[실시간 영상 스트리밍 코드]

```
1  from flask import Flask, render_template, Response
2  from camera import Camera
3
4  app = Flask(__name__)
5  @app.route('/')
6
7  def index():
8    return render_template('index.html')
9
10 def gen(camera):
11   while True:
12     frame = camera.get_frame()
13     yield (b'--frame\r\n'b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n')
14
15 @app.route('/video_feed')
16
17 def video_feed():
18   return Response(gen(Camera()), mimetype='multipart/x-mixedreplace; boundary=frame')
19
20 if __name__ == '__main__':
21   app.run(host='0.0.0.0', debug=True, threaded=True)
22   camera.start_preview(fullscreen=False,window=(100,20,640,480))
```

① 1 ~ 4번 줄에서는 Flask 클래스 불러오기 및 Flask 클래스 객체 app 생성, 이어서 5번 줄에서는 생성한 객체의 route를 설정, 즉 URL 설정

② 스트리밍 실행시 보여지는 화면으로 연결, index.html 파일이 보여지게 된다.

③ 사진형식 설정

④ 객체의 run함수를 이용해서 어플리케이션을 실행하도록 한다. host='0.0.0.0'으로 변경하여서 외부에서 접근 가능하도록 설정(app.run() 으로 설정시 로컬 서버에서 실행)

[파이카메라 사진 촬영 코드]

```
1  from picamera import PiCamera
2  from time import sleep
3
4  camera = PiCamera()
5  camera.resolution = (960, 720)
6  camera.start_preview(fullscreen=False, window=(450,100,640,480))
7  camera.rotation = 180
8  i = 0
9  sleep(3)
10
11 while True :
12   sleep(2)
13   camera.capture('/home/pi/Desktop/image%s.jpg' % i)
14   i = i + 1
15
16 camera.stop_preview()
```

① 라즈베리파이에 연결한 파이카메라 사용을 위해 PiCamera 클래스 불러오기 및 시간 동기화를 위한 time 클래스 불러옴

② 사진 크기 설정(640 x 480 해상도 , 60, 90fps)

③ 사진을 찍어서 저장

④ 사진 촬영 정지

[인식 값에 따른 물 펌프 제어 코드]

```
import serial
from time import sleep

port = "/dev/ttyACM0"
serialFromArduino = serial.Serial(port, 9600)
serialFromArduino.flushInput()

z= 1

while True:

  f = open("/home/pi/Value"+str(z)+".txt", "r")
  z = z+1
  sleep(2)
  line = f.readline()

  if not line : break
    line = int(line)

  if line > 55:
    print("Person Detect")
    print("")
    serialFromArduino.write('1')

  else:

    print("Person Not Detect")
    print("")

f.close()
```

① 라즈베리파이와 아두이노 간의 시리얼 통신을 위한 설정

(유선 연결인 Serial 통신이 가장 속도가 빨라 사용하였음)

② 실시간 사람 인식률을 저장한 텍스트 파일을 계속해서 읽어 사람 인식률 확인

더 이상 읽을 값이 없으면 반복을 끝낸다.

③ 사람 인식률이 55% 이상일 경우 아두이노로 신호를 보내 물 펌프 동작을 제어한다.

[사진 전송 및 인식률 수신]

```
import socket
import base64
from time import sleep
import os

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.connect(('192.168.0.12', 50030))

z = 0
i = 0

while True:

    k = ""
    data = open("/home/pi/Desktop/image" + str(i) + ".jpg", "rb")
    encoded_string = base64.b64encode(data.read())
    size= len(encoded_string)

    print("Image Transmission : "+str(size) +" bytes")

    size = str(size)
    sock.send(size)
    sleep(1)
    sock.send(encoded_string)
    sleep(2)

    i = i + 1
    k = sock.recv(4)
    k = int(k)
    f = open("Value"+str(z)+".txt", 'w')
    f.write(str(k)+'\n')
    z = z+1

    f.close()
```

① 서버 컴퓨터와 소켓 통신을 하기 위한 연결 설정

② 사진을 전송하는 과정에서 사진 데이터를 text 형식으로 변경해주는 base64 모듈을 사용하여 Text 형식으로 전송, 사진 전송 전에 누락없이 사진을 받기 위해 사진 크기를 먼저 전송한다.

③ 사진 전송 후 서버 쪽에서 실시간으로 detect object processing을 해주기 때문에 인식률 값을 다시 받아서 텍스트 파일에 저장한다.

[YOLO 코드]

```
from ctypes import *
import math
import random
import socket
import base64
import os
from time import sleep


server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
server_socket.bind(('192.168.0.12', 50030))
server_socket.listen(1)
client_socket, addr = server_socket.accept()

i = 0


def sample(probs):
    s = sum(probs)
    probs = [a/s for a in probs]
    r = random.uniform(0, 1)
    for i in range(len(probs)):
        r = r - probs[i]
        if r <= 0:
            return i
    return len(probs)-1

def c_array(ctype, values):
    arr = (ctype*len(values))()
    arr[:] = values
    return arr

class BOX(Structure):
    _fields_ = [("x", c_float),
                ("y", c_float),
                ("w", c_float),
                ("h", c_float)]

class DETECTION(Structure):
    _fields_ = [("bbox", BOX),
                ("classes", c_int),
```

```
                ("prob", POINTER(c_float)),
                ("mask", POINTER(c_float)),
                ("objectness", c_float),
                ("sort_class", c_int)]


class IMAGE(Structure):
    _fields_ = [("w", c_int),
                ("h", c_int),
                ("c", c_int),
                ("data", POINTER(c_float))]

class METADATA(Structure):
    _fields_ = [("classes", c_int),
                ("names", POINTER(c_char_p))]



#lib = CDLL("/home/pjreddie/documents/darknet/libdarknet.so", RTLD_GLOBAL)
lib = CDLL("/home/hbb/darknet/libdarknet.so", RTLD_GLOBAL)
lib.network_width.argtypes = [c_void_p]
lib.network_width.restype = c_int
lib.network_height.argtypes = [c_void_p]
lib.network_height.restype = c_int

predict = lib.network_predict
predict.argtypes = [c_void_p, POINTER(c_float)]
predict.restype = POINTER(c_float)

set_gpu = lib.cuda_set_device
set_gpu.argtypes = [c_int]

make_image = lib.make_image
make_image.argtypes = [c_int, c_int, c_int]
make_image.restype = IMAGE

get_network_boxes = lib.get_network_boxes
get_network_boxes.argtypes = [c_void_p, c_int, c_int, c_float, c_float, POINTER(c_int)]
get_network_boxes.restype = POINTER(DETECTION)

make_network_boxes = lib.make_network_boxes
make_network_boxes.argtypes = [c_void_p]
make_network_boxes.restype = POINTER(DETECTION)

```

```
free_detections = lib.free_detections
free_detections.argtypes = [POINTER(DETECTION), c_int]

free_ptrs = lib.free_ptrs
free_ptrs.argtypes = [POINTER(c_void_p), c_int]

network_predict = lib.network_predict
network_predict.argtypes = [c_void_p, POINTER(c_float)]

reset_rnn = lib.reset_rnn
reset_rnn.argtypes = [c_void_p]

load_net = lib.load_network
load_net.argtypes = [c_char_p, c_char_p, c_int]
load_net.restype = c_void_p

do_nms_obj = lib.do_nms_obj
do_nms_obj.argtypes = [POINTER(DETECTION), c_int, c_int, c_float]

do_nms_sort = lib.do_nms_sort
do_nms_sort.argtypes = [POINTER(DETECTION), c_int, c_int, c_float]

free_image = lib.free_image
free_image.argtypes = [IMAGE]

letterbox_image = lib.letterbox_image
letterbox_image.argtypes = [IMAGE, c_int, c_int]
letterbox_image.restype = IMAGE

load_meta = lib.get_metadata
lib.get_metadata.argtypes = [c_char_p]
lib.get_metadata.restype = METADATA

load_image = lib.load_image_color
load_image.argtypes = [c_char_p, c_int, c_int]
load_image.restype = IMAGE

rgbgr_image = lib.rgbgr_image
rgbgr_image.argtypes = [IMAGE]

predict_image = lib.network_predict_image
predict_image.argtypes = [c_void_p, IMAGE]
predict_image.restype = POINTER(c_float)

```

```
131 def classify(net, meta, im):
132     out = predict_image(net, im)
133     res = []
134     for i in range(meta.classes):
135         res.append((meta.names[i], out[i]))
136     res = sorted(res, key=lambda x: -x[1])
137     return res
138 
139 def detect(net, meta, image, thresh=.5, hier_thresh=.5, nms=.45):
140     im = load_image(image, 0, 0)
141     num = c_int(0)
142     pnum = pointer(num)
143     predict_image(net, im)
144     dets = get_network_boxes(net, im.w, im.h, thresh, hier_thresh, None, 0, pnum)
145     num = pnum[0]
146     if (nms): do_nms_obj(dets, num, meta.classes, nms);
147 
148     res = []
149     for j in range(num):
150         for i in range(meta.classes):
151             if dets[j].prob[i] > 0:
152                 b = dets[j].bbox
153                 res.append((meta.names[i], dets[j].prob[i], (b.x, b.y, b.w, b.h)))
154     res = sorted(res, key=lambda x: -x[1])
155     free_image(im)
156     free_detections(dets, num)
157     return res
158 
159 def isNumber(s):
160     try:
161         int(s)
162         return True
163     except ValueError:
164         return False
165 
166 if __name__ == "__main__":
167     while True:
168         size = ""
169         data = ""
170         size = client_socket.recv(6)
171 
172         if(isNumber(size) == False):
173             continue
174 
```

```
175         if(len(size) != 6):
176             continue
177 
178         size = int(size)
179 
180         while(len(data) != size):
181             data += client_socket.recv(100000)
182 
183         print("Receive Image : " + str(size)+" bytes")
184         f = open("yolo" + str(i) + ".jpg", "wb")
185         f.write(base64.b64decode(data))
186         net = load_net("cfg/yolov3-tiny.cfg", "yolov3-tiny.weights", 0)
187         meta = load_meta("cfg/coco.data")
188         r = detect(net, meta, "/home/hbb/darknet/yolo"+str(i)+".jpg")
189         h = len(r)
190         j = 0
191 
192         if h == 0:
193             i += 1
194 
195         client_socket.send("0")
196         continue
197 
198         else:
199             for j in range(0, h):
200                 if(r[j][0] == 'person'):
201                     r = r[j][1]
202 
203         r = r*100;
204         r = int(r)
205         print("Person Recognition Rate : "+str(r)+"%")
206         r = str(r)
207         client_socket.send(r)
208         sleep(1)
209 
210         f.close()
211 
212         i += 1
```

① 라즈베리파이와 서버 컴퓨터 간의 통신을 위한 소켓 연결 설정 단계

② 사진 전송 전에 사진 크기를 전송한다. 사진 크기를 먼저 받는 이유는 사진 데이터 누락없이 다 받기 위해서 그 데이터를 다 받을 때까지 반복하기 위함이다.

③ 사진 크기를 먼저 받는다.

④ 사진을 받은 후 base64로 디코딩하여 text 형식에서 바이너리 형식으로 변경 후 YOLO API를 사용하여 사진을 분석하여 사람 인식을 하는 과정.

⑤ YOLO API의 결과로 사람 인식률을 받아서 사람 인식률에 따른 물 펌프 제어를 위해 인식률 전송

[아두이노 신호 전송 코드]

```
1 import serial
2 from time import sleep
3 port = "/dev/ttyACM0"
4 serialFromArduino = serial.Serial(port, 9600)
5 serialFromArduino.flushInput()
6 z= 1
7
8 while True:
9     f = open("/home/pi/Value"+str(z)+".txt", "r")
10    z = z+1
11    sleep(2)
12    line = f.readline()
13
14    if not line : break
15        line = int(line)
16
17    if line > 55:
18            print("Person Detect")
19            print("")
20            serialFromArduino.write('1')
21    else:
22            print("Person Not Detect")
23            print("")
24            serialFromArduino.write('2')
25
26    f.close()
```

① 사람 인식률이 저장된 파일을 읽는다.

② 사람 인식률이 55% 이상이면 아두이노에게 물 펌프 작동을 멈추라는 의미로 '1'이라는 신호를 전송한다. 55% 이하일 경우는 드론과 사람 사이의 거리가 먼 거리이기 때문에 해충 기피제를 분사하여도 상관이 없기 때문에 계속해서 분사하라는 의미로 '2'를 전송해줍니다.

[아두이노 물 펌프 제어 코드]

```
int pinnum=13;

void setup()
{
  pinMode(pinnum,OUTPUT);
  Serial.begin(9600);
  Serial.println("1 to turn off");
}

void loop()
{
  if(Serial.available())
    {
        int res =Serial.parseInt();

        if(res==1)
        {
              digitalWrite(pinnum, LOW);
              delay(1000);
        }

        else if(res==2)
        {
             digitalWrite(pinnum, HIGH);
             delay(1000);
             digitalWrite(pinnum, LOW);
             delay(1000);
        }
    }
}
```

① 아두이노와 물 펌프 간의 통신을 위한 연결 설정

② 아두이노 코드에서 loop문을 계속해서 반복하는데 라즈베리파이에서 아두이노로 사람이 사람이 있을 경우 1을 전송하고 없을 경우 2를 전송하므로 그에 따른 물 펌프 동작을 제어

[LIDAR 센서 및 Servo Motor 제어]

```
#include <Servo.h>

Servo myservo;
int m =20;

void setup()
{
    myservo.attach(9);//서브모터 제어핀
    Serial.begin(9600);
}

void loop()
{
    myservo.write(m); // 각도 0 도로 이동
    delay(1000);

    if(m<100)
        m+=10;

    else
        m=20;

    int sensorValue = analogRead(A0);
    float distance = 12343.85 * pow( sensorValue, -1.15);
    delay(1000);
}
```

① Servo Motor 제어핀 설정 및 Serial 연결 설정

② Servo Motor 각도 설정 및 Lidar 센서를 통한 거리 측정

[display11.java – 첫 번째 메뉴의 첫 번째 화면]

```java
package hufsice.hufsbugblock;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class display11 extends AppCompatActivity {

    home home = new home();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display11);

        Button b5 = (Button)findViewById(R.id.button5);
        b5.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display12.class);
                startActivity(intent);
            }
        });

    }
}
```

[display12.java – 첫 번째 메뉴의 두 번째 화면]

```java
package hufsice.hufsbugblock;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class display12 extends AppCompatActivity {

    home home = new home();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display12);

        Button b6 = (Button)findViewById(R.id.button6);
        b6.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display13.class);
                startActivity(intent);
            }
        });

    }
}
```

[display13.java – 첫 번째 메뉴의 세 번째 화면]

```java
package hufsice.hufsbugblock;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class display13 extends AppCompatActivity {

    home home = new home();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display13);

        Button b7 = (Button)findViewById(R.id.button7);
        b7.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display14.class);
                startActivity(intent);
            }
        });

    }
}

```

[display14.java – 첫 번째 메뉴의 네 번째 화면]

```java
package hufsice.hufsbugblock;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class display14 extends AppCompatActivity {

    home home = new home();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display14);

        Button b8 = (Button)findViewById(R.id.button8);
        b8.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display15.class);
                startActivity(intent);
            }
        });

    }
}
```

[display15.java – 첫 번째 메뉴의 다섯 번째 화면]

```cs
package hufsice.hufsbugblock;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class display15 extends AppCompatActivity {

    home home = new home();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display15);

        Button b9 = (Button)findViewById(R.id.button9);
        b9.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), home.class);
                startActivity(intent);
            }
        });

    }
}
```

[display21.java – 두 번째 메뉴의 첫 번째 화면]

```cs
package hufsice.hufsbugblock;
```

```
2
3  import android.content.Intent;
4  import android.support.v7.app.AppCompatActivity;
5  import android.os.Bundle;
6  import android.view.View;
7  import android.widget.Button;
8
9  public class display21 extends AppCompatActivity {
10
11     home home = new home();
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.display21);
17
18         Button b10 = (Button)findViewById(R.id.button10);
19         b10.setOnClickListener(new View.OnClickListener() {
20             @Override
21             public void onClick(View view) {
22                 Intent intent = new Intent(getApplicationContext(), display211.class);
23                 startActivity(intent);
24             }
25         });
26
27         Button b11 = (Button)findViewById(R.id.button11);
28         b11.setOnClickListener(new View.OnClickListener() {
29             @Override
30             public void onClick(View view) {
31                 Intent intent = new Intent(getApplicationContext(), display221.class);
32                 startActivity(intent);
33             }
34         });
35
36     }
37 }
38
```

[display31.java – 세 번째 메뉴의 첫 번째 화면, ConnectThread class를 포함한다]

```
1  package hufsice.hufsbugblock;
```

```
2
3  import android.app.Activity;
4  import android.content.Context;
5  import android.os.*;
6  import android.support.v7.app.AppCompatActivity;
7  import android.view.*;
8  import android.view.View.OnClickListener;
9  import android.widget.*;
10 import java.io.BufferedReader;
11 import java.io.InputStreamReader;
12 import java.net.InetSocketAddress;
13 import java.net.Socket;
14 import android.widget.Toast;
15
16 public class display31 extends AppCompatActivity implements OnClickListener {
17
18     EditText input;
19     @Override
20     protected void onCreate(Bundle savedInstanceState) {
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.display31);
23         input = (EditText)findViewById(R.id.serverText);
24         Button button = (Button)findViewById(R.id.socketButton);
25         button.setOnClickListener(this);
26     }
27
28     @Override
29     public void onClick(View v) {
30         if (v.getId() == R.id.socketButton){
31             ConnectThread th = new ConnectThread(this,"192.168.0.4");
32             th.start();
33         }
34     }
35 }
36
37 class ConnectThread extends Thread{
38
39     private String hostname;
40     private Context context;
41
42     public ConnectThread(Context context, String addr){
43         this.context = context;
44         hostname = addr;
45     }
```

```
46
47      public void run(){
48          try{
49              int port = 50006;
50              Socket socket = new Socket();
51              socket.setSoTimeout(10000);
52              socket.connect(new InetSocketAddress(hostname, port), 10000);
53              for (int c=0; c<100; c++) {
54                  BufferedReader in =
55                   new BufferedReader(new InputStreamReader(socket.getInputStream()));
56                  final String data = in.readLine();
57                  ((Activity)context).runOnUiThread(new Runnable() {
58
59                      @Override
60                      public void run() {
61                          Toast.makeText(context, data +" 앞에 장애물이 감지되었습니다." ,
62                          Toast.LENGTH_SHORT).show();
63                      }
64                  });
65              }
66              socket.close();
67          }catch(Exception e){
68              e.printStackTrace();
69          }
70      }
71  }
```

[display211.java – 두 번째 메뉴의 첫 번째 메뉴의 첫 번째 화면, WebClient class를 포함한다]

```
1  package hufsice.hufsbugblock;
2
```

```java
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.view.View;
import android.widget.EditText;
import android.view.View.OnClickListener;

public class display211 extends AppCompatActivity{

    WebView webview;

    @Override
    protected void onCreate(Bundle savedInstanceState){
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display211);
        webview = (WebView)findViewById(R.id.webView);
        webview.setWebViewClient(new WebClient()); // 응용프로그램에서 직접 url 처리
        WebSettings set = webview.getSettings();
        set.setJavaScriptEnabled(true);
        set.setBuiltInZoomControls(true);
        webview.loadUrl("http://www.naver.com");

        findViewById(R.id.button4).setOnClickListener(onclick);
    }

    OnClickListener onclick = new OnClickListener() {
        @Override
        public void onClick(View v) {
            System.out.println("클릭");
            String url= null;
            EditText editText = (EditText)findViewById(R.id.editText);
            url = editText.getText().toString();
            webview.loadUrl(url);
        }
    };

}


class WebClient extends WebViewClient {
    public boolean shouldOverrideUrlLoading(WebView view, String url) {
        view.loadUrl(url);
```

```
47          return true;
48      }
49  }
```

[display221.java – 두 번째 메뉴의 두 번째 메뉴의 첫 번째 화면]

```java
package hufsice.hufsbugblock;

import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.InputSource;

import java.net.URL;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;

public class display221 extends AppCompatActivity {
    TextView textview;
    Document doc = null;
    LinearLayout layout = null;
    XML weatherXml = new XML();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display221);

        weatherXml.execute("http://www.kma.go.kr/wid/queryDFSRSS.jsp?zone=1159068000");
        textview = (TextView) findViewById(R.id.WeatherInfo);

        Button b12 = (Button)findViewById(R.id.button12);
        b12.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), home.class);
                startActivity(intent);
            }
        });
```

```
45      }
46
47
48
49      private class XML extends AsyncTask<String, Void, Document> {
50
51          @Override
52          protected Document doInBackground(String... urls) {
53              URL url;
54              try {
55                  url = new URL(urls[0]);
56                  DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
57                  DocumentBuilder db = dbf.newDocumentBuilder();
58                  doc = db.parse(new InputSource(url.openStream()));
59                  doc.getDocumentElement().normalize();
60              } catch (Exception e) {
61                  Toast.makeText(getBaseContext(),
62  "Parsing Error", Toast.LENGTH_SHORT).show();
63              }
64              return doc;
65          }
66
67          @Override
68          protected void onPostExecute(Document doc) {
69              String str = "";
70              NodeList nodeList = doc.getElementsByTagName("data");
71
72              str += "\n\n  #  날씨 정보 \n\n";
73              Node node = nodeList.item(0);
74              Element element = (Element) node;
75
76              NodeList timeInfo = element.getElementsByTagName("hour");
77              str += "  측정 기준 시간: "+  timeInfo.item(0).getChildNodes().item(0)
78  .getNodeValue() +"시\n";
79
80              NodeList temperatures = element.getElementsByTagName("temp");
81              str += "  온도 = "+  temperatures.item(0).getChildNodes().item(0)
82
83  .getNodeValue() +" °C\n";
84
85              NodeList weatherCondition = element.getElementsByTagName("wfKor");
86              str += "  날씨 = "+  weatherCondition.item(0).getChildNodes().item(0)
87  .getNodeValue() +"\n";
88
```

```
            NodeList humidity = element.getElementsByTagName("reh");
            str += "  습도 = "+  humidity.item(0).getChildNodes().item(0)
.getNodeValue() +"%\n";

            NodeList windDirection = element.getElementsByTagName("wdKor");
            str += "  풍향 = "+  windDirection.item(0).getChildNodes().item(0)
.getNodeValue() +"쪽\n";

            NodeList windSpeed = element.getElementsByTagName("wd");
            str += "  풍속 = "+  windSpeed.item(0).getChildNodes().item(0)
.getNodeValue() +"m/s";

            double humidityValue = Double.parseDouble(windSpeed.item(0)
.getChildNodes().item(0).getNodeValue());
            if(humidityValue>=5 && humidityValue<=6) {
                str += " (드론 비행 주의)";
            }else if(humidityValue>=7 && humidityValue<=8) {
                str += " (드론 비행 위험)";
            }else if(humidityValue>=9) {
                str += " (드론 비행 금지)";
            }else {
                str += "";
            }

            double temperatureValue = Double.parseDouble(temperatures.item(0)
.getChildNodes().item(0).getNodeValue());
            String weatherInformation = weatherCondition.item(0).getChildNodes()
.item(0).getNodeValue();

            if(temperatureValue>=29 || temperatureValue<=15 ||
!weatherInformation.equals("맑음 ") || !weatherInformation.equals("구름 조금")) {
                str += "\n\n\n    ※주의사항※";
            }




            if(temperatureValue>=29 || temperatureValue<=-15) {
                str += "\n\n   29 ˚C 이상 혹은 -
15 ˚C 이하에서의 드론 비행은 드론의 파손을 야기할 수 있습니다.";
            }

```

```
            if(!weatherInformation.equals("맑음 ") ||
!weatherInformation.equals("구름 조금")) {
                str += "\n\n   현재 기상 상황이 좋지 않습니다.";
            }

            if(temperatureValue>=29 || temperatureValue<=15 ||
!weatherInformation.equals("맑음 ") || !weatherInformation.equals("구름 조금")) {
                str += "\n\n   비행 드론을 삼가 주시기 바랍니다.";
            }

            textview.setText(str);
            super.onPostExecute(doc);
        }

        final protected double temperatureValues(Document doc) {

            NodeList nodeList = doc.getElementsByTagName("data");
            Node node = nodeList.item(0);
            Element element = (Element) node;

            NodeList temperatures = element.getElementsByTagName("temp");
            String str =  temperatures.item(0).getChildNodes().item(0).getNodeValue();

            double tempValue = Double.parseDouble(str);
            return tempValue;
        }
    }
}
```

[home.java – 어플을 실행하자마자 나오는 메인 화면]

```java
package hufsice.hufsbugblock;

import android.content.Intent;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;


public class home extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.home);

        Button b1 = (Button)findViewById(R.id.button);
        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display11.class);
                startActivity(intent);
            }
        });

        Button b2 = (Button)findViewById(R.id.button2);
        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display21.class);
                startActivity(intent);
            }
        });

        Button b3 = (Button)findViewById(R.id.button3);
        b3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), display31.class);
                startActivity(intent);
```

```
            }
        });
    }

    int information = 0;

    public void toast(){

        Handler handler = new Handler();

        if(information == 1) {
            Toast toastView = Toast.makeText(this, "위험합니다!", Toast.LENGTH_SHORT);
            toastView.show();
            information = 0;
        }
        handler.postDelayed(new Runnable() {
            public void run() {
            }
        }, 5000);
    }
}
```

[AndroidManifest.xml]

```
<?xml version="1.0" encoding="UTF-8"?>

-<manifest package="hufsice.hufsbugblock" xmlns:android="http://schemas.android.com/
apk/res/android">
<!-- To auto-complete the email text field in the login form with the user's emails -->

<uses-permission android:name="android.permission.GET_ACCOUNTS"/>
<uses-permission android:name="android.permission.READ_PROFILE"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>

-<application android:theme="@style/AppTheme" android:supportsRtl="true" android:roundIcon=
"@mipmap/ic_launcher_round" android:label="@string/app_name" android:icon=
"@drawable/dronelogo" android:allowBackup="true">

-<activity android:name=".home">

-<intent-filter>

<action android:name="android.intent.action.MAIN"/>

<category android:name="android.intent.category.LAUNCHER"/>

</intent-filter>

</activity>
<activity android:name=".display11"/>
<activity android:name=".display12"/>
<activity android:name=".display13"/>
<activity android:name=".display14"/>
<activity android:name=".display15"/>
```

```
<activity android:name=".display21"/>
<activity android:name=".display221"/>
<activity android:name=".display31"/>
<activity android:name=".display211"> </activity>

</application>
</manifest>
```

[colors.xml]

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#ea3f20</color>
    <color name="colorPrimaryDark">#f98777</color>
    <color name="colorAccent">#FF4081</color>
    <color name="colorMintBlue">#e3f2f4</color>
    <color name="colorSkyBlue">#B8F2EC</color>
    <color name="colorText">#000000</color>
    <color name="colorButton">#0BE04E</color>
</resources>
```

cs

[dimens.xml]

```
<resources>
    <!-- Default screen margins, per the Android Design guidelines. -->
    <dimen name="activity_horizontal_margin">16dp</dimen>
    <dimen name="activity_vertical_margin">16dp</dimen>
</resources>
```

cs

[display11.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android=
"http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display11">

    <TextView
        android:id="@+id/textView"
        android:textColor="@color/colorText"
        android:layout_width="280dp"
        android:layout_height="0dp"
        android:layout_marginBottom="13dp"
        android:layout_marginEnd="43dp"
        android:layout_marginLeft="43dp"
        android:layout_marginRight="43dp"
        android:layout_marginStart="43dp"
        android:layout_marginTop="24dp"
        android:textSize="14dp"
        android:text="@string/str5"
        app:layout_constraintBottom_toTopOf="@+id/button5"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/button5"
        android:textColor="@color/colorText"
        android:background="@drawable/buttondesign"
        android:layout_width="wrap_content"                                    cs
```

```
        android:layout_height="wrap_content"
        android:layout_marginBottom="28dp"
        android:layout_marginEnd="48dp"
        android:layout_marginRight="48dp"
        android:text="Next"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />
</android.support.constraint.ConstraintLayout>
```

[display12.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android=
"http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display11">

    <TextView
        android:id="@+id/textView"
        android:layout_width="280dp"
        android:layout_height="396dp"
        android:layout_marginEnd="42dp"
        android:layout_marginLeft="42dp"
        android:layout_marginRight="42dp"
        android:layout_marginStart="42dp"
        android:layout_marginTop="24dp"
        android:textSize="16dp"
        android:textColor="@color/colorText"
        android:text="@string/str6"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"                    cs
```

```
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/button6"
        android:background="@drawable/buttondesign"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="28dp"
        android:layout_marginEnd="48dp"
        android:layout_marginRight="48dp"
        android:text="Next"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />
</android.support.constraint.ConstraintLayout>
```

[display13.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display11">

    <TextView
        android:id="@+id/textView"
        android:textColor="@color/colorText"
        android:layout_width="280dp"
        android:layout_height="28dp"
        android:layout_marginEnd="42dp"
        android:layout_marginLeft="42dp"
        android:layout_marginRight="42dp"
        android:layout_marginStart="42dp"
        android:layout_marginTop="24dp"
        android:textSize="16dp"
        android:text="@string/str7"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
26
27      <Button
28          android:id="@+id/button7"
29          android:layout_width="wrap_content"
30          android:layout_height="wrap_content"
31          android:layout_marginBottom="28dp"
32          android:layout_marginEnd="48dp"
33          android:layout_marginRight="48dp"
34          android:background="@drawable/buttondesign"
35          android:text="Next"
36          app:layout_constraintBottom_toBottomOf="parent"
37          app:layout_constraintEnd_toEndOf="parent" />
38
39      <ImageView
40          android:id="@+id/imageView"
41          android:layout_width="280dp"
42          android:layout_height="321dp"
43          android:layout_marginEnd="42dp"
44          android:layout_marginLeft="42dp"
45          android:layout_marginRight="42dp"
46          android:layout_marginStart="42dp"
47          android:layout_marginTop="16dp"
48          app:layout_constraintEnd_toEndOf="parent"
49          app:layout_constraintHorizontal_bias="0.0"
50          app:layout_constraintStart_toStartOf="parent"
51          app:layout_constraintTop_toBottomOf="@+id/textView"
```

```
52              app:srcCompat="@drawable/prohibit" />
53 </android.support.constraint.ConstraintLayout>
```

[display14.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display11">

    <TextView
        android:id="@+id/textView"
        android:textColor="@color/colorText"
        android:layout_width="280dp"
        android:layout_height="100dp"
        android:layout_marginEnd="42dp"
        android:layout_marginLeft="42dp"
        android:layout_marginRight="42dp"
        android:layout_marginStart="42dp"
        android:layout_marginTop="24dp"
        android:textSize="16dp"
        android:text="@string/str8"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
26
27      <Button
28          android:id="@+id/button8"
29          android:background="@drawable/buttondesign"
30          android:layout_width="wrap_content"
31          android:layout_height="wrap_content"
32          android:layout_marginBottom="28dp"
33          android:layout_marginEnd="48dp"
34          android:layout_marginRight="48dp"
35          android:text="Next"
36          app:layout_constraintBottom_toBottomOf="parent"
37          app:layout_constraintEnd_toEndOf="parent" />
38
39      <ImageView
40          android:id="@+id/imageView"
41          android:layout_width="280dp"
42          android:layout_height="276dp"
43          android:layout_marginEnd="42dp"
44          android:layout_marginLeft="42dp"
45          android:layout_marginRight="42dp"
46          android:layout_marginStart="42dp"
47          android:layout_marginTop="12dp"
48          app:layout_constraintEnd_toEndOf="parent"
49          app:layout_constraintHorizontal_bias="0.0"
50          app:layout_constraintStart_toStartOf="parent"
51          app:layout_constraintTop_toBottomOf="@+id/textView"
```

```
52              app:srcCompat="@drawable/restrict" />
53 </android.support.constraint.ConstraintLayout>
```

[display15.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display11">

    <TextView
        android:id="@+id/textView"
        android:textColor="@color/colorText"
        android:layout_width="280dp"
        android:layout_height="26dp"
        android:layout_marginEnd="42dp"
        android:layout_marginLeft="42dp"
        android:layout_marginRight="42dp"
        android:layout_marginStart="42dp"
        android:layout_marginTop="24dp"
        android:textSize="16dp"
        android:text="@string/str9"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
26
27      <TextView
28          android:id="@+id/textView2"
29          android:textColor="@color/colorText"
30          android:layout_width="280dp"
31          android:layout_height="900dp"
32          android:layout_marginEnd="42dp"
33          android:layout_marginLeft="42dp"
34          android:layout_marginRight="42dp"
35          android:layout_marginStart="42dp"
36          android:layout_marginTop="16dp"
37          android:textSize="16dp"
38          android:text="@string/str10"
39          app:layout_constraintEnd_toEndOf="parent"
40          app:layout_constraintStart_toStartOf="parent"
41          app:layout_constraintTop_toBottomOf="@+id/imageView" />
42
43      <Button
44          android:id="@+id/button9"
45          android:layout_width="wrap_content"
46          android:layout_height="wrap_content"
47          android:layout_marginBottom="28dp"
48          android:layout_marginEnd="48dp"
49          android:layout_marginRight="48dp"
50          android:text="Home"
51          android:background="@drawable/buttondesign"
```

```
52              app:layout_constraintBottom_toBottomOf="parent"
53              app:layout_constraintEnd_toEndOf="parent" />
54
55      <ImageView
56              android:id="@+id/imageView"
57              android:layout_width="280dp"
58              android:layout_height="260dp"
59              android:layout_marginEnd="42dp"
60              android:layout_marginLeft="42dp"
61              android:layout_marginRight="42dp"
62              android:layout_marginStart="42dp"
63              android:layout_marginTop="12dp"
64              app:layout_constraintEnd_toEndOf="parent"
65              app:layout_constraintStart_toStartOf="parent"
66              app:layout_constraintTop_toBottomOf="@+id/textView"
67              app:srcCompat="@drawable/control" />
68
69  </android.support.constraint.ConstraintLayout>
```

[display21.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display21"
    tools:layout_editor_absoluteY="81dp">

    <Button
        android:id="@+id/button10"
        android:layout_width="380dp"
        android:layout_height="120dp"
        android:layout_marginEnd="55dp"
        android:layout_marginStart="55dp"
        android:background="@drawable/buttondesign"
        android:text="@string/str11"
        android:textColor="@color/colorText"
        android:textSize="25dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
```

```
            android:id="@+id/button11"
            android:layout_width="380dp"
            android:layout_height="120dp"

            android:layout_marginEnd="55dp"
            android:layout_marginStart="55dp"
            android:background="@drawable/buttondesign"
            android:text="@string/str12"
            android:textColor="@color/colorText"
            android:textSize="25dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/button10"
            app:layout_constraintVertical_bias="0.0" />

</android.support.constraint.ConstraintLayout>
```

[display31.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_height="match_parent"
    android:layout_width="match_parent"
    android:background="@color/colorMintBlue">

    <TextView
        android:id="@+id/textSign"
        android:textColor="@color/colorText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="143dp"
        android:textSize="16dp"
        android:text="아래의 버튼을 누르시면 소켓 통신할 수 있습니다."
        tools:context=".MainActivity" />

    <Button
        android:id="@+id/socketButton"
        android:background="@drawable/buttondesign"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
26            android:layout_alignParentTop="true"
27            android:layout_centerHorizontal="true"
28            android:layout_marginTop="205dp"
29            android:text="소켓 연결"
30            android:textSize="20sp"
31            tools:context=".MainActivity" />
32
33        <EditText
34            android:id="@+id/serverText"
35            android:layout_width="wrap_content"
36            android:layout_height="wrap_content"
37            android:layout_alignParentBottom="true"
38            android:layout_centerHorizontal="true"
39            android:layout_marginBottom="172dp"
40            android:text="192.168.0.4"
41            android:textSize="18dp"
42            tools:context=".MainActivity" />
43
44    </RelativeLayout>
```

[display211.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    android:orientation="vertical"
    tools:context="display211">

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="10dp"
        android:layout_marginLeft="10dp"
        android:layout_marginRight="10dp"
        android:layout_marginTop="10dp"
        android:text="http://" />

    <WebView
        android:id="@+id/webView"
        android:layout_width="match_parent"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="10dp"
        android:layout_marginLeft="10dp"
```

```
            android:layout_marginRight="10dp"
            android:layout_height="332dp" />


    <Button
            android:id="@+id/button4"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:layout_marginBottom="10dp"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:text= "@string/str4" />

</LinearLayout>
```

[display221.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".display221">

    <Button
        android:id="@+id/button12"
        android:background="@drawable/buttondesign"
        android:textSize="25dp"
        android:textColor="@color/colorText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="28dp"
        android:layout_marginEnd="48dp"
        android:layout_marginRight="48dp"
        android:text="Home"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />

    <TextView
```

```
            android:id="@+id/WeatherInfo"
            android:layout_width="301dp"
            android:layout_height="500dp"
            android:layout_marginEnd="10dp"
            android:textColor="@color/colorText"
            android:textSize="18dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.575"
            app:layout_constraintStart_toStartOf="parent"
            tools:ignore="MissingConstraints"
            tools:layout_editor_absoluteY="5dp" />
</android.support.constraint.ConstraintLayout>
```

[home.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorMintBlue"
    tools:context=".home">

    <Button
        android:background="@drawable/buttondesign"
        android:textSize="25dp"
        android:textColor="@color/colorText"
        android:id="@+id/button"
        android:layout_width="380dp"
        android:layout_height="120dp"
        android:layout_marginEnd="55dp"
        android:layout_marginLeft="55dp"
        android:layout_marginRight="55dp"
        android:layout_marginStart="55dp"
        android:layout_marginTop="2dp"
        android:text="@string/str1"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
26
27      <Button
28          android:id="@+id/button2"
29          android:layout_width="380dp"
30          android:layout_height="120dp"
31          android:layout_marginEnd="55dp"
32          android:layout_marginStart="55dp"
33          android:background="@drawable/buttondesign"
34          android:text="@string/str2"
35          android:textColor="@color/colorText"
36          android:textSize="25dp"
37          app:layout_constraintEnd_toEndOf="parent"
38          app:layout_constraintStart_toStartOf="parent"
39          app:layout_constraintTop_toBottomOf="@+id/button" />
40
41      <Button
42          android:id="@+id/button3"
43          android:layout_width="380dp"
44          android:layout_height="120dp"
45          android:layout_marginEnd="55dp"
46          android:layout_marginStart="55dp"
47          android:background="@drawable/buttondesign"
48          android:text="@string/str3"
49          android:textColor="@color/colorText"
50          android:textSize="25dp"
51          app:layout_constraintEnd_toEndOf="parent"
```

```
52          app:layout_constraintStart_toStartOf="parent"
53          app:layout_constraintTop_toBottomOf="@+id/button2" />
54
55  </android.support.constraint.ConstraintLayout>
```

[strings.xml]

```xml
<resources>
    <string name="app_name">Hufs Bug Block</string>
    <string name="str1">드론 운행시 주의사항</string>
    <string name="str2">실시간 스트리밍 및 날씨</string>
    <string name="str3">소켓통신 서버에 접속하기</string>
    <string name="str4">URL</string>
    <string name="str5">#드론 조종사 준수사항(항공법) ₩n₩n1. 비행 중에는 장치를 육안으로 항상 확인할 수 있어야 합니다. (안개, 황사 등 시야가 좋지 않은 경우) ₩n₩n2. 사람이 많이 모인 곳 상공에서는 비행을 금지합니다. (경기장, 각종 페스티벌
     등 인파가 많이 모인 곳) ₩n₩n3. 사고나 분실에 대비해 장치는 소유자 이름과 연락처를 기재하도록 합니다. ₩n₩n4. 일몰 후부터 일몰 전까지, 야간 비행은 불법입니다. (야간 : 일몰 후부터 일출 전까지) ₩n₩n5. 비행 중 낙하물을 투하하지 않습니다
    . (인명 또는 재산에 피해가 우려되는 투하물) ₩n₩n6. 음주 상태에서는 드론 조종을 금지합니다. ₩n₩n7. 비행 전 연료량, 배터리 잔량, 통신 상태 등에 대하여 점검이 필요합니다. ₩n₩n[준수사항 미준수시 200만원 이하의 과태료가 부과될 수 있습니다
    .]</string>
    <string name="str6"># 드론 비행 공역 ₩n₩n비행 전 허가가 필요합니다. ₩n₩n1. 비행장 주변 관제권(반경 9.3km, 지방항공청 또는 국방부에서 허가 받은 경우 비행가능) ₩n₩n2. 비행금지구역(서울 강북지역, 휴전선, 원전 주변)(지방항공청 또는 국방부
        에서 허가 받은 경우 비행가능) ₩n₩n3. 고도 150m 이상(수면, 지면 또는 구조물 최상단 기준, 지방항공청 또는 국방부에서 허가 받은 경우 비행가능)₩n₩n
        [비행승인 없이 비행시 200만원 이하 벌금형에 처해 질 수 있습니다.]</string>
```

```
21    <string name="str7"># 비행금지구역</string>
22    <string name="str8"># 비행제한구역(고도 150m 미만, 시계거리 내에서는 허가 없이 비행가
능, 단, 서울특별시 내의 비행제한구역은 무조건 비행 승인을 받아야 함.)</string>
23    <string name="str9"># 관제권(공항, 군사용 비행장)</string>
24    <string name="str10">#위 내용은 12kg 이하 취미용 드론에 해당, 12kg 이상은 추가적
25 인 법 적용₩n₩n[출처 : 국토교통부]</string>
26    <string name="str11">실시간 스트리밍</string>
27    <string name="str12">실시간 기상정보</string>
28    <string name="str13"></string>
29    <string name="title_activity_display221">Sign in</string>
30
31    <!-- Strings related to login -->
32    <string name="prompt_email">Email</string>
33    <string name="prompt_password">Password (optional)</string>
34    <string name="action_sign_in">Sign in or register</string>
35    <string name="action_sign_in_short">Sign in</string>
      <string name="error_invalid_email">This email address is invalid</string>
      <string name="error_invalid_password">This password is too short</string>
      <string name="error_incorrect_password">This password is incorrect</string>
      <string name="error_field_required">This field is required</string>
      <string name="permission_rationale">"Contacts permissions are needed for providing email
          completions."
```

```
    </string>

</resources>
```

[styles.xml]

```
<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>

</resources>
```

cs