약용작물 병충해 모니터링 AI 서비스

- 정밀농업 기술을 적용한 약용작물 데이터 서비스 플랫폼 상용화 -

AI 서비스 작성	(연세대학교)	허 진 경
	(리퓨터)	
AI 서비스 버전	가이드라인 버전 V1.0 2021년 10월 11일	

로고	약용작물 병충해 모니터링 AI 서비스 - 정밀농업 기술을 적용한 약용작물 데이터 서비스 플랫폼 상용화 -	문서번호	
	약용작물 데이터셋	제정일자	2021.10.11

제·개정 이력	관리구분	

개정번호	개정일자	변경내용	작 성	검 토	승 인
1	2021.10.11	초안 작성	허진경	허진경	허진경
2	2021.12.08	시스템 아키텍처 추가, 개발환경 사양 수정, 소스코드 추가, 오류 수정	허진경	허진경	허진경

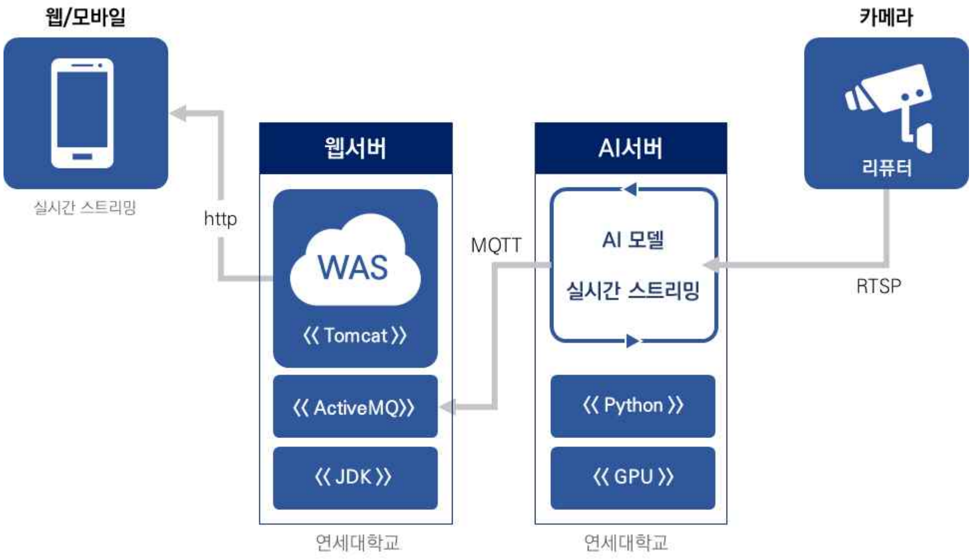
<제목 차례>

1. 스트리밍 서비스 개요

1.1. AI 실증서비스 개발 개요

카메라로부터 실시간으로 전송된 데이터를 AI 서비스를 통해 사람, 자동차 객체를 인식하고 대상 물체를 탐지한 후 화면에 그 결과를 실시간으로 송출한다.

1.2. 시스템 아키텍처



1.3. 개발환경 및 사양

서버 실행환경	
CPU	Intel(R) Core(TM) i9-10900X CPU @ 3.70GHz
Memory	128GB
GPU	GeForce RTX 3090 Ti * 2
Storage	SSD 1TB
OS	linux Ubuntu 18.04.5 LTS

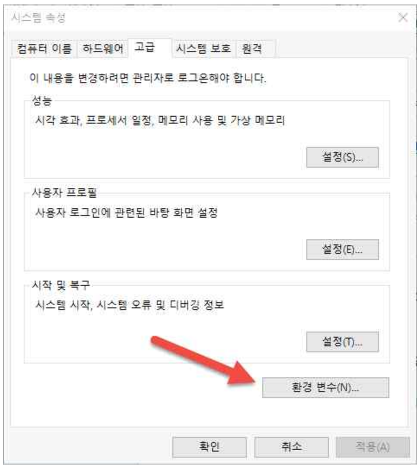
개발환경	
CPU	11th Gen Inter(R) Core(TM) i9-11900 2.50GHz
GPU	GeForce RTX 2080 SUPER
RAM	64GB
OS	Windows 10 Pro 20H2

언어 및 도구		
개발 언어	Python, JAVA, HTML5(HTML, Javascript, CSS)	
개발 도구	자바	Java SE 1.8 전자정부프레임워크(Eclipse) 3.9 SpringFramework
	파이썬	Anaconda 2021.05 & 2021.11 (Jupyter notebook 6.4.5) PyCharm community edition 2021.3
	WAS, Message Queue	Apache-tomcat-9 ActiveMQ 5.16.3
라이브러리	파이썬	OpenCV 4.5.4 YOLOv5 Tensorflow 2.7 paho-mqtt 1.6 torch 1.10.0, torchvision 0.11.1
	자바스크립트	JQuery 3.6.0 paho-mqtt javascript client 1.0.3
실행 파일명	PyCharm - ai_service.py Eclipse - streaming_service 프로젝트 폴더	

2. 윈도우 기반 개발환경 설정 및 테스트

2.1. JDK 설치 및 환경변수 설정

① JDK를 다운받아 설치한다.(java.sun.com -> JDK 1.8 다운로드 후 설치)
② 내 PC 우클릭 후 속성 클릭
③ 고급시스템 설정 클릭
④ 환경변수 클릭



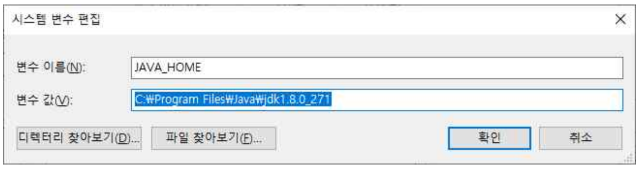
이 내용을 변경하려면 관리자로 로그온해야 합니다.

성능
시각 효과, 프로세서 일정, 메모리 사용 및 가상 메모리

사용자 프로필
사용자 로그인에 관련된 바탕 화면 설정

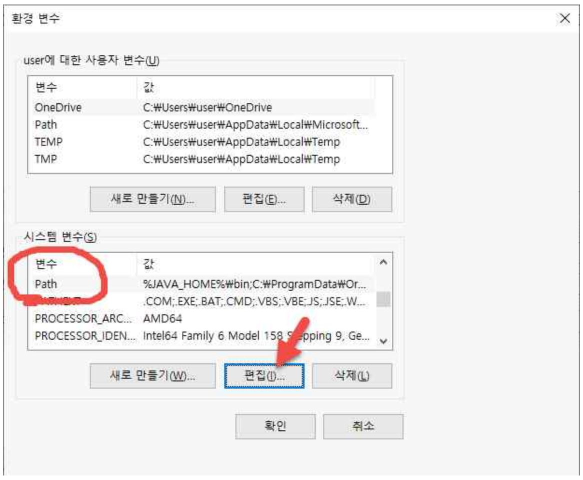
시작 및 복구
시스템 시작, 시스템 오류 및 디버깅 정보

⑤ 시스템변수에 새로 만들기 클릭 후 아래 사진과 같이 입력 후 확인 (윈도우 10 경우)



변수 이름(N): JAVA_HOME
변수 값(V): C:₩Program Files₩Java₩jdk1.8.0_271

⑥ Path 변수 찾아서 편집 클릭



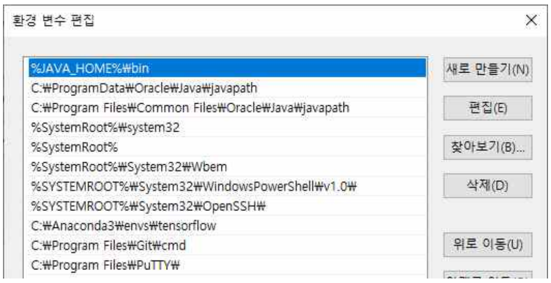
user에 대한 사용자 변수(U)

변수	값
OneDrive	C:₩Users₩user₩OneDrive
Path	C:₩Users₩user₩AppData₩Local₩Microsoft...
TEMP	C:₩Users₩user₩AppData₩Local₩Temp
TMP	C:₩Users₩user₩AppData₩Local₩Temp

시스템 변수(S)

변수	값
Path	%JAVA_HOME%₩bin;C:₩ProgramData₩Or...
[illegible]	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.W...
PROCESSOR_ARC...	AMD64
PROCESSOR_IDEN...	Intel64 Family 6 Model 158 Stepping 9, Ge...

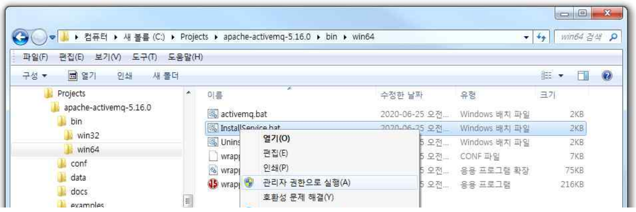
⑦ [새로 만들기] 클릭하여 %JAVA_HOME%₩bin 입력 후 맨위로 이동


환경 변수 편집

%JAVA_HOME%₩bin
C:₩ProgramData₩Oracle₩Java₩javapath
C:₩Program Files₩Common Files₩Oracle₩Java₩javapath
%SystemRoot%₩system32
%SystemRoot%
%SystemRoot%₩System32₩Wbem
%SYSTEMROOT%₩System32₩WindowsPowerShell₩v1.0₩
%SYSTEMROOT%₩System32₩OpenSSH₩
C:₩Anaconda3₩envs₩tensorflow
C:₩Program Files₩Git₩cmd
C:₩Program Files₩PuTTY₩

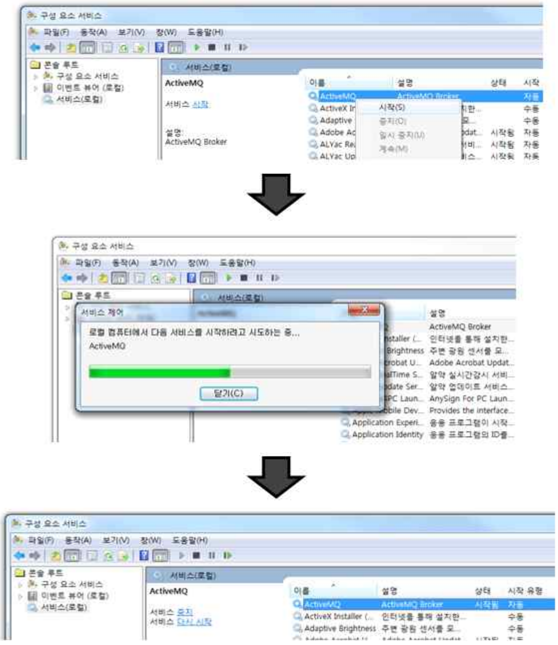
2.2. activemq 설치

① 다운로드 : http://activemq.apache.org/components/classic/download/
② Windows용 zip 파일 다운받은 후 압축 풀어 설치하기
③ *설치한 ActiveMQ폴더*₩bin₩win64₩InstallService.bat 관리자 권한으로 실행


컴퓨터 ▸ 새 볼륨 (C:) ▸ Projects ▸ apache-activemq-5.16.0 ▸ bin ▸ win64

이름	수정한 날짜	유형	크기
activemq.bat	2020-06-25 오전...	Windows 배치 파일	2KB
InstallService.bat	2020-06-25 오전...	Windows 배치 파일	2KB
Unins	5 오전...	Windows 배치 파일	2KB
wrap	5 오전...	CONF 파일	7KB
wrap	5 오전...	응용 프로그램 확장	75KB
wrap	5 오전...	응용 프로그램	216KB

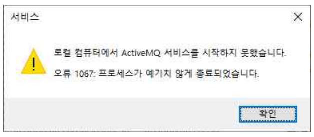
④ 윈도우 검색에 ‘서비스’ 검색 및 클릭 -> ActiveMQ 우클릭 및 시작


서비스 제어
로컬 컴퓨터에서 다음 서비스를 시작하려고 시도하는 중...
ActiveMQ

ActiveMQ
설명:
ActiveMQ Broker

이름	설명	상태	시작 유형
ActiveMQ	ActiveMQ Broker	시작됨	자동
ActiveX Installer (...	인터넷을 통해 설치한...		수동
Adaptive Brightness	주변 광원 센서를 모...		수동

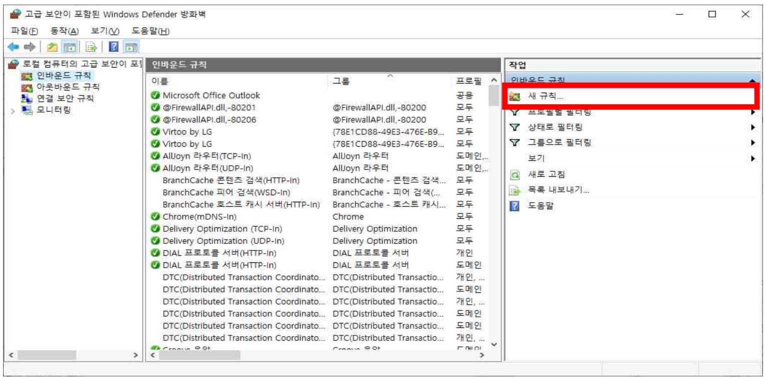
cf) 1067 오류가 발생했을 경우는 Java를 설치하고 JAVA_HOME 환경변수를 설정해야 함


서비스

로컬 컴퓨터에서 ActiveMQ 서비스를 시작하지 못했습니다.
오류 1067: 프로세스가 예기치 않게 종료되었습니다.

2.3. 방화벽 설정

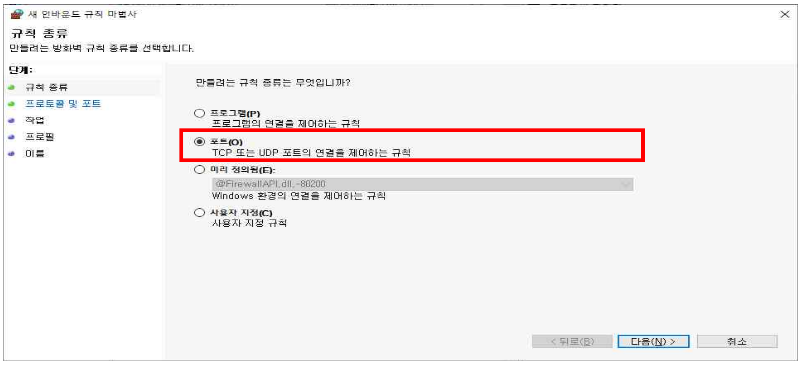
① Windows Defender 방화벽 → 고급설정 → 인바운드 규칙 → 새 규칙
7, 윈도우 10 순서로 나타냄



인바운드 규칙

이름	그룹	프로필
Microsoft Office Outlook		공용
@FirewallAPI.dll,-80201	@FirewallAPI.dll,-80200	모두
@FirewallAPI.dll,-80206	@FirewallAPI.dll,-80200	모두
Virtoo by LG	{78E1CD88-49E3-476E-B9...	모두
Virtoo by LG	{78E1CD88-49E3-476E-B9...	모두
AllJoyn 라우터(TCP-In)	AllJoyn 라우터	도메인,...
AllJoyn 라우터(UDP-In)	AllJoyn 라우터	도메인,...
BranchCache 콘텐츠 검색(HTTP-In)	BranchCache - 콘텐츠 검색...	모두
BranchCache 피어 검색(WSD-In)	BranchCache - 피어 검색(...	모두
BranchCache 호스트 캐시 서버(HTTP-In)	BranchCache - 호스트 캐시...	모두
Chrome(mDNS-In)	Chrome	모두
Delivery Optimization (TCP-In)	Delivery Optimization	모두
Delivery Optimization (UDP-In)	Delivery Optimization	모두
DIAL 프로토콜 서버(HTTP-In)	DIAL 프로토콜 서버	개인
DIAL 프로토콜 서버(HTTP-In)	DIAL 프로토콜 서버	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	개인, ...
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	개인, ...
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	개인, ...

② '규칙 종류 → 프로토콜 및 포트 → 작업 → 프로필 → 이름' 순서로 설정
- 규칙 종류 : 포트 클릭 → 다음



규칙 종류
만들려는 방화벽 규칙 종류를 선택합니다.

만들려는 규칙 종류는 무엇입니까?

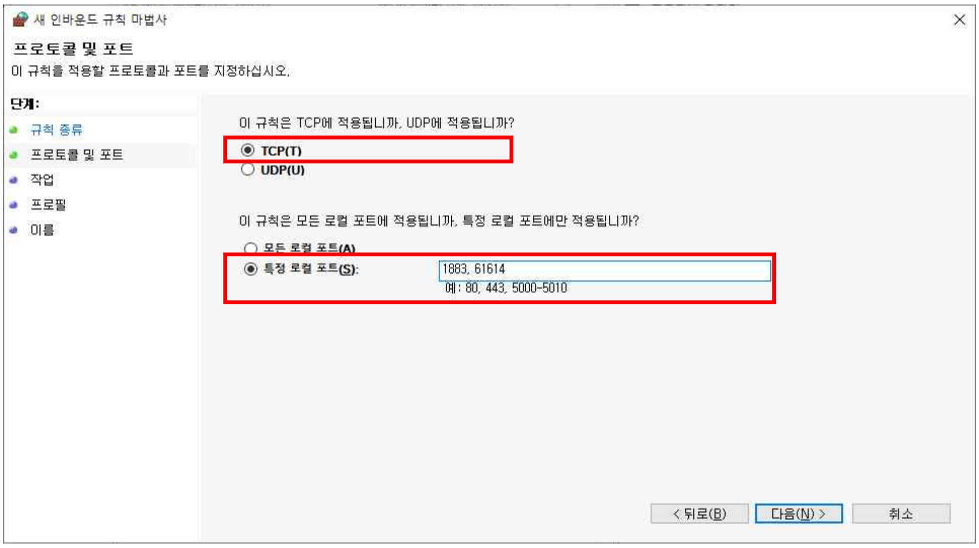
○ 프로그램(P)
프로그램의 연결을 제어하는 규칙

● 포트(O)
TCP 또는 UDP 포트의 연결을 제어하는 규칙

○ 미리 정의됨(E):
@FirewallAPI.dll,-80200
Windows 환경의 연결을 제어하는 규칙

○ 사용자 지정(C)
사용자 지정 규칙

- 프로토콜 및 포트 : TCP 클릭 → 특정 로컬포트 : 1883, 61614 → 다음



프로토콜 및 포트
이 규칙을 적용할 프로토콜과 포트를 지정하십시오.

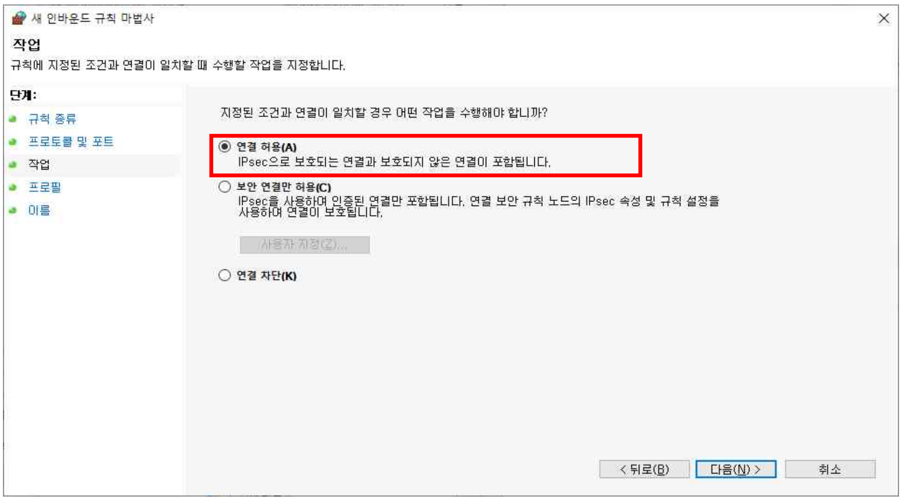
이 규칙은 TCP에 적용됩니까, UDP에 적용됩니까?

● TCP(T)
○ UDP(U)

이 규칙은 모든 로컬 포트에 적용됩니까, 특정 로컬 포트에만 적용됩니까?

○ 모든 로컬 포트(A)
● 특정 로컬 포트(S): 1883, 61614
예: 80, 443, 5000-5010

- 작업 : 연결 허용 → 다음



작업

규칙에 지정된 조건과 연결이 일치할 때 수행할 작업을 지정합니다.

단계:
- 규칙 종류
- 프로토콜 및 포트
- 작업
- 프로필
- 이름

지정된 조건과 연결이 일치할 경우 어떤 작업을 수행해야 합니까?

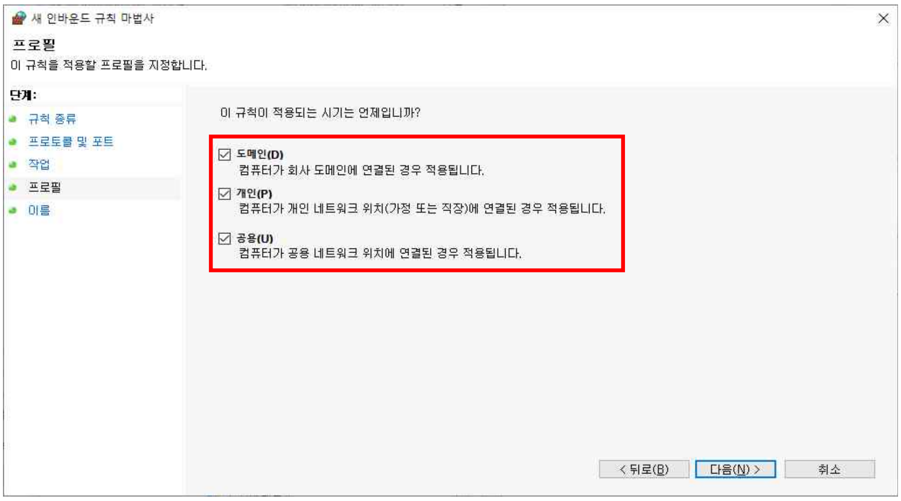
● 연결 허용(A)
IPsec으로 보호되는 연결과 보호되지 않은 연결이 포함됩니다.

○ 보안 연결만 허용(C)
IPsec를 사용하여 인증된 연결만 포함됩니다. 연결 보안 규칙 노드의 IPsec 속성 및 규칙 설정을 사용하여 연결이 보호됩니다.

사용자 지정(Z)...

○ 연결 차단(K)

- 프로필 : 모두 체크(도메인/개인/공용 체크) → 다음



프로필

이 규칙을 적용할 프로필을 지정합니다.

단계:
- 규칙 종류
- 프로토콜 및 포트
- 작업
- 프로필
- 이름

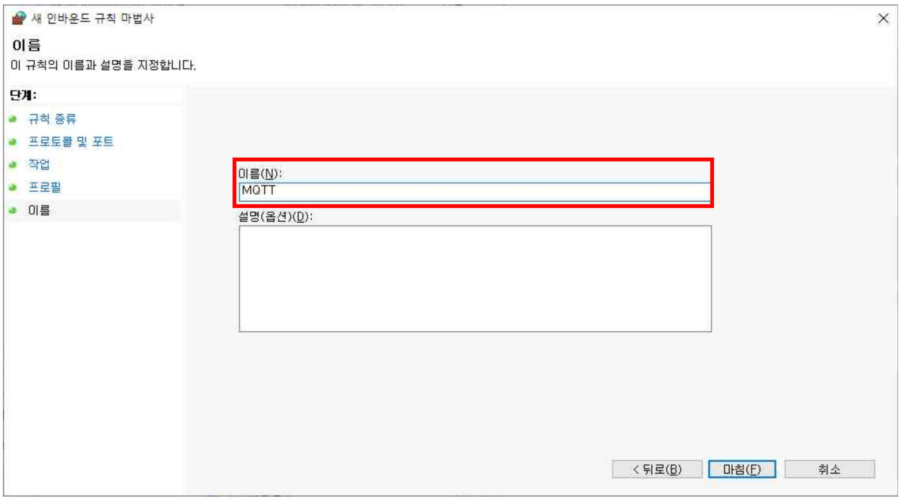
이 규칙이 적용되는 시기는 언제입니까?

☑ 도메인(D)
컴퓨터가 회사 도메인에 연결된 경우 적용됩니다.

☑ 개인(P)
컴퓨터가 개인 네트워크 위치(가정 또는 직장)에 연결된 경우 적용됩니다.

☑ 공용(U)
컴퓨터가 공용 네트워크 위치에 연결된 경우 적용됩니다.

- 이름 : ‘MQTT’ 작성 → 마침



이름

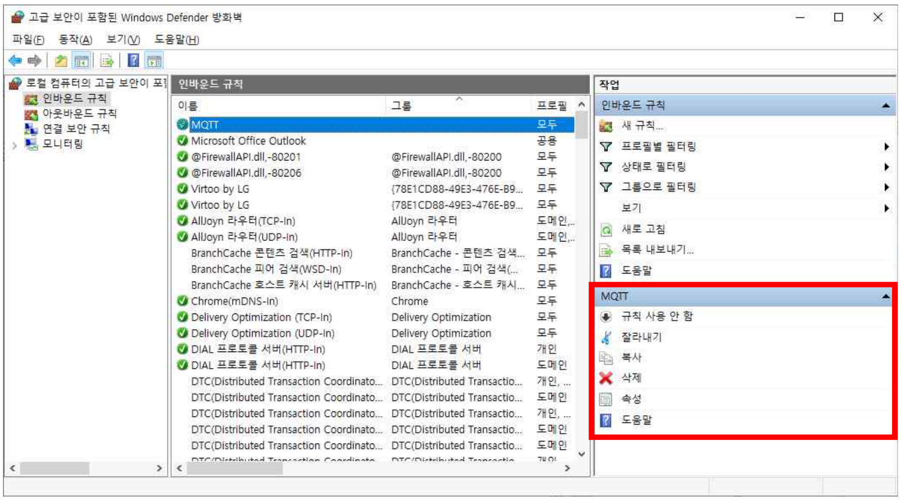
이 규칙의 이름과 설명을 지정합니다.

단계:
- 규칙 종류
- 프로토콜 및 포트
- 작업
- 프로필
- 이름

이름(N): MQTT

설명(옵션)(D):

③ 새 규칙이 추가된 결과



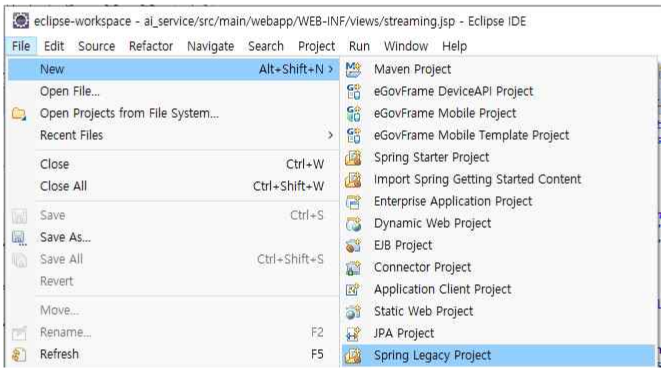
이름	그룹	프로필
MQTT		모두
Microsoft Office Outlook		공용
@FirewallAPI.dll,-80201	@FirewallAPI.dll,-80200	모두
@FirewallAPI.dll,-80206	@FirewallAPI.dll,-80200	모두
Virtoo by LG	{78E1CD88-49E3-476E-B9...	모두
Virtoo by LG	{78E1CD88-49E3-476E-B9...	모두
AllJoyn 라우터(TCP-In)	AllJoyn 라우터	도메인,...
AllJoyn 라우터(UDP-In)	AllJoyn 라우터	도메인,...
BranchCache 콘텐츠 검색(HTTP-In)	BranchCache - 콘텐츠 검색...	모두
BranchCache 피어 검색(WSD-In)	BranchCache - 피어 검색(...	모두
BranchCache 호스트 캐시 서버(HTTP-In)	BranchCache - 호스트 캐시...	모두
Chrome(mDNS-In)	Chrome	모두
Delivery Optimization (TCP-In)	Delivery Optimization	모두
Delivery Optimization (UDP-In)	Delivery Optimization	모두
DIAL 프로토콜 서버(HTTP-In)	DIAL 프로토콜 서버	개인
DIAL 프로토콜 서버(HTTP-In)	DIAL 프로토콜 서버	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	개인, ...
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	개인, ...
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	도메인
DTC(Distributed Transaction Coordinato...	DTC(Distributed Transactio...	도메인

2.4. 이클립스에서 프로젝트 작성 및 실행

① 이클립스 실행

② Spring Legacy Projecct로 새 프로젝트 생성

- 프로젝트 패키지명과 프로젝트 이름은 원하는 이름으로 입력


eclipse-workspace - ai_service/src/main/webapp/WEB-INF/views/streaming.jsp - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
New Alt+Shift+N >
Open File...
Open Projects from File System...
Recent Files >
Close Ctrl+W
Close All Ctrl+Shift+W
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename... F2
Refresh F5
Maven Project
eGovFrame DeviceAPI Project
eGovFrame Mobile Project
eGovFrame Mobile Template Project
Spring Starter Project
Import Spring Getting Started Content
Enterprise Application Project
Dynamic Web Project
EJB Project
Connector Project
Application Client Project
Static Web Project
JPA Project
Spring Legacy Project

③ 필요한 자바스크립트 라이브러리

- jquery(jquery.com)
- paho-mqtt(https://projects.eclipse.org/projects/iot.paho/downloads)
- 두 파일 모두 프로젝트의 resources 폴더 안에 있음

④ home.jsp 파일

```
<%@ page contentType="text/html; charset=utf-8" %>
<html>
<head>
    <title>Home</title>
</head>
<body>
```

```
<h1>AI 서비스</h1>
<a href="ai">약용작물 병충해 모니터링 AI 서비스</a>
</body>
</html>
```

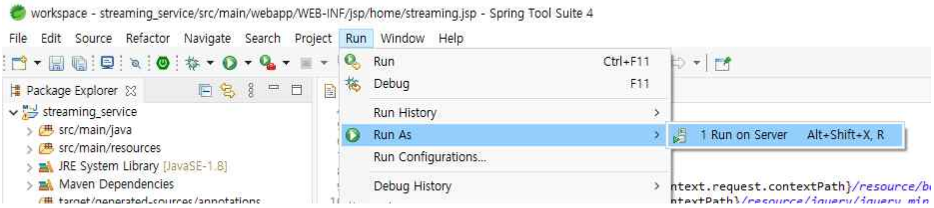
⑤ streaming.jsp 파일

```
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="height=device-height">
    <script type="text/javascript" src='<c:url value="/jquery/jquery-3.6.0.min.js"/> '></script>
    <script type="text/javascript" src='<c:url value="/paho-mqtt/paho-mqtt.js"/> '></script>
    <script type="text/javascript">
    $(function() {
        //클라이언트 인스턴스 생성(MQTT 서버의 주소를 입력해야 함)
        client = new Paho.MQTT.Client("localhost", 61614, new Date().getTime().toString());

        // 이벤트 핸들러 설정
        client.onConnectionLost = onConnectionLost;
        client.onMessageArrived = onMessageArrived;

        // 클라이언트 연결, 연결 성공하면 onConnect() 함수를 실행함
        client.connect({onSuccess:onConnect});
    })

    // 클라이언트가 연결되었을 때 호출되는 콜백함수
    function onConnect() {
        console.log("mqtt broker subscriber connected");
        client.subscribe("/#");
    }

    // 클라이언트가 커넥션을 읽을 때 호출되는 콜백함수
    function onConnectionLost(responseObject) {
      if (responseObject.errorCode !== 0) {
        console.log("onConnectionLost:"+responseObject.errorMessage);
      }
    }

    // 메시지가 도착했을 때 호출되는 콜백함수
    function onMessageArrived(message) {
        if(message.destinationName == "/camerapub") {
            $("#cameraView").attr("src", "data:image/jpg;base64, " + message.payloadString);
        }
    }
    </script>
    <title>약용작물 병충해 모니터링 AI 서비스</title>
    <style>
    div {
        width: 100%;
        height: 100%;
    }
    img#cameraView {
        max-width: 100%;
        max-height: 100%;
        bottom: 0;
        left: 0;
        margin: auto;
        overflow: auto;
        position: fixed;
        right: 0;
        top: 0;
        }
    </style>
</head>
<body>
    <!--h5 class="alert alert-info">AI 실증 서비스</h5-->
    <div align="center">
        <img id="cameraView" width=100% height=100%/>
    </div>
</body>
</html>
```

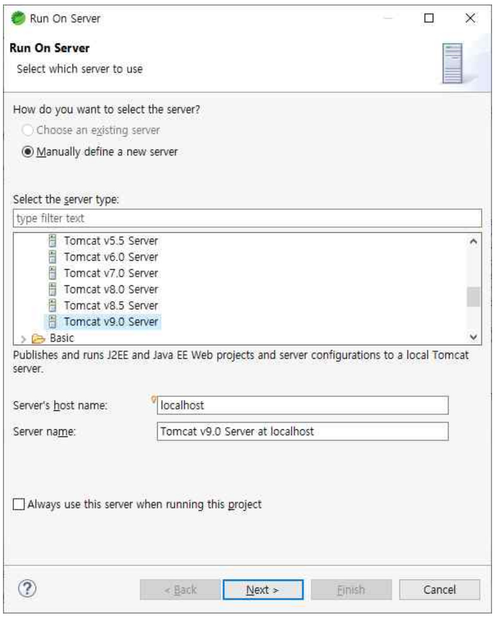
⑥ servlet-context.xml 파일

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans xmlns="http://www.springframework.org/schema/mvc"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:beans="http://www.springframework.org/schema/beans"
    xmlns:context="http://www.springframework.org/schema/context"
    xsi:schemaLocation="http://www.springframework.org/schema/mvc
https://www.springframework.org/schema/mvc/spring-mvc.xsd
        http://www.springframework.org/schema/beans
https://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/context
https://www.springframework.org/schema/context/spring-context.xsd">

    <annotation-driven />

    <resources mapping="/resources/**" location="/resources/" />

    <beans:bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <beans:property name="prefix" value="/WEB-INF/views/" />
        <beans:property name="suffix" value=".jsp" />
    </beans:bean>

    <context:component-scan base-package="com.coderby.myapp" />

    <view-controller view-name="streaming" path="/ai"/>

    <resources location="/WEB-INF/resources/" mapping="/**"/>
    <resources location="/WEB-INF/resources/images/" mapping="/images/**"/>
    <resources location="/WEB-INF/resources/jquey/" mapping="/jquery/**"/>
    <resources location="/WEB-INF/resources/paho-mqtt/" mapping="/paho-mqtt/**"/>
</beans:beans>
```

⑦ Run -> Run AS -> Run on Server 클릭


workspace - streaming_service/src/main/webapp/WEB-INF/jsp/home/streaming.jsp - Spring Tool Suite 4

⑧ Apache - Tomcat v9.0 Server 선택 후 next


Run On Server

Select which server to use

How do you want to select the server?

○ Choose an existing server

● Manually define a new server

Select the server type: Tomcat v9.0 Server

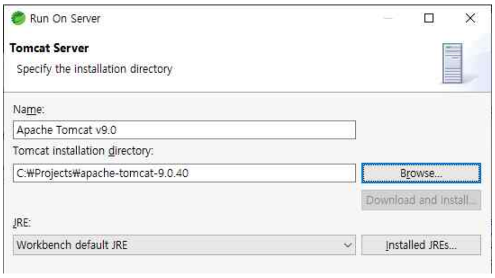
Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name: localhost

Server name: Tomcat v9.0 Server at localhost

☐ Always use this server when running this project

⑨ Browse 클릭 후 톰캣 설치된 폴더 선택 후 Finish 클릭


Run On Server

Tomcat Server

Specify the installation directory

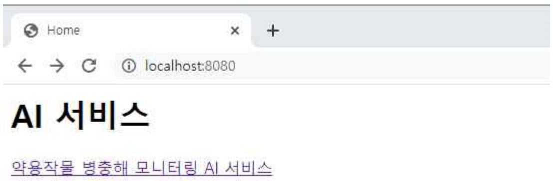
Name: Apache Tomcat v9.0

Tomcat installation directory: C:₩Projects₩apache-tomcat-9.0.40

JRE: Workbench default JRE

※ 크롬에서 보고 싶다면 메뉴목록에 Window -> Web Browser -> Chrome 선택

⑰ 크롬브라우저에서 http://localhost:8080/에 접속하면 링크 보임(컨텍스트 이름이 /여야 함)


localhost:8080

AI 서비스

약용작물 병충해 모니터링 AI 서비스

2.5. 스트리밍 서비스 테스트

1) 톰캣 서버 실행

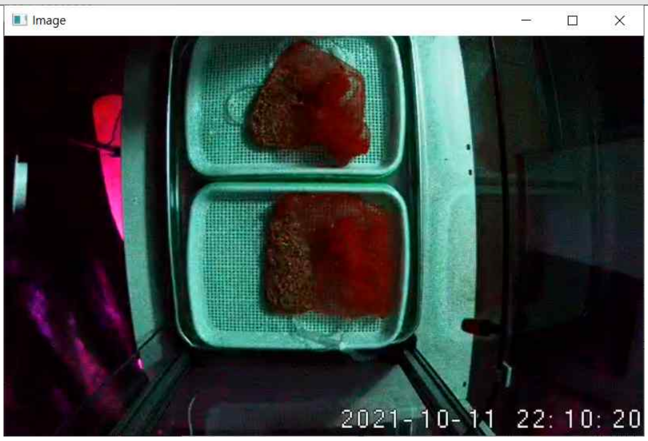
① $ ~/tomcat/webapps/ROOT 폴더 제외하고 모두 삭제
② ai_service.war 파일을 압축 풀어 ROOT 폴더에 복사
③ WEB-INF/jsp/streaming.jsp 파일 열어서 22라인 아이피 주소를 MQTT 서버 주소로 변경
④ 톰캣 서버 실행

```
$ ~/tomcat/bin/startup.sh
```

2) REST 서비스 연결 테스트

① 파이썬 소스코드(주피터 노트북에서 테스트)

```
import cv2
vid = cv2.VideoCapture("rtsp://reputer:reputer@letsfarm01.iptime.org:12110/stream1")
while True:
    ret, frame = vid.read()
    if not ret:
        print('Video has ended or failed, try a different video format!')
        break

    cv2.imshow("Image", frame)
    if cv2.waitKey(1)&0xFF == 27: # ESC
        break
cv2.waitKey()
cv2.destroyAllWindows()
```



- 카메라별 RTSP 주소
재배기 121번 정면 : rtsp://letsfarm01.iptime.org:12110/stream1
재배기 121번 사선 : rtsp://letsfarm01.iptime.org:12120/stream1
재배기 123번 정면 : rtsp://letsfarm01.iptime.org:12310/stream1
재배기 123번 사선 : rtsp://letsfarm01.iptime.org:12320/stream1
- RTSP 계정 정보
ID : reputer PW : reputer

3) 스트리밍 서비스 연결 테스트

1. mqtt broker 실행

2. tomcat 실행

3. 파이썬 MQTT 라이브러리 설치
   pip install paho-mqtt

4. mqtt/CameraPublisher.py 파일이 있어야 함.

```
import cv2
import paho.mqtt.client as mqtt
import threading
import base64
import time


class ImageMqttPublisher:
    def __init__(self, brokerIp=None, brokerPort=1883, pubTopic=None):
        self.brokerIp = brokerIp
        self.brokerPort = brokerPort
        self.pubTopic = pubTopic
        self.client = None

    def connect(self):
        thread = threading.Thread(target=self.__run, daemon=True)
        thread.start()

    def __run(self):
        self.client = mqtt.Client()
        self.client.on_connect = self.__on_connect
        self.client.on_disconnect = self.__on_disconnect
        self.client.connect(self.brokerIp, self.brokerPort)
        self.client.loop_forever()

    def __on_connect(self, client, userdata, flags, rc):
        print("ImageMqttClient mqtt broker connected")

    def __on_disconnect(self, client, userdata, rc):
        print("ImageMqttClient mqtt broker disconnected")

    def disconnect(self):
        self.client.disconnect()

    def sendBase64(self, frame):
        if self.client is None:
            return
        # MQTT Broker가 연결되어 있지 않을 경우
        if not self.client.is_connected():
            return
        # JPEG 포맷으로 인코딩
        retval, bytes = cv2.imencode(".jpg", frame)
        # 인코딩이 실패났을 경우
        if not retval:
            print("image encoding fail")
            return
        # Base64 문자열로 인코딩
        b64_bytes = base64.b64encode(bytes)
        # MQTT Broker에 보내기
        self.client.publish(self.pubTopic, b64_bytes, retain=True)


if __name__ == "__main__":
    videoCapture =
cv2.VideoCapture("rtsp://reputer:reputer@reputer01.iptimecam.com:20001/stream_ch00_1")
    imageMqttPusblisher = ImageMqttPublisher("localhost", 1883, "/camerapub")
    imageMqttPusblisher.connect()
    t = 0
    while True:
        if videoCapture.isOpened():
            retval, frame = videoCapture.read()
            frame = cv2.resize(frame, (640,480))
            if not retval:
                print("video capture fail")
                break
            imageMqttPusblisher.sendBase64(frame)
            print("\r", t, end="")
            t += 1
        else:
            break

    imageMqttPusblisher.disconnect()
    videoCapture.release()
```

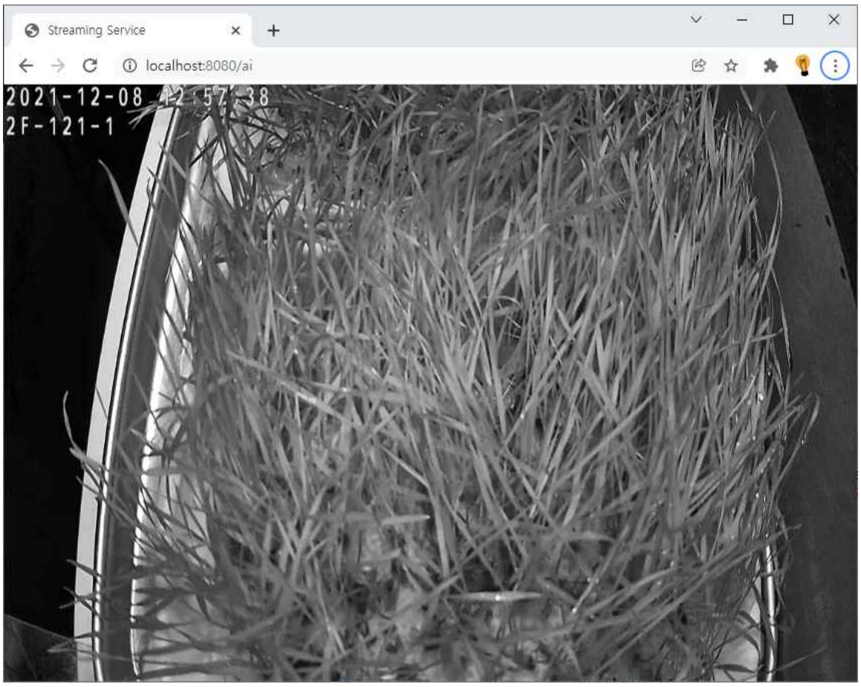
4. 스크립트 실행(ESC키 누르면 스트리밍 서비스 종료)

```
import cv2
vid = cv2.VideoCapture("rtsp://reputer:reputer@letsfarm01.iptime.org:12110/stream1")
from mqtt.Camerapublisher import ImageMqttPublisher

imageMqttPusblisher = ImageMqttPublisher("localhost", 1883, "/camerapub")
imageMqttPusblisher.connect()
while True:
    ret, frame = vid.read()
    if not ret:
        print('Video has ended or failed, try a different video format!')
        break

#     cv2.imshow("Image", frame)
    imageMqttPusblisher.sendBase64(frame)
    if cv2.waitKey(1)&0xFF == 27: # ESC
        break

imageMqttPusblisher.disconnect()
vid.release()
cv2.destroyAllWindows()
```

5. 브라우저 테스트

http://localhost:8080/ 접속 후 링크를 클릭하면 카메라 연결된 화면 보임

또는 http://localhost:8080/ai 접속



3. 리눅스기반 개발환경 설정 및 실행

3.1. 기본 설정

1) 방화벽 설정

http : 8080, 80,
ws : 61614, 61624
mqtt : 1883, 1893

2) JDK 설치

```
$ tar xf jdk1.8.0_271.tar.gz

$ vi ~/.bashrc
export JAVA_HOME=~/jdk1.8.0_271
export PATH=$PATH:JAVA_HOME/bin

$ source ~/.bashrc
```

3.2. MQTT Broker

1) MQTT broker 설치(리눅스 CentOS 7)

1) 다운로드 : http://activemq.apache.org/components/classic/download/
2) Linux tar.gz 파일 다운받은 후 압축 풀어 설치하기

```
$ wget http://dlcdn.apache.org//activemq/5.16.3/apache-activemq-5.16.3-bin.tar.gz
$ tar xf apache-activemq-5.16.3-bin.tar.gz
$ mv apache-activemq-5.16.3 activemq-5.16
```

2) Active MQ 콘솔 명령어로 테스트

```
$ ~/activemq-5.16/bin/linux-x86-64/activemq console
Running ActiveMQ Broker...
wrapper  | WARNING: Could not write lock file /var/lock/subsys/ActiveMQ: Permission denied
wrapper  | --> Wrapper Started as Console
wrapper  | Launching a JVM...
jvm 1    | Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
jvm 1    |   Copyright 1999-2006 Tanuki Software, Inc.  All Rights Reserved.
jvm 1    |
jvm 1    | Java Runtime: Oracle Corporation 1.8.0_271 /home/oem/jdk1.8.0_271/jre
jvm 1    |   Heap sizes: current=1005056k  free=947383k  max=1005056k
jvm 1    |     JVM args: -Dactivemq.home=../.. -Dactivemq.base=../..
-Djavax.net.ssl.keyStorePassword=password -Djavax.net.ssl.trustStorePassword=password
```

```
-Djavax.net.ssl.keyStore=../../conf/broker.ks -Djavax.net.ssl.trustStore=../../conf/broker.ts
-Dcom.sun.management.jmxremote -Dorg.apache.activemq.UseDedicatedTaskRunner=true
-Djava.util.logging.config.file=logging.properties -Dactivemq.conf=../../conf
-Dactivemq.data=../../data -Djava.security.auth.login.config=../../conf/login.config -Xmx1024m
-Djava.library.path=../../bin/linux-x86-64/ -Dwrapper.key=0AoE7HUjT1x8RuI_ -Dwrapper.port=32000
-Dwrapper.jvm.port.min=31000 -Dwrapper.jvm.port.max=31999 -Dwrapper.pid=17939
-Dwrapper.version=3.2.3 -Dwrapper.native_library=wrapper -Dwrapper.cpu.timeout=10 -Dwrapper.jvmid=1
jvm 1    ¦ Extensions classpath:
jvm 1    ¦   [../../lib,../../lib/camel,../../lib/optional,../../lib/web,../../lib/extra]
jvm 1    ¦ ACTIVEMQ_HOME: ../..
jvm 1    ¦ ACTIVEMQ_BASE: ../..
jvm 1    ¦ ACTIVEMQ_CONF: ../../conf
jvm 1    ¦ ACTIVEMQ_DATA: ../../data
jvm 1    ¦ Loading message broker from: xbean:activemq.xml
jvm 1    ¦  INFO ¦ Refreshing org.apache.activemq.xbean.XBeanBrokerFactory$1@4528880e: startup date
[Sun Jan 03 21:58:38 KST 2021]; root of context hierarchy
jvm 1    ¦  INFO ¦ Using Persistence Adapter:
KahaDBPersistenceAdapter[/home/oem/activemq-5.16/bin/linux-x86-64/../../data/kahadb]
jvm 1    ¦  INFO ¦ KahaDB is version 7
jvm 1    ¦  INFO ¦
PListStore:[/home/oem/activemq-5.16/bin/linux-x86-64/../../data/localhost/tmp_storage] started
jvm 1   ¦  INFO ¦ Apache ActiveMQ 5.16.0 (localhost, ID:reputer-34903-1609678719577-0:1) is starting
jvm 1    ¦  INFO ¦ Listening for connections at:
tcp://reputer:61616?maximumConnections=1000&wireFormat.maxFrameSize=104857600
jvm 1    ¦  INFO ¦ Connector openwire started
jvm 1    ¦  INFO ¦ Listening for connections at:
amqp://reputer:5672?maximumConnections=1000&wireFormat.maxFrameSize=104857600
jvm 1    ¦  INFO ¦ Connector amqp started
jvm 1    ¦  INFO ¦ Listening for connections at:
stomp://reputer:61613?maximumConnections=1000&wireFormat.maxFrameSize=104857600
jvm 1    ¦  INFO ¦ Connector stomp started
jvm 1    ¦  INFO ¦ Listening for connections at:
mqtt://reputer:1883?maximumConnections=1000&wireFormat.maxFrameSize=104857600
jvm 1    ¦  INFO ¦ Connector mqtt started
jvm 1    ¦  INFO ¦ Starting Jetty server
jvm 1    ¦  INFO ¦ Creating Jetty connector
jvm 1    ¦  WARN ¦ ServletContext@o.e.j.s.ServletContextHandler@193421b8{/,null,STARTING} has
uncovered http methods for path: /
jvm 1    ¦  INFO ¦ Listening for connections at
ws://reputer:61614?maximumConnections=1000&wireFormat.maxFrameSize=104857600
jvm 1    ¦  INFO ¦ Connector ws started
jvm 1    ¦  INFO ¦ Apache ActiveMQ 5.16.0 (localhost, ID:reputer-34903-1609678719577-0:1) started
jvm 1    ¦  INFO ¦ For help or more information please see: http://activemq.apache.org
jvm 1    ¦  INFO ¦ ActiveMQ WebConsole available at http://127.0.0.1:8161/
jvm 1    ¦  INFO ¦ ActiveMQ Jolokia REST API available at http://127.0.0.1:8161/api/jolokia/
```

Ctrl+C로 중단

3) Active MQ 서비스 시작 및 확인

```
$ ~/activemq-5.16/bin/linux-x86-64/activemq start
Starting ActiveMQ Broker...

$ jps
```

```
12464 WrapperSimpleApp
9415 Bootstrap
12619 Jps
```

3.3. 웹 애플리케이션

1) tomcat 설치

tomcat.apache.org에서 tomcat9 tar.gz 파일 다운받아 압축풀어 설치

```
$ tar -xf apache-tomcat-9.0.41.tar.gz
$ mv apache-tomcat-9.0.41 tomcat
```

2) 서버 실행

$ ~/tomcat/webapps/ROOT 폴더 제외하고 모두 삭제
streaming_service.war 파일을 압축 풀어 ROOT 폴더에 복사
WEB-INF/jsp/streaming.jsp 파일 열어서 22라인 아이피 주소를 222.107.208.146로 변경

```
$ ~/tomcat/bin/startup.sh
```

3) REST 서비스 테스트

- 카메라별 RTSP 주소

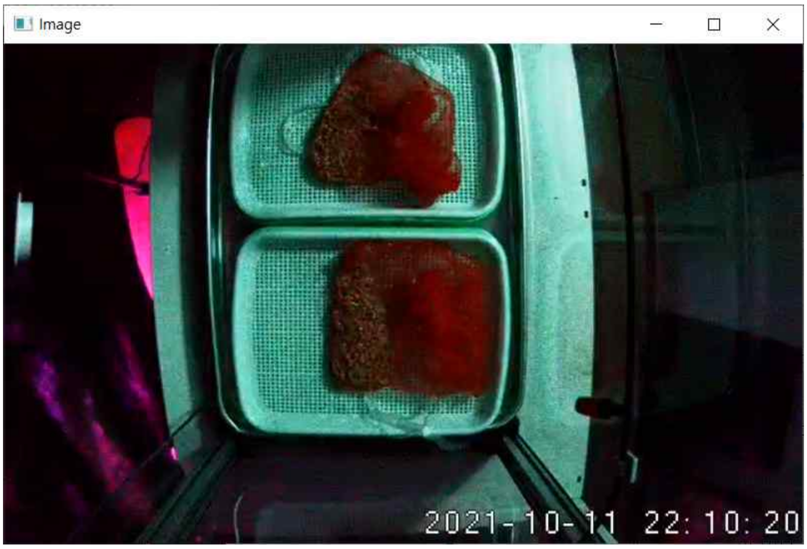
재배기 121번 정면 : rtsp://letsfarm01.iptime.org:12110/stream1
재배기 121번 사선 : rtsp://letsfarm01.iptime.org:12120/stream1
재배기 123번 정면 : rtsp://letsfarm01.iptime.org:12310/stream1
재배기 123번 사선 : rtsp://letsfarm01.iptime.org:12320/stream1

- RTSP 계정 정보

ID : reputer PW : reputer

- 파이썬 소스코드(주피터 노트북에서 테스트)

```
import cv2
vid = cv2.VideoCapture("rtsp://reputer:reputer@letsfarm01.iptime.org:12110/stream1")
while True:
    ret, frame = vid.read()
    if not ret:
        print('Video has ended or failed, try a different video format!')
        break


    cv2.imshow("Image", frame)
    if cv2.waitKey(1)&0xFF == 27: # ESC
        break
cv2.waitKey()
cv2.destroyAllWindows()
```



4) 스트리밍 서비스 테스트

1. mqtt broker 실행

2. tomcat 실행

3. 파이썬 MQTT 라이브러리 설치
   pip install paho-mqtt

4. mqtt/CameraPublisher.py

```
import cv2
import paho.mqtt.client as mqtt
import threading
import base64
import time


class ImageMqttPublisher:
    def __init__(self, brokerIp=None, brokerPort=1883, pubTopic=None):
        self.brokerIp = brokerIp
        self.brokerPort = brokerPort
        self.pubTopic = pubTopic
        self.client = None

    def connect(self):
        thread = threading.Thread(target=self.__run, daemon=True)
        thread.start()

    def __run(self):
        self.client = mqtt.Client()
        self.client.on_connect = self.__on_connect
        self.client.on_disconnect = self.__on_disconnect
        self.client.connect(self.brokerIp, self.brokerPort)
        self.client.loop_forever()

    def __on_connect(self, client, userdata, flags, rc):
        print("ImageMqttClient mqtt broker connected")

    def __on_disconnect(self, client, userdata, rc):
        print("ImageMqttClient mqtt broker disconnected")

    def disconnect(self):
        self.client.disconnect()

    def sendBase64(self, frame):
        if self.client is None:
            return
        # MQTT Broker가 연결되어 있지 않을 경우
        if not self.client.is_connected():
            return
        # JPEG 포맷으로 인코딩
        retval, bytes = cv2.imencode(".jpg", frame)
        # 인코딩이 실패났을 경우
        if not retval:
            print("image encoding fail")
            return
        # Base64 문자열로 인코딩
        b64_bytes = base64.b64encode(bytes)
        # MQTT Broker에 보내기
        self.client.publish(self.pubTopic, b64_bytes, retain=True)


if __name__ == "__main__":
    videoCapture =
cv2.VideoCapture("rtsp://reputer:reputer@reputer01.iptimecam.com:20001/stream_ch00_1")
    imageMqttPusblisher = ImageMqttPublisher("localhost", 1883, "/camerapub")
    imageMqttPusblisher.connect()
    t = 0
    while True:
        if videoCapture.isOpened():
            retval, frame = videoCapture.read()
            frame = cv2.resize(frame, (640,480))
            if not retval:
                print("video capture fail")
                break
            imageMqttPusblisher.sendBase64(frame)
            print("\r", t, end="")
            t += 1
        else:
            break

    imageMqttPusblisher.disconnect()
    videoCapture.release()
```

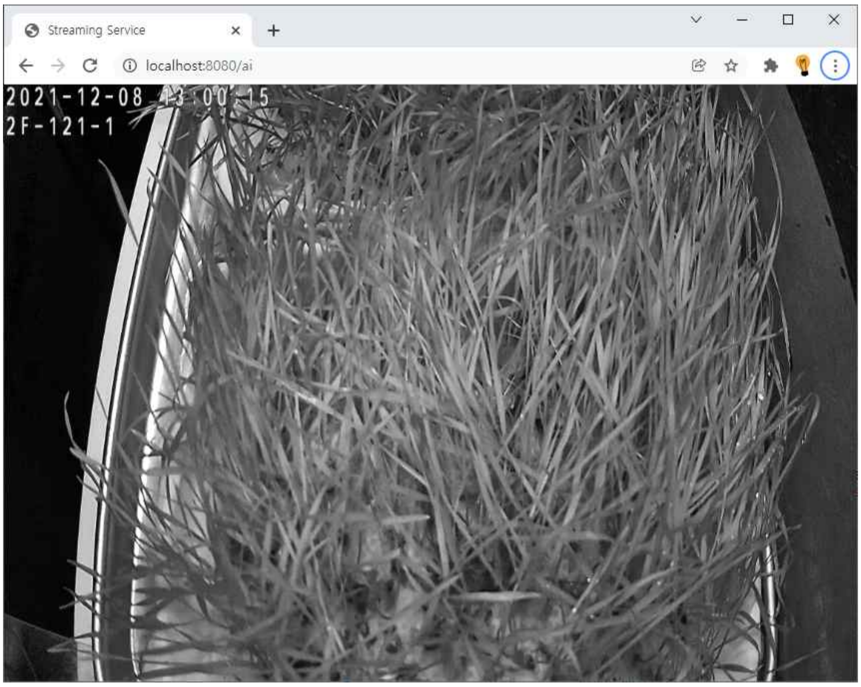
5. 스크립트 실행(ESC키 누르면 스트리밍 서비스 종료)

```
import cv2
vid = cv2.VideoCapture("rtsp://reputer:reputer@letsfarm01.iptime.org:12110/stream1")
from mqtt.Camerapublisher import ImageMqttPublisher

```

```
imageMqttPusblisher = ImageMqttPublisher("localhost", 1883, "/camerapub")
imageMqttPusblisher.connect()
while True:
    ret, frame = vid.read()
    if not ret:
        print('Video has ended or failed, try a different video format!')
        break

#     cv2.imshow("Image", frame)
    imageMqttPusblisher.sendBase64(frame)
#    if cv2.waitKey(1)&0xFF == 27: # ESC
#        break

imageMqttPusblisher.disconnect()
vid.release()
cv2.destroyAllWindows()
```

6. 브라우저 테스트

http://localhost:8080/ai



4. 소스코드

4.1. 자바

1) servlet-context.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans xmlns="http://www.springframework.org/schema/mvc"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:beans="http://www.springframework.org/schema/beans"
    xmlns:context="http://www.springframework.org/schema/context"
    xsi:schemaLocation="http://www.springframework.org/schema/mvc
https://www.springframework.org/schema/mvc/spring-mvc.xsd
        http://www.springframework.org/schema/beans
https://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/context
https://www.springframework.org/schema/context/spring-context.xsd">

    <!-- DispatcherServlet Context: defines this servlet's request-processing infrastructure -->

    <!-- Enables the Spring MVC @Controller programming model -->
    <annotation-driven />

    <!-- Handles HTTP GET requests for /resources/** by efficiently serving up static resources in
the ${webappRoot}/resources directory -->
    <resources mapping="/resources/**" location="/resources/" />

    <!-- Resolves views selected for rendering by @Controllers to .jsp resources in the
/WEB-INF/views directory -->
    <beans:bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <beans:property name="prefix" value="/WEB-INF/views/" />
        <beans:property name="suffix" value=".jsp" />
    </beans:bean>

    <context:component-scan base-package="com.coderby.myapp" />

    <view-controller view-name="streaming" path="/ai"/>

    <resources location="/WEB-INF/resources/" mapping="/**"/>
    <resources location="/WEB-INF/resources/images/" mapping="/images/**"/>
    <resources location="/WEB-INF/resources/jquey/" mapping="/jquery/**"/>
    <resources location="/WEB-INF/resources/paho-mqtt/" mapping="/paho-mqtt/**"/>
</beans:beans>
```

2) streaming.jsp

```
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!doctype html>
<html>
```

```
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="height=device-height">
    <script type="text/javascript" src='<c:url value="/jquery/jquery-3.6.0.min.js"/> '></script>
    <script type="text/javascript" src='<c:url value="/paho-mqtt/paho-mqtt.js"/> '></script>
    <script type="text/javascript">
    $(function() {
        //클라이언트 인스턴스 생성
        client = new Paho.MQTT.Client("localhost", 61614, new Date().getTime().toString());

        // 이벤트 핸들러 설정
        client.onConnectionLost = onConnectionLost;
        client.onMessageArrived = onMessageArrived;

        // 클라이언트 연결
        client.connect({onSuccess:onConnect});
    })

    // 클라이언트가 연결되었을 때 호출되는 콜백함수
    function onConnect() {
        console.log("mqtt broker subscriber connected");
        client.subscribe("/#");
    }

    // called when the client loses its connection
    function onConnectionLost(responseObject) {
      if (responseObject.errorCode !== 0) {
        console.log("onConnectionLost:"+responseObject.errorMessage);
      }
    }

    // called when a message arrives
    function onMessageArrived(message) {
        if(message.destinationName == "/camerapub") {
            $("#cameraView").attr("src", "data:image/jpg;base64, " + message.payloadString);
        }
    }
    </script>
    <title>약용작물 병충해 모니터링 AI 서비스</title>
    <style>
    div {
        width: 100%;
        height: 100%;
    }
    img#cameraView {
        max-width: 100%;
        max-height: 100%;
        bottom: 0;
        left: 0;
        margin: auto;
        overflow: auto;
        position: fixed;
        right: 0;
        top: 0;
        }
    </style>
```

```
</head>
<body>
    <!--h5 class="alert alert-info">AI 실증 서비스</h5-->
    <div align="center">
        <img id="cameraView" width=100% height=100%/>
    </div>
</body>
</html>
```

3) 전체화면으로 영상이 보이도록 수정하는 스타일

```
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="height=device-height">
...생략...
    <title>약용작물 병충해 모니터링 AI 서비스</title>
    <style>
    div {
        width: 100%;
        height: 100%;
    }
    img#cameraView {
        max-width: 100%;
        max-height: 100%;
        bottom: 0;
        left: 0;
        margin: auto;
        overflow: auto;
        position: fixed;
        right: 0;
        top: 0;
        }
    </style>
</head>
<body>
    <!--h5 class="alert alert-info">AI 실증 서비스</h5-->
    <div align="center">
        <img id="cameraView" width=100% height=100%/>
    </div>
</body>
</html>
```

4.2. 파이썬

1) ai.py

```
# -*- coding: utf-8 -*-
"""
@created 2021년 10월 11일 월요일
@author 허진경
"""
import os
os.environ['CUDA_VISIBLE_DEVICES'] = '1'
# comment out below line to enable tensorflow logging outputs
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '3'
import time
import tensorflow as tf
physical_devices = tf.config.experimental.list_physical_devices('GPU')
if len(physical_devices) > 0:
    tf.config.experimental.set_memory_growth(physical_devices[0], True)
from absl import app, flags
from absl.flags import FLAGS
import core.utils as utils
from tensorflow.python.saved_model import tag_constants
from core.config import cfg
import cv2
import numpy as np
from tensorflow.compat.v1 import ConfigProto
from shapely.geometry import LineString
from tools import preprocessing
from mqtt.Camerapublisher import ImageMqttPublisher
# ------------------------------------------------
def mask_polylines(cap, camera_name, bus):
    pts = []  # 점찍은 좌표를 저장할 리스트
    # setMouseCallback 함수
    def draw_circle(event, x, y, flags, param):
        if event == cv2.EVENT_LBUTTONDBLCLK:
            ix, iy = x, y
            cv2.circle(img, (ix, iy), 3, (0, 255, 0), -1)
            pts.append([ix, iy])
    cv2.namedWindow("image")
    cv2.setMouseCallback("image", draw_circle)
    ret, img = cap.read()
    h,w,c = img.shape
    cv2.rectangle(img, (185, 130), (400 + 100 * int(bus), 230), (0, 0, 0), -1)
    cv2.putText(img, f'{"Bus" * int(bus)} Roi', (200, 200), cv2.FONT_HERSHEY_COMPLEX, 2, (255, 0,
255), 5)
    if h > 1000:
        img = cv2.resize(img, (int(w / 2), int(h / 2)))
    while True:
        cv2.imshow("image", img)  # 동영상 이미지를 보여주는 화면
        # cv2.imshow("bg_mask",bg_mask) #마스크로 저장할 이미지를 보여주는 화면
        k = cv2.waitKey(1) & 0xFF
        if k == ord("s"):
            pts = np.array(pts, dtype=np.int32)
            cv2.polylines(img, [pts], True, (0, 0, 255), 2)
        elif k == 27:
            break
    cv2.destroyAllWindows()
    if h>1000:
        pts *= 2
    # numpy 파일로 저장
    if bus:
        np.save("./pts/"+"bus_"+camera_name,pts)
    else:
        np.save("./pts/" + camera_name, pts)
    return pts

def borderline(pts):
    border = []
    for i in range(len(pts) - 1):
        border.append(LineString([pts[i], pts[i + 1]]))
    border.append(LineString([pts[-1], pts[0]]))
    return border

def point_in_border(point, pts):
    border = borderline(pts)
    com_line = LineString([(0, point[1]), (point[0], point[1])])
    count = 0
    for line in border:
        if com_line.intersection(line):
            count += 1
    if count % 2 == 0:
        return False
    else:
        return True
# ---------------------------------------------------------------------------
flags.DEFINE_string('video', './video/F20009_1_202011260930.avi', 'path to input folder')
flags.DEFINE_string('camera', '165.032.105.25', 'camera ip address')
flags.DEFINE_float('iou', 0.1, 'iou threshold')
flags.DEFINE_float('score', 0.01, 'score threshold')
flags.DEFINE_boolean('dont_show', True, 'show video outputww')
flags.DEFINE_string('IP', '165.132.105.25', 'server ip address')
# rtsp://reputer:reputer@rtsp://reputer01.iptimecam.com:20001/stream_ch00_1 # 재배기 1(정면)

def main(_argv):
    print("start")
    s_t = time.time()
    video_path = FLAGS.video
    # load configuration for object detector
    config = ConfigProto()
    config.gpu_options.allow_growth = True
    STRIDES, ANCHORS, NUM_CLASS, XYSCALE = utils.load_config(FLAGS)
    input_size = 608
    # otherwise load standard tensorflow saved model
    saved_model_loaded = tf.saved_model.load('./checkpoints/yolov4-608',
tags=[tag_constants.SERVING])
    infer = saved_model_loaded.signatures['serving_default']
    video_name = FLAGS.camera # 165.132.105.25
    camera_name = FLAGS.camera # 165.132.105.25 person 111.202.15.53, car F18003_3, person F20009_1
    # begin video capture
    vid = cv2.VideoCapture("rtsp://reputer:reputer@reputer01.iptimecam.com:20001/stream_ch00_1")
    try:
        pts = np.load("./pts/" + camera_name + ".npy")
    except:
        mask_polylines(vid, camera_name, False)
        pts = np.load("./pts/" + camera_name + ".npy")
    print("car video")
    nms_max_overlap = 0.95
    color = {'person': (131, 224, 112), 'bicycle': (51, 221, 255), 'motorbike': (61, 61, 245)}

    allowed_classes = color.keys()
    width = int(vid.get(cv2.CAP_PROP_FRAME_WIDTH))
    height = int(vid.get(cv2.CAP_PROP_FRAME_HEIGHT))
    vid_fps = int(vid.get(cv2.CAP_PROP_FPS))
    codec = cv2.VideoWriter_fourcc(*'XVID')
#    out = cv2.VideoWriter(video_path[:-len(v)] + "detection" + f'/{video_name}_detection.avi',
codec, vid_fps,
 #                          (width, height))
    frame_cnt = 0
#    print(video_path)
    imageMqttPusblisher = ImageMqttPublisher(FLAGS.IP, 1883, "/camerapub")
    imageMqttPusblisher.connect()
    # while video is running
    while True:
        start_time = time.time()
        ret, frame = vid.read()
        if not ret:
            print('Video has ended or failed, try a different video format!')
            break
        # save frame_roi
        frame_roi = frame.copy()
        cv2.polylines(frame_roi, [pts], True, (0, 0, 255), 2)
        frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
        image_data = cv2.resize(frame, (input_size, input_size))  # ,interpolation = cv2.INTER_AREA
        image_data = image_data / 255.
        image_data = image_data[np.newaxis, ...].astype(np.float32)
        # run detections on tflite if flag is set
        batch_data = tf.constant(image_data)
        pred_bbox = infer(batch_data)
        for key, value in pred_bbox.items():
            boxes = value[:, :, 0:4]
            pred_conf = value[:, :, 4:]
        boxes, scores, classes, valid_detections = tf.image.combined_non_max_suppression(
            boxes=tf.reshape(boxes, (tf.shape(boxes)[0], -1, 1, 4)),
            scores=tf.reshape(
                pred_conf, (tf.shape(pred_conf)[0], -1, tf.shape(pred_conf)[-1])),
            max_output_size_per_class=50,
            max_total_size=50,
            iou_threshold=FLAGS.iou,
            score_threshold=FLAGS.score
        )
        # convert data to numpy arrays and slice out unused elements
        num_objects = valid_detections.numpy()[0]
        bboxes = boxes.numpy()[0]
        bboxes = bboxes[0:int(num_objects)]
        scores = scores.numpy()[0]
        scores = scores[0:int(num_objects)]
        classes = classes.numpy()[0]
        classes = classes[0:int(num_objects)]
        # format bounding boxes from normalized ymin, xmin, ymax, xmax ---> xmin, ymin, width, height
        original_h, original_w, _ = frame.shape
        bboxes = utils.format_boxes(bboxes, original_h, original_w)
        mw = bboxes[:, 2].mean()
        # read in all class names from config
        class_names = utils.read_class_names(cfg.YOLO.CLASSES)
        # loop through objects and use class index to get class name, allow only classes in
allowed_classes list
        names = []
        deleted_indx = []
        for i in range(num_objects):
            class_indx = int(classes[i])
            class_name = class_names[class_indx]
            w = bboxes[i][2] - bboxes[i][0]
            h = bboxes[i][3] - bboxes[i][1]
            if class_name not in allowed_classes:
                deleted_indx.append(i)
            elif (w > 300) or (h > 300):
                deleted_indx.append(i)
            else:
                names.append(class_name)
        names = np.array(names)
        # delete detections that are not in allowed_classes
        bboxes = np.delete(bboxes, deleted_indx, axis=0)
        scores = np.delete(scores, deleted_indx, axis=0)
        classes = np.delete(classes, deleted_indx, axis=0)
        indices = preprocessing.non_max_suppression(bboxes, classes, nms_max_overlap, scores)
        if frame_cnt % 3 == 0:
            count = {cn:0 for cn in color.keys()}
        for ind in indices:
            bbox = bboxes[ind]
            class_name = names[ind]
            x, y, w, h = bbox
            if (w > 300) or (h > 300):
                continue
            center = (x + int(w / 2), y + int(h / 2))
            if point_in_border(center, pts):
                if FLAGS.person == False:
                    if point_in_border(center,bpts):
                        if w > mw:
                            class_name = "bus"
                # draw bbox on screen
                cv2.rectangle(frame, (int(x), int(y)), (int(x) + int(w), int(y) + int(h)),
color[class_name], 1)
                cv2.putText(frame, class_name, (int(bbox[0]), int(bbox[1]) + 10), 0, 0.5,
                            (0, 0, 0), 2)
                if frame_cnt%3 == 0:
                    count[class_name] += 1
        text_height = 50
        cv2.rectangle(frame, (50, 0), (450, 20 + len(color.keys())*50), (0, 0, 0), -1)
        for cn in color.keys():
            cv2.putText(frame, f'{cn} : {count[cn]}', (60, text_height), cv2.FONT_HERSHEY_COMPLEX,
1.5, color[cn], 2)
            text_height += 50
        result = np.asarray(frame)
        result = cv2.cvtColor(frame, cv2.COLOR_RGB2BGR)
        # out.write(result)
        result = cv2.resize(result, None, fx=0.5, fy= 0.5)
        imageMqttPusblisher.sendBase64(result)
        # calculate frames per second of running detections
        if not FLAGS.dont_show:
            if height>720:
                result = cv2.resize(result, (int(width/2), int(height/2)))
            cv2.imshow("Output Video", result)
        # if output flag is set, save video file
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break_flag = 0
            # out.release()
            break
        fps = 1.0 / (time.time() - start_time)
        # print("{0} {1} : {2:.2f}fps".format(FLAGS.camera, frame_cnt, fps), end='\r')
        frame_cnt += 1
    imageMqttPusblisher.disconnect()
    # xml 생성
    vid.release()
    # out.release()
    cv2.destroyAllWindows()
    print(time.time() - s_t)
    cv2.destroyAllWindows()
if __name__ == '__main__':
    try:
        app.run(main)
    except SystemExit:
        pass
```

5. 문제 해결

5.1. ActiveMQ 서버 재실행

리눅스에서 ActiveMQ 서버가 동작하지 않을 경우 서버 재실행해야 함, 종료 후 다시 실행시키기 전에 WrapperSimpleApp 프로세스가 살아 있으면 kill -9 명령으로 종료시켜야 함

```
(base) oem@reputer:~/activemq-5.16/bin$ ./activemq stop
INFO: Loading '/home/oem/activemq-5.16//bin/env'
INFO: Using java '/home/oem/jdk1.8.0_271/bin/java'
INFO: Waiting at least 30 seconds for regular process termination of pid '2380' :
Java Runtime: Oracle Corporation 1.8.0_271 /home/oem/jdk1.8.0_271/jre
  Heap sizes: current=62976k  free=61992k  max=932352k
    JVM args: -Xms64M -Xmx1G -Djava.util.logging.config.file=logging.properties
-Djava.security.auth.login.config=/home/oem/activemq-5.16//conf/login.config
-Dactivemq.classpath=/home/oem/activemq-5.16//conf:/home/oem/activemq-5.16//../lib/:
-Dactivemq.home=/home/oem/activemq-5.16/ -Dactivemq.base=/home/oem/activemq-5.16/
-Dactivemq.conf=/home/oem/activemq-5.16//conf -Dactivemq.data=/home/oem/activemq-5.16//data
Extensions classpath:

[/home/oem/activemq-5.16/lib,/home/oem/activemq-5.16/lib/camel,/home/oem/activemq-5.16/lib/optiona
l,/home/oem/activemq-5.16/lib/web,/home/oem/activemq-5.16/lib/extra]
ACTIVEMQ_HOME: /home/oem/activemq-5.16
ACTIVEMQ_BASE: /home/oem/activemq-5.16
ACTIVEMQ_CONF: /home/oem/activemq-5.16/conf
ACTIVEMQ_DATA: /home/oem/activemq-5.16/data
Connecting to pid: 2380
Stopping broker: localhost
.. TERMINATED

(base) oem@reputer:~/activemq-5.16/bin$ ./activemq start
INFO: Loading '/home/oem/activemq-5.16//bin/env'
INFO: Using java '/home/oem/jdk1.8.0_271/bin/java'
INFO: Starting - inspect logfiles specified in logging.properties and log4j.properties to get details
INFO: pidfile created : '/home/oem/activemq-5.16//data/activemq.pid' (pid '8175')
(base) oem@reputer:~/activemq-5.16/bin$
```