

| 기타 강좌 | >

봇 개발자에서 앱 개발자로 넘어가보기 2단계, 앱으로 똑같이 만들어 보기



Dark Tornado 카페스텝 1:1 채팅
 2020.07.31. 09:33 조회 244

댓글 17 URL 복사

이전 게시물 : <https://cafe.naver.com/nameyee/22017>



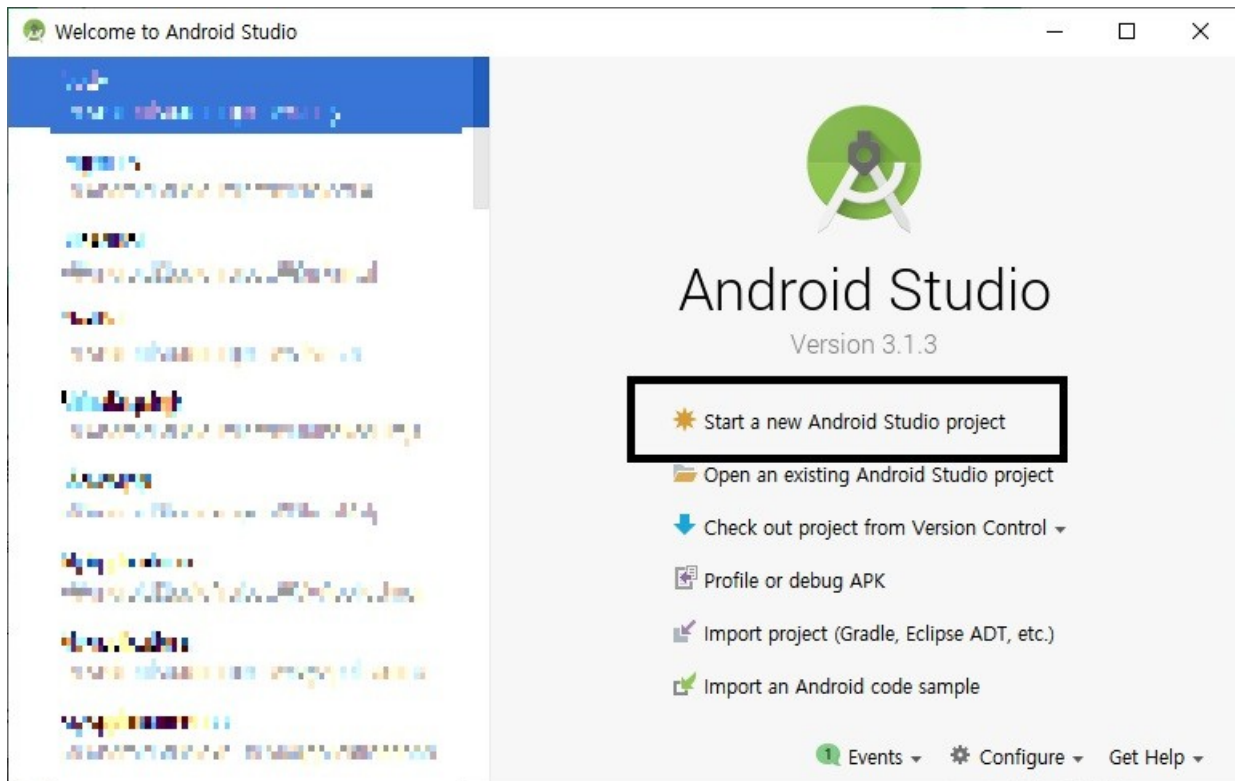
컴퓨터에서는 이클립스와 안드로이드 스튜디오, 폰(안드로이드)에서는 AIDE 등을 사용할 수 있지만, 여기서는 안드로이드 스튜디오를 사용할겁니다.

이 게시글은 기본적인 자바 문법은 알고 있다고 가정하고 작성되었습니다.

이전 글의 서론에서 이어지는 이야기입니다만, 필자의 경우 ModPE 스크립트 구동을 지원하는 앱을 뜯다가 자바에 문법에 나름 익숙해진 상태에서 넘어왔습니다.

JDK 설치, 안드로이드 스튜디오 설치, 안드로이드 SDK 설치하는 검색하면 나오니 알아서 다 하라는 거시네요.

하여튼 새로운 프로젝트를 만들어볼겁니다.



앱 이름이나 도메인 이름은 알아서 만드세요.

Create New Project

Create Android Project

Application name
My Application

Company domain
darktornado.com

Project location
C:\Users\DarkTornado\AndroidStudioProjects\MyApplication

Package name
com.darktornado.myapplication Edit

☐ Include C++ support
☒ Include Kotlin support

Previous Next Cancel Finish

위에서 설정한 내용에 따라 패키지명이 결정됩니다.

Create New Project

Create Android Project

Application name
WeatherInfo

Company domain
kbot2app.com

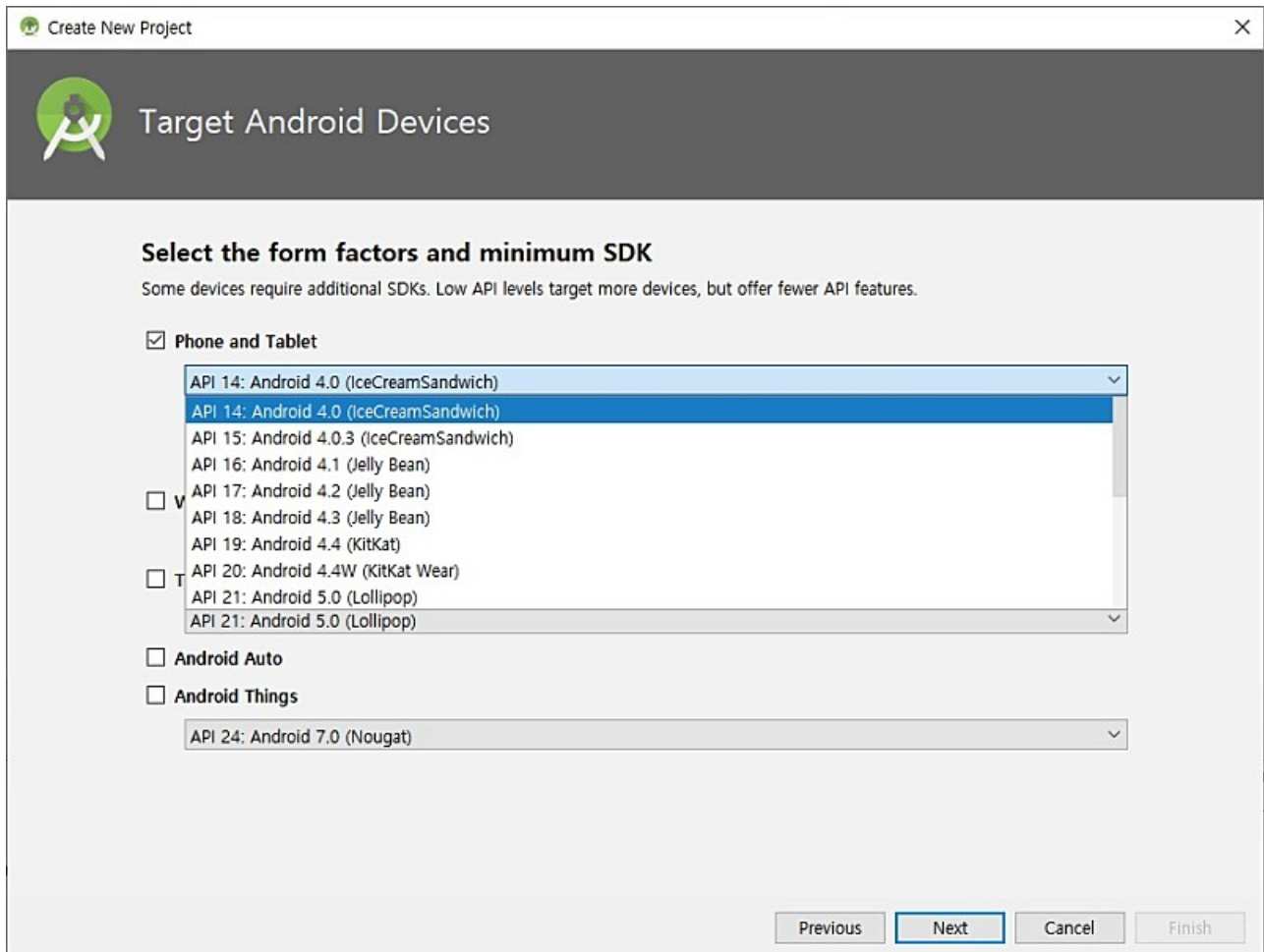
Project location
C:\Users\DarkTornado\AndroidStudioProjects\WeatherInfo

Package name
com.kbot2app.weatherinfo Edit

☐ Include C++ support
☐ Include Kotlin support

Previous Next Cancel Finish

해당 앱이 돌아갈 때 필요한 안드로이드의 최소 버전입니다.



Create New Project

Target Android Devices

Select the form factors and minimum SDK
Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ **Phone and Tablet**

API 14: Android 4.0 (IceCreamSandwich)
API 14: Android 4.0 (IceCreamSandwich)
API 15: Android 4.0.3 (IceCreamSandwich)
API 16: Android 4.1 (Jelly Bean)
API 17: Android 4.2 (Jelly Bean)
API 18: Android 4.3 (Jelly Bean)
API 19: Android 4.4 (KitKat)
API 20: Android 4.4W (KitKat Wear)
API 21: Android 5.0 (Lollipop)
API 21: Android 5.0 (Lollipop)

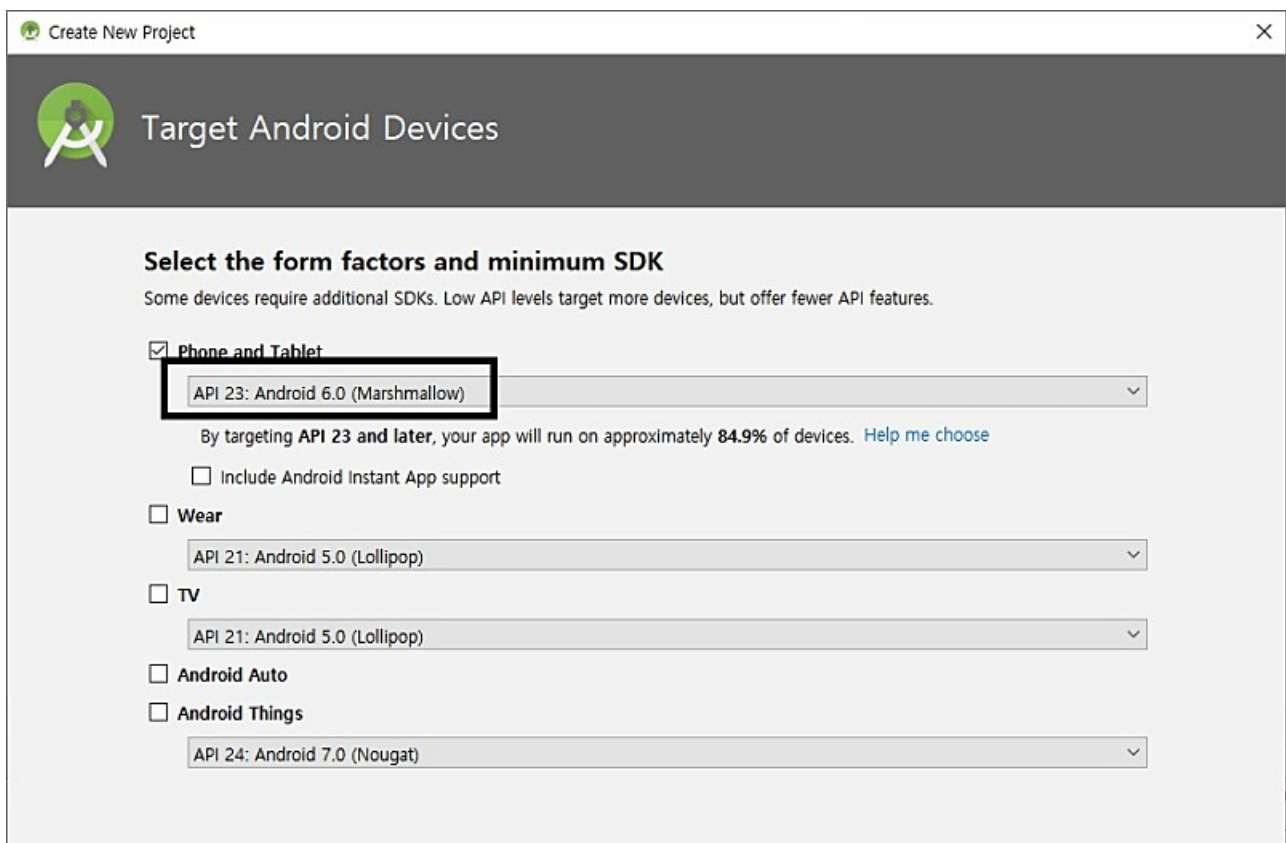
☐ **Android Auto**

☐ **Android Things**

API 24: Android 7.0 (Nougat)

Previous Next Cancel Finish

사실, 여기서 구현하려는 기능은 진저 이하로 해도 상관없지만, 이번에는 마시멜로로 해본 거시예오.



Create New Project

Target Android Devices

Select the form factors and minimum SDK
Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ **Phone and Tablet**

API 23: Android 6.0 (Marshmallow)

By targeting API 23 and later, your app will run on approximately 84.9% of devices. [Help me choose](#)

☐ Include Android Instant App support

☐ **Wear**

API 21: Android 5.0 (Lollipop)

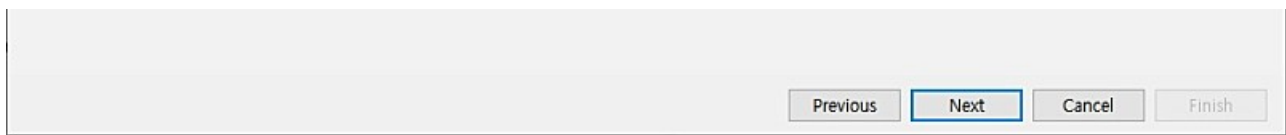
☐ **TV**

API 21: Android 5.0 (Lollipop)

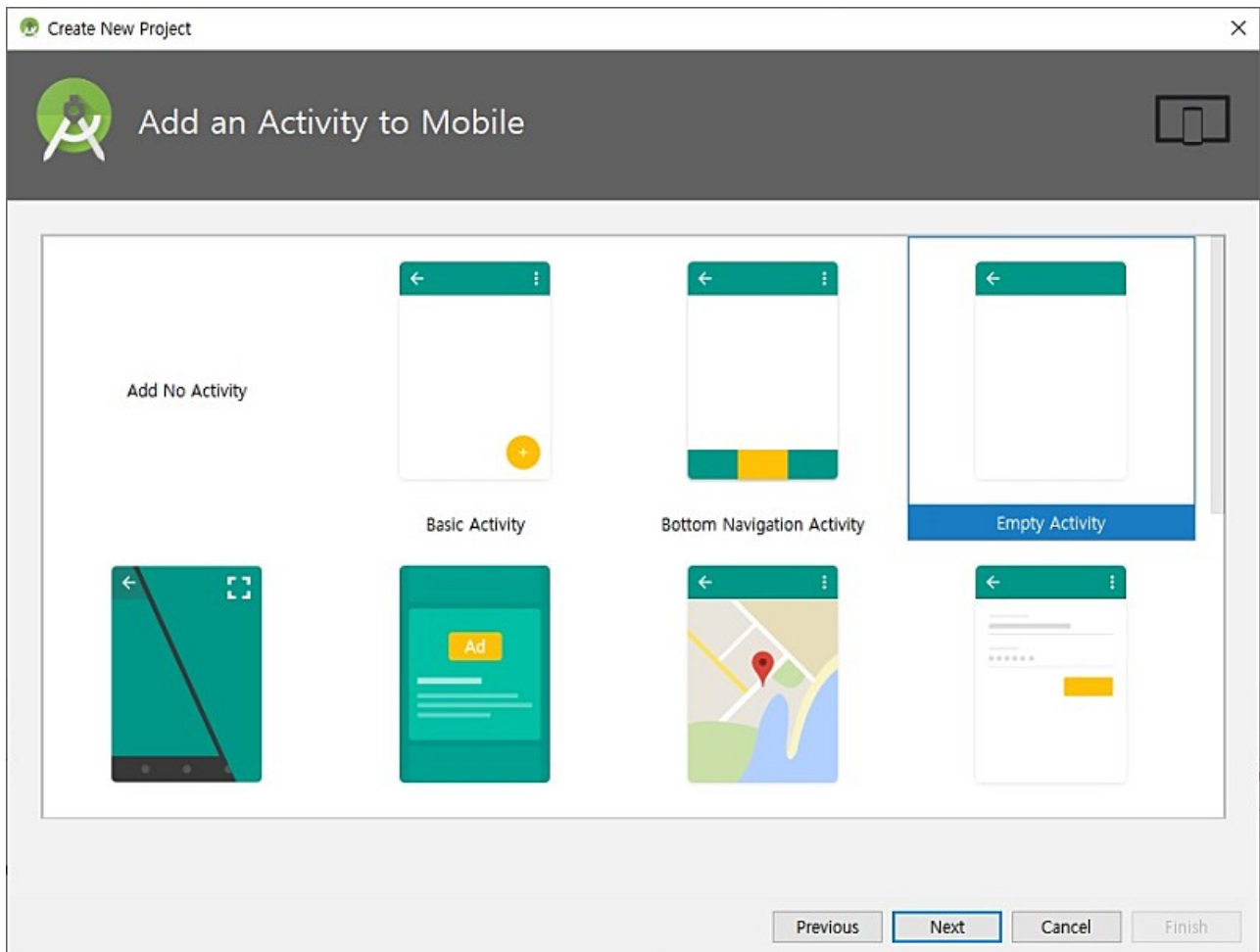
☐ **Android Auto**

☐ **Android Things**

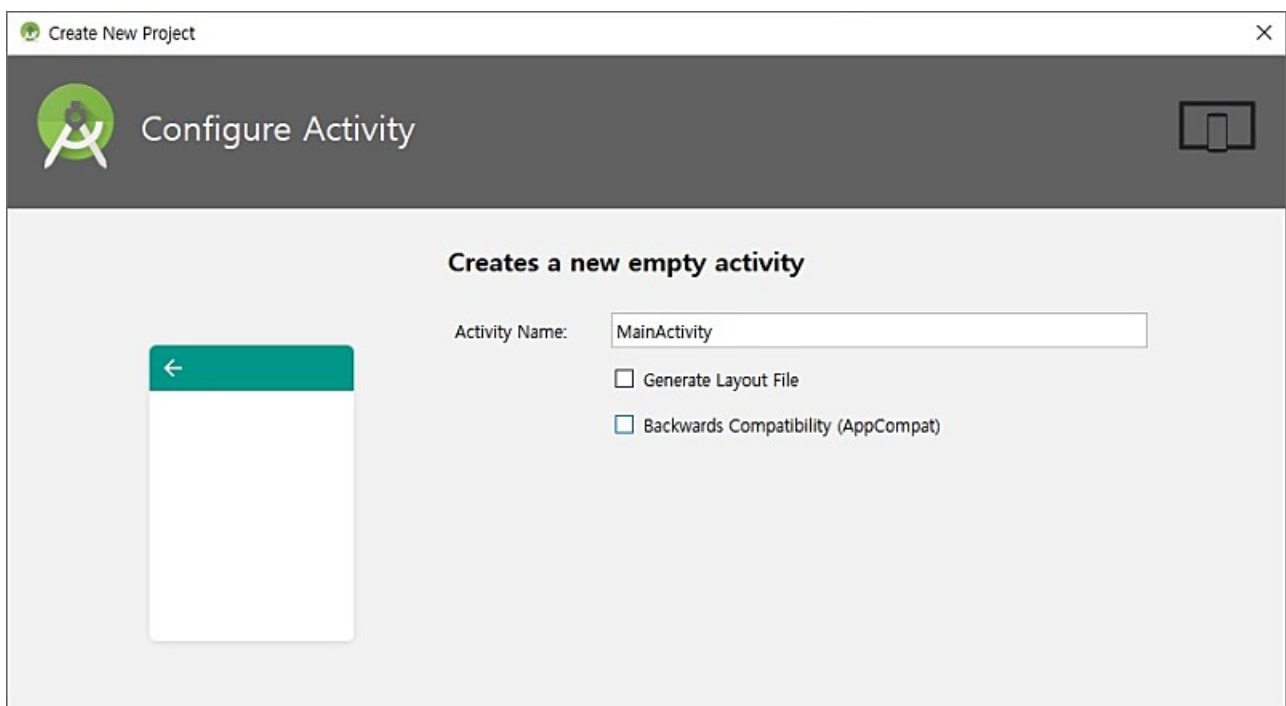
API 24: Android 7.0 (Nougat)



안스 최신버전에는 없다는 소문을 들었던 것 같지만, 알아서 EmptyActivity 선택.



xml은 사용하지 않을 것이니, 레이아웃 파일 생성은 날려먹고, AppCompatActivity도 여기서는 굳이 필요하지 않으니 체크 해제.



If false, this activity base class will be Activity instead of AppCompatActivity

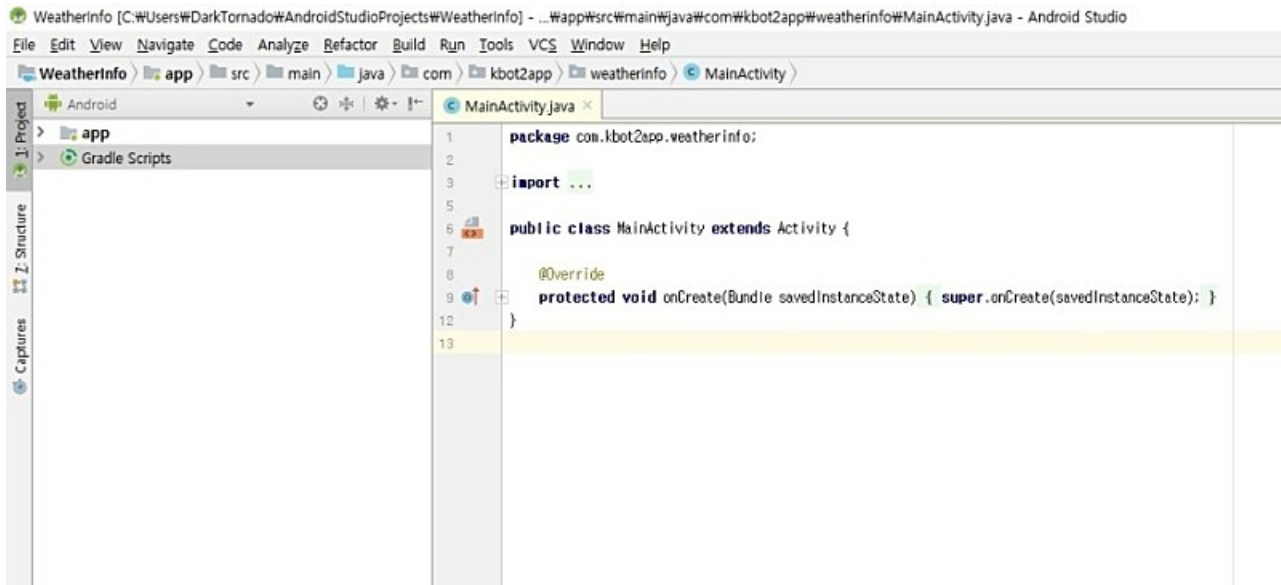
Previous

Next

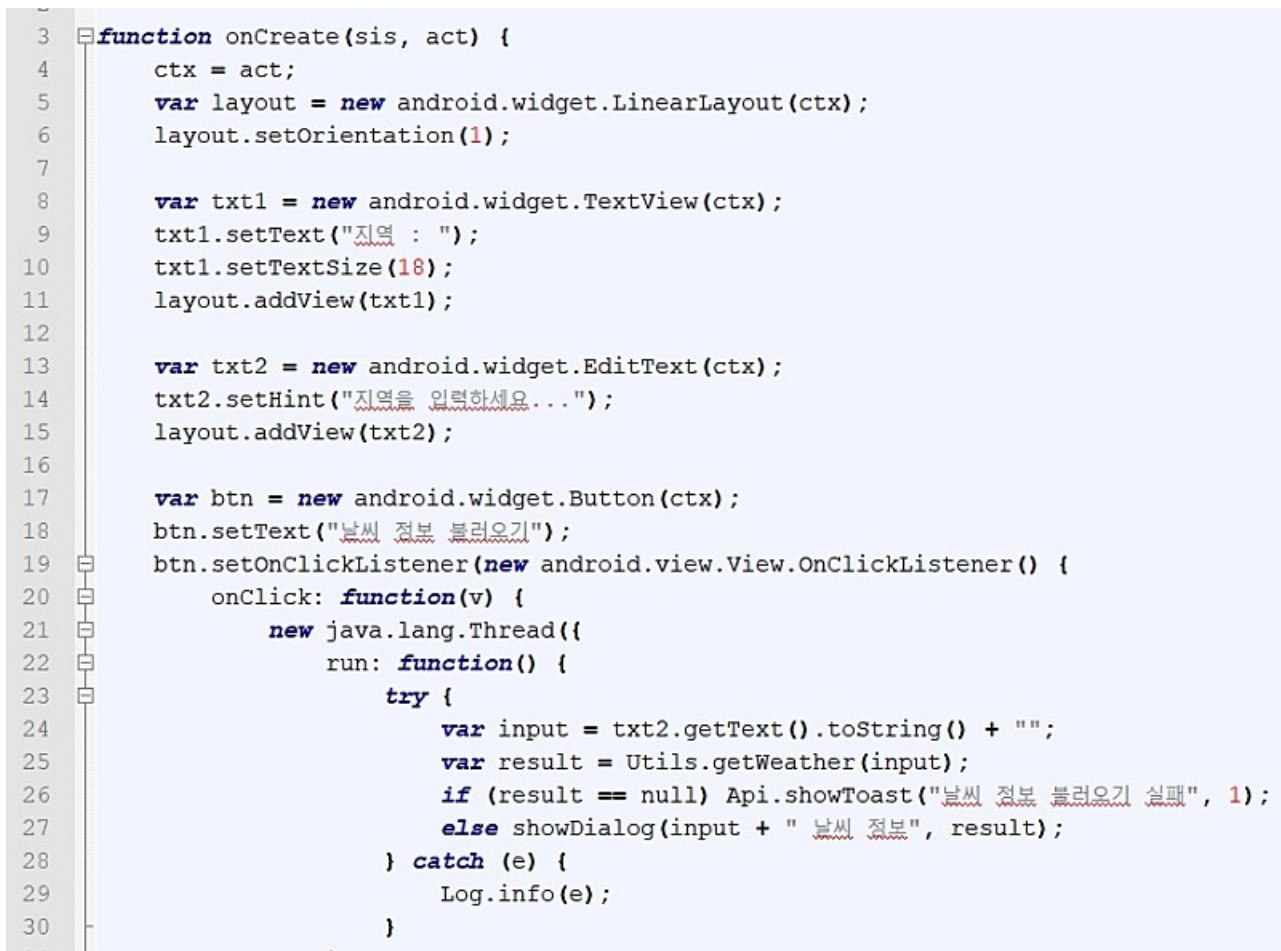
Cancel

Finish

08



이전에 만들었던 소스에 있는 onCreate에 있는걸



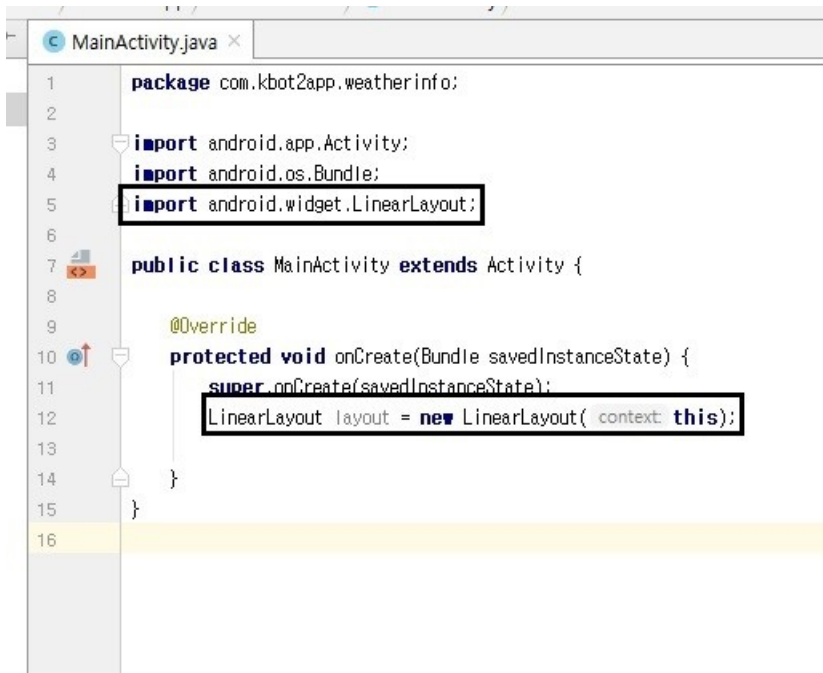
```

31         },
32         }).start();
33     }
34 });
35 layout.addView(btn);
36
37 act.setContentView(layout);
38 }
39

```

자바 문법으로 여기다가 옮길겁니다.

일단, 굳이 안하고 뺄것해도 되지만, import가 필요하니, LinearLayout을 import 시켜주고, 변수 선언시 var 대신 해당 변수에 들어갈 값의 자료형으로 바꾸어주고, Context에는 액티비티 자기 자신(?)인 this 투입.

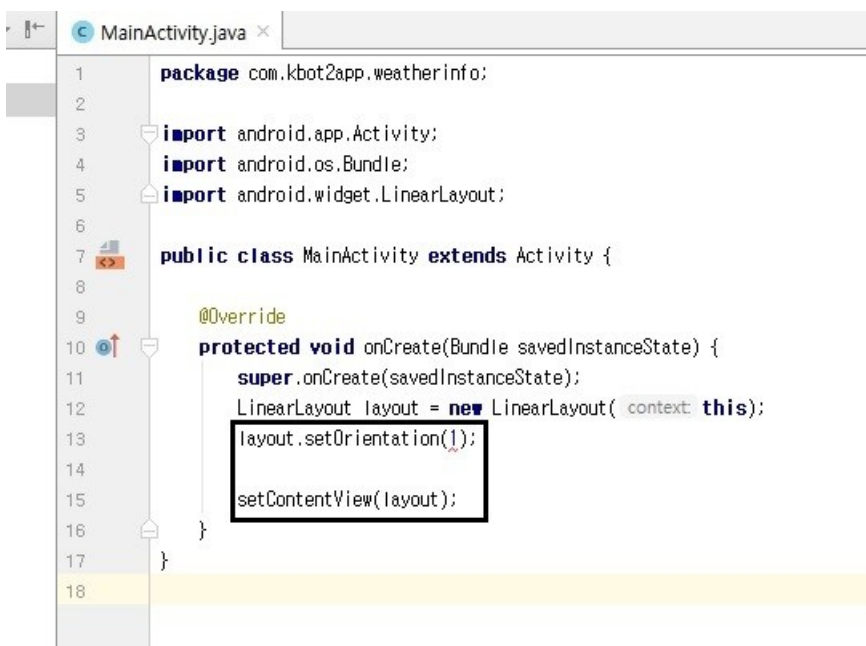


```

1 package com.kbot2app.weatherinfo;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.widget.LinearLayout;
6
7 public class MainActivity extends Activity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        LinearLayout layout = new LinearLayout(context, this);
13    }
14 }
15
16

```

레이아웃 세로정렬 후 그 레이아웃을 액티비티의 뷰로 설정.



```

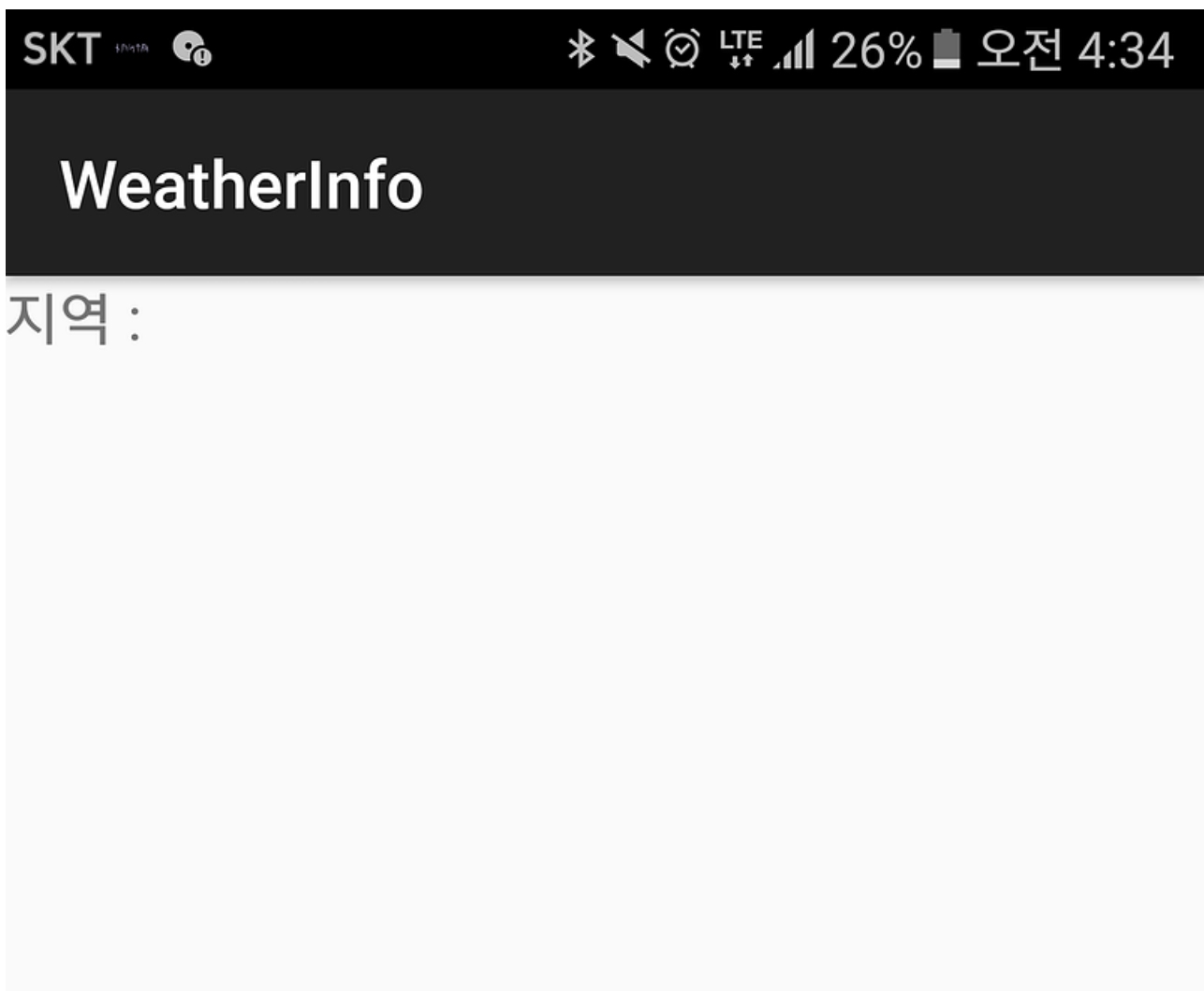
1 package com.kbot2app.weatherinfo;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.widget.LinearLayout;
6
7 public class MainActivity extends Activity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        LinearLayout layout = new LinearLayout(context, this);
13        layout.setOrientation(1);
14        setContentView(layout);
15    }
16 }
17
18

```

TextView도 같은 방식으로 넣어보았습니다.


```
MainActivity.java x
1 package com.kbot2app.weatherinfo;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.widget.LinearLayout;
6 import android.widget.TextView;
7
8 public class MainActivity extends Activity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         LinearLayout layout = new LinearLayout( context: this);
14         layout.setOrientation(1);
15
16         TextView txt1 = new TextView( context: this);
17         txt1.setText("지역 : ");
18         txt1.setTextSize(18);
19         layout.addView(txt1);
20
21         setContentView(layout);
22     }
23 }
24
```

저 상태에서 앱을 빌드하면 이렇게 나옵니다.



같은 방식으로 EditText와 Button도 넣었습니다.

```
1 package com.kbot2app.weatherinfo;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import android.widget.EditText;
8 import android.widget.LinearLayout;
9 import android.widget.TextView;
10
11 public class MainActivity extends Activity {
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         LinearLayout layout = new LinearLayout( context: this);
17         layout.setOrientation(1);
18
19         TextView txt1 = new TextView( context: this);
20         txt1.setText("지역 : ");
21         txt1.setTextSize(18);
22         layout.addView(txt1);
```

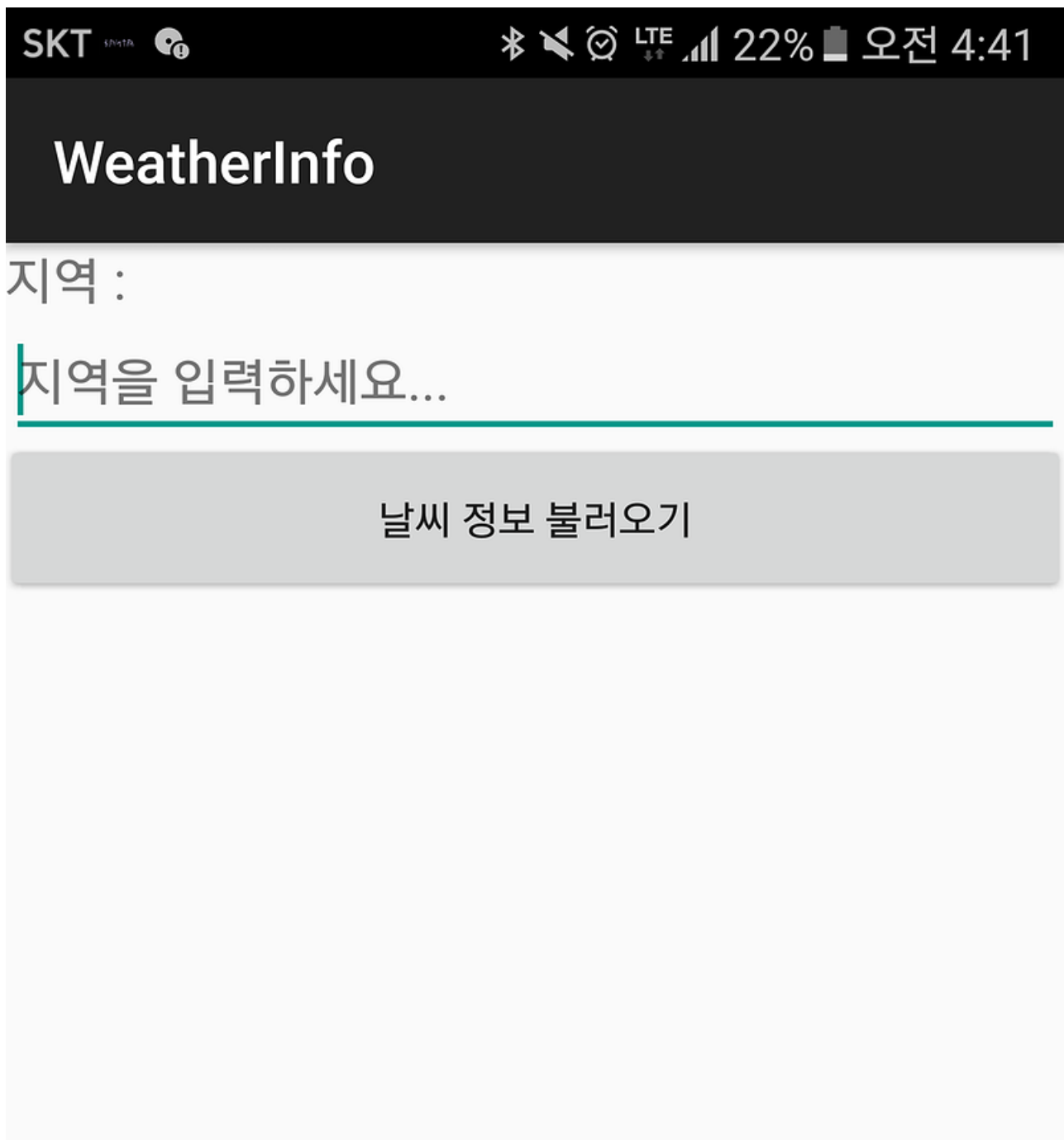


```

23
24 EditText txt2 = new EditText( context: this);
25 txt2.setHint("지역을 입력하세요...");
26 layout.addView(txt2);
27
28 Button btn = new Button( context: this);
29 btn.setText("날씨 정보 불러오기");
30 btn.setOnClickListener(new View.OnClickListener() {
31     @Override
32     public void onClick(View view) {
33
34     }
35 });
36 layout.addView(btn);
37
38 setContentView(layout);
39 }
40
41

```

일단 키면 눈에 보이는 부분은 완성했습니다.



이전 글에서 메인 UI에서는 파싱이 불가능하다는 것을 알게 되었을 겁니다.
저 안에서 새로 쓰레드를 열면 뭔가 블록이 깊어지는(?) 관계로, 다른 함수로 분리시켰습니다.

```
23
24     EditText txt2 = new EditText( context: this);
25     txt2.setHint("지역을 입력하세요...");
26     layout.addView(txt2);
27
28     Button btn = new Button( context: this);
29     btn.setText("날씨 정보 불러오기");
30     btn.setOnClickListener(new View.OnClickListener() {
31         @Override
32         public void onClick(View view) {
33             loadWeather(txt2.getText().toString());
34         }
35     });
36     layout.addView(btn);
37
38     setContentView(layout);
39 }
40
41 private void loadWeather(String input){
42     new Thread(new Runnable() {
43         @Override
44         public void run() {
45
46         }
47     }).start();
48 }
49
50
51
52
```

그런데, OnClickListener 내부에서 txt2 변수가지고 빼애앵거립니다.

앞에 final을 붙여주거나 전역변수로 바꾸어주면 해결됩니다. 상수 쓰라고 빼애앵거리는 거시예오.

```
21 txt1.setTextSize(10);
22 layout.addView(txt1);
23
24 final EditText txt2 = new EditText( context: this);
25 txt2.setHint("지역을 입력하세요...");
26 layout.addView(txt2);
27
28 Button btn = new Button( context: this);
29 btn.setText("날씨 정보 불러오기");
30 btn.setOnClickListener(new View.OnClickListener() {
31     @Override
32     public void onClick(View view) {
33         loadWeather(txt2.getText().toString());
34     }
35 });
36 layout.addView(btn);
37
38 setContentView(layout);
39 }
40
41 private void loadWeather(String input){
42     new Thread(new Runnable() {
43         @Override
```

기존 카톡봇 소스를 자바 문법으로 그대로 넣었는데, 안드로이드 스튜디오가 빼애앵거립니다.

```
1 private void loadWeather(String input){
2     new Thread(new Runnable() {
3         @Override
4         public void run() {
5             String result = Utils.getWeather(input);
6             if (result == null) Api.showToast("날씨 정보 불러오기 실패", 1);
7             else showDialog( id: input + " 날씨 정보", result);
8         }
9     }).start();
10 }
```

input을 상수로 바꾸고, 날씨를 읽어오는 함수도 일단 정의는 해두고,
Api.showToast():는 카톡봇에서 지원하는 함수이니, toast라는 토스트 메시지를 띄우는 함수를 만들고,
showDialog():라는 다이얼로그를 띄우는 함수로 바꾸어주었더니 잠잠해졌습니다.

```
1 private void loadWeather(final String input){
2     new Thread(new Runnable() {
3         @Override
4         public void run() {
5             String result = getWeatherInfo(input);
6             if (result == null) toast( msg: "날씨 정보 불러오기 실패");
7             else showDialog( title: input + " 날씨 정보", result);
8         }
9     }).start();
10 }
11
12 private String getWeatherInfo(String pos){
13
14 }
15
16 public void showDialog(String title, String msg) {
17     try {
18         AlertDialog.Builder dialog = new AlertDialog.Builder( context: this);
19         dialog.setTitle(title);
20         dialog.setMessage(msg);
```

```

        dialog.setNegativeButton( text: "닫기", listener: null);
        dialog.show();
    } catch (Exception e) {
        toast(e.toString());
    }
}

private void toast(String msg){
    Toast.makeText( context: this, msg, Toast.LENGTH_SHORT).show();
}
}

```

위에 import 부분입니다.

```

import android.app.Activity;
import android.app.AlertDialog;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        LinearLayout layout = new LinearLayout( context: this);
        layout.setOrientation(1);

        TextView txt1 = new TextView( context: this);
        txt1.setText("지역 : ");
        txt1.setTextSize(18);
    }
}

```

이제 날씨 정보를 읽어오는 부분을 바꾸어봅시다.

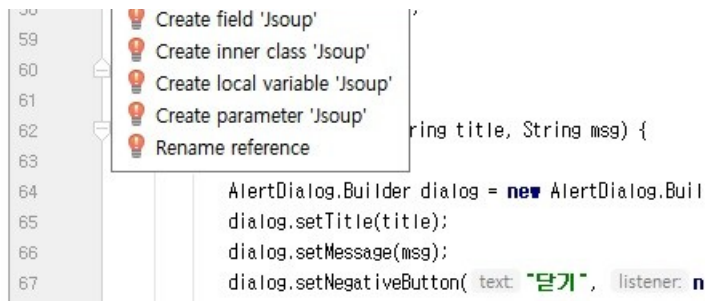
```

Utils.getWeather = function(pos) {
    try {
        var data = org.jsoup.Jsoup.connect("https://m.search.naver.com/search.naver?query=" + pos.replace(/ /g, "+") + "+날씨").get();
        data = data.select("div.status_wrap");
        var temp = data.select("strong").get(0).text();
        temp = temp.replace("°C", "°").replace("°", "℃");
        var hum = data.select("li.type_humidity").select("span").get(0).text();
        var state = data.select("div.weather_main").get(0).text();
        var dust = data.select("li.sign1").get(0).text();
        dust = dust.select("span.figure_result").text() + " (" + dust.select("span.figure_result").text() + ")";
        var result = "상태 : " + state + "\n" + temp + "\n" + hum + "\n" + dust;
        return result;
    } catch (e) {
        Log.error("날씨 정보 불러오기 실패\n" + e);
        return null;
    }
};

```

않아;; Jsoup이 없습니다.





원래 Jsoup 라이브러리는 외부 라이브러리입니다.

요즘 존재하는 카톡봇들이 다 지원해서 Jsoup이 원래 자바스크립트에 있는건줄 아는데, Jsoup은 자바로 만들어진 라이브러리로, 기본 제공이 아닙니다.

그러니, 라이브러리를 받아서 넣어줍니다.

Jsoup 사이트 주소 : <https://jsoup.org/>



사이트 접속 후 Download 클릭

jsoup

News

Bugs

Discussion

Download

API Reference

Cookbook

Try jsoup

jsoup » jsoup: Java HTML Parser

jsoup: Java HTML Parser

jsoup is a Java library for working with real-world HTML. It provides a very convenient API for fetching URLs and extracting and manipulating data, using the best of HTML5 DOM methods and CSS selectors.

jsoup implements the **WHATWG HTML5** specification, and parses HTML to the same DOM as modern browsers do.

- scrape and parse HTML from a URL, file, or string
- find and extract data, using DOM traversal or CSS selectors
- manipulate the HTML elements, attributes, and text
- clean user-submitted content against a safe white-list, to prevent XSS attacks
- output tidy HTML

jsoup is designed to deal with all varieties of HTML found in the wild; from pristine and validating, to invalid tag-soup; jsoup will create a sensible parse tree.

Example

Fetch the **Wikipedia** homepage, parse it to a DOM, and select the headlines from the *In the news* section into a list of **Elements** (online sample, full source):

```
Document doc = Jsoup.connect("https://en.wikipedia.org/").get();
log(doc.title());
Elements newsHeadlines = doc.select("#mp-itn b a");
```

Cookbook contents

Introduction

1. Parsing and traversing a Document

Input

2. Parse a document from a String
3. Parsing a body fragment
4. Load a Document from a URL
5. Load a Document from a File

Extracting data

6. Use DOM methods to navigate a document
7. Use selector-syntax to find elements
8. Extract attributes, text, and HTML from elements
9. Working with URLs
10. Example program: list links

Modifying data

11. Set attribute values
12. Set the HTML of an element

core library 앞에 있는 저거(?) 클릭

jsoup

News

Bugs

Discussion

Download

API Reference

Cookbook

Try jsoup

jsoup » Download and install jsoup

Download and install jsoup

jsoup is available as a downloadable .jar java library. The current release version is 1.13.1.

- **jsoup-1.13.1.jar** core library
- jsoup-1.13.1-sources.jar optional sources jar
- jsoup-1.13.1-javadoc.jar optional javadoc jar

What's new

See the 1.13.1 release announcement for the latest changes, or the changelog for the full history.

Previous releases of jsoup are also available.

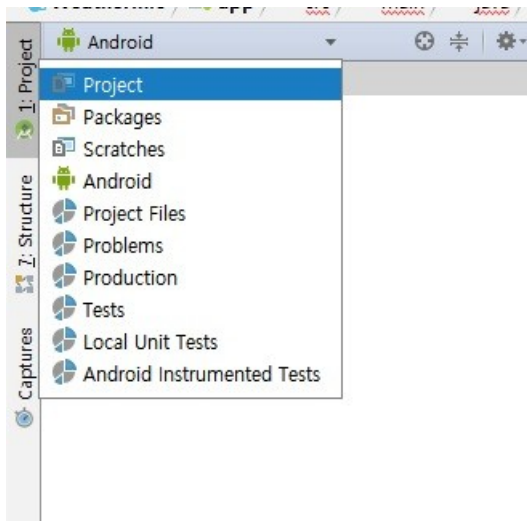
Maven

If you use Maven to manage the dependencies in your Java project, you do not need to download; just place the following into your POM's <dependencies> section:

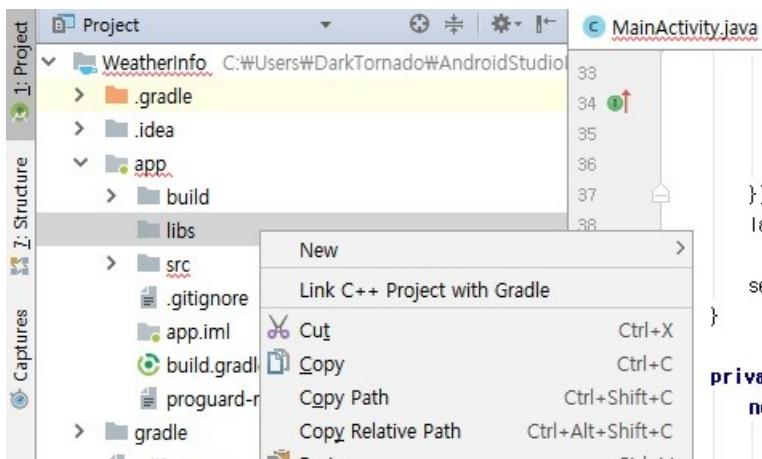
```
<dependency>
<!-- jsoup HTML parser library @ https://jsoup.org/ -->
<groupId>org.jsoup</groupId>
<artifactId>jsoup</artifactId>
<version>1.13.1</version>
```

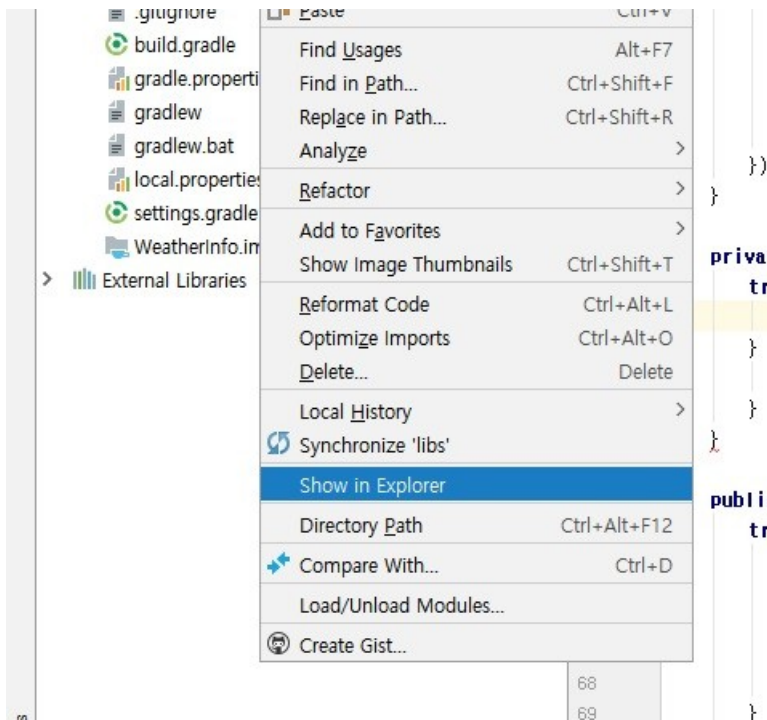
그러면 JSoup 라이브러리가 다운로드되기 시작할겁니다.

다운로드가 끝나면 이제 적용해야지요. 안스 오른쪽에 있는 파일 트리 비스무리하게 생긴거에 있는 Android를 Project로 변경

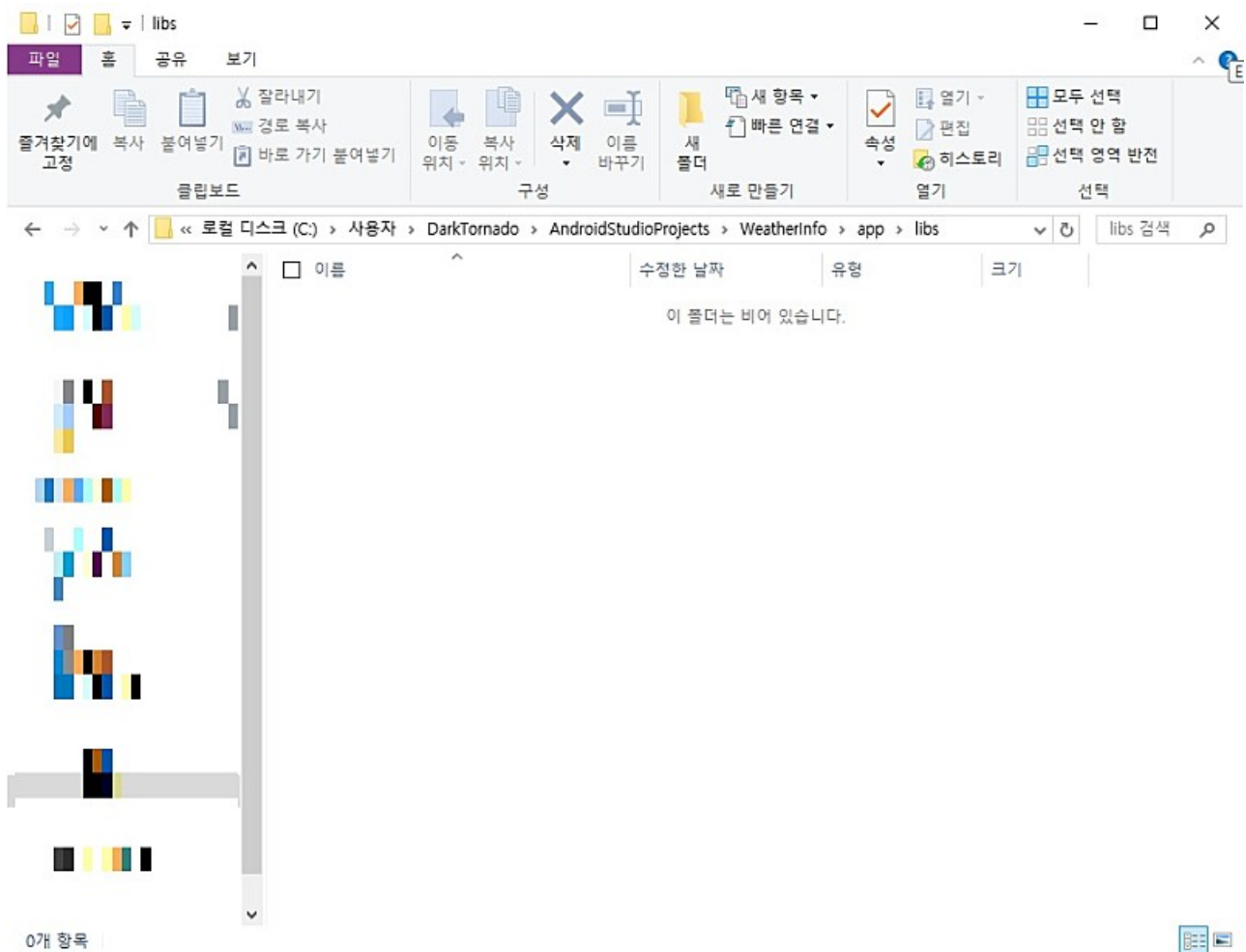


Project로 바뀌었으면 어떻게 잘 열어서(?) libs 폴더에 우클릭해서 Show in Explorer를 누르면 직접 찾아가든 해서



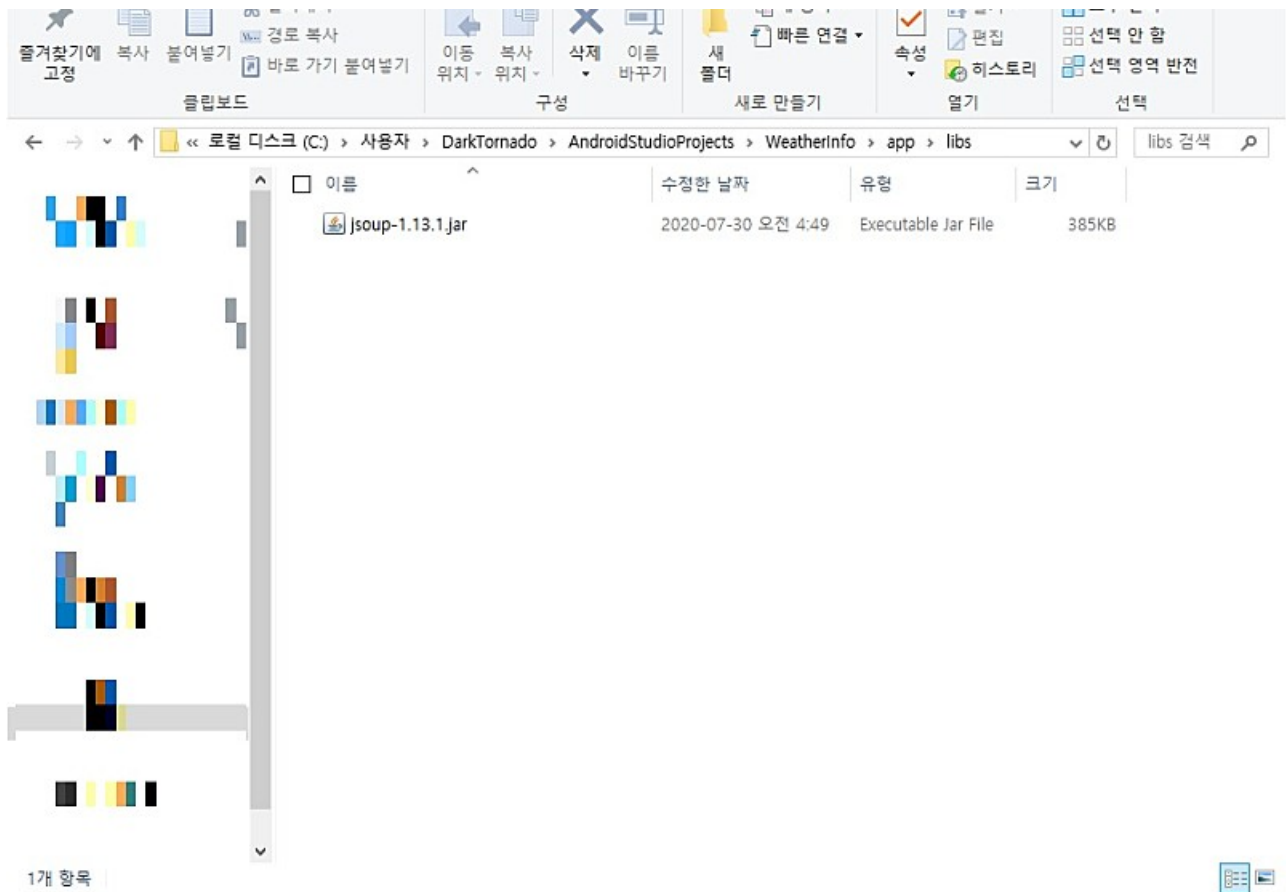


저 lib 폴더에 들어갑니다.

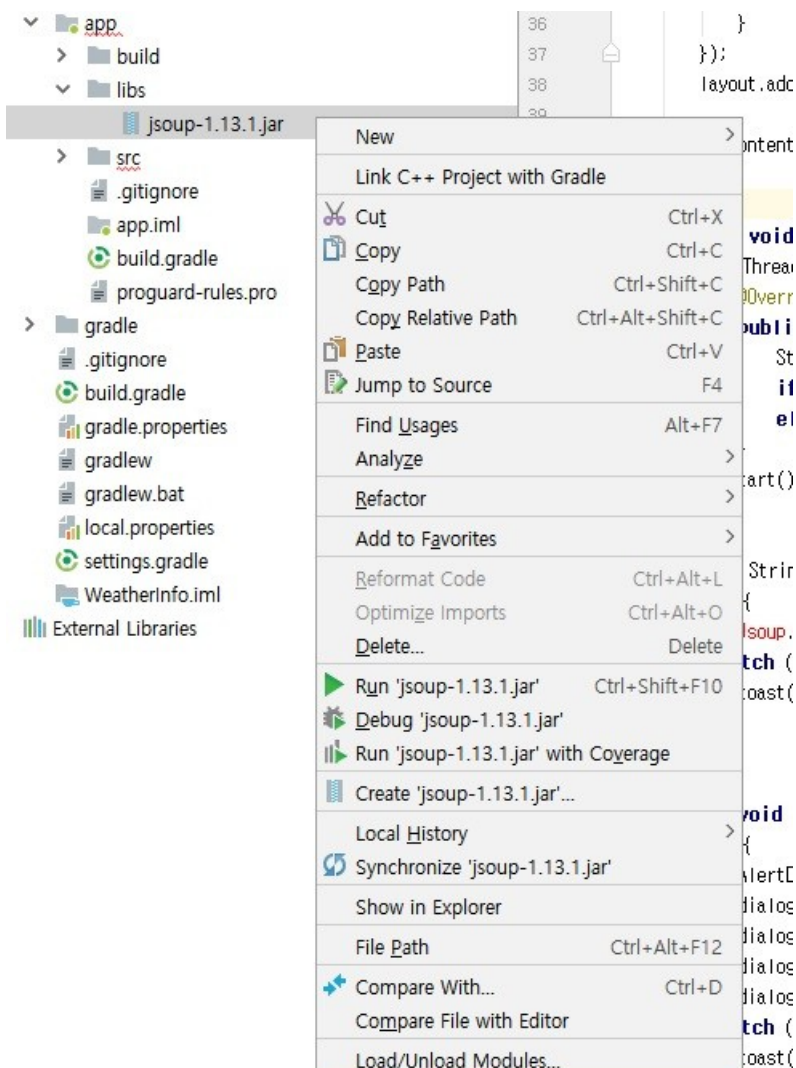


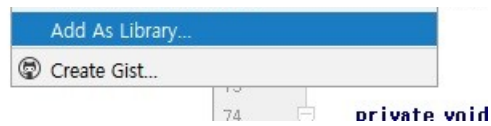
그리고 아까 받았던 JSoup 라이브러리가 들어간 .jar 파일을 저기다가 넣어줍니다.





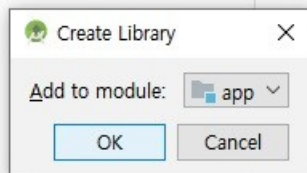
그러면 이렇게 뜬 것에서 우클릭을 하고, Add As Library...



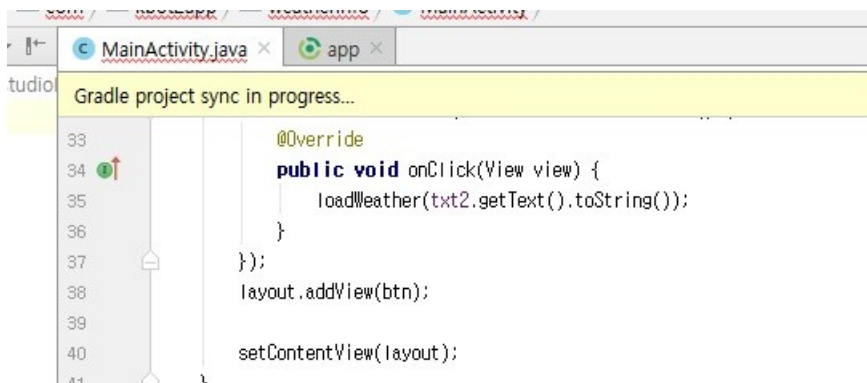


OK

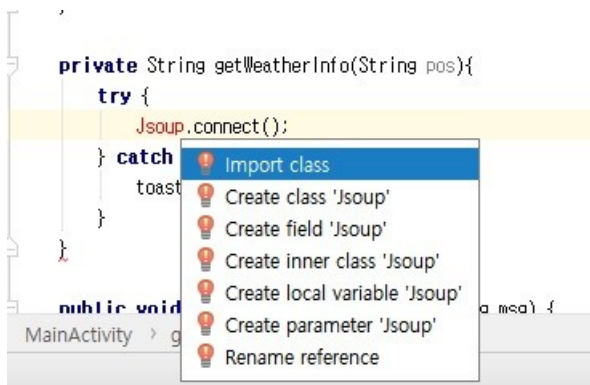
```
    불러오기 실패");
    , result);
```



그럼 무언가가 막 돌아갈겁니다. 동컴의 경우 오래걸릴지도 모릅니다.



이제는 Jsoup이라는게 있다고 뜹니다! 아까는 없던 ~~import class~~ 등장



cts#WeatherInfo

일단, 날씨 정보를 긁어오는 부분 중 가장 위에 있던 줄을 자바로 작성하였습니다.
 .get(): 메소드로 요청을 날린 것의 결과물의 자료형은 Document입니다.



```

private String getWeatherInfo(String pos){
    try {
        Document data = Jsoup.connect( url: "https://m.search.naver.com/search.naver?query=" + pos.replace(/_/g, "-") + "+날씨").get();
    } catch (Exception e) {
        toast(e.toString());
    }
}

public void showDialog(String title, String msg) {
    try {
        AlertDialog.Builder dialog = new AlertDialog.Builder( context: this);
        dialog.setTitle(title);
        dialog.setMessage(msg);
        dialog.setNegativeButton( text: "닫기", listener: null);
        dialog.show();
    } catch (Exception e) {
        toast(e.toString());
    }
}

```

위에 import 부분입니다.

```

8      import android.widget.EditText;
9      import android.widget.LinearLayout;
10     import android.widget.TextView;
11     import android.widget.Toast;
12
13     import org.jsoup.Jsoup;
14     import org.jsoup.nodes.Document;
15
16     public class MainActivity extends Activity {
17
18         @Override
19         protected void onCreate(Bundle savedInstanceState) {
20             super.onCreate(savedInstanceState);
21             LinearLayout layout = new LinearLayout( context: this);
22             layout.setOrientation(1);
23
24             TextView txt1 = new TextView( context: this);
25             txt1.setText("지역 : ");

```

저번 글에서 언급한 바와 같이, 자바에서는 .replace(문자열1, 문자열2);를 하면 모든 문자열1을 문자열2로 바꾸니, 여기서도 자바식(?)으로 바꾸어줍니다.

```

    }).start();
}

private String getWeatherInfo(String pos){
    try {
        Document data = Jsoup.connect( url: "https://m.search.naver.com/search.naver?query=" + pos.replace( target: "-", replacement: "+") + "+날씨").get();
    } catch (Exception e) {
        toast(e.toString());
    }
}

public void showDialog(String title, String msg) {
    try {
        AlertDialog.Builder dialog = new AlertDialog.Builder( context: this);
        dialog.setTitle(title);
        dialog.setMessage(msg);
        dialog.setNegativeButton( text: "닫기", listener: null);
    }
}

```

.select():으로 특정 태그만 선택해서 넣고 싶은데, JS와는 달리, 다른 자료형인 값을 넣는게 안되는 거 같습니다.

변수 data에 저장될 자료형은 Document인데, Elements라는 자료형을 넣으려고 시도하니 오류 뿜뿜

```

private String getWeatherInfo(String pos){
    try {
        Document data = Jsoup.connect( url: "https://m.search.naver.com/search.naver?query=" + pos.replace( target: "-", replacement: "+") + "+날씨").get();
        data = data.select( cssQuery: "div.status_wrap");
    }
}

```

Incompatible types.
Required: `org.jsoup.nodes.Document`
Found: `org.jsoup.select.Elements`

어차피 메소드 체이닝을 지원하지 않으니 뒤에다가 붙이고, 처음부터 자료형이 Elements였던 것으로 바꾸어주면 됩니다.
한 줄이 너무 길어지는 관계로, url 부분은 따로 변수를 만들어서 분리시켰습니다.

```

}

private String getWeatherInfo(String pos){
    try {
        String url = "https://m.search.naver.com/search.naver?query=" + pos.replace( target: " ", replacement: "+") + "+날씨";
        Elements data = Jsoup.connect(url).get().select( cssQuery: "div.status_wrap");

    } catch (Exception e) {
        toast(e.toString());
    }
}

public void showDialog(String title, String msg) {
    try {
        AlertDialog.Builder dialog = new AlertDialog.Builder( context: this);
        dialog.setTitle(title);
        dialog.setMessage(msg);
    }
}

```

위에 import 부분도 수정

```

import androidx.fragment.app.Fragment;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;

import org.jsoup.Jsoup;
import org.jsoup.select.Elements;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        LinearLayout layout = new LinearLayout( context: this);
        layout.setOrientation(1);

        TextView txt1 = new TextView( context: this);
    }
}

```

온도 정보와 습도 정보, 상태는 그냥 var만 String으로 바꾸어주니 해결되었습니다.

```

private String getWeatherInfo(String pos){
    try {
        String url = "https://m.search.naver.com/search.naver?query=" + pos.replace( target: " ", replacement: "+") + "+날씨";
        Elements data = Jsoup.connect(url).get().select( cssQuery: "div.status_wrap");

        String temp = data.select("strong").get(0).text();
        temp = temp.replace( target: "현재 온도", replacement: "온도 : ").replace( target: "°", replacement: "℃");
        String hum = data.select("li.type_humidity").select("span").get(0).text();
        String state = data.select("div.weather_main").get(0).text();

    } catch (Exception e) {
        toast(e.toString());
    }
}

```

미세먼지 부분도 어찌저찌 잘 해결.

```

    }).start();
}

private String getWeatherInfo(String pos){
    try {
        String url = "https://m.search.naver.com/search.naver?query=" + pos.replace( target: " ", replacement: "+") + "+날씨";
        Elements data = Jsoup.connect(url).get().select( cssQuery: "div.status_wrap");

        String temp = data.select("strong").get(0).text();
        temp = temp.replace( target: "현재 온도", replacement: "온도 : ").replace( target: "° ", replacement: "℃");
        String hum = data.select("li.type_humidity").select("span").get(0).text();
        String state = data.select("div.weather_main").get(0).text();

        Element _dust = data.select("li.sign1").get(0);
        String dust = _dust.select( cssQuery: "span.figure_text").text() + " (" + _dust.select( cssQuery: "span.figure_result").text() + ")";

    } catch (Exception e) {
        toast(e.toString());
    }
}

public void showDialog(String title, String msg) {
    try {
        AlertDialog.Builder dialog = new AlertDialog.Builder( context: this);
        dialog.setTitle(title);
        dialog.setMessage(msg);
    }
}

```

위에 import 부분에 추가된 부분 사실 자동 완성시키거나 단축기(Alt + Enter) 사용시 알아서 import되는...음

```

import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import org.jsoup.Jsoup;
import org.jsoup.nodes.Element;
import org.jsoup.select.Elements;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

결과물을 반환시키는 부분과, 오류 발생시 null을 반환하는 부분을 추가했습니다.

```

private String getWeatherInfo(String pos){
    try {
        String url = "https://m.search.naver.com/search.naver?query=" + pos.replace( target: " ", replacement: "+") + "+날씨";
        Elements data = Jsoup.connect(url).get().select( cssQuery: "div.status_wrap");

        String temp = data.select("strong").get(0).text();
        temp = temp.replace( target: "현재 온도", replacement: "온도 : ").replace( target: "° ", replacement: "℃");
        String hum = data.select("li.type_humidity").select("span").get(0).text();
        String state = data.select("div.weather_main").get(0).text();

        Element _dust = data.select("li.sign1").get(0);
        String dust = _dust.select( cssQuery: "span.figure_text").text() + " (" + _dust.select( cssQuery: "span.figure_result").text() + ")";

        return "상태 : " + state + "\n" + temp + "\n습도 : " + hum + "\n미세먼지 : " + dust;
    } catch (Exception e) {
        toast(e.toString());
        return null;
    }
}

```

저번 글에서 언급하였던 바와 같이, 메인 스크린(UI 스크린)에서만 UI에 접근해서 이것저것 할 수 있습니다.

Api.UIThread(); 같은건 카톡봇에만 있어서 여기서는 사용할 수 없습니다.
대신, runOnUiThread();라는게 있습니다.

```
        setContentView(layout);
    }

    private void loadWeather(final String input){
        new Thread(new Runnable() {
            @Override
            public void run() {
                final String result = getWeatherInfo(input);
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        if (result == null) toast( msg: "날씨 정보 불러오기 실패");
                        else showDialog( title: input + " 날씨 정보", result);
                    }
                });
            }
        }).start();
    }

    private String getWeatherInfo(String pos){
        try {
            String url = "https://m.search.naver.com/search.naver?query=" + pos.replace(
            Elements data = Jsoup.connect(url).get().select( cssQuery: "div.status_wrap");
        }
    }
}
```

UI 쓰레드가 아니면 토스트도 못 띄웁니다.

오류 발생시 토스트 메시지로 오류 내용을 출력시키도록 만들었는데, 뭔가 메인 쓰레드가 아닌 곳에서 오류가 날 것 같은 켜리적 갓심이 드니, 토스트 메시지를 띄우는 부분도 UI 쓰레드에서 돌아가도록 만들었습니다.

```
        dialog.setTitle(title);
        dialog.setMessage(msg);
        dialog.setNegativeButton( text: "닫기", listener: null);
        dialog.show();
    } catch (Exception e) {
        toast(e.toString());
    }
}

private void toast(final String msg){
    runOnUiThread(new Runnable() {
        @Override
        public void run() {
            Toast.makeText(getApplicationContext(), msg, Toast.LENGTH_SHORT).show();
        }
    });
}
}
```

아니나다를까 오류가 뜹니다.



지역 :

서울

날씨 정보 불러오기

java.lang.SecurityException: Permission
denied (missing INTERNET permission?)

뭐야 영어로 머라머라 뜨는데 오류인가 어떻게 하지 빼애앵. 거기기만 하면 오류를 고칠 수 없습니다.
오류 내용 대강 읽어보니 권한이 없어서 저런 것 같습니다. 인터넷 권한이 없는거 아니냐는 문구도 있네요.

안드로이드 앱에는 권한이라는 것이 있습니다.

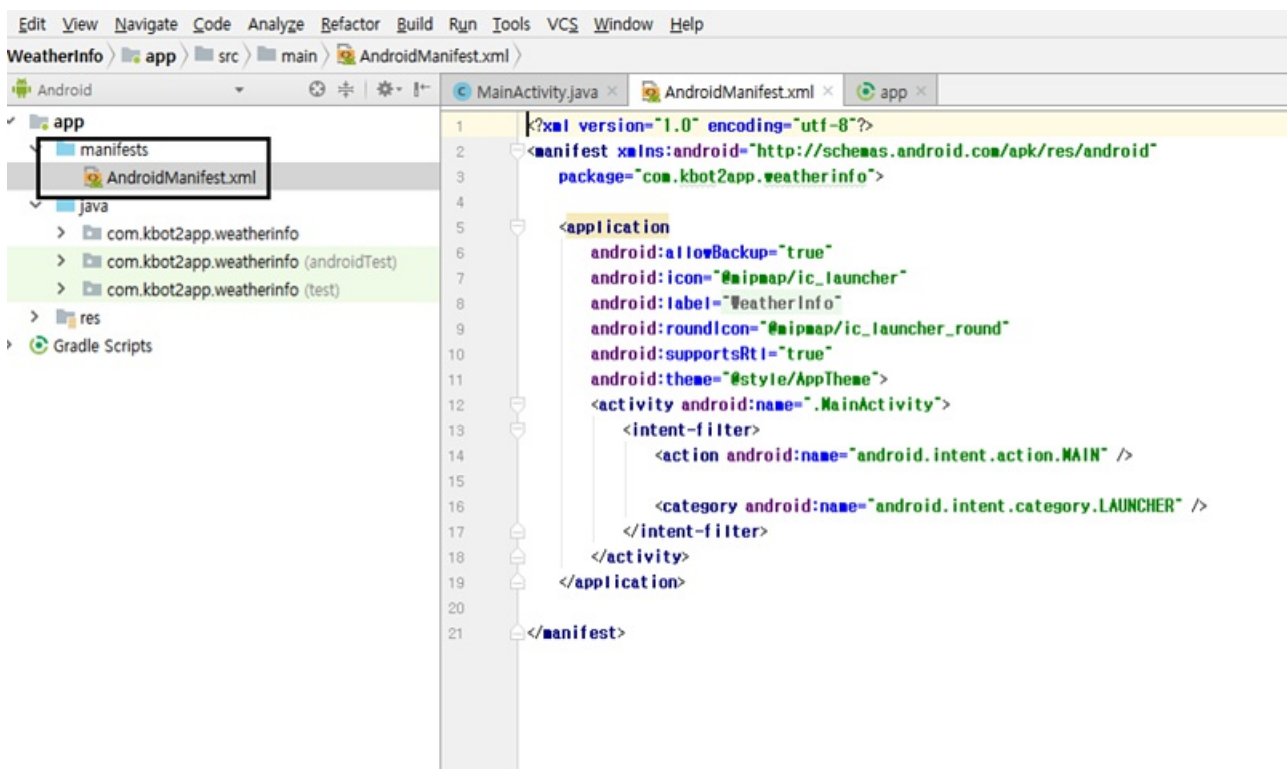
앱이 특정한 동작을 할 때 필요한 권한이 있어야 정상적으로 그 동작을 수행할 수 있습니다.

위의 경우, 인터넷 권한이 없기 때문에 웹에서부터 정보를 뜯어올(?)수 없는 상태입니다.

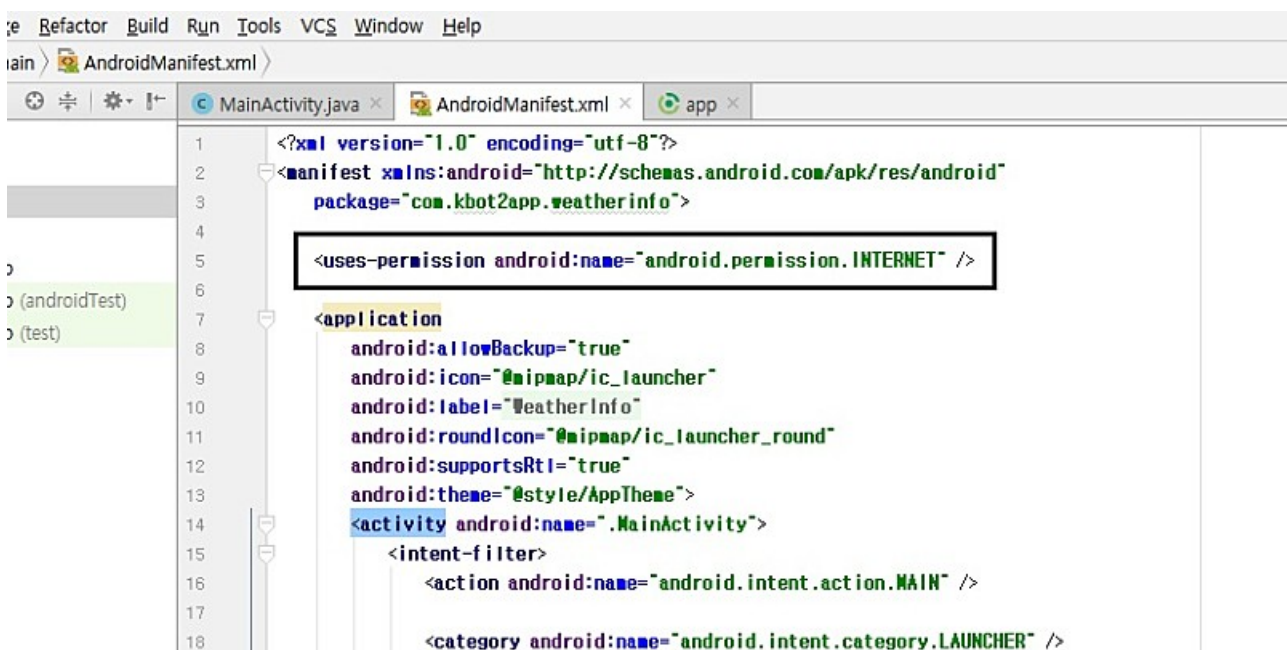
인터넷과 관련된 것을 하려면 인터넷 접근 권한이 필요합니다.

일부 권한들은 위험한 권한 또는 아주 위험한 권한으로 분류되는데, 뭇 구동 앱 설치시 권한 허용해달라고 뜨는 것들이 그런 권한들입니다.
위험한 또는 아주 위험한 권한은 사용자가 직접 허용해주어야 작동합니다.

인터넷 권한의 경우, 위험한 권한 등으로 분류되지는 않기에, AndroidManifest.xml 파일에 적어주기만 하면 됩니다.



다음



```

19 </intent-filter>
20 </activity>
21 </application>
22
23 </manifest>

```

아, 여기서 android:label 부분이 앱 이름입니다. 저거 바꾸면 앱 이름도 바뀌는 거시네요.
아이콘 바꾸는건 귀찮으니 패스

AndroidManifest.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3       package="com.kbot2app.weatherinfo">
4
5     <uses-permission android:name="android.permission.INTERNET" />
6
7     <application
8         android:allowBackup="true"
9         android:icon="@mipmap/ic_launcher"
10        android:label="@string/app_name"
11        android:roundIcon="@mipmap/ic_launcher_round"
12        android:supportRtl="true"
13        android:theme="@style/AppTheme">
14        <activity android:name=".MainActivity">
15            <intent-filter>
16                <action android:name="android.intent.action.MAIN" />
17
18                <category android:name="android.intent.category.LAUNCHER" />
19            </intent-filter>
20        </activity>
21    </application>
22
23 </manifest>

```

완성품(?)을 .apk 파일로 뽑아봅시다.

Android Studio Build menu

```

Tornado#AndroidStudioProjects#WeatherInfo - ...App#src#main#java#com#kbot2app#weatherinfo#MainActivity
File Edit View Build Tools Window Help
src main java Make Project Ctrl+F9 MainActivity
Make Module 'app'
Clean Project
Rebuild Project
Refresh Linked C++ Projects
Edit Build Types...
Edit Flavors...
Edit Libraries and Dependencies...
Select Build Variant...
Build APK(s)
Generate Signed APK...
Analyze APK...
Deploy Module to App Engine...

```

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

```

```

TextView txt1 = new TextView(context, this);
txt1.setText("기온 : ");
txt1.setTextSize(18);
layout.addView(txt1);

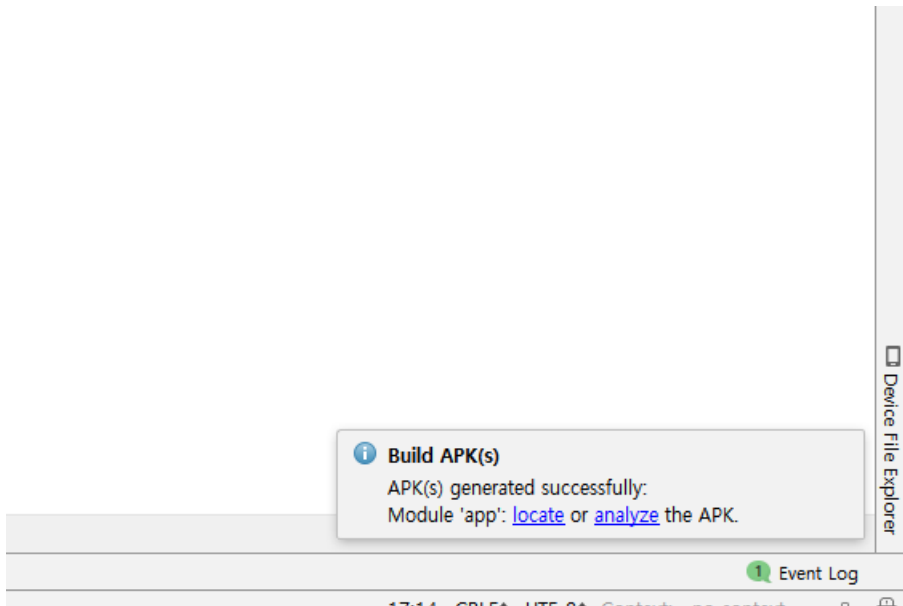
final EditText txt2 = new EditText(context, this);

```

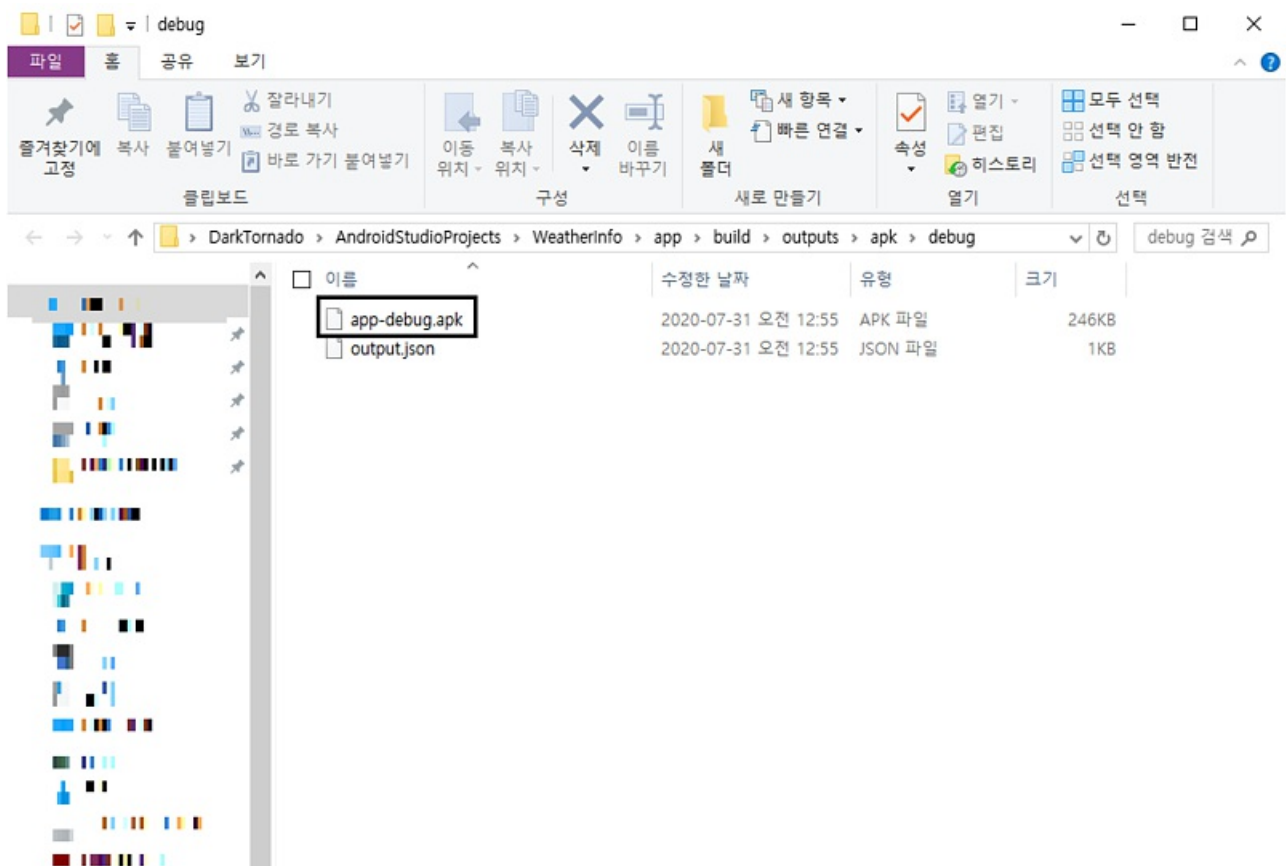
무엇인가가 돌아가다가,



이렇게 뜨면 끝난겁니다. locate 부분을 누르면 나오는



여기에서 app-debug.apk 파일이 서명이 되지 않은 .apk 파일입니다. Play 스토어에는 못올려요.





하어튼, 읍시 잘 작동합니다.

SKT TALK TALK TALK 5:10 PM

43% 오전 12:53

날씨 정보 불러오기

지역 :
서울

날씨 정보 불러오기

서울 날씨 정보

상태 : 맑음
온도 : 24℃
습도 : 86%
미세먼지 : 좋음 (11)

달기

필자는 여백이 없는 것이 몹시 불편한 관계로, 여백을 추가했습니다.

SKT TALK TALK SHINTEL

54% 오전 1:04

날씨 정보 불러오기

지역 :

지역을 입력하세요...

날씨 정보 불러오기

© 2020 Dark Tornado, All rights reserved.

앱 프로젝트는 깃허브에서 확인할 수 있는 거시예요.

<https://github.com/DarkTornado/Kbot2App>



DarkTornado/Kbot2App

카카오톡 봇 to 안드로이드 앱. Contribute to DarkTornado/Kbot2App development by creating an account on GitHub.
github.com

참고로, JSoup에는 MIT 라이선스가 적용되어 있으니, 알아서 잘 지키라는 거시예요.



Dark Tornado님의 게시글 더보기 >

❤️ 좋아요 11 💬 댓글 17

🔗 공유 신고

댓글 등록순 최신순

댓글알림



윤정성 님
정성 곳

2020.07.31. 09:44 답글쓰기

⋮



Dark Tornado 작성자

앗!

2020.08.03. 06:57 답글쓰기

⋮



이청연
좋은글감사합니다~

2020.07.31. 09:48 답글쓰기

⋮



도미 doami2005
감사합니다

2020.07.31. 09:52 답글쓰기

⋮

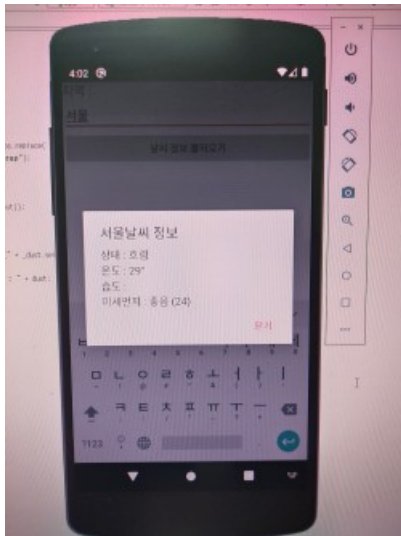


도미 doami2005

성공했습니다 감사합니다

날씨 불러올 때 java.net.SocketException socket failed EPERM 이라 뜨시는 분들은 앱 삭제 후 재설치 하시면 됩니다
30분 날렸네요

어라 다시보니 습도가 안나오네 오타친듯



2020.07.31. 13:05 답글쓰기



Dark Tornado 작성자

도미 doami2005 사실, 깃허브에 소스 있...

2020.08.03. 06:57 답글쓰기



맨들

Good

2020.07.31. 10:08 답글쓰기



Kiri



2020.07.31. 12:03 답글쓰기



NenkaLab

딕토상 Darcula 안써요?

2020.07.31. 13:04 답글쓰기



Dark Tornado 작성자

처음 선택할 때 허연걸 골라버린 거시예오

2020.08.03. 06:58 답글쓰기



NenkaLab

Dark Tornado 어서 설정을 가는 거시예오!

2020.08.03. 08:22 답글쓰기



Dark Tornado 작성자

NenkaLab 어림없는 거시예오


2020.08.03. 10:26 답글쓰기

$e^0 + 1$
= 0


Steve28

정.성.추


2020.07.31. 15:32 답글쓰기

- 


nolbo
 허억 이런 꿀강좌를 싸롱해요 닥토님
 2020.07.31. 19:01 답글쓰기

⋮
- 

마른얼음
 -음엑 그래들을 써주세요-
 2020.07.31. 22:08 답글쓰기

⋮
- 

Dark Tornado 작성자
 어라
 2020.08.03. 06:58 답글쓰기

⋮
- 

Sherry123
 감사합니다
 2020.08.04. 00:19 답글쓰기

⋮

Hibot

댓글을 남겨보세요



등록

 글쓰기 답글

목록 ▲ TOP

'| 기타 강좌 |' 게시판 글

구글 커스텀 검색 API써보기 🤖 [3] N

안녕하세요 [5]

봇 개발자에서 앱 개발자로 넘어가보기 2단계, 앱으로 똑같이 만들어보기 🤖 [17]

봇 개발자에서 앱 개발자로 넘어가보기 1단계, 스크립트 액티비티 사용 🤖 [7]

[스압] 원하는 기능 만드는 법 (날씨, 끝말잇기 등등) 🤖 [21]

이 게시판 새글 구독하기

Hibot	09:01
초보입니다CL	2020.08.02.
Dark Tornado	2020.07.31.
Dark Tornado	2020.07.30.
Dark Tornado	2020.07.28.
전체보기	

1 2 3

이 카페 인기글



ㅋㅋㅋㅋ....ㅋㅋㅋㅋㅋㅋㅋ ㅋㅋㅋㅋㅋㅋ

뽕뽕 배우는 인간

♡0 💬4

hajuhee01

♡0 💬10



사자성어 퀴즈 소스



카톡을 한폰에서 두개쓰는데 답장도 두계정

전부 다 합니다 ㅠㅠ

민심이 이상하게 흐르네요.

범이야

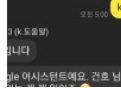
♡0 💬16

♡6 💬17

카링 사용하는데

CiBot

♡0 💬10



어시스턴트

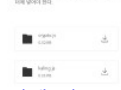
어시스턴트

어시스턴트

어시스턴트

어시스턴트

어시스턴트



이게 뭐죠? 모듈...