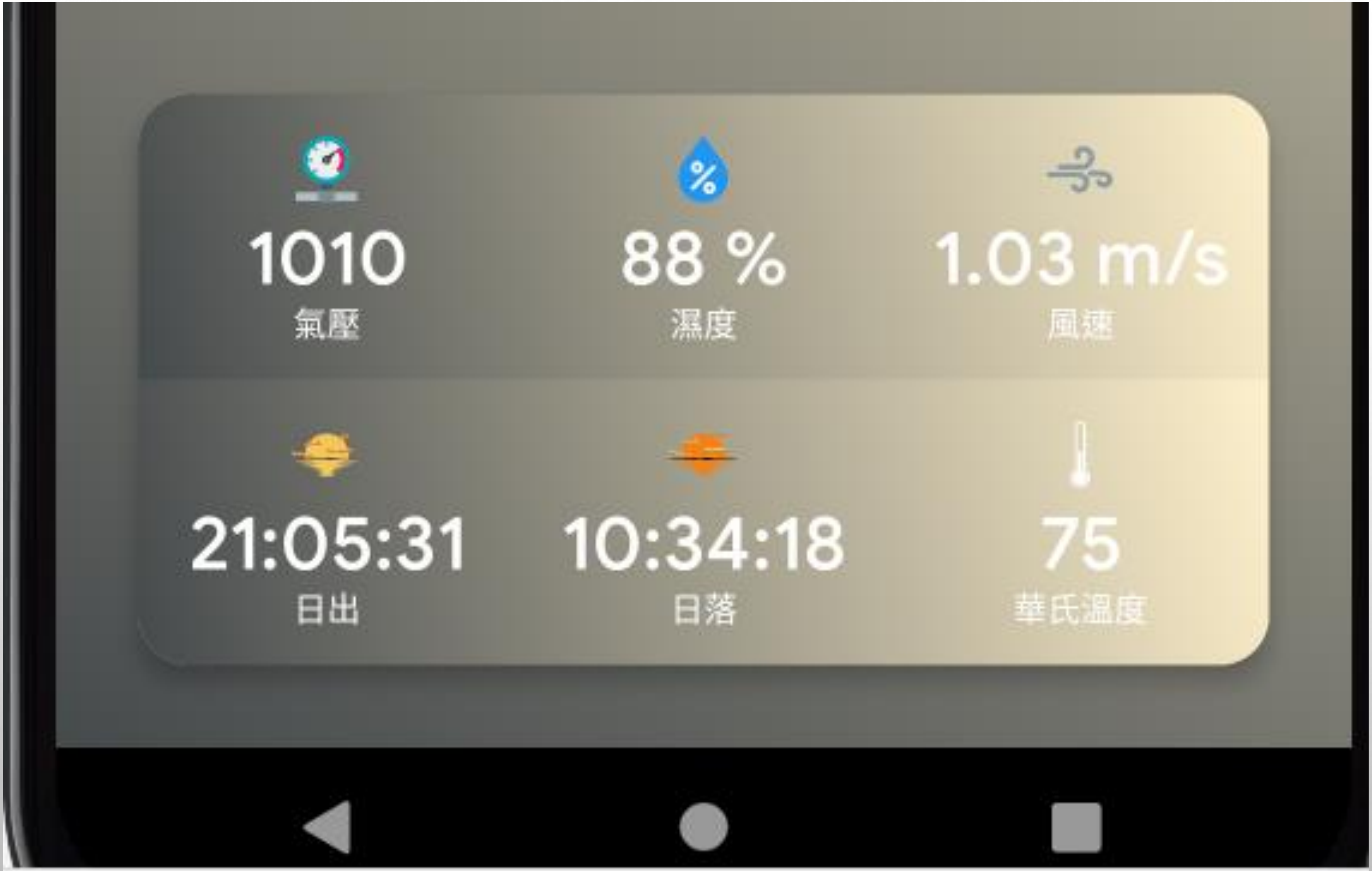
Day 26.9° Night 21.9°

24.0°

體感溫度 24.7°

Clouds

- 可利用搜索框搜索地名
- 右上角的小地圖為定位按鈕
- 依天氣些切換顯示天氣圖片(第12頁)



1010
氣壓

88 %
濕度

1.03 m/s
風速

21:05:31
日出

10:34:18
日落

75
華氏溫度

- 包含6項天氣屬性

利用 OPENWEATHERMAP

- 經緯度:

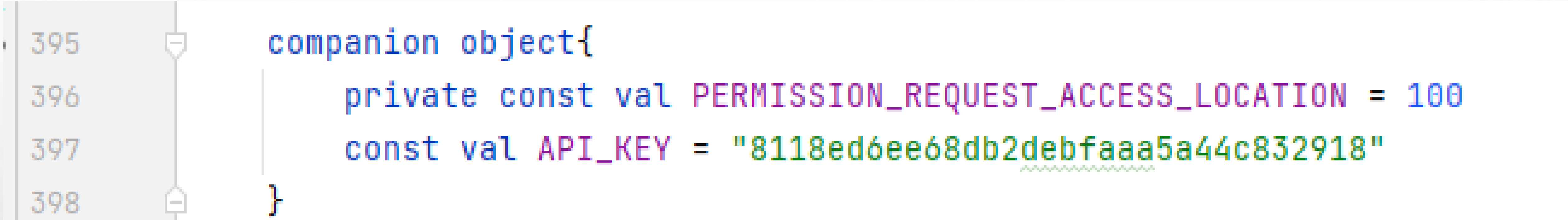
HTTPS://API.OPENWEATHERMAP.ORG/DATA/2.5/WEATHER?LAT={LAT}&LON={LON}&APPID={API KEY}

HTTPS://API.OPENWEATHERMAP.ORG/DATA/2.5/WEATHER?LAT=25.0478&LON=121.5319&APPID=8118ED6EE68DB2DEBFAAA5A44C832918

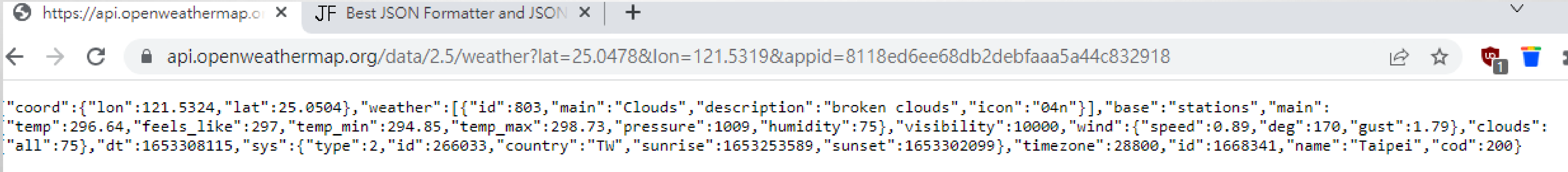
- 城市名:

HTTPS://API.OPENWEATHERMAP.ORG/DATA/2.5/WEATHER?Q={CITY NAME}&APPID={API KEY}

HTTPS://API.OPENWEATHERMAP.ORG/DATA/2.5/WEATHER?Q=TAIPEI&UNITS=METRIC&APPID=8118ED6EE68DB2DEBFAAA5A44C832918

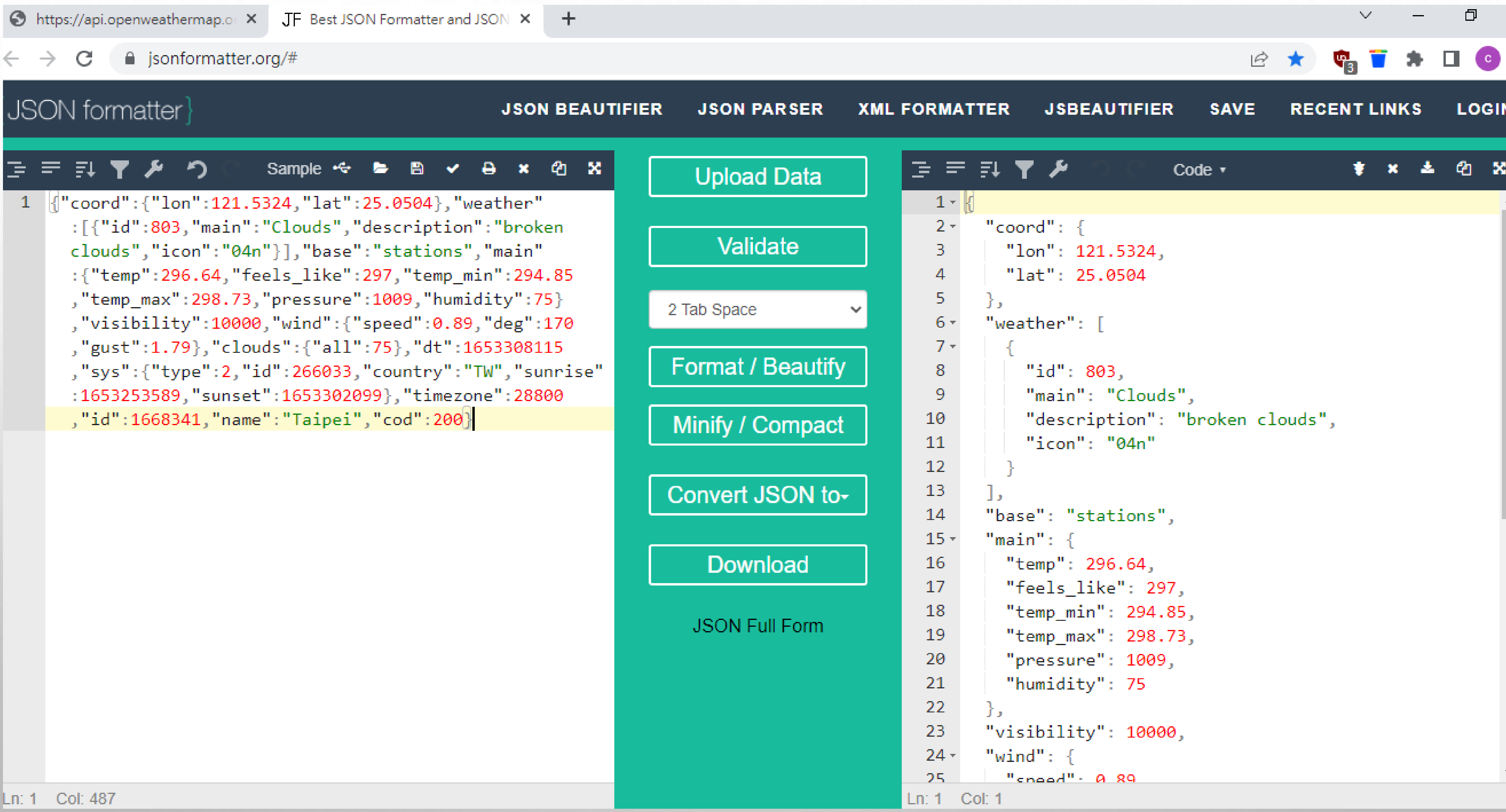


```
    companion object{
        private const val PERMISSION_REQUEST_ACCESS_LOCATION = 100
        const val API_KEY = "8118ed6ee68db2debfaaa5a44c832918"
    }
```



api.openweathermap.org/data/2.5/weather?lat=25.0478&lon=121.5319&appid=8118ed6ee68db2debfaaa5a44c832918

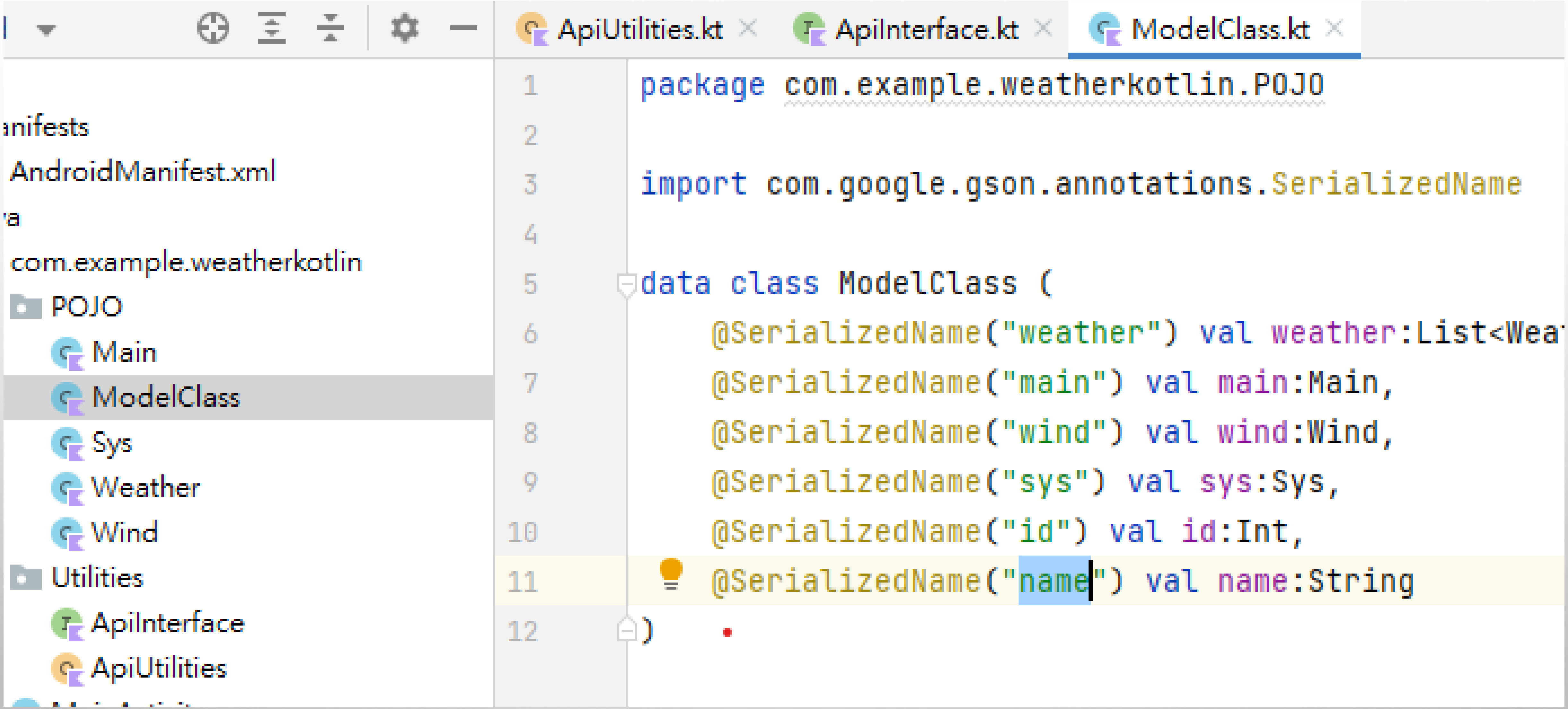
```
"coord":{"lon":121.5324,"lat":25.0504},"weather":[{"id":803,"main":"Clouds","description":"broken clouds","icon":"04n"}],"base":"stations","main":
"temp":296.64,"feels_like":297,"temp_min":294.85,"temp_max":298.73,"pressure":1009,"humidity":75},"visibility":10000,"wind":{"speed":0.89,"deg":170,"gust":1.79},"clouds":
"all":75},"dt":1653308115,"sys":{"type":2,"id":266033,"country":"TW","sunrise":1653253589,"sunset":1653302099},"timezone":28800,"id":1668341,"name":"Taipei","cod":200}
```



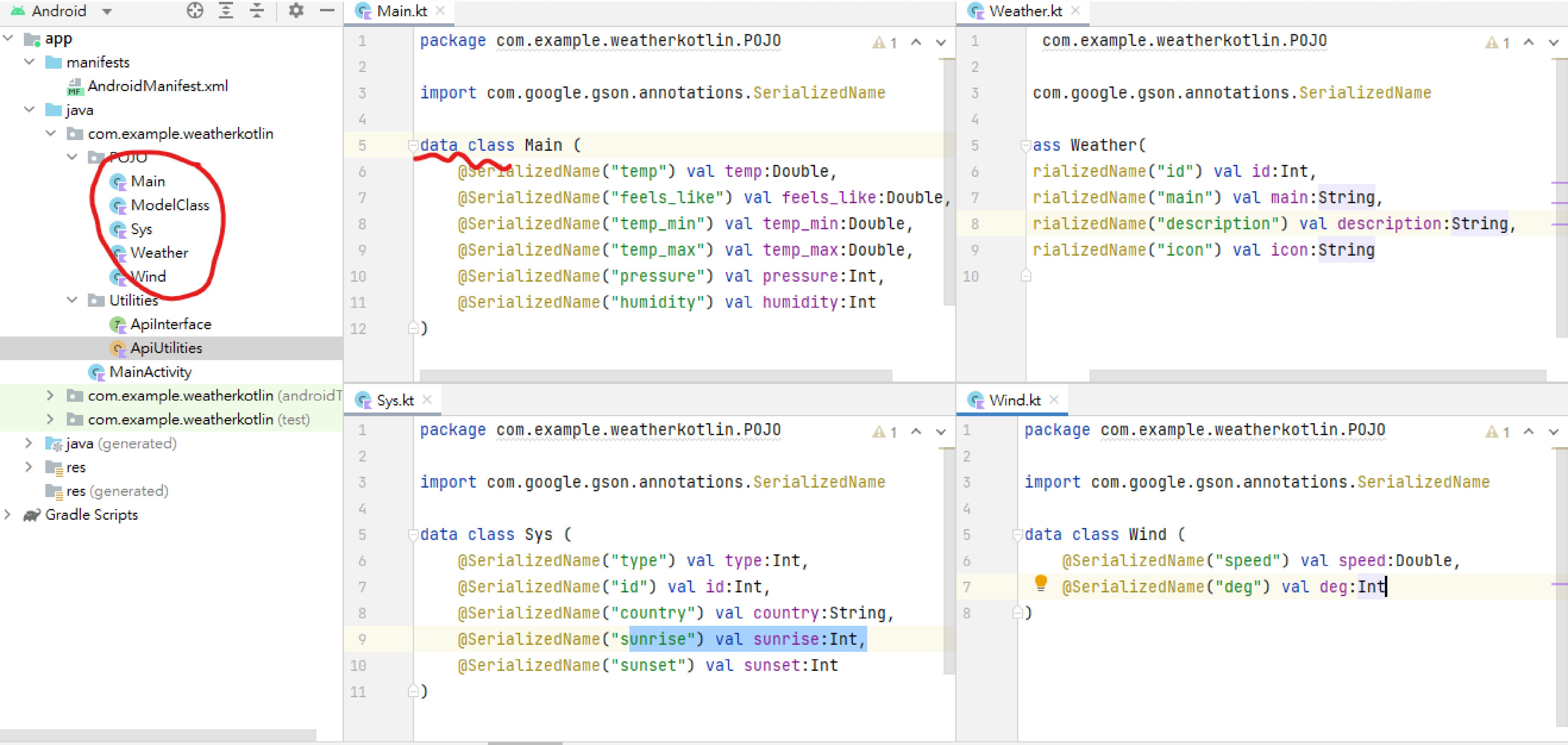
2 Tab Space

```
{"coord":{"lon":121.5324,"lat":25.0504},"weather":[{"id":803,"main":"Clouds","description":"broken clouds","icon":"04n"}],"base":"stations","main":{"temp":296.64,"feels_like":297,"temp_min":294.85,"temp_max":298.73,"pressure":1009,"humidity":75},"visibility":10000,"wind":{"speed":0.89,"deg":170,"gust":1.79},"clouds":{"all":75},"dt":1653308115,"sys":{"type":2,"id":266033,"country":"TW","sunrise":1653253589,"sunset":1653302099},"timezone":28800,"id":1668341,"name":"Taipei","cod":200}
```

```
{
  "coord": {
    "lon": 121.5324,
    "lat": 25.0504
  },
  "weather": [
    {
      "id": 803,
      "main": "Clouds",
      "description": "broken clouds",
      "icon": "04n"
    }
  ],
  "base": "stations",
  "main": {
    "temp": 296.64,
    "feels_like": 297,
    "temp_min": 294.85,
    "temp_max": 298.73,
    "pressure": 1009,
    "humidity": 75
  },
  "visibility": 10000,
  "wind": {
    "speed": 0.89
```



```kotlin
package com.example.weatherkotlin.POJO

import com.google.gson.annotations.SerializedName

data class ModelClass (
    @SerializedName("weather") val weather:List<Wea
    @SerializedName("main") val main:Main,
    @SerializedName("wind") val wind:Wind,
    @SerializedName("sys") val sys:Sys,
    @SerializedName("id") val id:Int,
    @SerializedName("name") val name:String
)
```



Main.kt

```kotlin
package com.example.weatherkotlin.POJO

import com.google.gson.annotations.SerializedName

data class Main (
    @SerializedName("temp") val temp:Double,
    @SerializedName("feels_like") val feels_like:Double,
    @SerializedName("temp_min") val temp_min:Double,
    @SerializedName("temp_max") val temp_max:Double,
    @SerializedName("pressure") val pressure:Int,
    @SerializedName("humidity") val humidity:Int
)
```

Weather.kt

```kotlin
 com.example.weatherkotlin.POJO

com.google.gson.annotations.SerializedName

ass Weather(
rializedName("id") val id:Int,
rializedName("main") val main:String,
rializedName("description") val description:String,
rializedName("icon") val icon:String

```

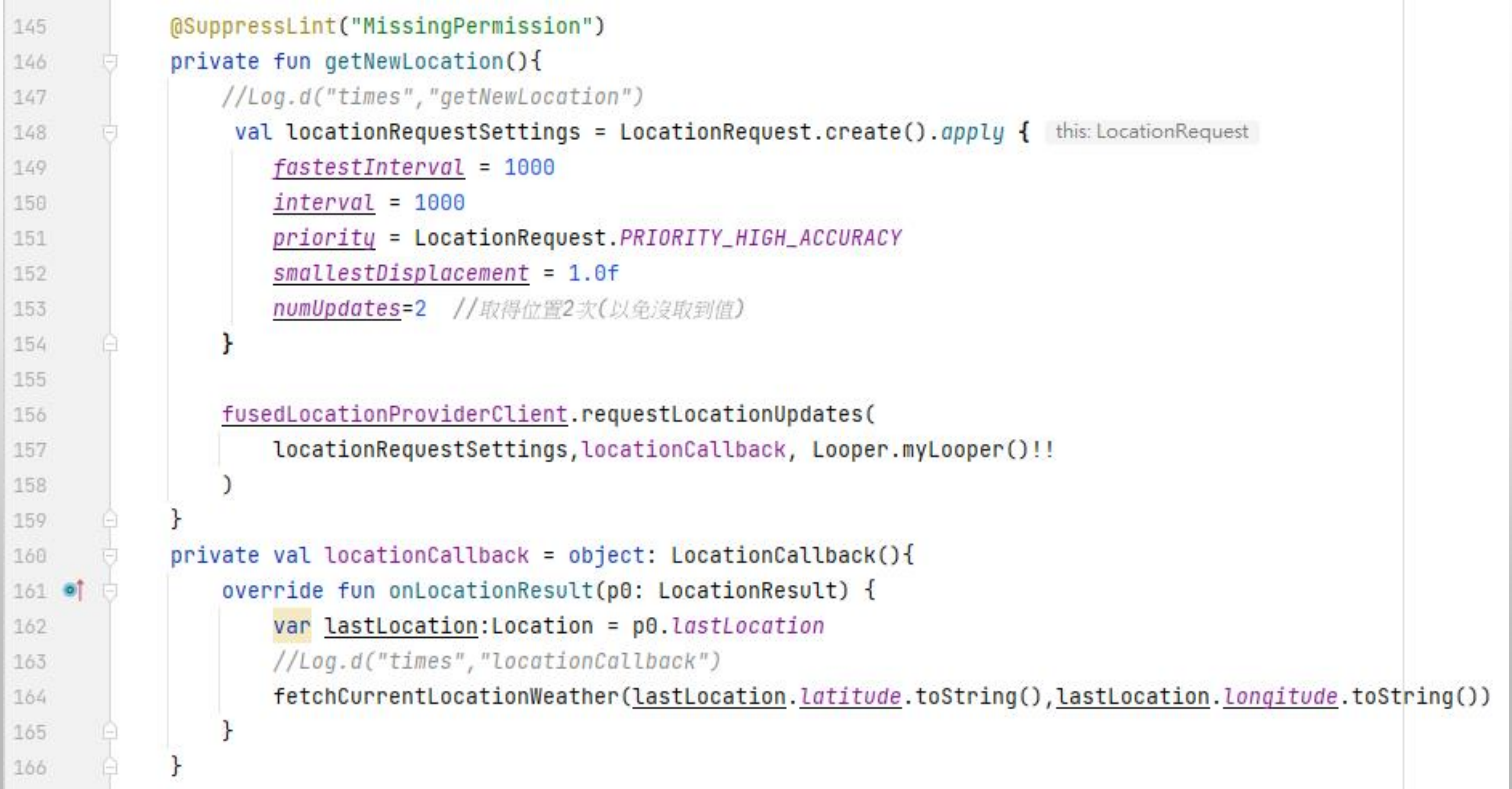
Sys.kt

```kotlin
package com.example.weatherkotlin.POJO

import com.google.gson.annotations.SerializedName

data class Sys (
    @SerializedName("type") val type:Int,
    @SerializedName("id") val id:Int,
    @SerializedName("country") val country:String,
    @SerializedName("sunrise") val sunrise:Int,
    @SerializedName("sunset") val sunset:Int
)
```

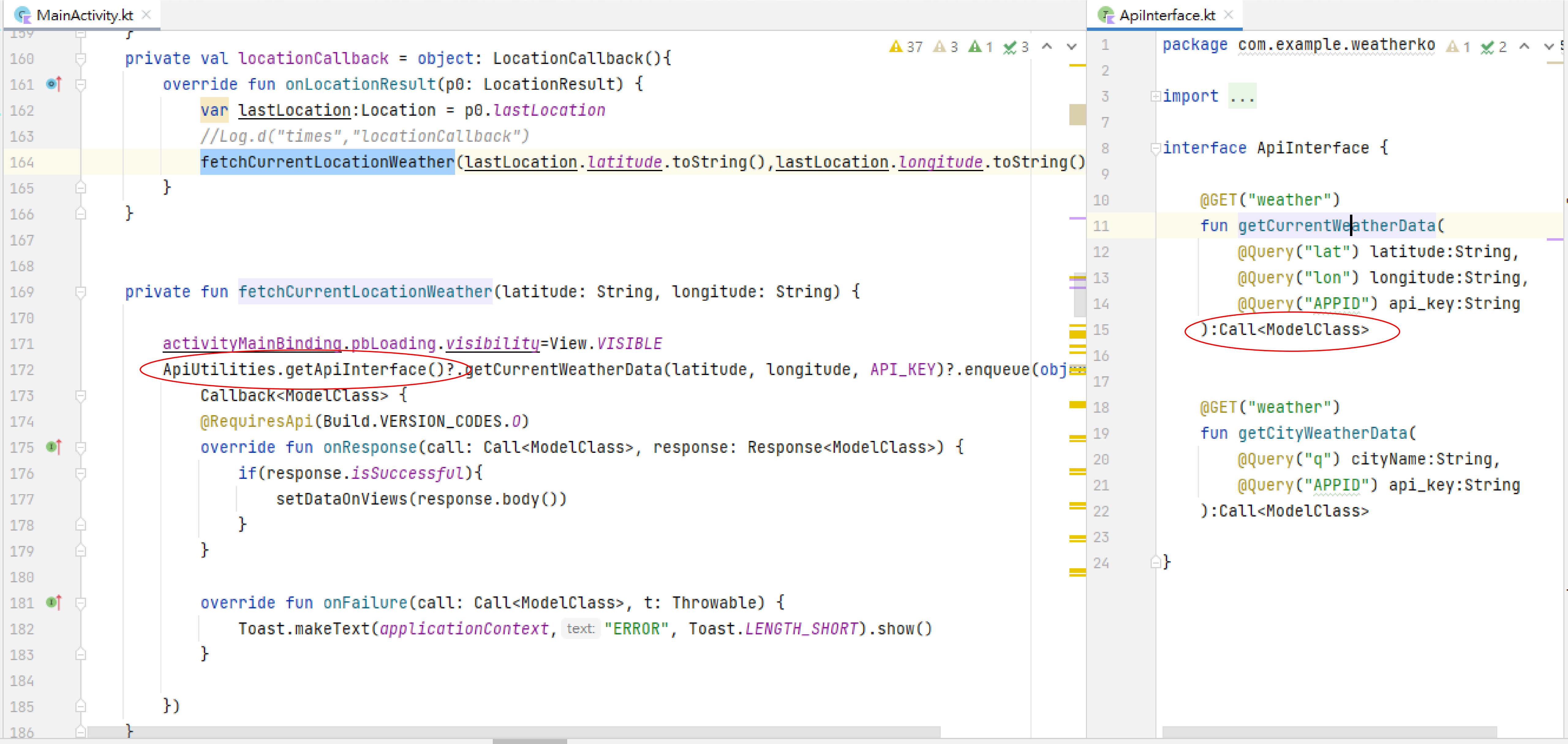
Wind.kt

```kotlin
package com.example.weatherkotlin.POJO

import com.google.gson.annotations.SerializedName

data class Wind (
    @SerializedName("speed") val speed:Double,
    @SerializedName("deg") val deg:Int
)
```

定位, 取得”新位置” (非系統暫存定位位置)

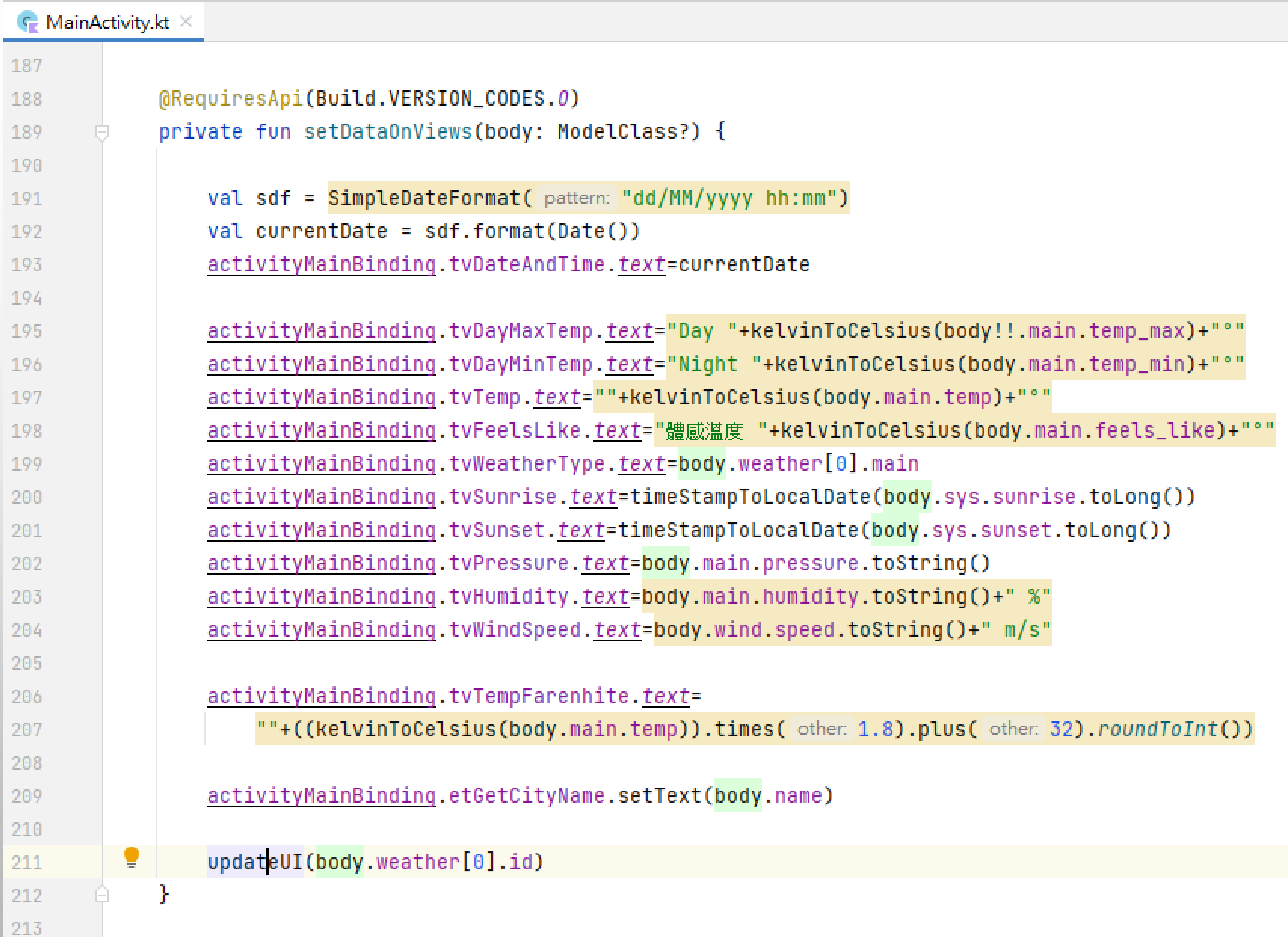


```
@SuppressLint("MissingPermission")
private fun getNewLocation(){
    //Log.d("times","getNewLocation")
     val locationRequestSettings = LocationRequest.create().apply {
        fastestInterval = 1000
        interval = 1000
        priority = LocationRequest.PRIORITY_HIGH_ACCURACY
        smallestDisplacement = 1.0f
        numUpdates=2  //取得位置2次(以免沒取到值)
    }

    fusedLocationProviderClient.requestLocationUpdates(
        locationRequestSettings,locationCallback, Looper.myLooper()!!
    )
}
private val locationCallback = object: LocationCallback(){
    override fun onLocationResult(p0: LocationResult) {
        var lastLocation:Location = p0.lastLocation
        //Log.d("times","locationCallback")
        fetchCurrentLocationWeather(lastLocation.latitude.toString(),lastLocation.longitude.toString())
    }
}
```

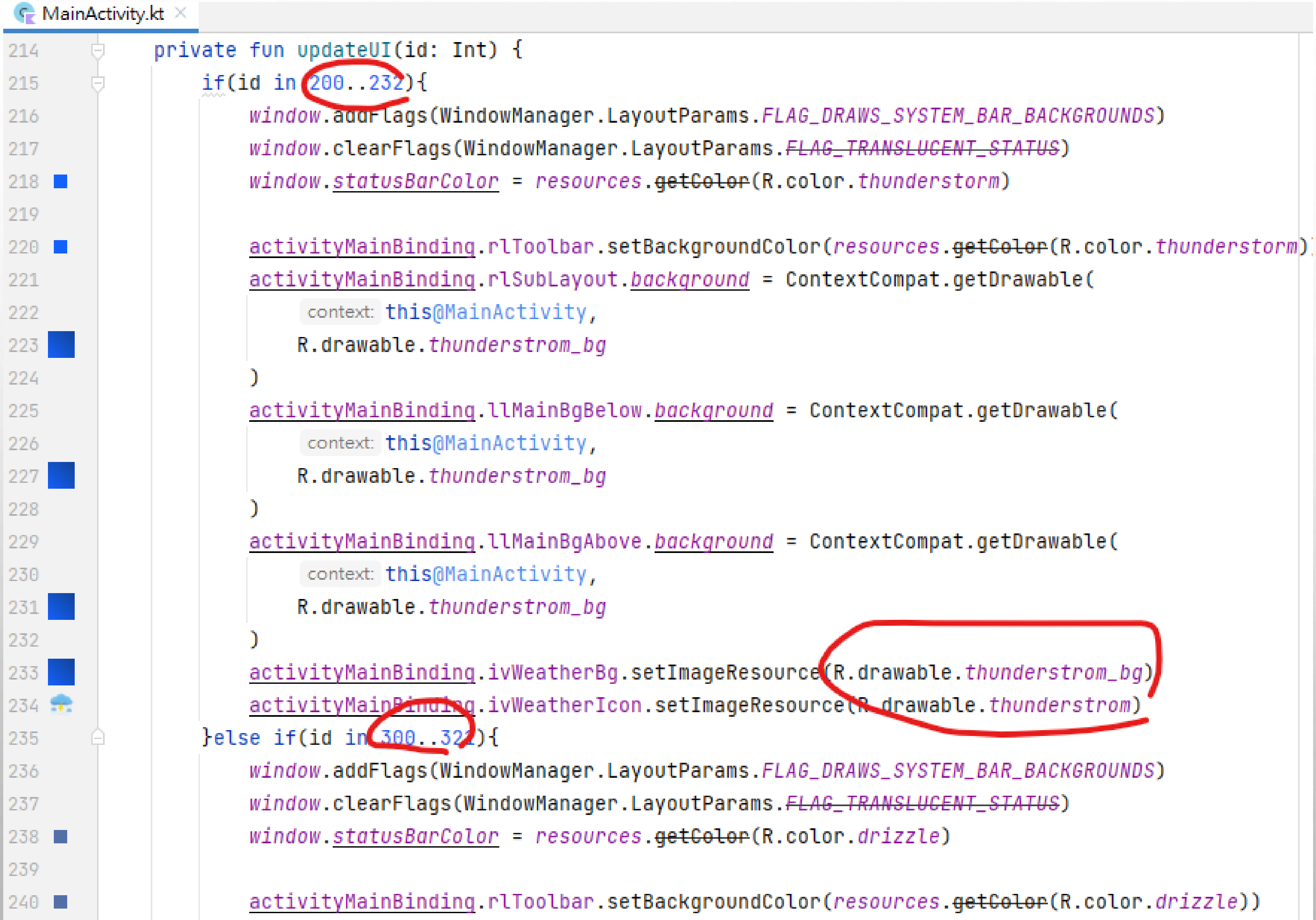


```kotlin
    }
    private val locationCallback = object: LocationCallback(){
        override fun onLocationResult(p0: LocationResult) {
            var lastLocation:Location = p0.lastLocation
            //Log.d("times","locationCallback")
            fetchCurrentLocationWeather(lastLocation.latitude.toString(),lastLocation.longitude.toString()
        }
    }


    private fun fetchCurrentLocationWeather(latitude: String, longitude: String) {

        activityMainBinding.pbLoading.visibility=View.VISIBLE
        ApiUtilities.getApiInterface()?.getCurrentWeatherData(latitude, longitude, API_KEY)?.enqueue(obj
            Callback<ModelClass> {
            @RequiresApi(Build.VERSION_CODES.O)
            override fun onResponse(call: Call<ModelClass>, response: Response<ModelClass>) {
                if(response.isSuccessful){
                    setDataOnViews(response.body())
                }
            }

            override fun onFailure(call: Call<ModelClass>, t: Throwable) {
                Toast.makeText(applicationContext, text: "ERROR", Toast.LENGTH_SHORT).show()
            }

        })
    }
```

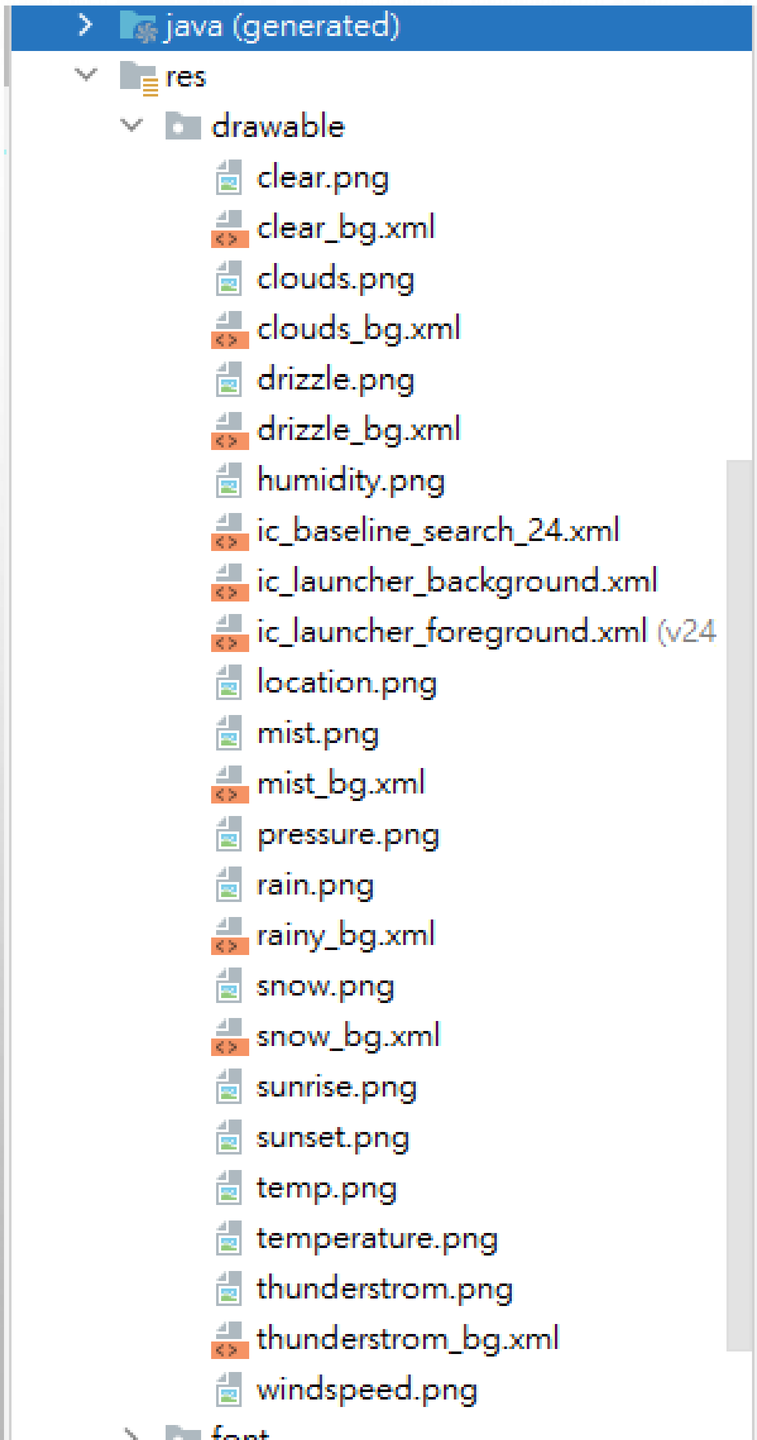
```kotlin
package com.example.weatherko

import ...

interface ApiInterface {

    @GET("weather")
    fun getCurrentWeatherData(
        @Query("lat") latitude:String,
        @Query("lon") longitude:String,
        @Query("APPID") api_key:String
    ):Call<ModelClass>


    @GET("weather")
    fun getCityWeatherData(
        @Query("q") cityName:String,
        @Query("APPID") api_key:String
    ):Call<ModelClass>

}
```



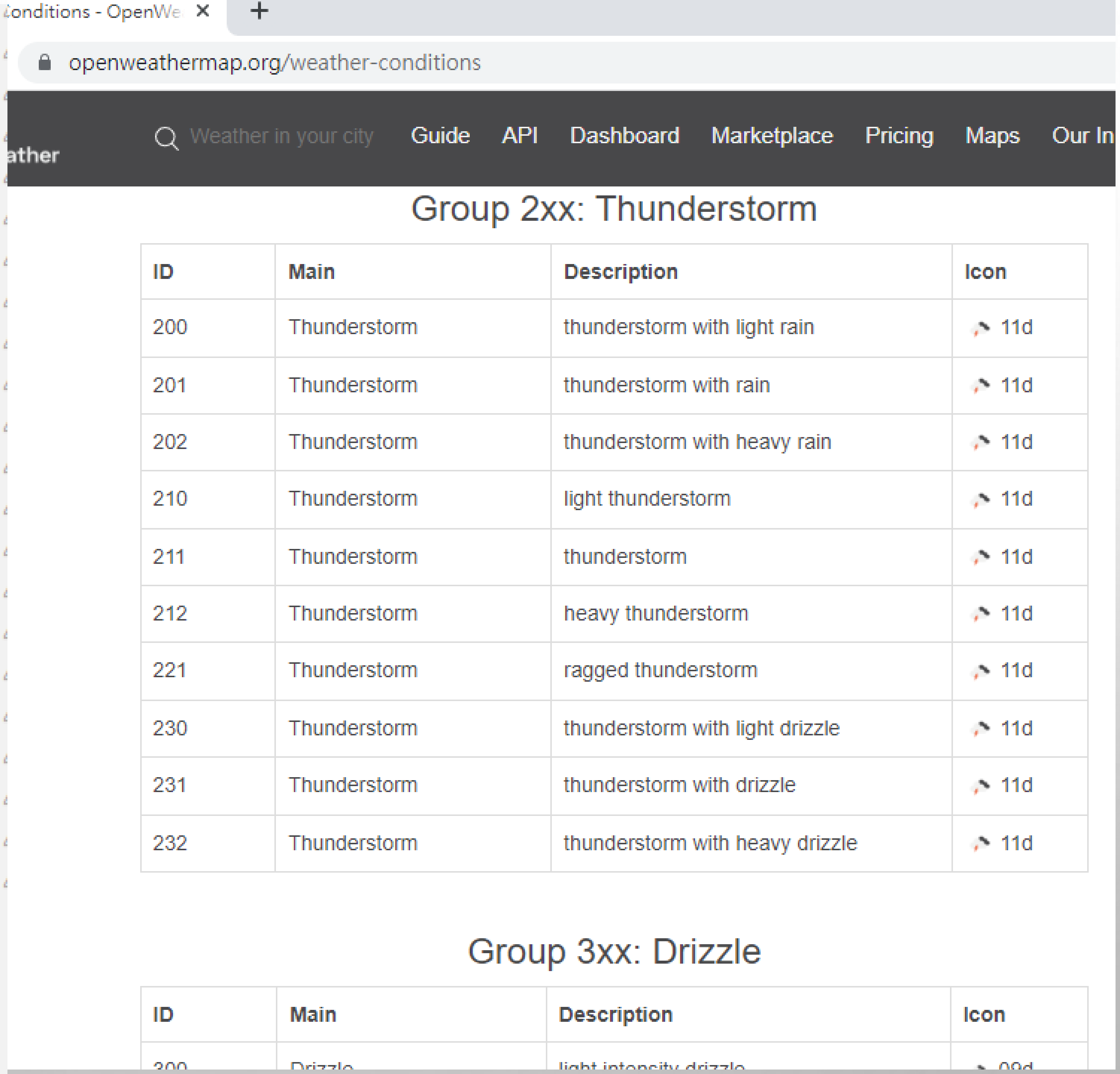
```
    @RequiresApi(Build.VERSION_CODES.O)
    private fun setDataOnViews(body: ModelClass?) {

        val sdf = SimpleDateFormat( pattern: "dd/MM/yyyy hh:mm")
        val currentDate = sdf.format(Date())
        activityMainBinding.tvDateAndTime.text=currentDate

        activityMainBinding.tvDayMaxTemp.text="Day "+kelvinToCelsius(body!!.main.temp_max)+"°"
        activityMainBinding.tvDayMinTemp.text="Night "+kelvinToCelsius(body.main.temp_min)+"°"
        activityMainBinding.tvTemp.text=""+kelvinToCelsius(body.main.temp)+"°"
        activityMainBinding.tvFeelsLike.text="體感溫度 "+kelvinToCelsius(body.main.feels_like)+"°"
        activityMainBinding.tvWeatherType.text=body.weather[0].main
        activityMainBinding.tvSunrise.text=timeStampToLocalDate(body.sys.sunrise.toLong())
        activityMainBinding.tvSunset.text=timeStampToLocalDate(body.sys.sunset.toLong())
        activityMainBinding.tvPressure.text=body.main.pressure.toString()
        activityMainBinding.tvHumidity.text=body.main.humidity.toString()+" %"
        activityMainBinding.tvWindSpeed.text=body.wind.speed.toString()+" m/s"

        activityMainBinding.tvTempFarenhite.text=
            ""+((kelvinToCelsius(body.main.temp)).times( other: 1.8).plus( other: 32).roundToInt())

        activityMainBinding.etGetCityName.setText(body.name)

        updateUI(body.weather[0].id)
    }

```



```
    private fun updateUI(id: Int) {
        if(id in 200..232){
            window.addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTEM_BAR_BACKGROUNDS)
            window.clearFlags(WindowManager.LayoutParams.~~FLAG_TRANSLUCENT_STATUS~~)
            window.statusBarColor = resources.~~getColor~~(R.color.thunderstorm)

            activityMainBinding.rlToolbar.setBackgroundColor(resources.~~getColor~~(R.color.thunderstorm))
            activityMainBinding.rlSubLayout.background = ContextCompat.getDrawable(
                context: this@MainActivity,
                R.drawable.thunderstrom_bg
            )
            activityMainBinding.llMainBgBelow.background = ContextCompat.getDrawable(
                context: this@MainActivity,
                R.drawable.thunderstrom_bg
            )
            activityMainBinding.llMainBgAbove.background = ContextCompat.getDrawable(
                context: this@MainActivity,
                R.drawable.thunderstrom_bg
            )
            activityMainBinding.ivWeatherBg.setImageResource(R.drawable.thunderstrom_bg)
            activityMainBinding.ivWeatherIcon.setImageResource(R.drawable.thunderstrom)
        }else if(id in 300..321){
            window.addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTEM_BAR_BACKGROUNDS)
            window.clearFlags(WindowManager.LayoutParams.~~FLAG_TRANSLUCENT_STATUS~~)
            window.statusBarColor = resources.~~getColor~~(R.color.drizzle)

            activityMainBinding.rlToolbar.setBackgroundColor(resources.~~getColor~~(R.color.drizzle))
```



java (generated)
res
drawable
clear.png
clear_bg.xml
clouds.png
clouds_bg.xml
drizzle.png
drizzle_bg.xml
humidity.png
ic_baseline_search_24.xml
ic_launcher_background.xml
ic_launcher_foreground.xml (v24
location.png
mist.png
mist_bg.xml
pressure.png
rain.png
rainy_bg.xml
snow.png
snow_bg.xml
sunrise.png
sunset.png
temp.png
temperature.png
thunderstrom.png
thunderstrom_bg.xml
windspeed.png
font



Conditions - OpenWe ✕ +
openweathermap.org/weather-conditions
ather
Weather in your city
Guide API Dashboard Marketplace Pricing Maps Our In

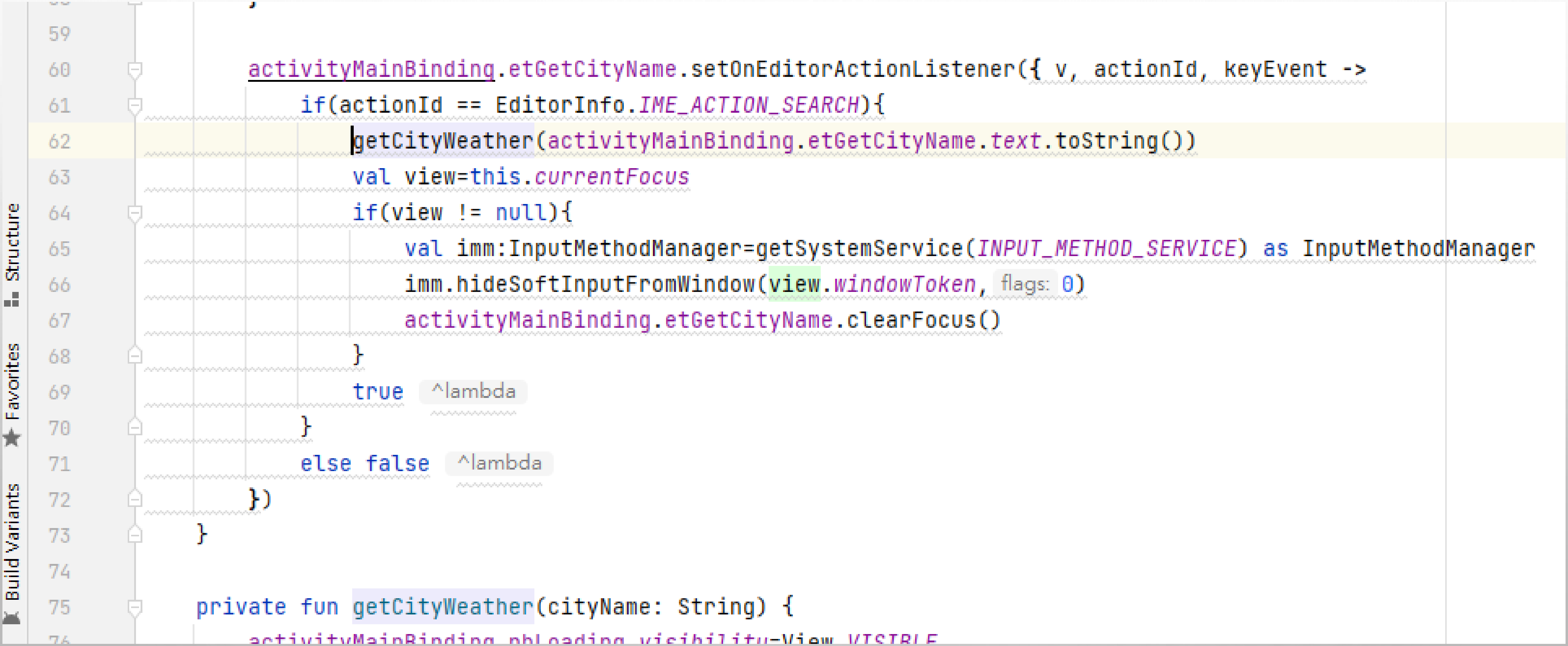
Group 2xx: Thunderstorm

ID	Main	Description	Icon
200	Thunderstorm	thunderstorm with light rain	11d
201	Thunderstorm	thunderstorm with rain	11d
202	Thunderstorm	thunderstorm with heavy rain	11d
210	Thunderstorm	light thunderstorm	11d
211	Thunderstorm	thunderstorm	11d
212	Thunderstorm	heavy thunderstorm	11d
221	Thunderstorm	ragged thunderstorm	11d
230	Thunderstorm	thunderstorm with light drizzle	11d
231	Thunderstorm	thunderstorm with drizzle	11d
232	Thunderstorm	thunderstorm with heavy drizzle	11d

Group 3xx: Drizzle

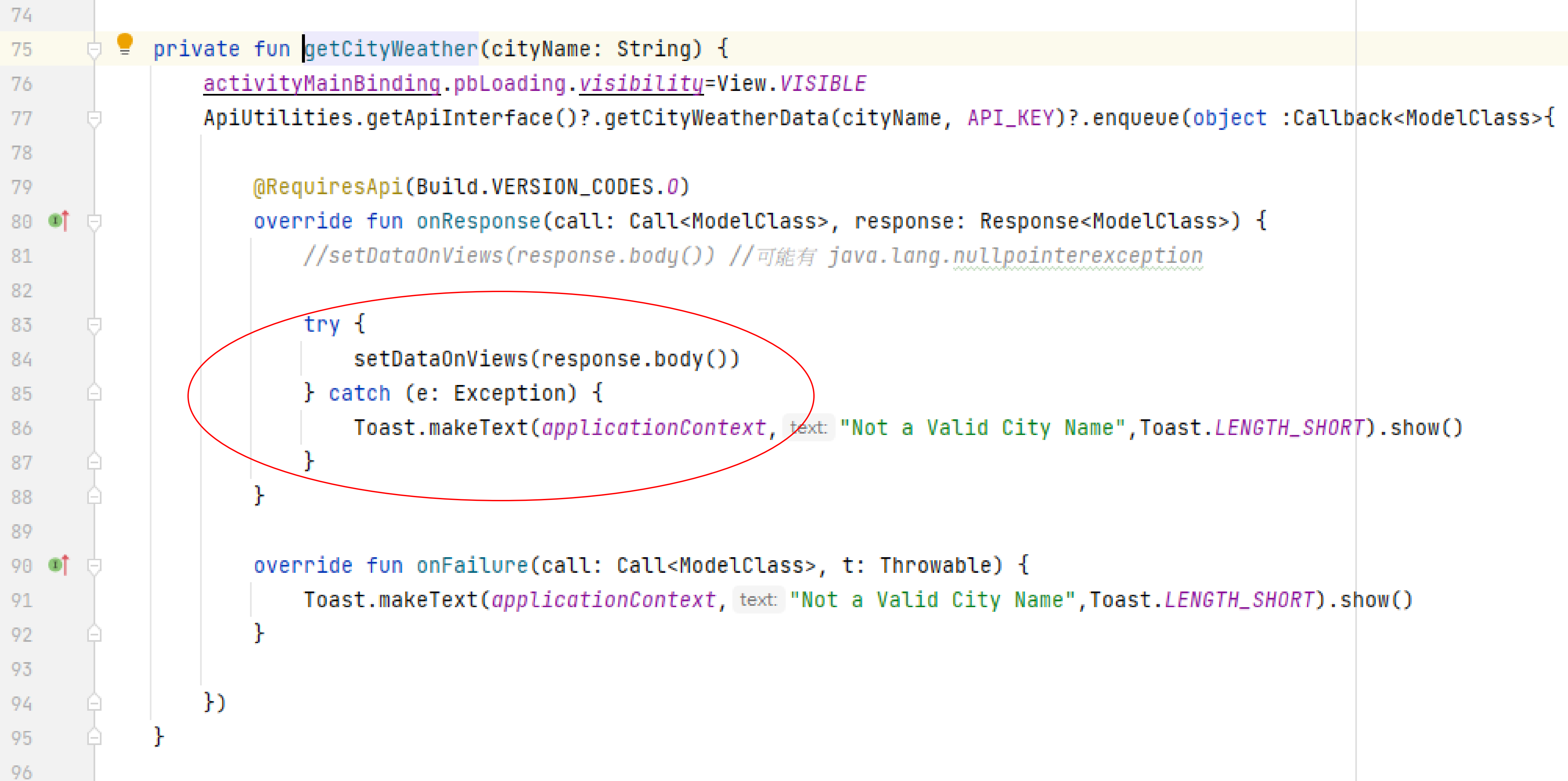
ID	Main	Description	Icon
300	Drizzle	light intensity drizzle	09d

搜尋框部分(輸入地名)



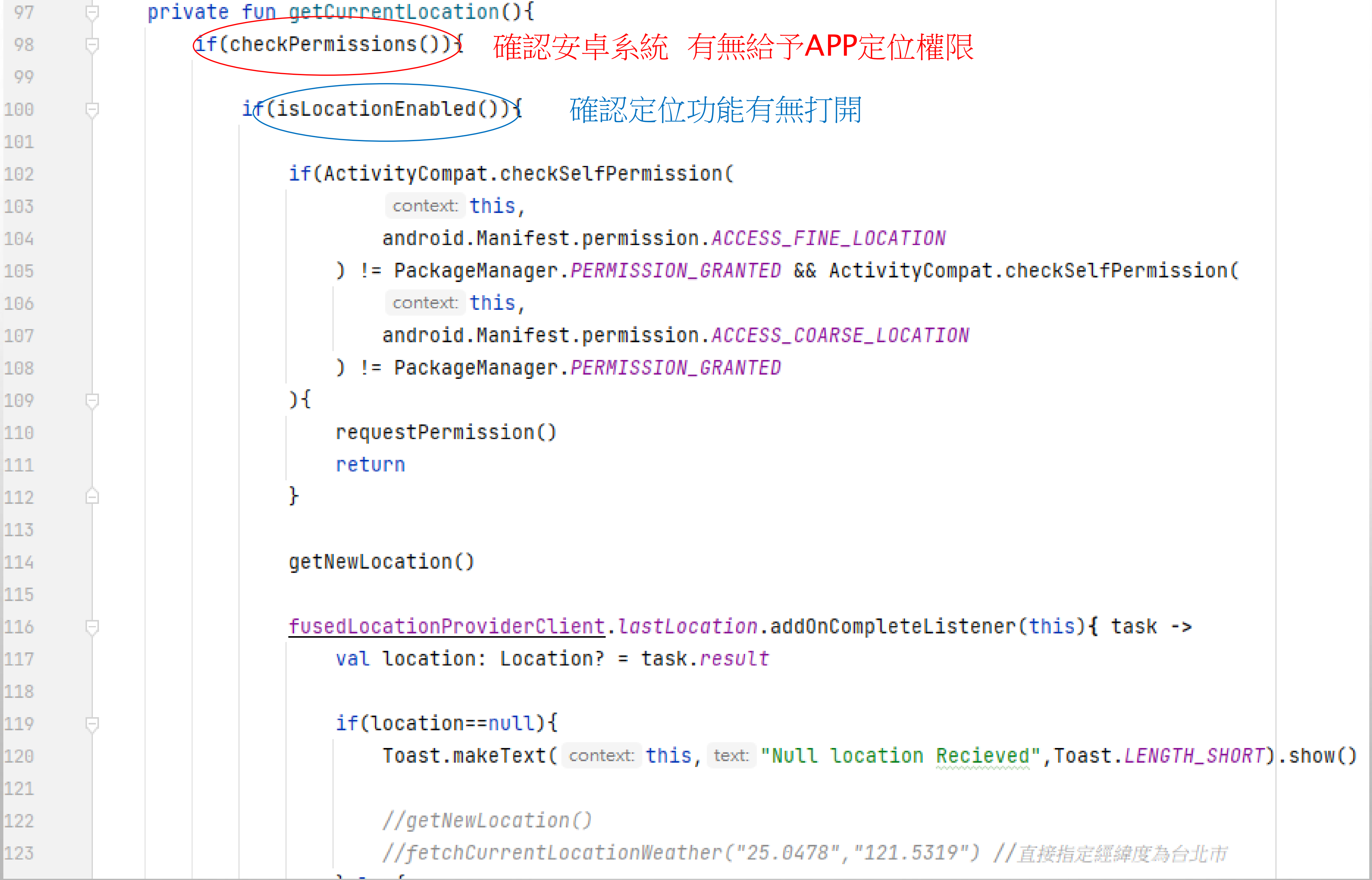
```
        }

        activityMainBinding.etGetCityName.setOnEditorActionListener({ v, actionId, keyEvent ->
            if(actionId == EditorInfo.IME_ACTION_SEARCH){
                getCityWeather(activityMainBinding.etGetCityName.text.toString())
                val view=this.currentFocus
                if(view != null){
                    val imm:InputMethodManager=getSystemService(INPUT_METHOD_SERVICE) as InputMethodManager
                    imm.hideSoftInputFromWindow(view.windowToken, flags: 0)
                    activityMainBinding.etGetCityName.clearFocus()
                }
                true ^lambda
            }
            else false ^lambda
        })
    }

    private fun getCityWeather(cityName: String) {
        activityMainBinding.pbLoading.visibility=View.VISIBLE
```

搜尋框避免輸入錯誤閃退(例外處理)



```
private fun getCityWeather(cityName: String) {
    activityMainBinding.pbLoading.visibility=View.VISIBLE
    ApiUtilities.getApiInterface()?.getCityWeatherData(cityName, API_KEY)?.enqueue(object :Callback<ModelClass>{

        @RequiresApi(Build.VERSION_CODES.O)
        override fun onResponse(call: Call<ModelClass>, response: Response<ModelClass>) {
            //setDataOnViews(response.body()) //可能有 java.lang.nullpointerexception

            try {
                setDataOnViews(response.body())
            } catch (e: Exception) {
                Toast.makeText(applicationContext, text: "Not a Valid City Name",Toast.LENGTH_SHORT).show()
            }
        }

        override fun onFailure(call: Call<ModelClass>, t: Throwable) {
            Toast.makeText(applicationContext, text: "Not a Valid City Name",Toast.LENGTH_SHORT).show()
        }

    })
}
```

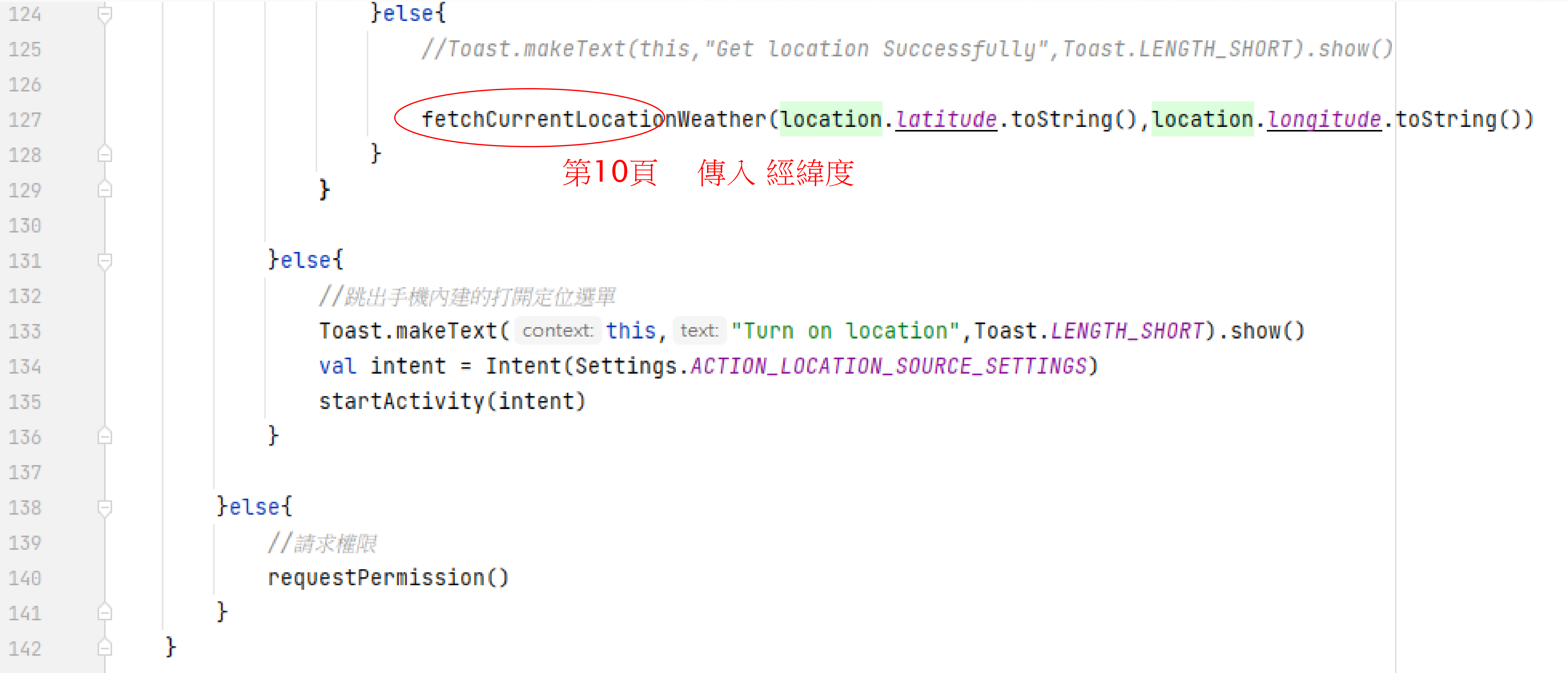
進行定位部分



```
private fun getCurrentLocation(){
    if(checkPermissions()){

        if(isLocationEnabled()){

            if(ActivityCompat.checkSelfPermission(
                    context: this,
                    android.Manifest.permission.ACCESS_FINE_LOCATION
                ) != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
                    context: this,
                    android.Manifest.permission.ACCESS_COARSE_LOCATION
                ) != PackageManager.PERMISSION_GRANTED
            ){
                requestPermission()
                return
            }

            getNewLocation()

            fusedLocationProviderClient.lastLocation.addOnCompleteListener(this){ task ->
                val location: Location? = task.result

                if(location==null){
                    Toast.makeText( context: this, text: "Null location Recieved",Toast.LENGTH_SHORT).show()

                    //getNewLocation()
                    //fetchCurrentLocationWeather("25.0478","121.5319") //直接指定經緯度為台北市
```

確認安卓系統 有無給予APP定位權限

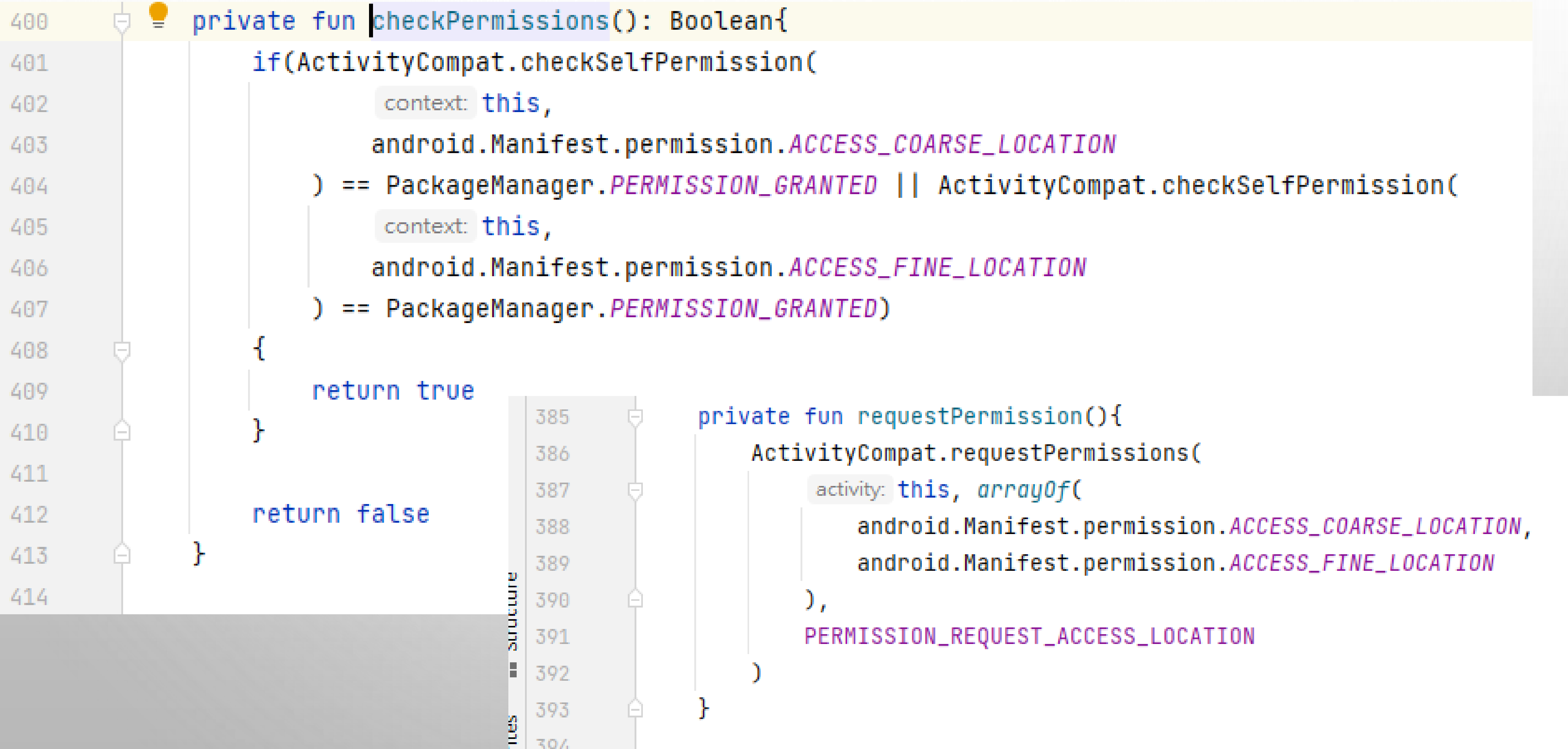
確認定位功能有無打開



```
                }else{
                    //Toast.makeText(this,"Get location Successfully",Toast.LENGTH_SHORT).show()

                    fetchCurrentLocationWeather(location.latitude.toString(),location.longitude.toString())
                }
            }

        }else{
            //跳出手機內建的打開定位選單
            Toast.makeText( context: this, text: "Turn on location",Toast.LENGTH_SHORT).show()
            val intent = Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS)
            startActivity(intent)
        }

    }else{
        //請求權限
        requestPermission()
    }
}
```

第10頁　傳入 經緯度

確認安卓系統　有無給予APP定位權限

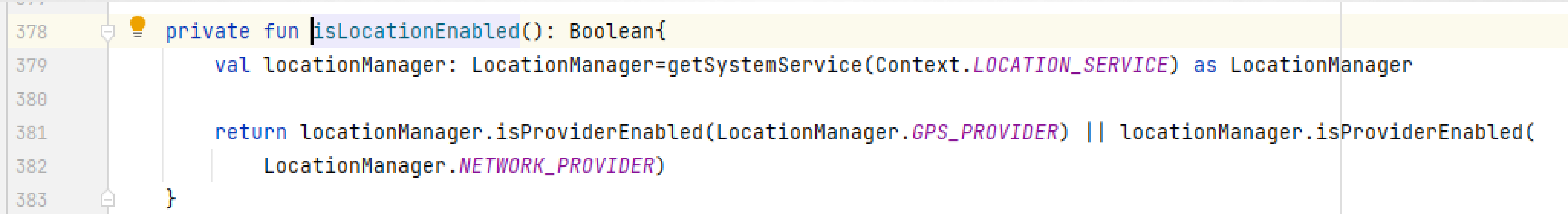


```
private fun checkPermissions(): Boolean{
    if(ActivityCompat.checkSelfPermission(
            context: this,
            android.Manifest.permission.ACCESS_COARSE_LOCATION
        ) == PackageManager.PERMISSION_GRANTED || ActivityCompat.checkSelfPermission(
            context: this,
            android.Manifest.permission.ACCESS_FINE_LOCATION
        ) == PackageManager.PERMISSION_GRANTED)
    {
        return true
    }

    return false
}
```

```
private fun requestPermission(){
    ActivityCompat.requestPermissions(
        activity: this, arrayOf(
            android.Manifest.permission.ACCESS_COARSE_LOCATION,
            android.Manifest.permission.ACCESS_FINE_LOCATION
        ),
        PERMISSION_REQUEST_ACCESS_LOCATION
    )
}
```

確認定位功能有無打開

- 定位有2種 (網路4G定位 或 GPS定位)



```
private fun isLocationEnabled(): Boolean{
    val locationManager: LocationManager=getSystemService(Context.LOCATION_SERVICE) as LocationManager

    return locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER) || locationManager.isProviderEnabled(
        LocationManager.NETWORK_PROVIDER)
}
```