|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | from flask import Flask, render\_template, Response  from camera import Camera  app = Flask(\_\_name\_\_)  @app.route('/')  def index():  return render\_template('index.html')  def gen(camera):  while True:  frame = camera.get\_frame()  yield (b'--frame\r\n'b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n')  @app.route('/video\_feed')  def video\_feed():  return Response(gen(Camera()), mimetype='multipart/x-mixedreplace; boundary=frame')  if \_\_name\_\_ == '\_\_main\_\_':  app.run(host='0.0.0.0', debug=True, threaded=True)  camera.start\_preview(fullscreen=False,window=(100,20,640,480)) | [cs](http://colorscripter.com/info#e) |

**[Streaming.py - 실시간 영상 스트리밍 코드]**

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)

**[capturePhoto.py - 파이카메라 사진 촬영 코드]**

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | from picamera import PiCamera  from time import sleep  camera = PiCamera()  camera.resolution = (960, 720)  camera.start\_preview(fullscreen=False, window=(450,100,640,480))  camera.rotation = 180  i = 0  sleep(3)  while True :  sleep(2)  camera.capture('/home/pi/Desktop/image%s.jpg' % i)  i = i + 1  camera.stop\_preview() | [cs](http://colorscripter.com/info#e) |

1. 라즈베리파이에 연결한 파이카메라 사용을 위해 PiCamera 클래스 불러오기 및

시간 동기화를 위한 time 클래스 불러옴

1. 사진 크기 설정(640 x 480 해상도 , 60, 90fps)
2. 사진을 찍어서 저장
3. 사진 촬영 정지

**[isUpperThan55per.py - 인식 값에 따른 물 펌프 제어 코드]**

|  |  |  |
| --- | --- | --- |
| 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 | 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() | [cs](http://colorscripter.com/info#e) |

1. 라즈베리파이와 아두이노 간의 시리얼 통신을 위한 설정

(유선 연결인 Serial 통신이 가장 속도가 빨라 사용하였음)

1. 실시간 사람 인식률을 저장한 텍스트 파일을 계속해서 읽어 사람 인식률 확인

더 이상 읽을 값이 없으면 반복을 끝낸다.

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

**[saveImage.py - 사진 전송 및 인식률 수신]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34 | 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() | [cs](http://colorscripter.com/info#e) |
|  |  |  |

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.

1. 사진 전송 후 서버 쪽에서 실시간으로 detect object processing을 해주기 때문에 인식률

값을 다시 받아서 텍스트 파일에 저장한다.

**[YOLO.py - YOLO 코드]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182  183  184  185  186  187  188  189  190  191  192  193  194  195  196  197  198  199  200  201  202  203  204  205  206  207  208  209  210  211  212 | 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)    def classify(net, meta, im):      out = predict\_image(net, im)      res = []      for i in range(meta.classes):          res.append((meta.names[i], out[i]))      res = sorted(res, key=lambda x: -x[1])      return res    def detect(net, meta, image, thresh=.5, hier\_thresh=.5, nms=.45):      im = load\_image(image, 0, 0)      num = c\_int(0)      pnum = pointer(num)      predict\_image(net, im)      dets = get\_network\_boxes(net, im.w, im.h, thresh, hier\_thresh, None, 0, pnum)      num = pnum[0]      if (nms): do\_nms\_obj(dets, num, meta.classes, nms);        res = []      for j in range(num):          for i in range(meta.classes):              if dets[j].prob[i] > 0:                  b = dets[j].bbox                  res.append((meta.names[i], dets[j].prob[i], (b.x, b.y, b.w, b.h)))      res = sorted(res, key=lambda x: -x[1])      free\_image(im)      free\_detections(dets, num)      return res    def isNumber(s):      try:          int(s)          return True      except ValueError:          return False    if \_\_name\_\_ == "\_\_main\_\_":      while True:          size = ""          data = ""          size = client\_socket.recv(6)            if(isNumber(size) == False):              continue            if(len(size) != 6):              continue            size = int(size)            while(len(data) != size):              data += client\_socket.recv(100000)            print("Receive Image : " + str(size)+" bytes")          f = open("yolo" + str(i) + ".jpg", "wb")          f.write(base64.b64decode(data))          net = load\_net("cfg/yolov3-tiny.cfg", "yolov3-tiny.weights", 0)          meta = load\_meta("cfg/coco.data")          r = detect(net, meta, "/home/hbb/darknet/yolo"+str(i)+".jpg")          h = len(r)          j = 0            if h == 0:              i += 1            client\_socket.send("0")          continue            else:              for j in range(0, h):                  if(r[j][0] == 'person'):                      r = r[j][1]            r = r\*100;          r = int(r)          print("Person Recognition Rate : "+str(r)+"%")          r = str(r)          client\_socket.send(r)          sleep(1)            f.close()            i += 1 | [cs](http://colorscripter.com/info#e) |

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.

1. 사진 크기를 먼저 받는다.
2. 사진을 받은 후 base64로 디코딩하여 text 형식에서 바이너리 형식으로 변경 후 YOLO API를

사용하여 사진을 분석하여 사람 인식을 하는 과정.

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

**[transmitArduino.c - 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.

**[waterpumpControl.c - 아두이노 물 펌프 제어 코드]**

|  |  |  |
| --- | --- | --- |
| 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 | 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);          }      }  } | [cs](http://colorscripter.com/info#e) |

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

**[servoLidar.c - LIDAR 센서 및 Servo Motor 제어]**

|  |  |  |
| --- | --- | --- |
| 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 | #include <Servo.h>    Servo myservo;  int m =20;    void setup()  {      myservo.attach(9);//서브모터 제어핀      Serial.begin(9600);  }    void loop()  {      myservo.write(m); // 각도 0도로 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);  } | [cs](http://colorscripter.com/info#e) |

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]**

|  |  |  |
| --- | --- | --- |
| 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 | 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;    ㅁ  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);              }          });        }  } | [cs](http://colorscripter.com/info#e) |

① AppCompatActivity는 안드로이드 하위버전을 지원하는 activity이다.

② onCreate 메소드는 activity가 처음으로 시작될 때 호출된다. 저장중인 Bundle 객체를 이용한다.

**[display12.java – 첫 번째 메뉴의 두 번째 화면]**

|  |  |  |
| --- | --- | --- |
| 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 | 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);              }          });        }  } | [cs](http://colorscripter.com/info#e) |

① Button에 객체를 생성하여 버튼의 정보를 저장한다. findViewById 메소드로 버튼을 가리킨다.

② setOnClickListener을 활용하여 이미 생성한 버튼의 객체를 다룬다.

**[display13.java – 첫 번째 메뉴의 세 번째 화면]**

|  |  |  |
| --- | --- | --- |
| 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 | 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);              }          });        }  } | [cs](http://colorscripter.com/info#e) |

**[display14.java – 첫 번째 메뉴의 네 번째 화면]**

|  |  |  |
| --- | --- | --- |
| 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 | 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);              }          });        }  } | [cs](http://colorscripter.com/info#e) |

**[display15.java – 첫 번째 메뉴의 다섯 번째 화면]**

|  |  |  |
| --- | --- | --- |
| 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 | 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);              }          });        }  } | [cs](http://colorscripter.com/info#e) |

**[display21.java – 두 번째 메뉴의 첫 번째 화면]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38 | 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 display21 extends AppCompatActivity {        home home = new home();        @Override      protected void onCreate(Bundle savedInstanceState) {          super.onCreate(savedInstanceState);          setContentView(R.layout.display21);            Button b10 = (Button)findViewById(R.id.button10);          b10.setOnClickListener(new View.OnClickListener() {              @Override              public void onClick(View view) {                  Intent intent = new Intent(getApplicationContext(), display211.class);                  startActivity(intent);              }          });            Button b11 = (Button)findViewById(R.id.button11);          b11.setOnClickListener(new View.OnClickListener() {              @Override              public void onClick(View view) {                  Intent intent = new Intent(getApplicationContext(), display221.class);                  startActivity(intent);              }          });        }  } | [cs](http://colorscripter.com/info#e) |

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

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71 | package hufsice.hufsbugblock;    import android.app.Activity;  import android.content.Context;  import android.os.\*;  import android.support.v7.app.AppCompatActivity;  import android.view.\*;  import android.view.View.OnClickListener;  import android.widget.\*;  import java.io.BufferedReader;  import java.io.InputStreamReader;  import java.net.InetSocketAddress;  import java.net.Socket;  import android.widget.Toast;    public class display31 extends AppCompatActivity implements OnClickListener {        EditText input;      @Override      protected void onCreate(Bundle savedInstanceState) {          super.onCreate(savedInstanceState);          setContentView(R.layout.display31);          input = (EditText)findViewById(R.id.serverText);          Button button = (Button)findViewById(R.id.socketButton);          button.setOnClickListener(this);      }        @Override      public void onClick(View v) {          if (v.getId() == R.id.socketButton){              ConnectThread th = new ConnectThread(this,"192.168.0.4");              th.start();          }      }  }    class ConnectThread extends Thread{        private String hostname;      private Context context;        public ConnectThread(Context context, String addr){          this.context = context;          hostname = addr;      }        public void run(){          try{              int port = 50006;              Socket socket = new Socket();              socket.setSoTimeout(10000);              socket.connect(new InetSocketAddress(hostname, port), 10000);              for (int c=0; c<100; c++) {                  BufferedReader in =  new BufferedReader(new InputStreamReader(socket.getInputStream()));                  final String data = in.readLine();                  ((Activity)context).runOnUiThread(new Runnable() {                        @Override                      public void run() {                          Toast.makeText(context, data +" 앞에 장애물이 감지되었습니다." ,  Toast.LENGTH\_SHORT).show();                      }                  });              }              socket.close();          }catch(Exception e){              e.printStackTrace();          }      }  } | [cs](http://colorscripter.com/info#e) |

① ip 주소인 192.168.0.4 네트워크 안에 있어야 작동가능.

② Socket 연결 Thread 시작.

③ run() 은 Thread 하위 클래스를 작동시키는 메소드이다.

④ BufferedReader은 문자열을 읽어들일 때 사용한다.

⑤ 드론 전방 4m 이내에 장애물이 감지될 시 디스플레이 하단에 경고 메시지를 출력한다.

⑥ Socket 연결을 닫는다.

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

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49 | package hufsice.hufsbugblock;    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);          return true;      }  } | [cs](http://colorscripter.com/info#e) |

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

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162 | 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);              }          });      }            private class XML extends AsyncTask<String, Void, Document> {            @Override          protected Document doInBackground(String... urls) {              URL url;              try {                  url = new URL(urls[0]);                  DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();                  DocumentBuilder db = dbf.newDocumentBuilder();                  doc = db.parse(new InputSource(url.openStream()));                  doc.getDocumentElement().normalize();              } catch (Exception e) {                  Toast.makeText(getBaseContext(),  "Parsing Error", Toast.LENGTH\_SHORT).show();              }              return doc;          }            @Override          protected void onPostExecute(Document doc) {              String str = "";              NodeList nodeList = doc.getElementsByTagName("data");                str += "\n\n  #  날씨 정보 \n\n";              Node node = nodeList.item(0);              Element element = (Element) node;                NodeList timeInfo = element.getElementsByTagName("hour");              str += "  측정 기준 시간: "+  timeInfo.item(0).getChildNodes().item(0)  .getNodeValue() +"시\n";                NodeList temperatures = element.getElementsByTagName("temp");              str += "  온도 = "+  temperatures.item(0).getChildNodes().item(0)  .getNodeValue() +"︒C\n";                NodeList weatherCondition = element.getElementsByTagName("wfKor");              str += "  날씨 = "+  weatherCondition.item(0).getChildNodes().item(0)  .getNodeValue() +"\n";                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;          }      }  } | [cs](http://colorscripter.com/info#e) |

① execute로 해당 url(기상청 RSS)에 접속하여 parsing한다. 해당 지역의 기상 정보를 가져올 수 있다.

② onPostExecute로 날씨 정보 및 기상 상황에 따른 드론 비행시 주의사항을 표시한다.

③ 기온에 따라 주의사항을 다르게 출력해야 하기 때문에 기온만 반환한다.

**[home.java – 어플을 실행하자마자 나오는 메인 화면]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63 | 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);      }  } | [cs](http://colorscripter.com/info#e) |

① information 변수는 라즈베리파이로부터 들어오는 신호 변수이다.

② toast를 출력하기 위해 handler 객체를 활용한다.

**[AndroidManifest.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41 | <?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> | [cs](http://colorscripter.com/info#e) |

① 어플리케이션에서 internet 사용을 허가해 주는 태그이다.

② 어플리케이션을 사용하는 동안에는 자동 잠금을 방지해주는 태그이다.

③ 어플리케이션에서 activity로 활용하기 위해서는 반드시 manifest 파일에 등록하는 절차가 필요하다.

**[colors.xml]**

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | <?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](http://colorscripter.com/info#e) |

① 변수에 color 번호를 입력하여 이름으로 하여금 색을 사용할 수 있도록 하는 xml파일이다. 기본적인 색도있고, 사용자가 임의로 설정도 가능하다.

**[dimens.xml]**

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5 | <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](http://colorscripter.com/info#e) |

**[display11.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42 | <?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"          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> | [cs](http://colorscripter.com/info#e) |

**[display12.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39 | <?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"          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> | [cs](http://colorscripter.com/info#e) |

**[display13.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53 | <?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" />  <Button  android:id="@+id/button7"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginBottom="28dp"  android:layout\_marginEnd="48dp"  android:layout\_marginRight="48dp"  android:background="@drawable/buttondesign"  android:text="Next"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent" />  <ImageView  android:id="@+id/imageView"  android:layout\_width="280dp"  android:layout\_height="321dp"  android:layout\_marginEnd="42dp"  android:layout\_marginLeft="42dp"  android:layout\_marginRight="42dp"  android:layout\_marginStart="42dp"  android:layout\_marginTop="16dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintHorizontal\_bias="0.0"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView"  app:srcCompat="@drawable/prohibit" />  </android.support.constraint.ConstraintLayout> | [cs](http://colorscripter.com/info#e) |

**[display14.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53 | <?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" />  <Button  android:id="@+id/button8"  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" />  <ImageView  android:id="@+id/imageView"  android:layout\_width="280dp"  android:layout\_height="276dp"  android:layout\_marginEnd="42dp"  android:layout\_marginLeft="42dp"  android:layout\_marginRight="42dp"  android:layout\_marginStart="42dp"  android:layout\_marginTop="12dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintHorizontal\_bias="0.0"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView"  app:srcCompat="@drawable/restrict" />  </android.support.constraint.ConstraintLayout> | [cs](http://colorscripter.com/info#e) |

**[display15.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69 | <?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" />  <TextView  android:id="@+id/textView2"  android:textColor="@color/colorText"  android:layout\_width="280dp"  android:layout\_height="900dp"  android:layout\_marginEnd="42dp"  android:layout\_marginLeft="42dp"  android:layout\_marginRight="42dp"  android:layout\_marginStart="42dp"  android:layout\_marginTop="16dp"  android:textSize="16dp"  android:text="@string/str10"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/imageView" />  <Button  android:id="@+id/button9"  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"  android:background="@drawable/buttondesign"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent" />  <ImageView  android:id="@+id/imageView"  android:layout\_width="280dp"  android:layout\_height="260dp"  android:layout\_marginEnd="42dp"  android:layout\_marginLeft="42dp"  android:layout\_marginRight="42dp"  android:layout\_marginStart="42dp"  android:layout\_marginTop="12dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView"  app:srcCompat="@drawable/control" />  </android.support.constraint.ConstraintLayout> | [cs](http://colorscripter.com/info#e) |

**[display21.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42 | <?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> | [cs](http://colorscripter.com/info#e) |

**[display31.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44 | <?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"  android:layout\_alignParentTop="true"  android:layout\_centerHorizontal="true"  android:layout\_marginTop="205dp"  android:text="소켓 연결"  android:textSize="20sp"  tools:context=".MainActivity" />  <EditText  android:id="@+id/serverText"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentBottom="true"  android:layout\_centerHorizontal="true"  android:layout\_marginBottom="172dp"  android:text="192.168.0.4"  android:textSize="18dp"  tools:context=".MainActivity" />  </RelativeLayout> | [cs](http://colorscripter.com/info#e) |

**[display211.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40 | <?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> | [cs](http://colorscripter.com/info#e) |

**[display221.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37 | <?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> | [cs](http://colorscripter.com/info#e) |

**[home.xml]**

|  |  |  |
| --- | --- | --- |
| 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  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55 | <?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" />        <Button          android:id="@+id/button2"          android:layout\_width="380dp"          android:layout\_height="120dp"          android:layout\_marginEnd="55dp"          android:layout\_marginStart="55dp"          android:background="@drawable/buttondesign"          android:text="@string/str2"          android:textColor="@color/colorText"          android:textSize="25dp"          app:layout\_constraintEnd\_toEndOf="parent"          app:layout\_constraintStart\_toStartOf="parent"          app:layout\_constraintTop\_toBottomOf="@+id/button" />        <Button          android:id="@+id/button3"          android:layout\_width="380dp"          android:layout\_height="120dp"          android:layout\_marginEnd="55dp"          android:layout\_marginStart="55dp"          android:background="@drawable/buttondesign"          android:text="@string/str3"          android:textColor="@color/colorText"          android:textSize="25dp"          app:layout\_constraintEnd\_toEndOf="parent"          app:layout\_constraintStart\_toStartOf="parent"          app:layout\_constraintTop\_toBottomOf="@+id/button2" />    </android.support.constraint.ConstraintLayout>  [*r*](http://colorscripter.com/info#e) | [cs](http://colorscripter.com/info#e) |

**[strings.xml]**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | 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  31  32  33  34  35 | <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>      <string name="str7"># 비행금지구역</string>      <string name="str8"># 비행제한구역(고도 150m 미만, 시계거리 내에서는 허가 없이 비행가능, 단, 서울특별시 내의 비행제한구역은 무조건 비행 승인을 받아야 함.)</string>      <string name="str9"># 관제권(공항, 군사용 비행장)</string>      <string name="str10">#위 내용은 12kg 이하 취미용 드론에 해당, 12kg 이상은 추가적인 법 적용\n\n[출처 : 국토교통부]</string>      <string name="str11">실시간 스트리밍</string>      <string name="str12">실시간 기상정보</string>      <string name="str13"></string>      <string name="title\_activity\_display221">Sign in</string>        <!-- Strings related to login -->      <string name="prompt\_email">Email</string>      <string name="prompt\_password">Password (optional)</string>      <string name="action\_sign\_in">Sign in or register</string>      <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> | [cs](http://colorscripter.com/info#e) | |  | |  |  |  | | --- | --- | --- | | 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  31  32  33  34  35 | <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>      <string name="str7"># 비행금지구역</string>      <string name="str8"># 비행제한구역(고도 150m 미만, 시계거리 내에서는 허가 없이 비행가능, 단, 서울특별시 내의 비행제한구역은 무조건 비행 승인을 받아야 함.)</string>      <string name="str9"># 관제권(공항, 군사용 비행장)</string>      <string name="str10">#위 내용은 12kg 이하 취미용 드론에 해당, 12kg 이상은 추가적인 법 적용\n\n[출처 : 국토교통부]</string>      <string name="str11">실시간 스트리밍</string>      <string name="str12">실시간 기상정보</string>      <string name="str13"></string>      <string name="title\_activity\_display221">Sign in</string>        <!-- Strings related to login -->      <string name="prompt\_email">Email</string>      <string name="prompt\_password">Password (optional)</string>      <string name="action\_sign\_in">Sign in or register</string>      <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>  [*Colored by Color Scripter*](http://colorscripter.com/info#e) | [cs](http://colorscripter.com/info#e) | |

① 안드로이드 스튜디오는 타 언어간 앱 개발 시에 문제가 될 수 있는 문자열을 다루는 것에 대해서 string xml 파일을 사용한다. 이 파일을 사용함으로서, 직접 string을 파일에 작성하지 않아도 되고, 그로 인해 호환성 문제도 해결되는 효과가 있기 때문이다.

**[styles.xml]**

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | <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](http://colorscripter.com/info#e) |