(https://www.youtube.com/watch?v=3YqbO2AlepM)

Dustom Object Detection Tensorflow Lite Raspberry Pi Bookworm

$ git clone https://github.com/freedomwebtech/tflite-custom-object-bookworm

```
pi@raspberrypi:~ $ git clone https://github.com/freedomwebtech/tflite-custom-obj
ect-bookworm
'tflite-custom-object-bookworm'에 복제합니다...
remote: Enumerating objects: 23, done.
remote: Counting objects: 100% (23/23), done.
remote: Compressing objects: 100% (22/22), done.
remote: Total 23 (delta 5), reused 0 (delta 0), pack-reused 0
오브젝트를 받는 중: 100% (23/23), 3.22 MiB | 5.02 MiB/s, 완료.
델타를 알아내는 중: 100% (5/5), 완료.
pi@raspberrypi:~ $
```

$ sudo rm /usr/lib/python3.11/EXTERNALLY-MANAGED

```
파일(F)  편집(E)  탭(T)  도움말(H)
 GNU nano 7.2                          rpi4.txt
sudo rm /usr/lib/python3.11/EXTERNALLY-MANAGED
```

$ sudo pip3 install opencv-python

```
pi@raspberrypi:~ $ sudo pip3 install opencv-python
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting opencv-python
  Downloading opencv_python-4.9.0.80-cp37-abi3-manylinux_2_17_aarch64.manylinux2
014_aarch64.whl (41.3 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 41.3/41.3 MB 2.9 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.21.2 in /usr/lib/python3/dist-packages (
from opencv-python) (1.24.2)
Installing collected packages: opencv-python
Successfully installed opencv-python-4.9.0.80
WARNING: Running pip as the 'root' user can result in broken permissions and con
flicting behaviour with the system package manager. It is recommended to use a v
irtual environment instead: https://pip.pypa.io/warnings/venv
pi@raspberrypi:~ $
```

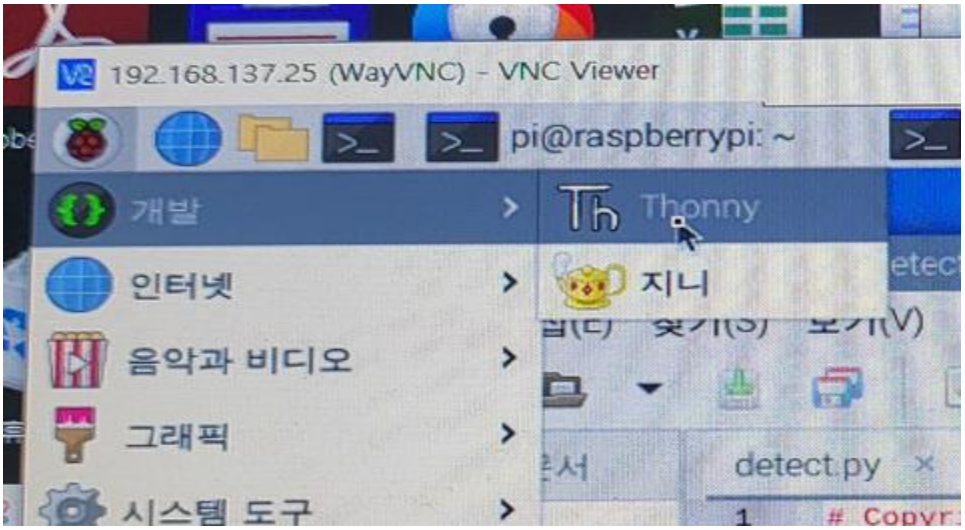
$ sudo pip3 install mediapipe

```
pi@raspberrypi:~ $ sudo pip3 install mediapipe
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting mediapipe
  Downloading mediapipe-0.10.13-cp311-cp311-manylinux_2_17_aarch64.manylinux201
4_aarch64.whl (33.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 33.0/33.0 MB 4.4 MB/s eta 0:00:00

Collecting absl-py
  Downloading https://www.piwheels.org/simple/absl-py/absl_py-2.1.0-py3-none-an
y.whl (133 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 133.7/133.7 kB 110.3 kB/s eta 0:00:0
0
Collecting attrs>=19.1.0
  Downloading https://www.piwheels.org/simple/attrs/attrs-23.2.0-py3-none-any.w
hl (60 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 60.8/60.8 kB 170.7 kB/s eta 0:00:0
0
Collecting flatbuffers>=2.0
  Downloading https://www.piwheels.org/simple/flatbuffers/flatbuffers-201810032
```

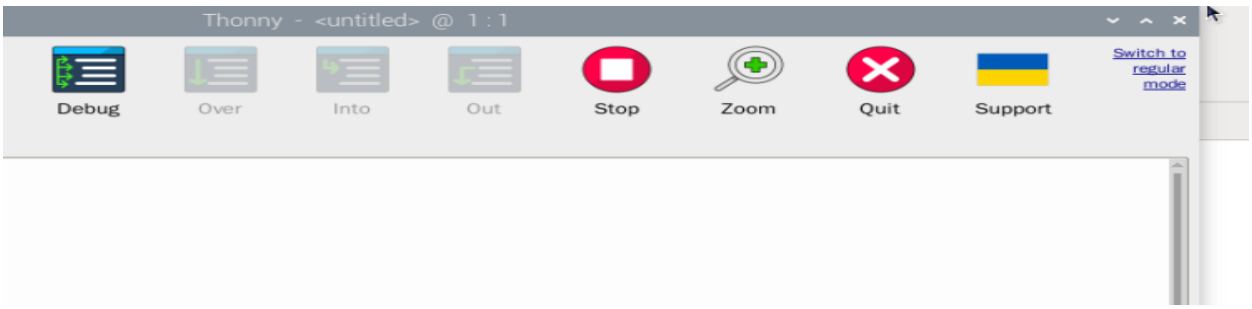
(중략)

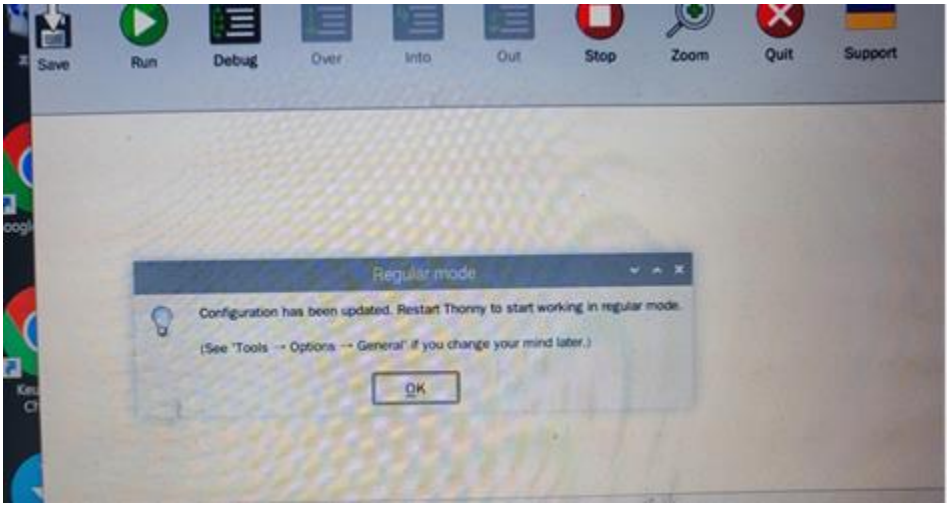
```
  Downloading https://www.piwheels.org/simple/pycparser/pycparser-2.22-py3-none
-any.whl (117 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 117.6/117.6 kB 3.7 MB/s eta 0:00:00
Requirement already satisfied: six>=1.5 in /usr/lib/python3/dist-packages (from
 python-dateutil>=2.7->matplotlib->mediapipe) (1.16.0)
Installing collected packages: flatbuffers, scipy, python-dateutil, pyparsing,
pycparser, protobuf, packaging, opt-einsum, opencv-contrib-python, ml-dtypes, k
iwisolver, fonttools, cycler, contourpy, attrs, absl-py, matplotlib, jaxlib, ja
x, CFFI, sounddevice, mediapipe
Successfully installed CFFI-1.16.0 absl-py-2.1.0 attrs-23.2.0 contourpy-1.2.1 c
ycler-0.12.1 flatbuffers-20181003210633 fonttools-4.51.0 jax-0.4.26 jaxlib-0.4.
26 kiwisolver-1.4.5 matplotlib-3.8.4 mediapipe-0.10.13 ml-dtypes-0.4.0 opencv-c
ontrib-python-4.9.0.80 opt-einsum-3.3.0 packaging-24.0 protobuf-4.25.3 pycparse
r-2.22 pyparsing-3.1.2 python-dateutil-2.9.0.post0 scipy-1.13.0 sounddevice-0.4
.6
WARNING: Running pip as the 'root' user can result in broken permissions and co
nflicting behaviour with the system package manager. It is recommended to use a
 virtual environment instead: https://pip.pypa.io/warnings/venv
pi@raspberrypi:~ $
```

. 라즈베리파이 버튼 중 ‘개발’에서 ‘Thonny’를 클릭

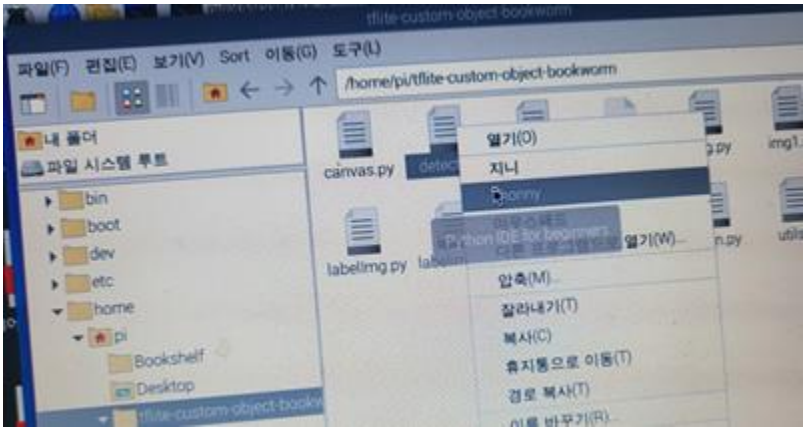


*** 우측 상단에서 ‘Switch to regular mode’를 클릭하고, 뜨는 창에서 ‘ok’ 눌러 종료**





*** Thonny 창을 닫음. 파일메니져에서 'detect.py' (USB 카메라용. detect1.py 는 파이 카메라용)를 우측 버턴에서 'Thonny'를 오픈**



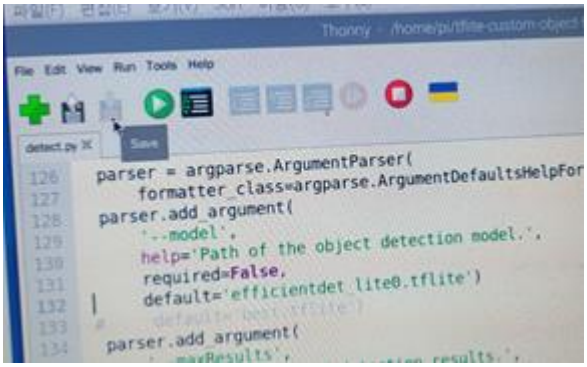
. detect.py 에서 #47 의 'VideoCapture(0)'에서 '0'는 USB 카메라용

```
45
46  # Start capturing video input from the camera
47  cap = cv2.VideoCapture(0)
48  cap.set(cv2.CAP_PROP_FRAME_WIDTH, width)
49  cap.set(cv2.CAP_PROP_FRAME_HEIGHT, height)
50
51  # Visualization parameters
```

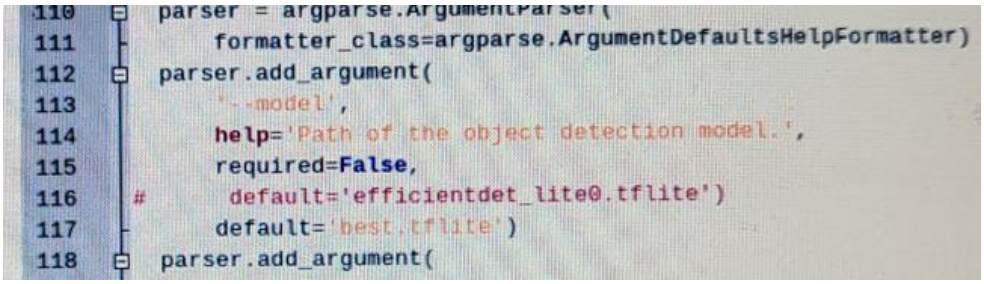
. # 133 를 comment 하고 (custom model 결과라서), #132 의 커멘트를 해제

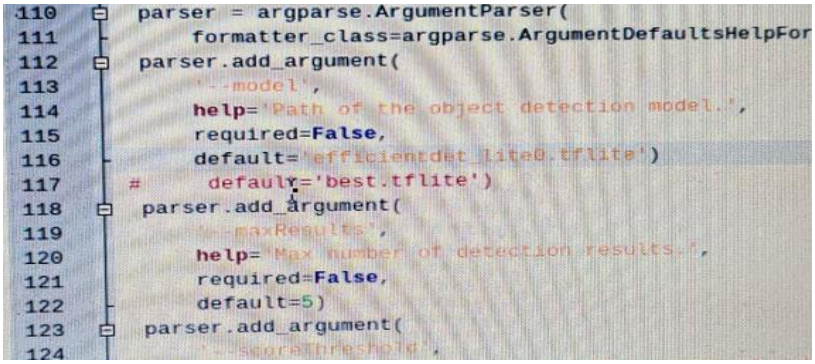
```
131       required=False,
132       default='efficientdet_lite0.tflite')
133  #     default='best.tflite')
134   parser.add_argument(
135       '--maxResults',
```

. 수정한 후, 상단의 저장 버튼을 클릭하여 저장

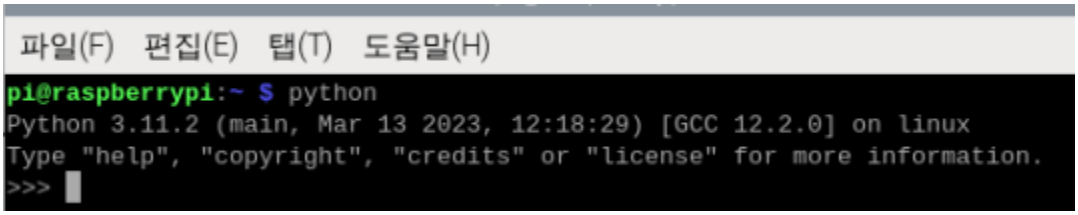


. picamera 경우는 ‘detect1.py’를 불러서 같은 방법으로 수정





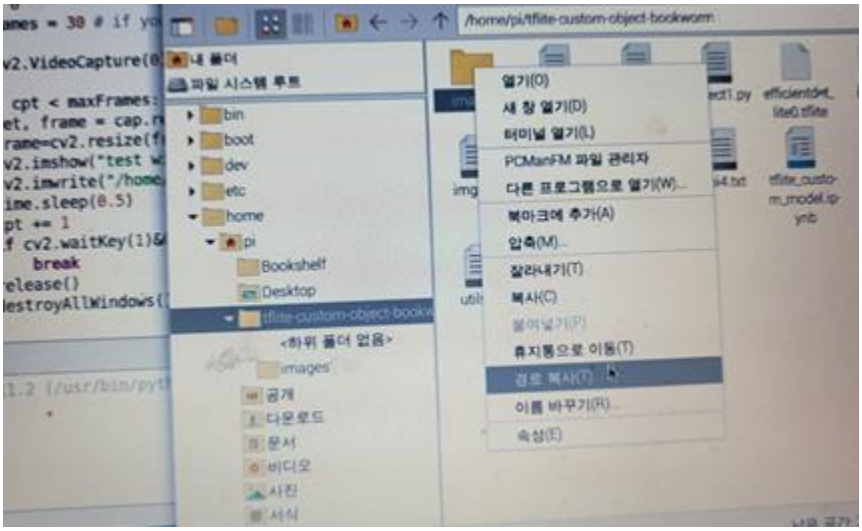
. 파이썬은 3.11 버전을 사용 중



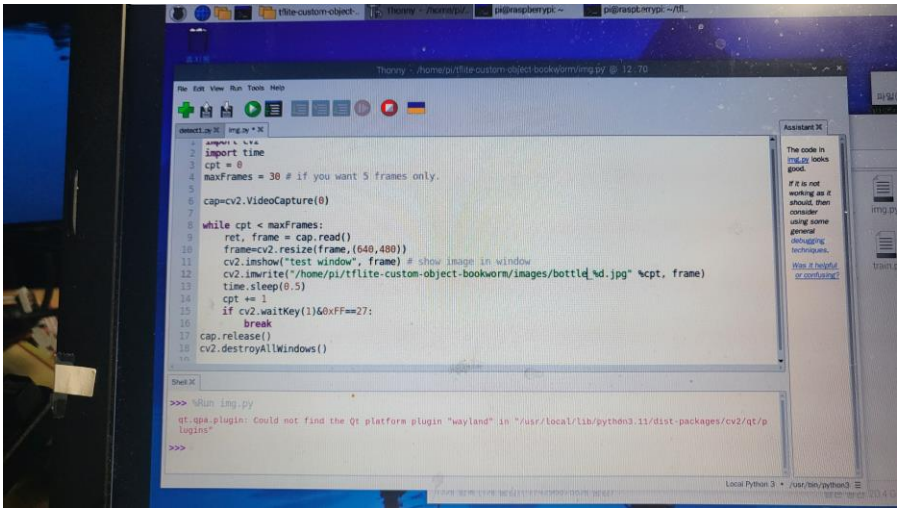
custom 이미지 만들기(img.py 이용)

$ mkdir images

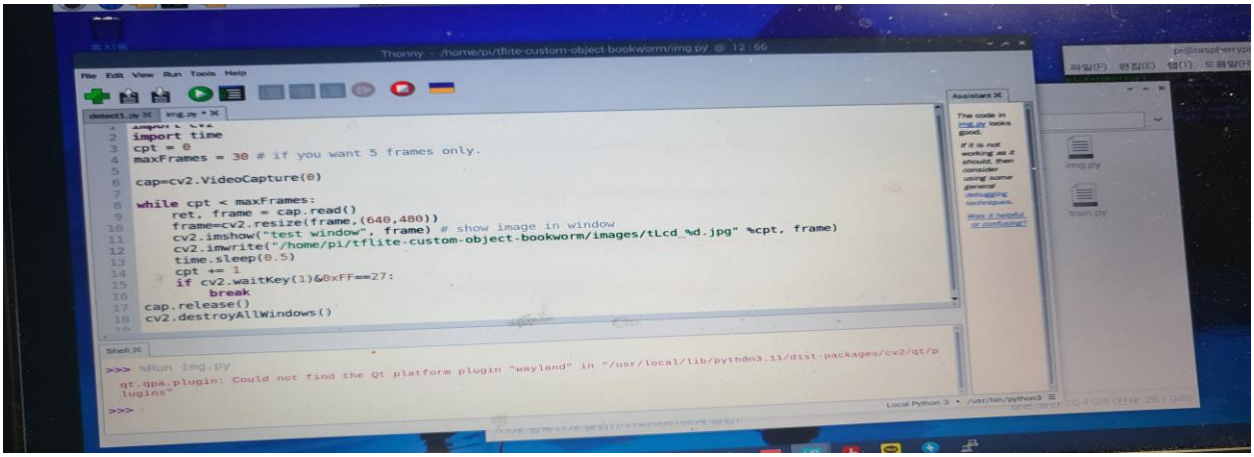
. images 폴더의 path 를 마우스 우측 버튼 이용 복사한 후 img.py 수정



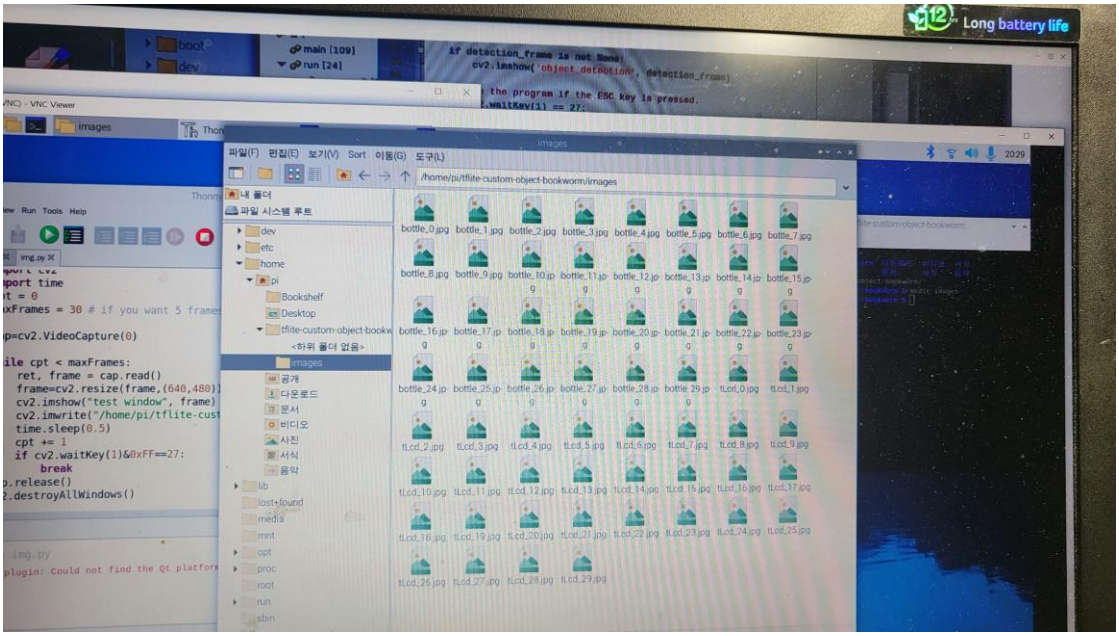
. img.py 에서 USB 카메라를 이용해서 '통'(bottle,각자가 원하는 object)을 천천히 움직여서 30 장 사진 얻음



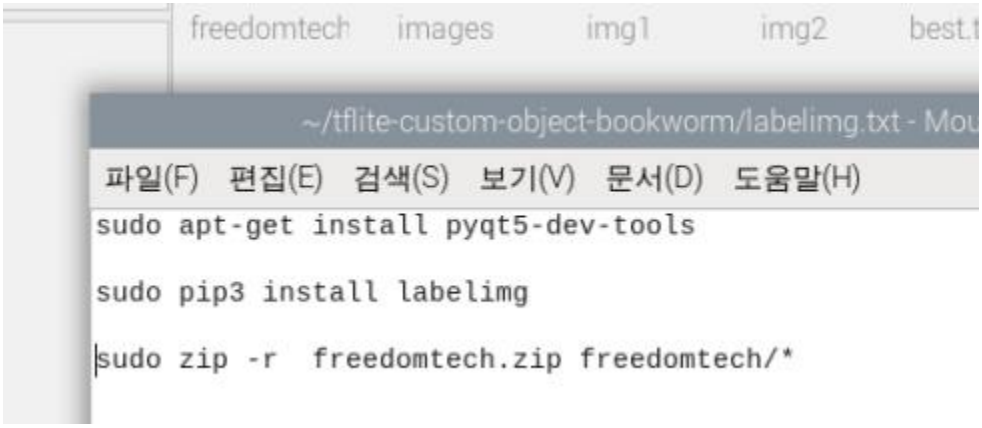
. img.py 에서 USB 카메라를 이용해서 'Text LCD'(각자가 원하는 object)를 천천히 움직여서 30 장 사진 얻음 (다른 img2 를 만들어서 별도로 하는 걸 추천. 삭제 때문에.)



. images 폴더에는 총 60 장 사진이 저장됨



. 이미지 라벨 설정(labelimg.txt 에서 첫 줄 복사)



$ sudo apt-get install pyqt5-dev-tools

```
pi@raspberrypi:~/tflite-custom-object-bookworm $ sudo apt-get install pyqt5-dev-tools
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음 패키지가 자동으로 설치되었지만 더 이상 필요하지 않습니다:
  libraspberrypi0
Use 'sudo apt autoremove' to remove it.
다음 새 패키지를 설치할 것입니다:
  pyqt5-dev-tools
0개 업그레이드, 1개 새로 설치, 0개 제거 및 0개 업그레이드 안 함.
177 kB바이트 아카이브를 받아야 합니다.
이 작업 후 479 kB바이트의 디스크 공간을 더 사용하게 됩니다.
받기:1 http://deb.debian.org/debian bookworm/main arm64 pyqt5-dev-tools arm64 5.15.9+dfsg-1 [177 kB]
내려받기 177 kB바이트, 소요시간 0초 (1,370 kB바이트/초)
Selecting previously unselected package pyqt5-dev-tools.
(데이터베이스 읽는중 ...현재 149209개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack .../pyqt5-dev-tools_5.15.9+dfsg-1_arm64.deb ...
Unpacking pyqt5-dev-tools (5.15.9+dfsg-1) ...
pyqt5-dev-tools (5.15.9+dfsg-1) 설정하는 중입니다 ...
Processing triggers for man-db (2.11.2-2) ...
pi@raspberrypi:~/tflite-custom-object-bookworm $
```

. 두번째 줄 실행 ($ sudo pip3 install labelimg)

```
pi@raspberrypi:~/tflite-custom-object-bookworm $ sudo pip3 install labelimg
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting labelimg
  Downloading https://www.piwheels.org/simple/labelimg/labelImg-1.8.6-py2.py3-none-any.whl (265 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 265.7/265.7 kB 282.8 kB/s eta 0:00:00
Requirement already satisfied: lxml in /usr/lib/python3/dist-packages (from labelimg) (4.9.2)
Requirement already satisfied: pyqt5 in /usr/lib/python3/dist-packages (from labelimg) (5.15.9)
Requirement already satisfied: PyQt5-sip<13,>=12.11 in /usr/lib/python3/dist-packages (from pyqt5->labelimg) (1
2.11.1)
Installing collected packages: labelimg
Successfully installed labelimg-1.8.6
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the sys
tem package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
pi@raspberrypi:~/tflite-custom-object-bookworm $
```

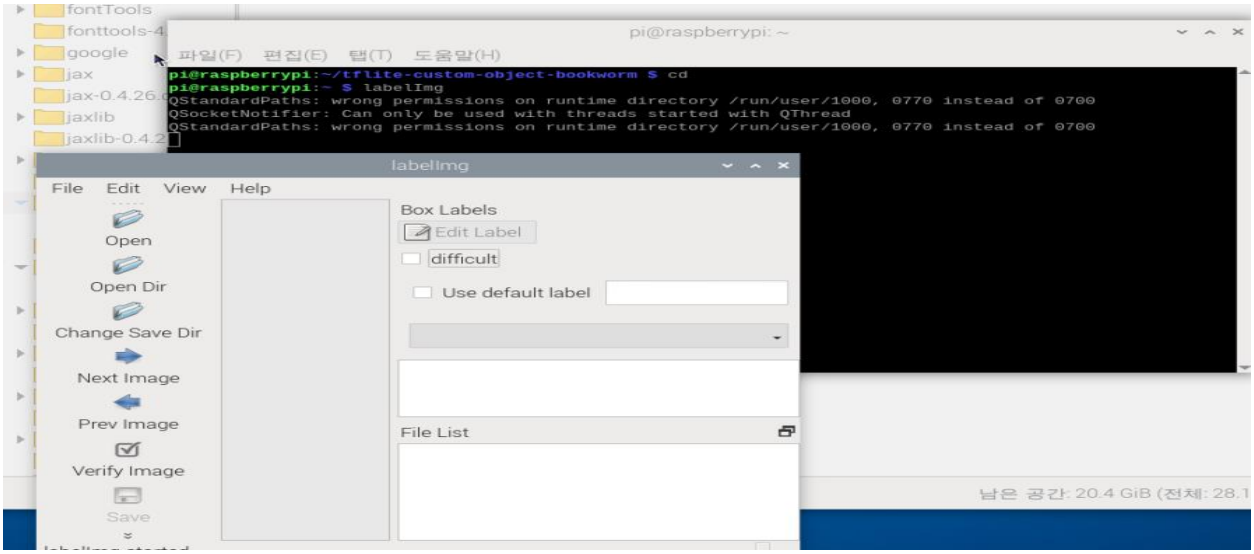
. canvas.py 삭제($ sudo rm –rf /usr/local/lib/python3.11/dist-packages/libs/canvas.py)하고 이곳에 새로운 canvas.py(repository 에 있는)를 복사.

```
파일(F)  편집(E)  탭(T)  도움말(H)
pi@raspberrypi:~ $ sudo rm -rf /usr/local/lib/python3.11/dist-packages/libs/canv
as.py
pi@raspberrypi:~ $ cd tflite-custom-object-bookworm/
pi@raspberrypi:~/tflite-custom-object-bookworm $ sudo mv canvas.py /usr/local/li
b/python3.11/dist-packages/libs
pi@raspberrypi:~/tflite-custom-object-bookworm $
```

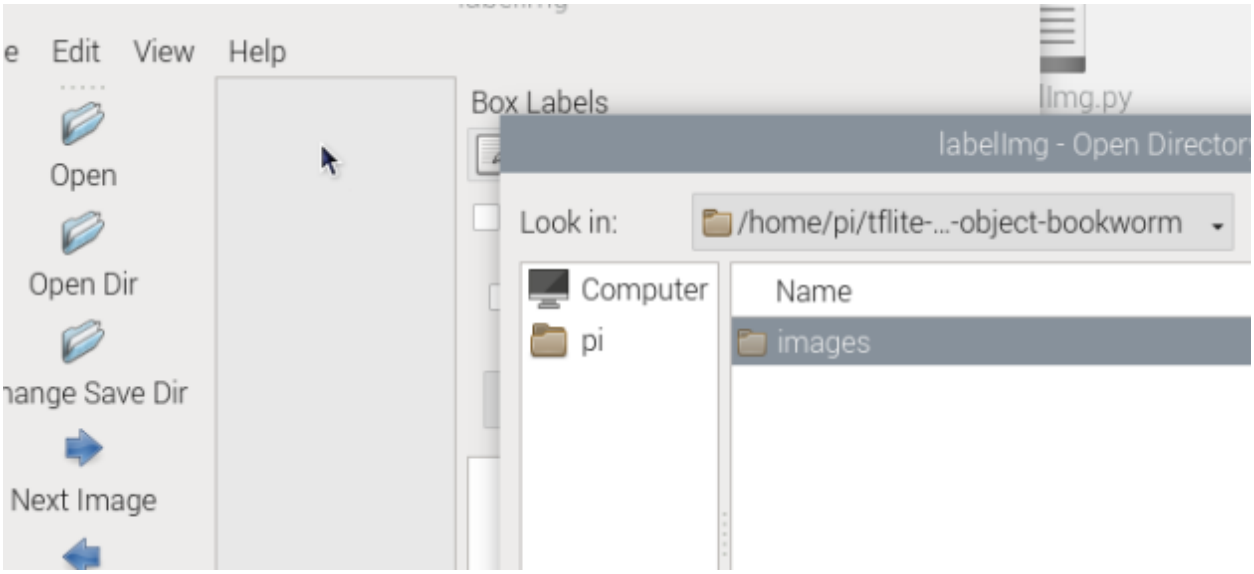
. labelimg.py (/usr/local/lib/python3.11/dist-packages/labelImg/labelImg.py)

```
pi@raspberrypi:~/tflite-custom-object-bookworm $ sudo rm -rf /usr/local/lib/python3.11/dist-packages/lab
elImg/labelImg.py
pi@raspberrypi:~/tflite-custom-object-bookworm $ sudo mv labelImg.py /usr/local/lib/python3.11/dist-pack
ages/labelImg
pi@raspberrypi:~/tflite-custom-object-bookworm $
```

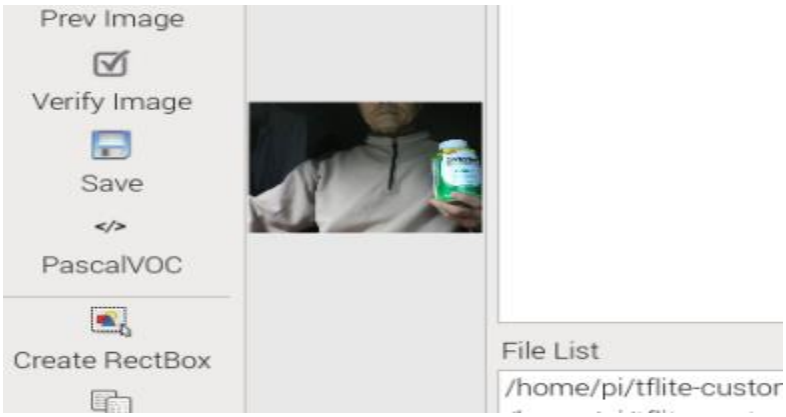
$ labelImg



. ‘Open Dir’과 ‘Change Save Dir’을 모두 img1 (또는 이미지가 있는 폴더)폴더로 지정

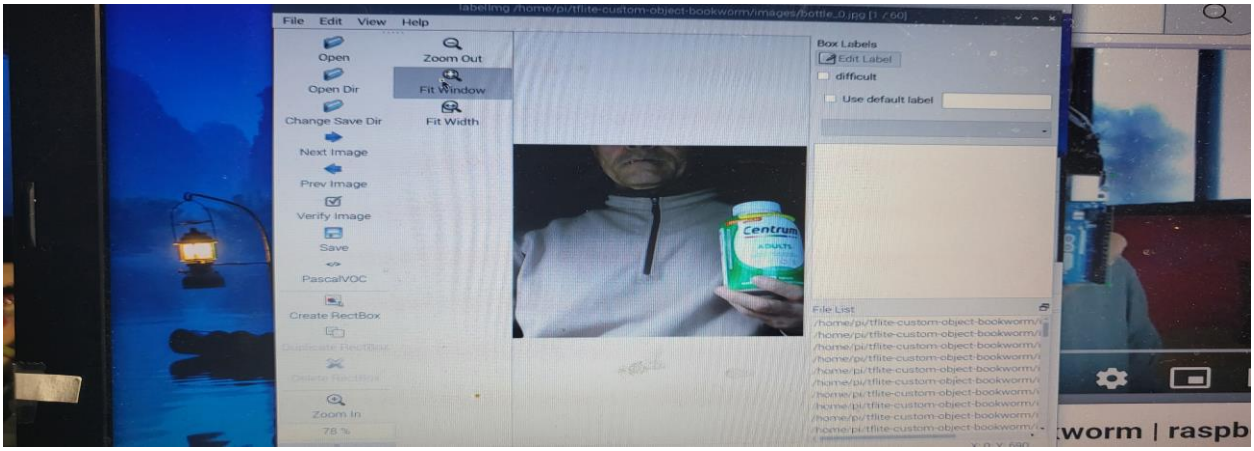


. format 은 ‘PascalVoc’ (TensorFlow 용)선택

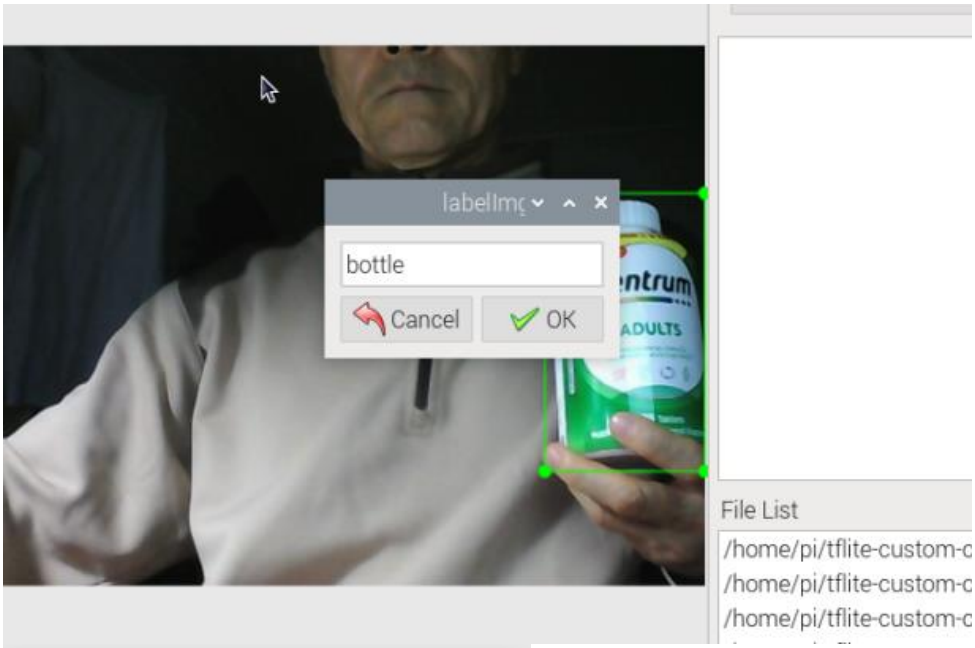


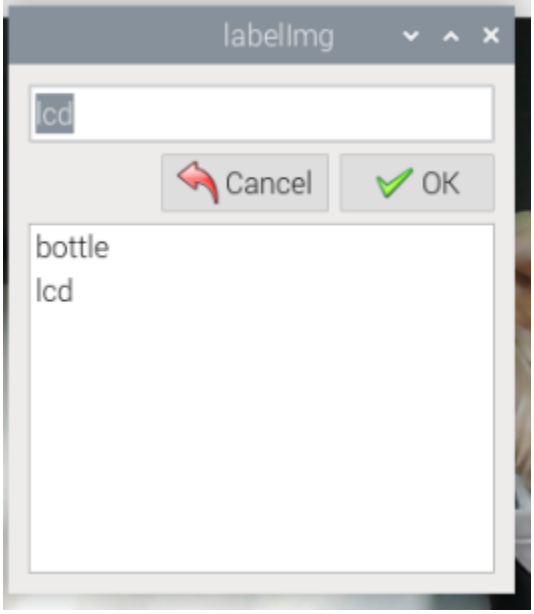
. ‘Create RectBox’를 선택하고 라벨링 시작

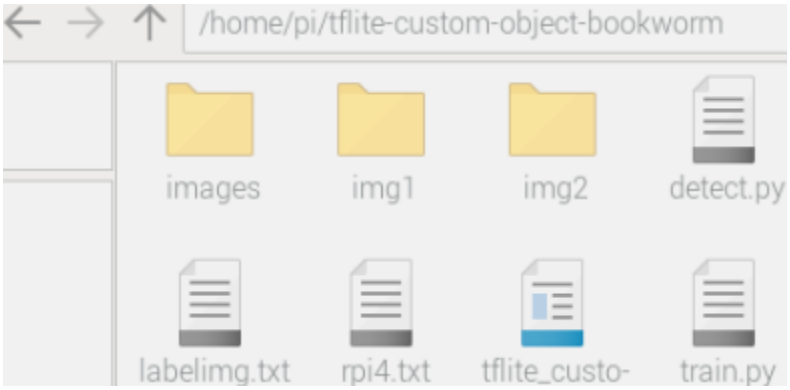
. 그림 사이즈가 작으면 좌측 하단 메뉴 중에서 ‘Zoom In’을 선택하고 뜨는 창에서 ‘Fit Window’를 선택



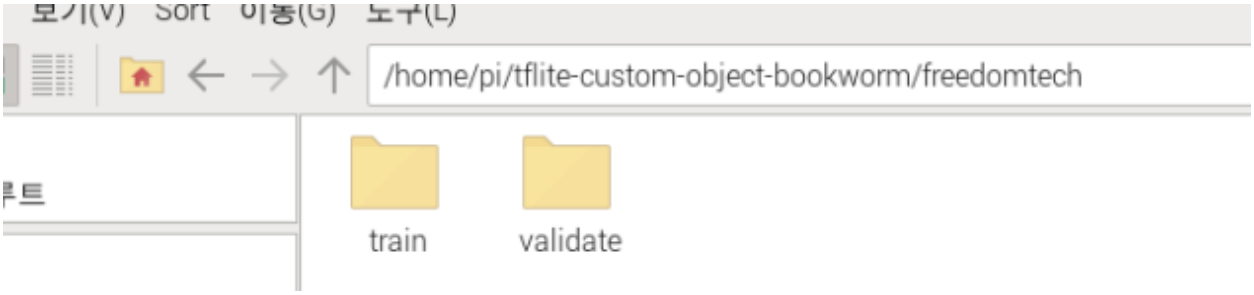
. 다시 ‘Create RectangleBox’를 누르고 직사각형으로 라벨링. 이름은 ‘bottle’로 (선택한 object 에 맞는 이름으로) 쓰고 ‘Ok’ 클릭. 좌측에서 ‘Save’ 버턴을 누르고 ‘Next Image’를 클릭
. 2 번째는 폴더를 img2(본인이 정한 이미지 폴더)를 만들고 ‘lcd’(선택한 object 에 맞는 이름으로)이름으로 라벨링 (마음에 안 들 경우 폴더 삭제가 쉬움). 나중에 images 폴더를 만들어서 합칩.





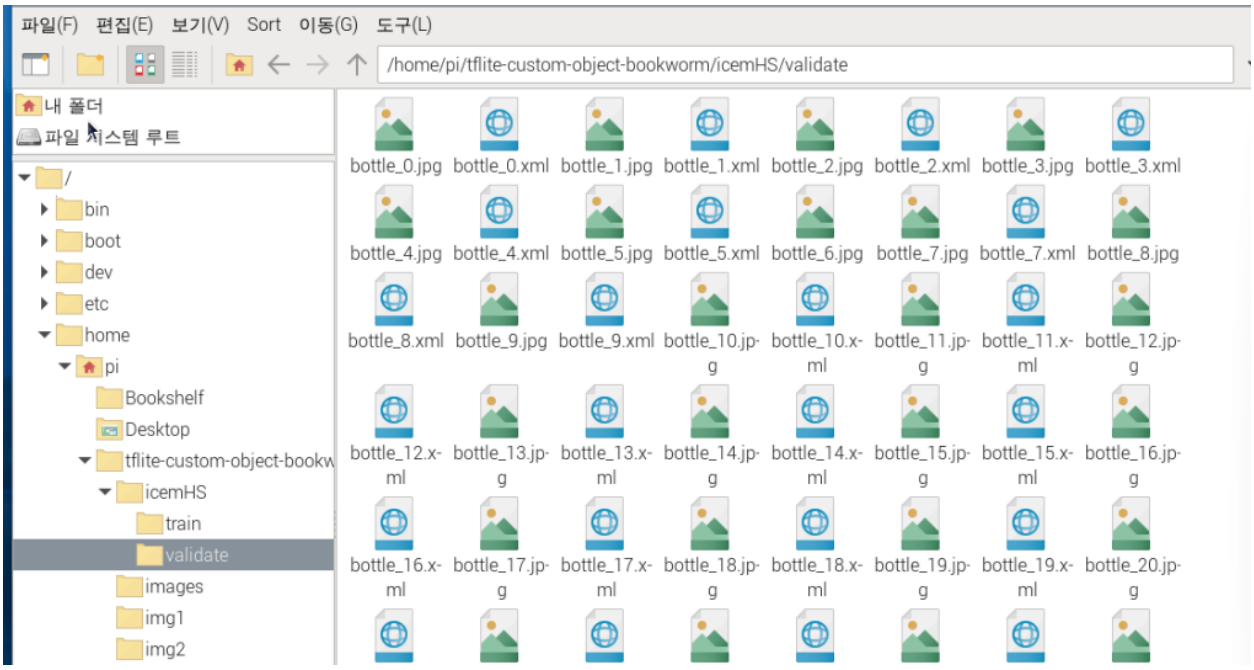


. feedomtech 폴더를 만들고, 하위 폴더로 ‘train’, ‘validate’ 폴더를 만듦

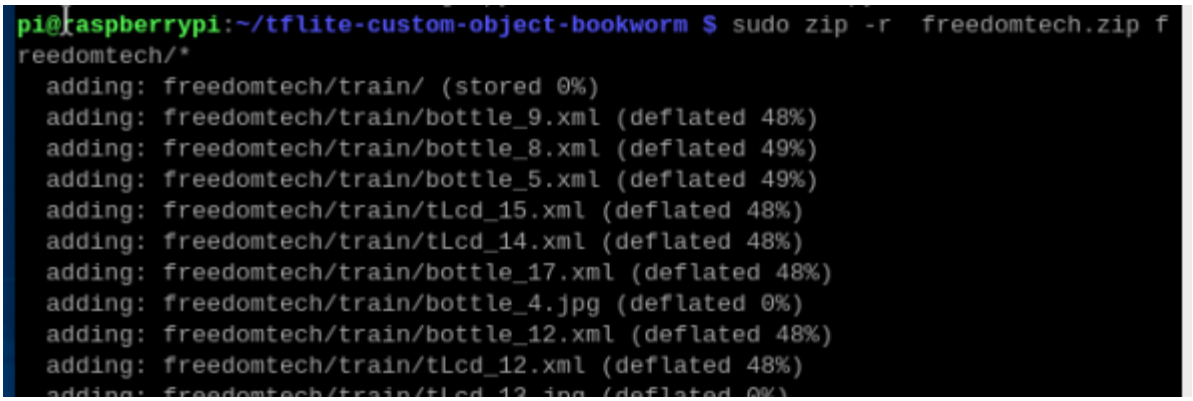


. 폴더 ‘train’, ‘validate’에 images 폴더 모두를 복사해 넣음

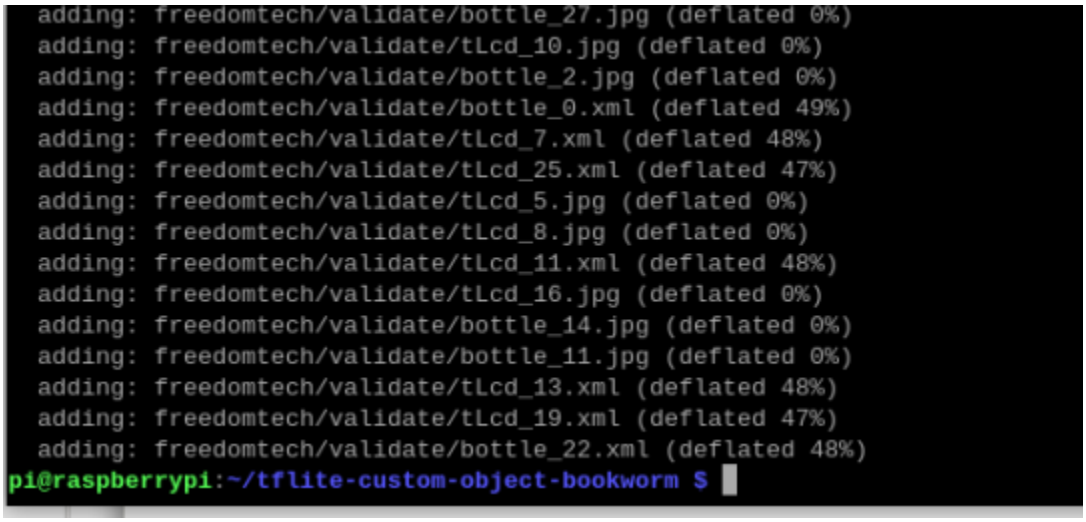
(‘편집’에 있는 ‘모두 선택(^A)’을 이용)



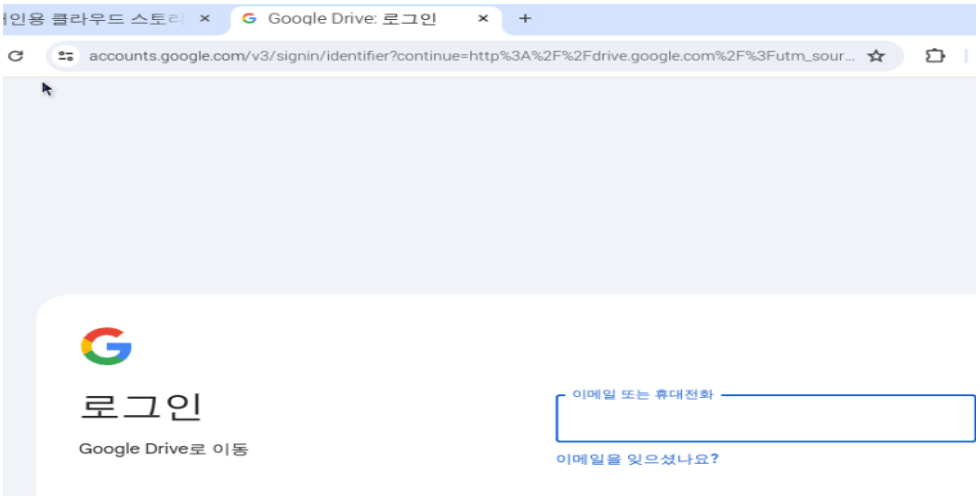
.labelimg.txt 의 세번째 명령 실행 ($ sudo zip -r freedomtech.zip freedomtech/*)



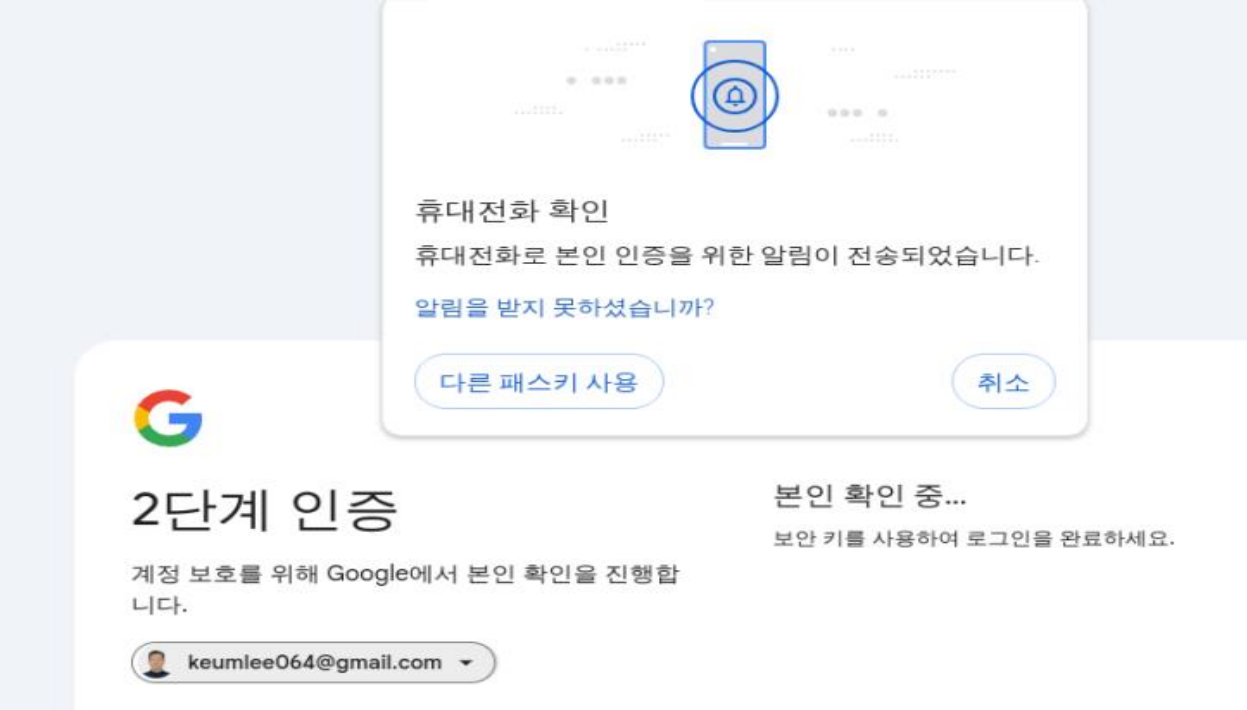
(중략)



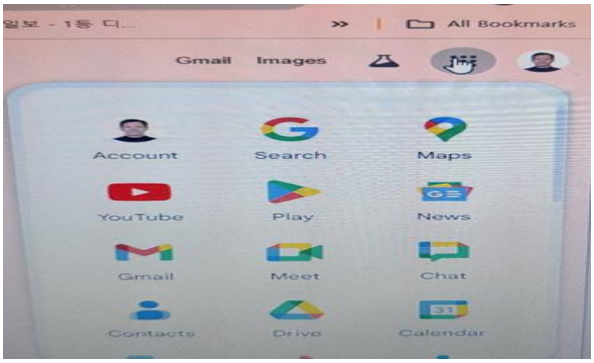
. 구글로그인.



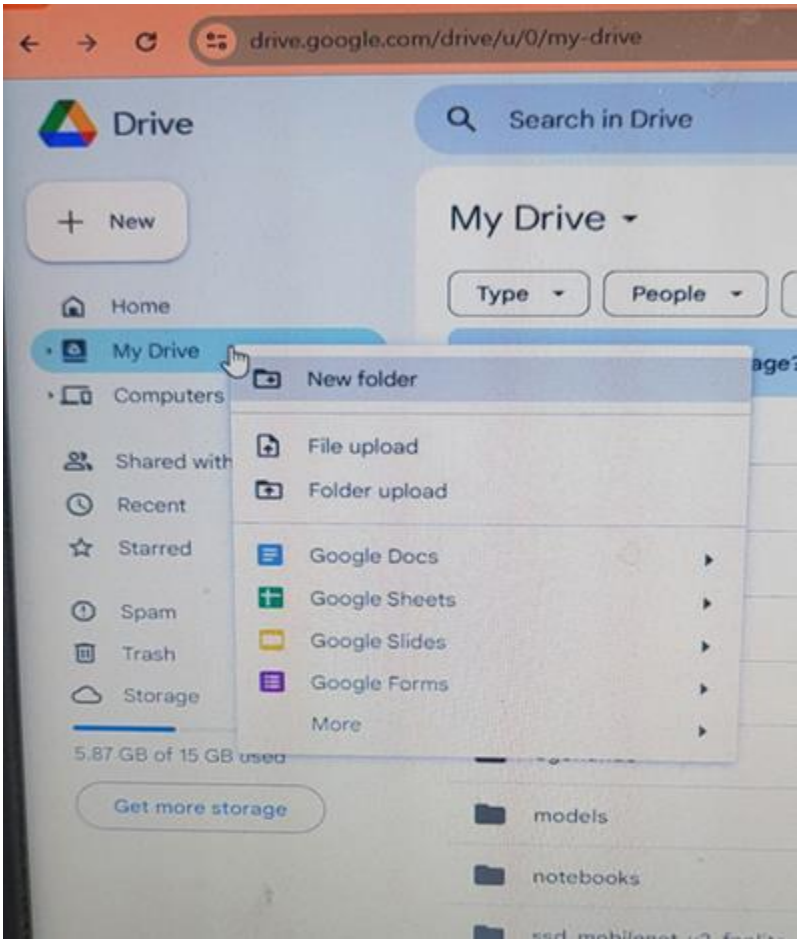
. 본인 인증절차를 거침



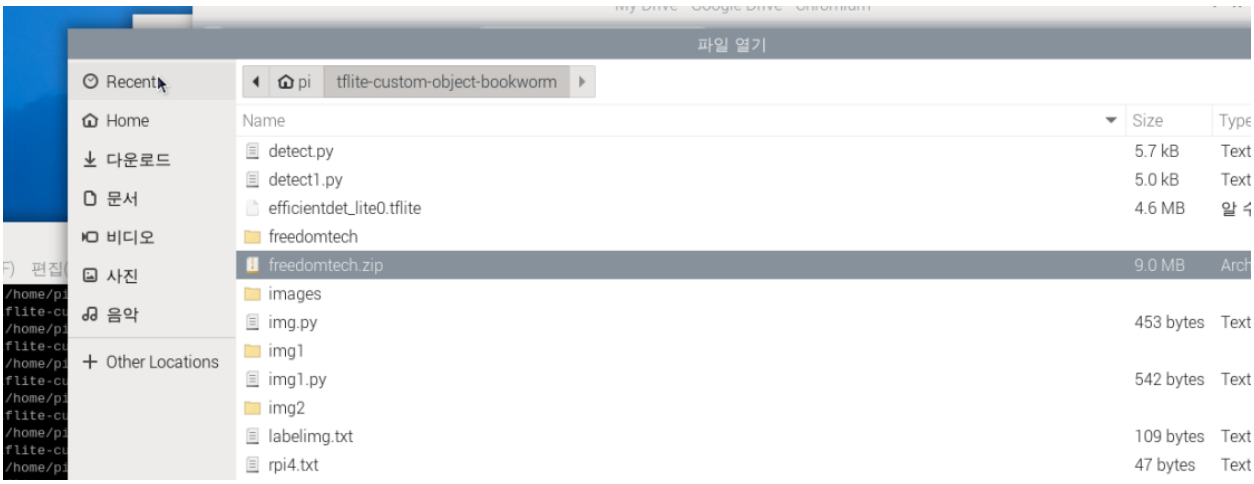
드라이버에 upload (구글 계정이 필요). 자기 이름의 좌측에 있는 점 9 개를 눌러서 뜨는 창에서 'Drive'를 클릭



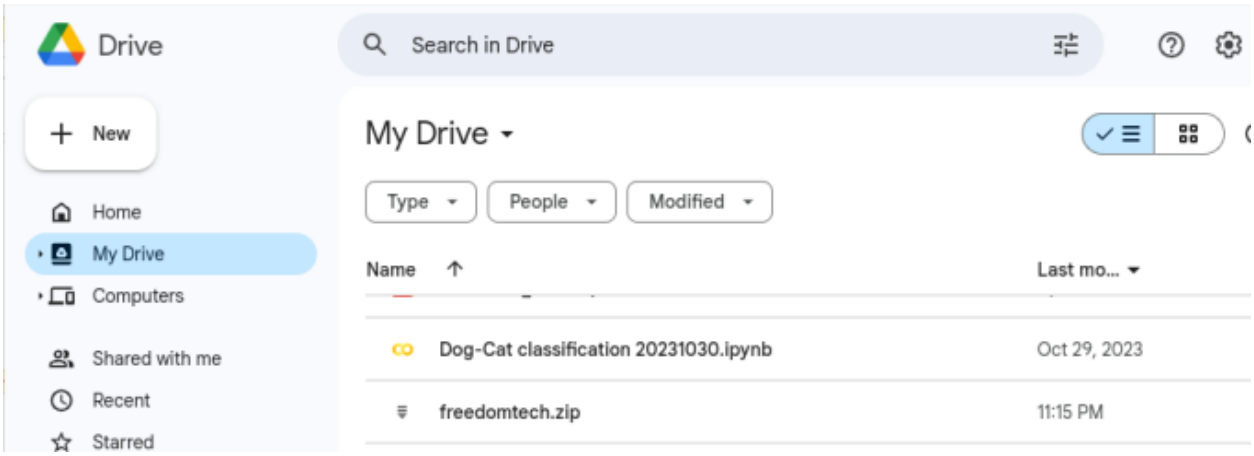
. 좌측 ‘My Drive’를 선택하고 뜨는 창에서 ‘File upload’를 클릭



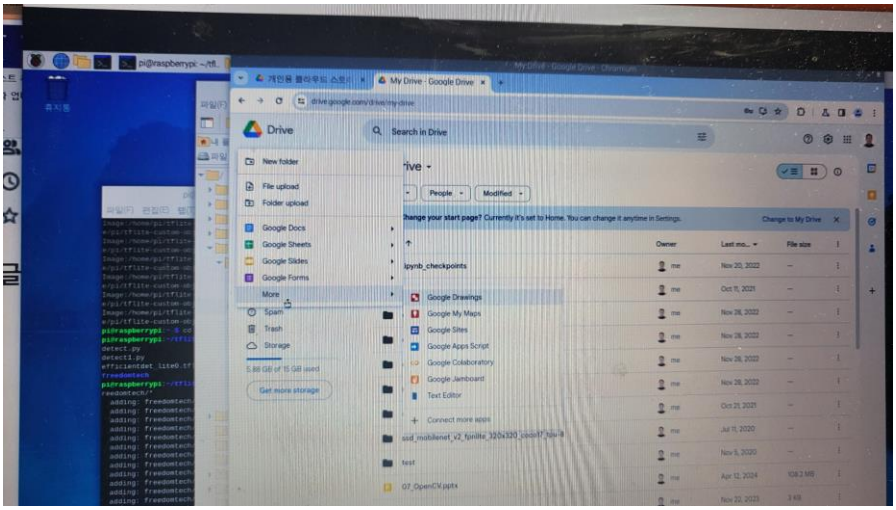
. 압축 파일 (freedomtech.zip)
을 선택하고 하단의 ‘open’ 클릭



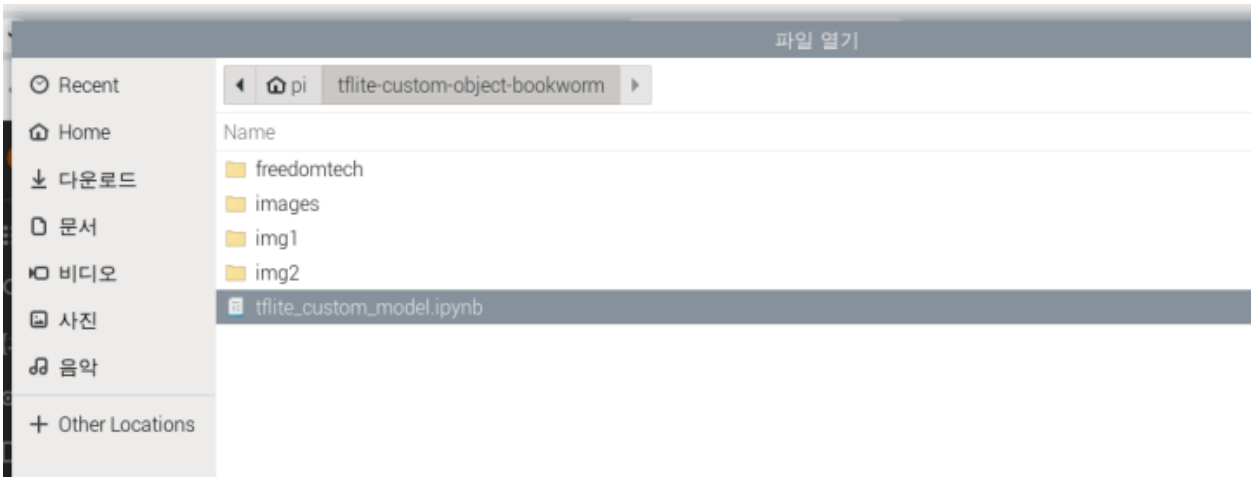
. 랩탑에서 구글 드라이버 염

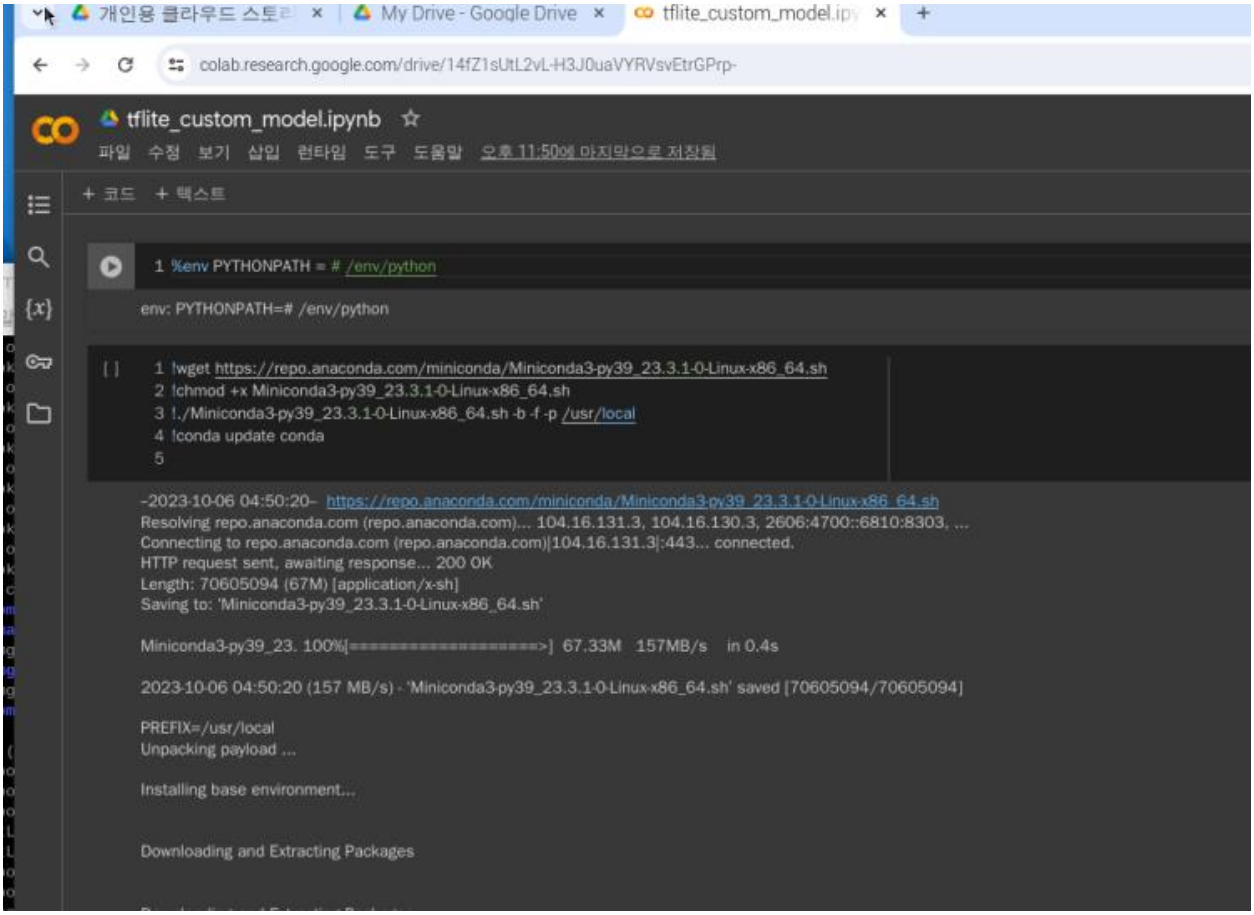


. 구글 Colab 오픈 (+New/ More / Google Colaboratory)

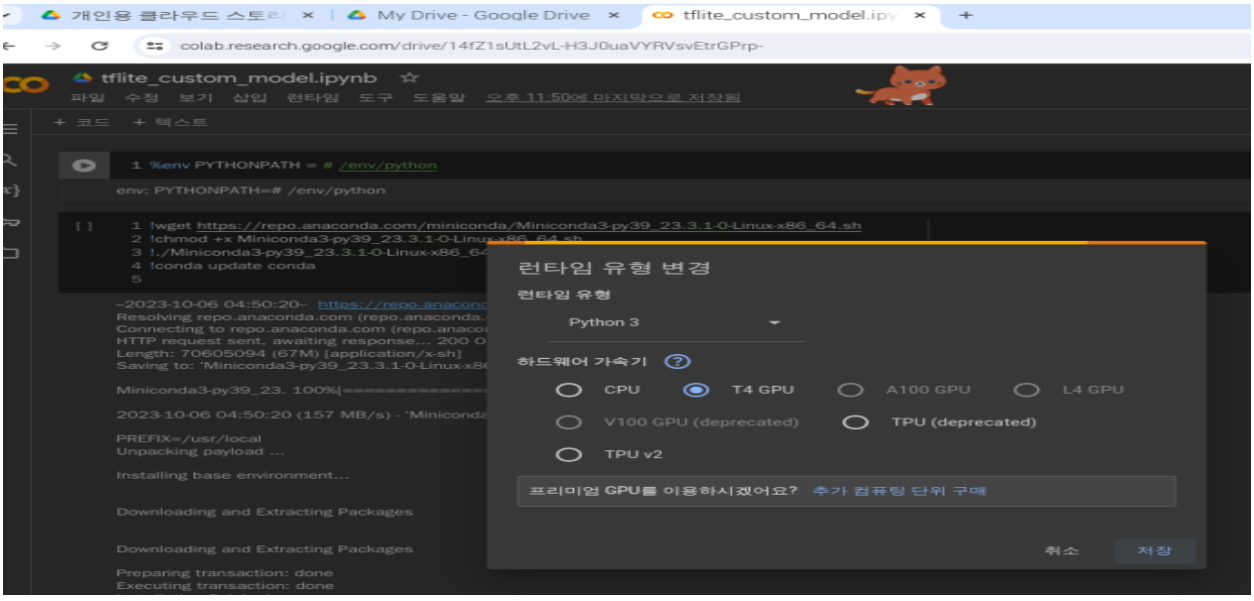


. Colab 에서 ‘노트열기’를 선택하고 ‘업로드’/’둘러보기’/Home 에서노프북 파일을 열기





. Colab 상단 메뉴에서 ‘런타임’을 눌러서 ‘런타임 유형 변경’을 눌러 뜨는 창에서 ‘T4 GPU’ 선택



(뜨는 창에서 ‘GPU 없이 연결’ 선택)

. cell 별로 실행

%env PYTHONPATH = # /env/python

```
[1]  1 %env PYTHONPATH = # /env/python

     env: PYTHONPATH=# /env/python
```

!wget https://repo.anaconda.com/miniconda/Miniconda3-py39_23.3.1-0-Linux-x86_64.sh
!chmod +x Miniconda3-py39_23.3.1-0-Linux-x86_64.sh
!./Miniconda3-py39_23.3.1-0-Linux-x86_64.sh -b -f -p /usr/local
!conda update conda

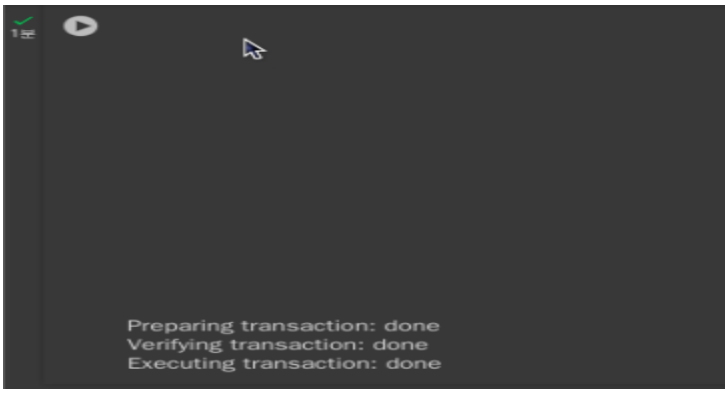
```
1 !wget https://repo.anaconda.com/miniconda/Miniconda3-py39_23.3.1-0-Linux-x86_64.sh
2 !chmod +x Miniconda3-py39_23.3.1-0-Linux-x86_64.sh
3 !./Miniconda3-py39_23.3.1-0-Linux-x86_64.sh -b -f -p /usr/local
4 !conda update conda
5

--2024-05-04 15:06:50--  https://repo.anaconda.com/miniconda/Miniconda3-py39_23.3.1-0-Linux-x86_64.sh
Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.32.241, 104.16.191.158, 2606:4700::6810:20f1, ...
Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.32.241|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 70605094 (67M) [application/x-sh]
Saving to: ‘Miniconda3-py39_23.3.1-0-Linux-x86_64.sh’

Miniconda3-py39_23. 100%[===================>]  67.33M  65.9MB/s    in 1.0s

2024-05-04 15:06:52 (65.9 MB/s) - ‘Miniconda3-py39_23.3.1-0-Linux-x86_64.sh’ saved [70605094/70605094]

PREFIX=/usr/local
Unpacking payload ...

Installing base environment...


Downloading and Extracting Packages
```

(중략. ‘y/n’를 물어 보면 ‘y’)

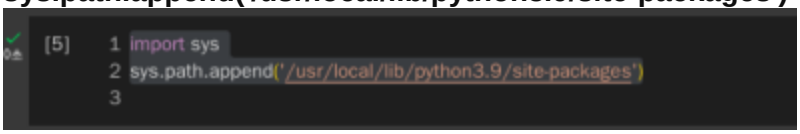
```
sqlite                3.41.1-h5eee18b_0 -> 3.45.3-h5eee18b_0
tqdm                  4.65.0-py39hb070fc8_0 -> 4.66.2-py39h2f386ee_0
tzdata                2023c-h04d1e81_0 -> 2024a-h04d1e81_0
urllib3               1.26.15-py39h06a4308_0 -> 2.1.0-py39h06a4308_1
xz                    5.2.10-h5eee18b_1 -> 5.4.6-h5eee18b_1
zlib                  1.2.13-h5eee18b_0 -> 1.2.13-h5eee18b_1
zstandard             0.19.0-py39h5eee18b_0 -> 0.22.0-py39h2c38b39_0

Proceed ([y]/n)? y
```

```
Proceed ([y]/n)? y

Downloading and Extracting Packages

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

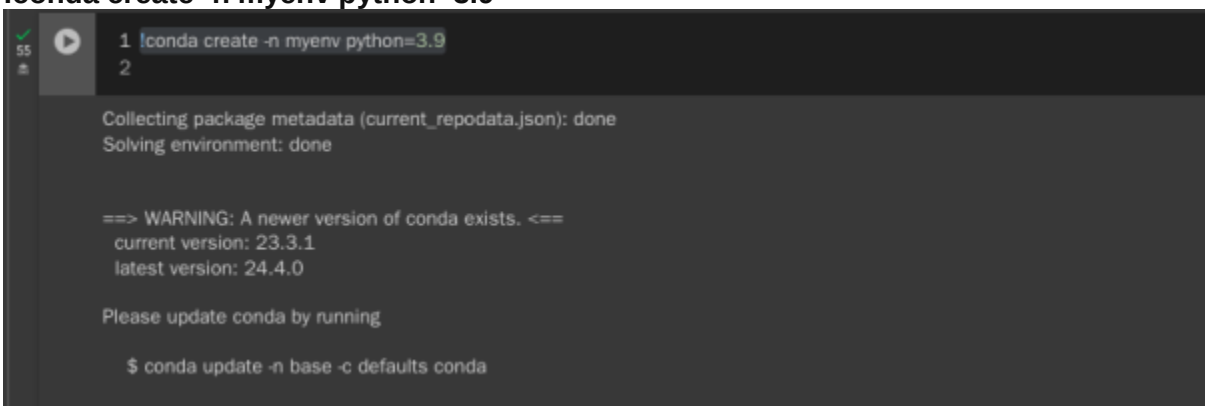
(중략)



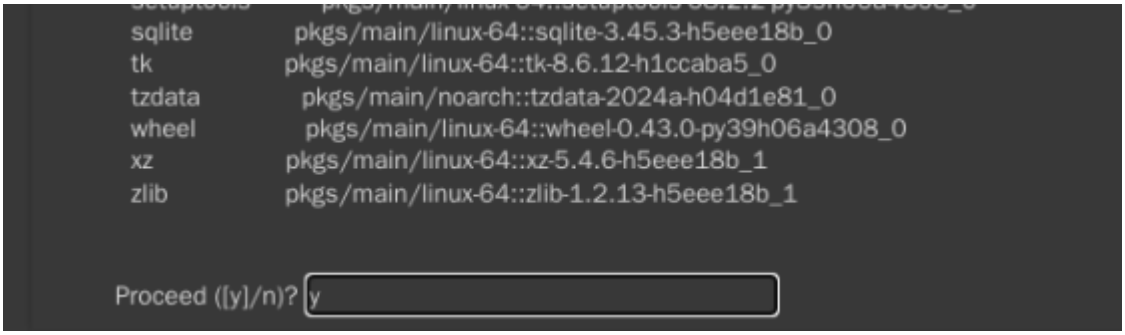
import sys
sys.path.append('/usr/local/lib/python3.9/site-packages')



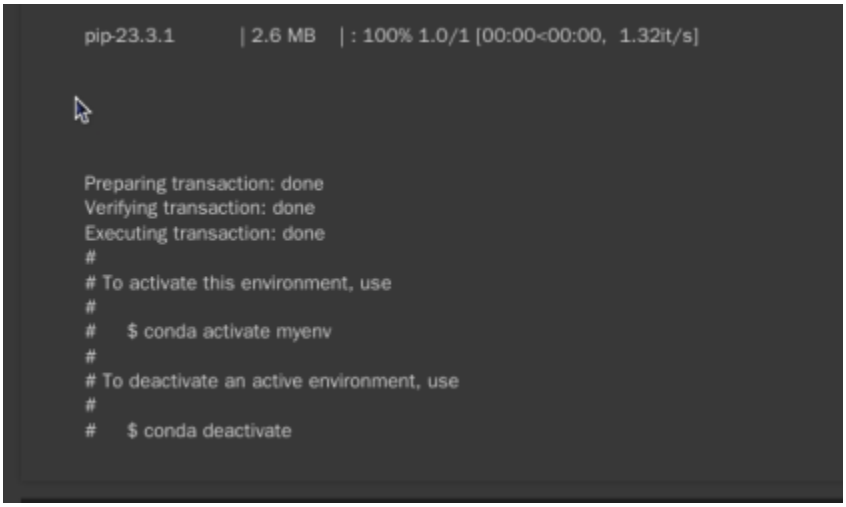
!conda create -n myenv python=3.9



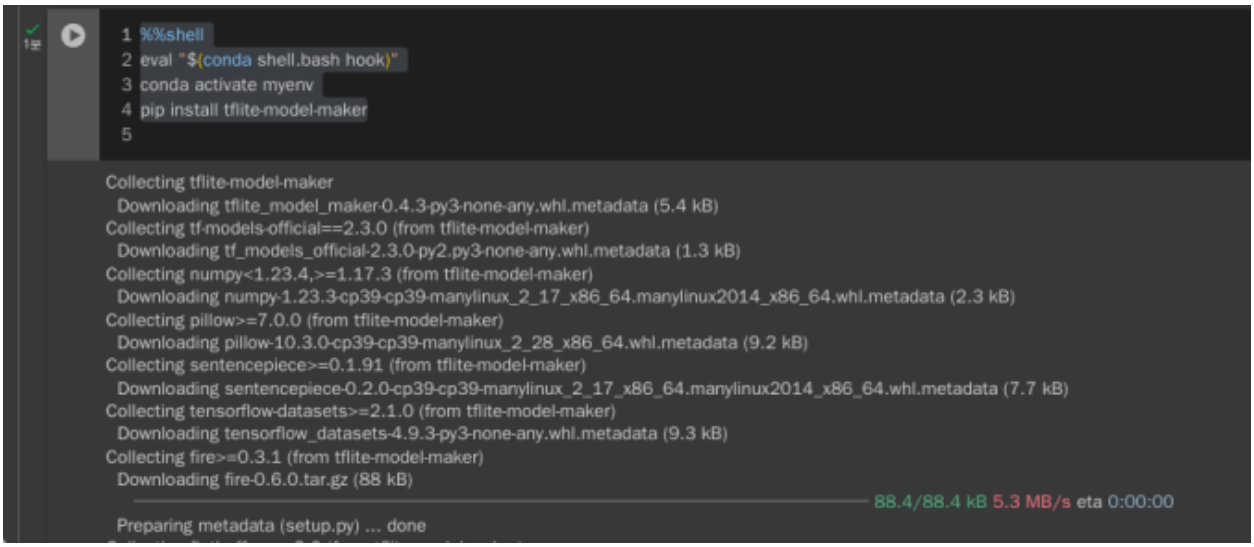
(중략. 'y/n' 물어보면 'y')



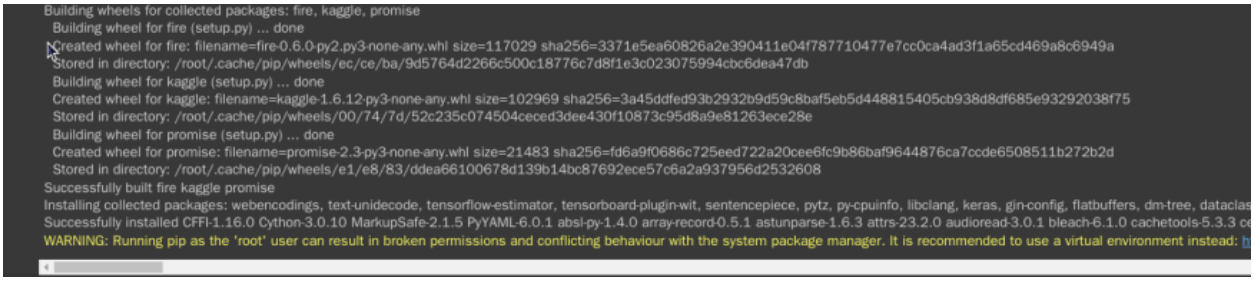
(중략)



%%shell
eval "$(conda shell.bash hook)"
conda activate myenv
pip install tflite-model-maker

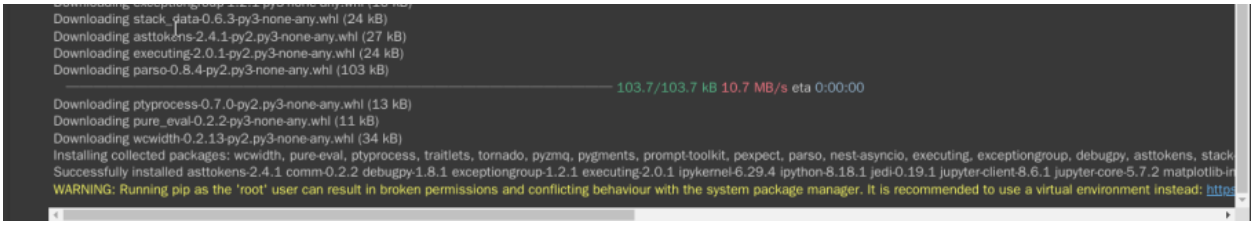


(중략)

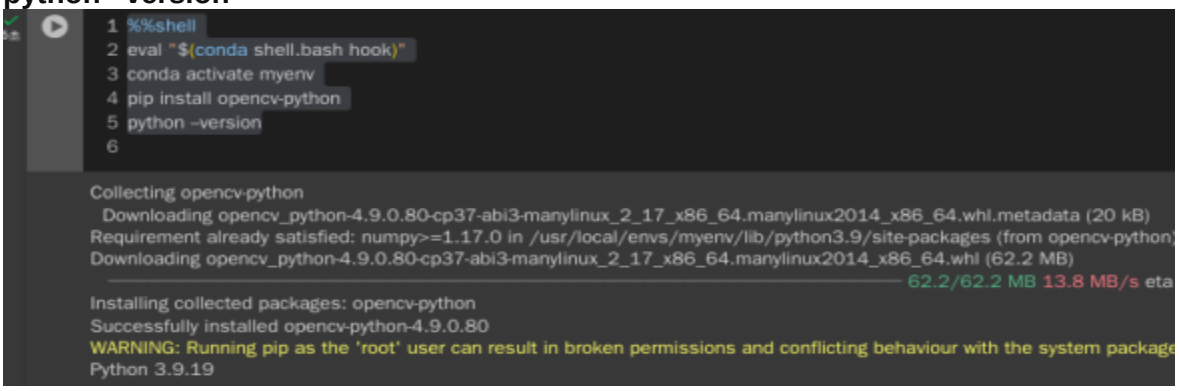


%%shell
eval "$(conda shell.bash hook)"
conda activate myenv
pip install ipykernel

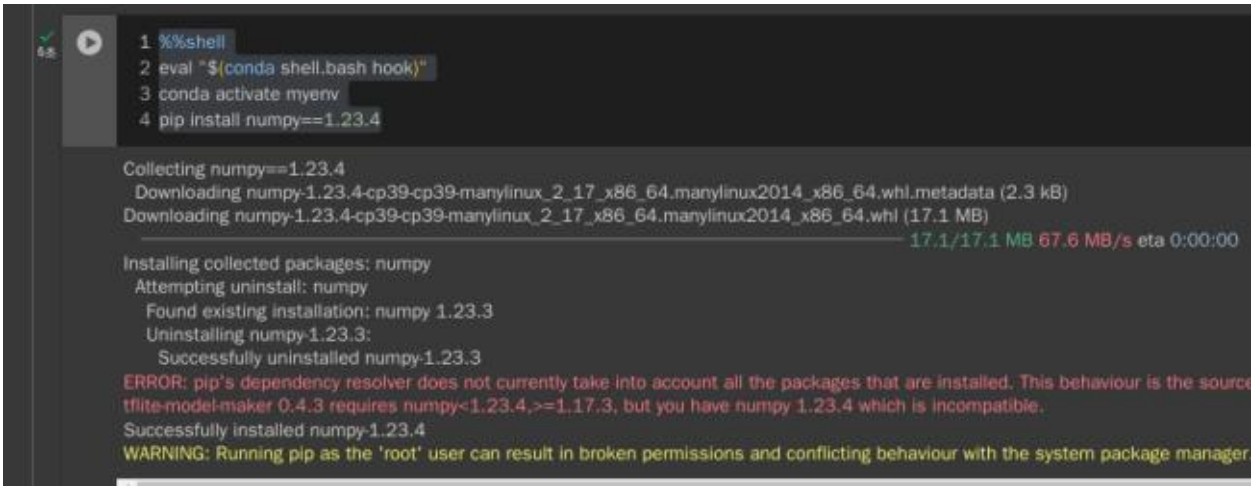
(중략)



%%shell
eval "$(conda shell.bash hook)"
conda activate myenv
pip install opencv-python
python –version

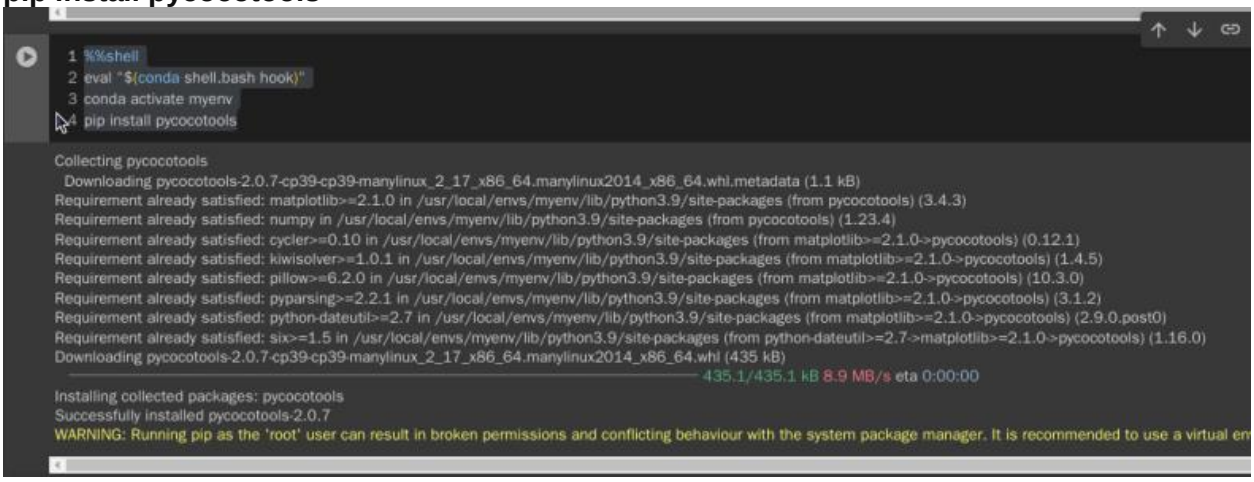


%%shell
eval "$(conda shell.bash hook)"
conda activate myenv
pip install numpy==1.23.4



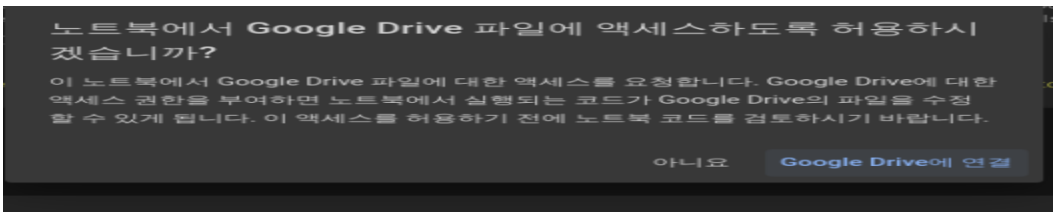
('ERROR'는 무시)

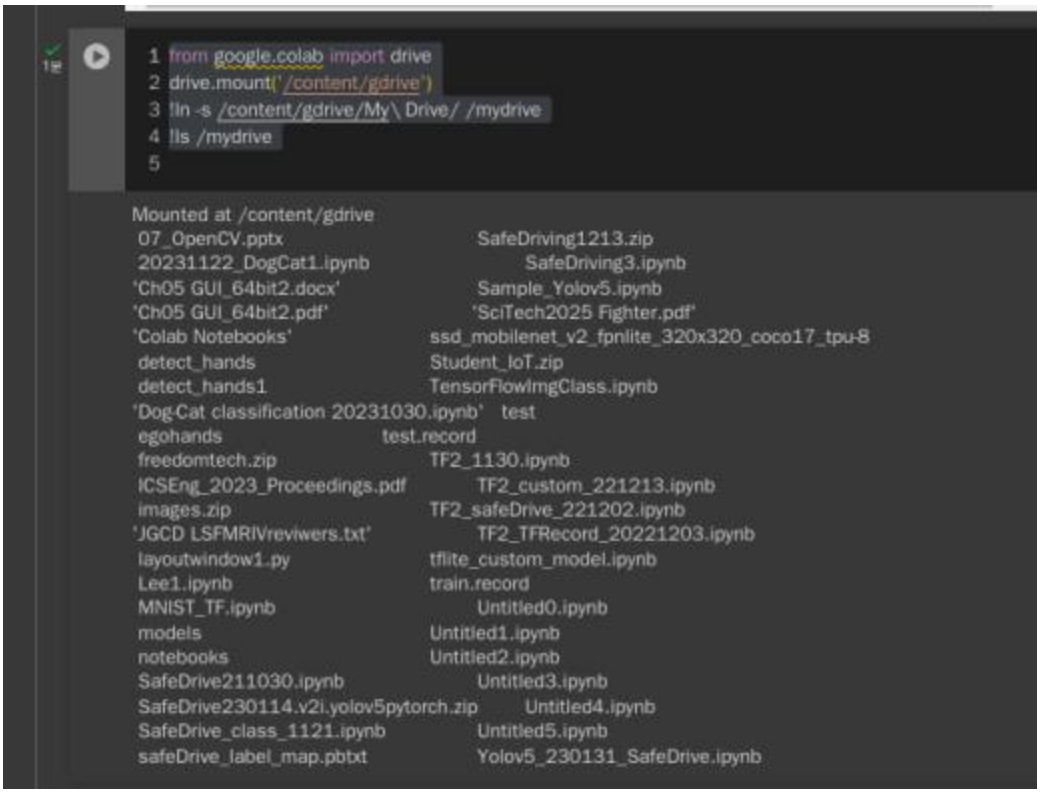
%%shell
eval "$(conda shell.bash hook)"
conda activate myenv
pip install pycocotools



from google.colab import drive
drive.mount('/content/gdrive')
!ln -s /content/gdrive/My\ Drive/ /mydrive
!ls /mydrive

. 승인 필요 (뜨는 창에서 '구글드라이브 연결' 선택 후, 내 구글 계정을 선택하고 계속 'continue')

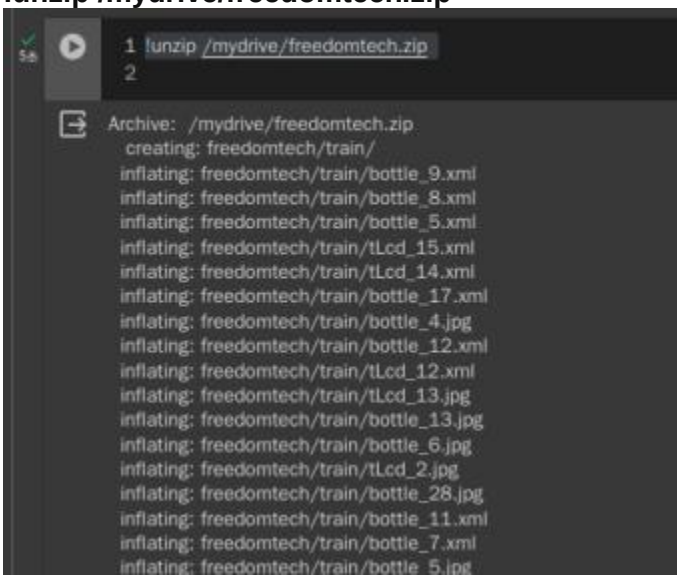




```
1 from google.colab import drive
2 drive.mount('/content/gdrive')
3 !ln -s /content/gdrive/My\ Drive/ /mydrive
4 !ls /mydrive
5
```

```
Mounted at /content/gdrive
07_OpenCV.pptx                          SafeDriving1213.zip
20231122_DogCat1.ipynb                  SafeDriving3.ipynb
'Ch05 GUI_64bit2.docx'                  Sample_Yolov5.ipynb
'Ch05 GUI_64bit2.pdf'                   'SciTech2025 Fighter.pdf'
'Colab Notebooks'                       ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8
detect_hands                            Student_IoT.zip
detect_hands1                           TensorFlowImgClass.ipynb
'Dog-Cat classification 20231030.ipynb'  test
egohands                                test.record
freedomtech.zip                         TF2_1130.ipynb
ICSEng_2023_Proceedings.pdf             TF2_custom_221213.ipynb
images.zip                              TF2_safeDrive_221202.ipynb
'JGCD LSFMRIVreviwers.txt'              TF2_TFRecord_20221203.ipynb
layoutwindow1.py                        tflite_custom_model.ipynb
Lee1.ipynb                              train.record
MNIST_TF.ipynb                          Untitled0.ipynb
models                                  Untitled1.ipynb
notebooks                               Untitled2.ipynb
SafeDrive211030.ipynb                   Untitled3.ipynb
SafeDrive230114.v2i.yolov5pytorch.zip   Untitled4.ipynb
SafeDrive_class_1121.ipynb              Untitled5.ipynb
safeDrive_label_map.pbtxt               Yolov5_230131_SafeDrive.ipynb
```

!unzip /mydrive/freedomtech.zip

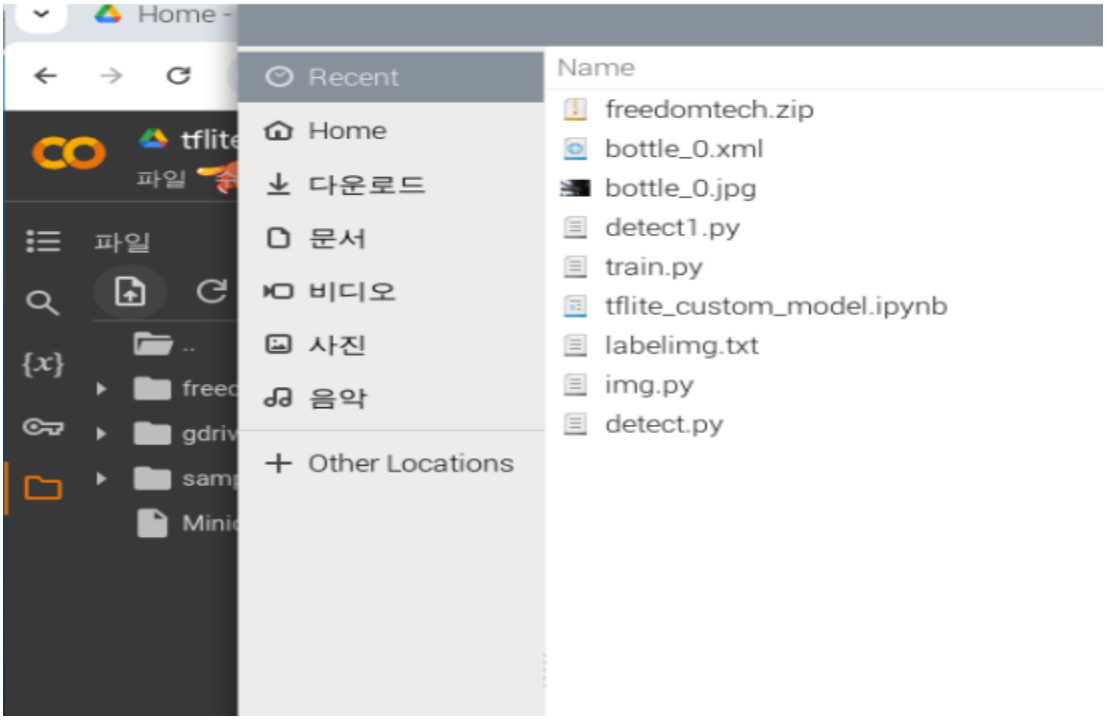


```
1 !unzip /mydrive/freedomtech.zip
2
```

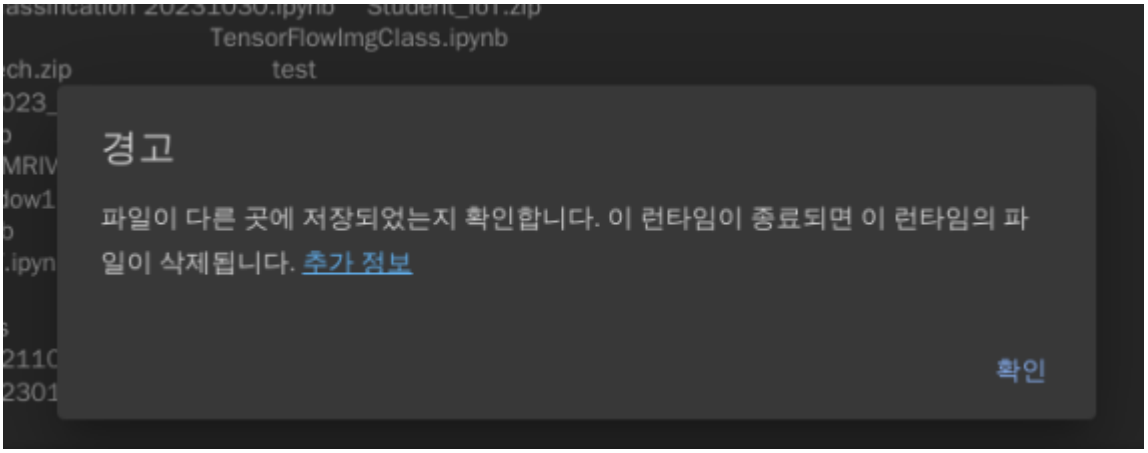
```
Archive:  /mydrive/freedomtech.zip
   creating: freedomtech/train/
  inflating: freedomtech/train/bottle_9.xml
  inflating: freedomtech/train/bottle_8.xml
  inflating: freedomtech/train/bottle_5.xml
  inflating: freedomtech/train/tLcd_15.xml
  inflating: freedomtech/train/tLcd_14.xml
  inflating: freedomtech/train/bottle_17.xml
  inflating: freedomtech/train/bottle_4.jpg
  inflating: freedomtech/train/bottle_12.xml
  inflating: freedomtech/train/tLcd_12.xml
  inflating: freedomtech/train/tLcd_13.jpg
  inflating: freedomtech/train/bottle_13.jpg
  inflating: freedomtech/train/bottle_6.jpg
  inflating: freedomtech/train/tLcd_2.jpg
  inflating: freedomtech/train/bottle_28.jpg
  inflating: freedomtech/train/bottle_11.xml
  inflating: freedomtech/train/bottle_7.xml
  inflating: freedomtech/train/bottle_5.jpg
```

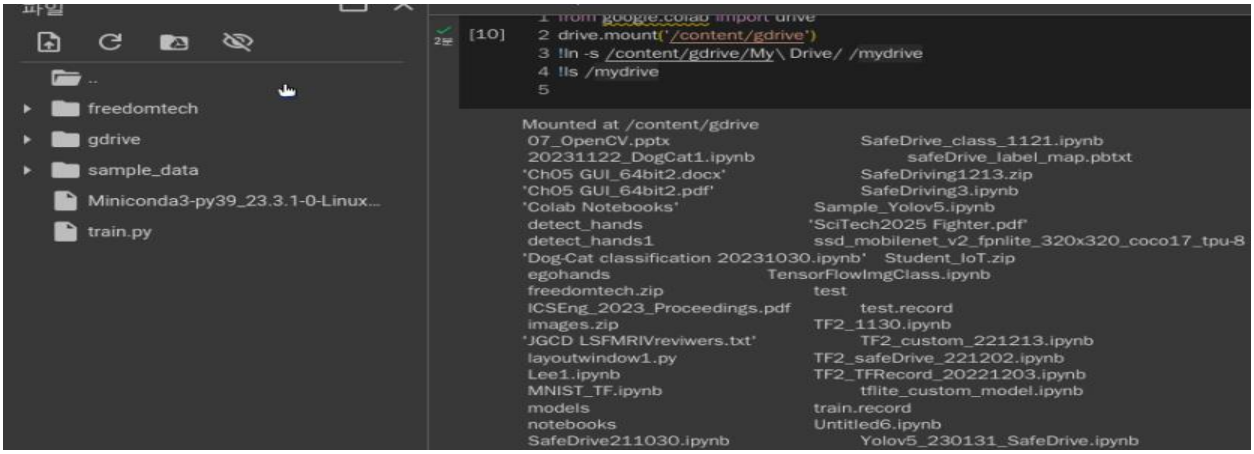
(중략. 좌측 폴더에 ‘freedomtech’ 만들어짐)

. 좌측 노란 폴더를 클릭(‘/content’가 최상위 폴더임)하고 상단의 위쪽 화살표(파일열기)를 클릭하여 train.py 를 선택하고 하단의 ‘open’를 클릭하여 train.py 를 읽어 들임

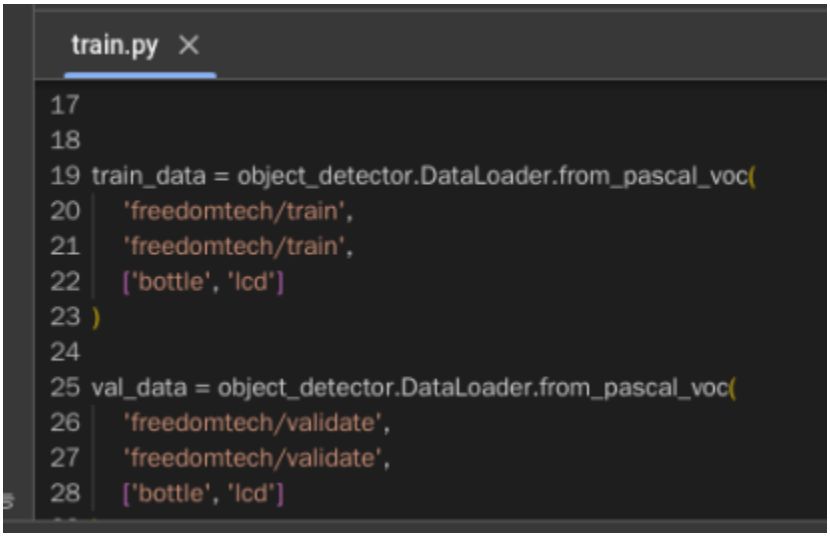


(뜨는 경고창은 '확인' 누르고 무시)





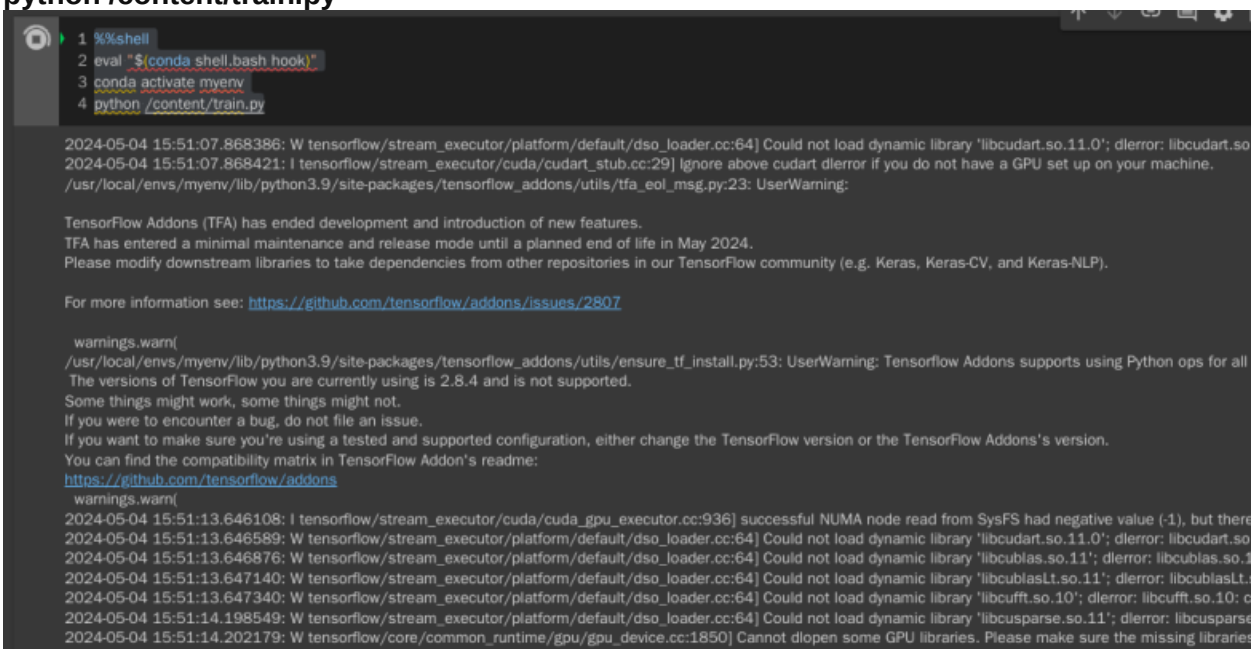
. train.py 를 클릭해서 class 이름 변경 (train_data 및 val_data 에서 'bottle', 'lcd'로 변경)



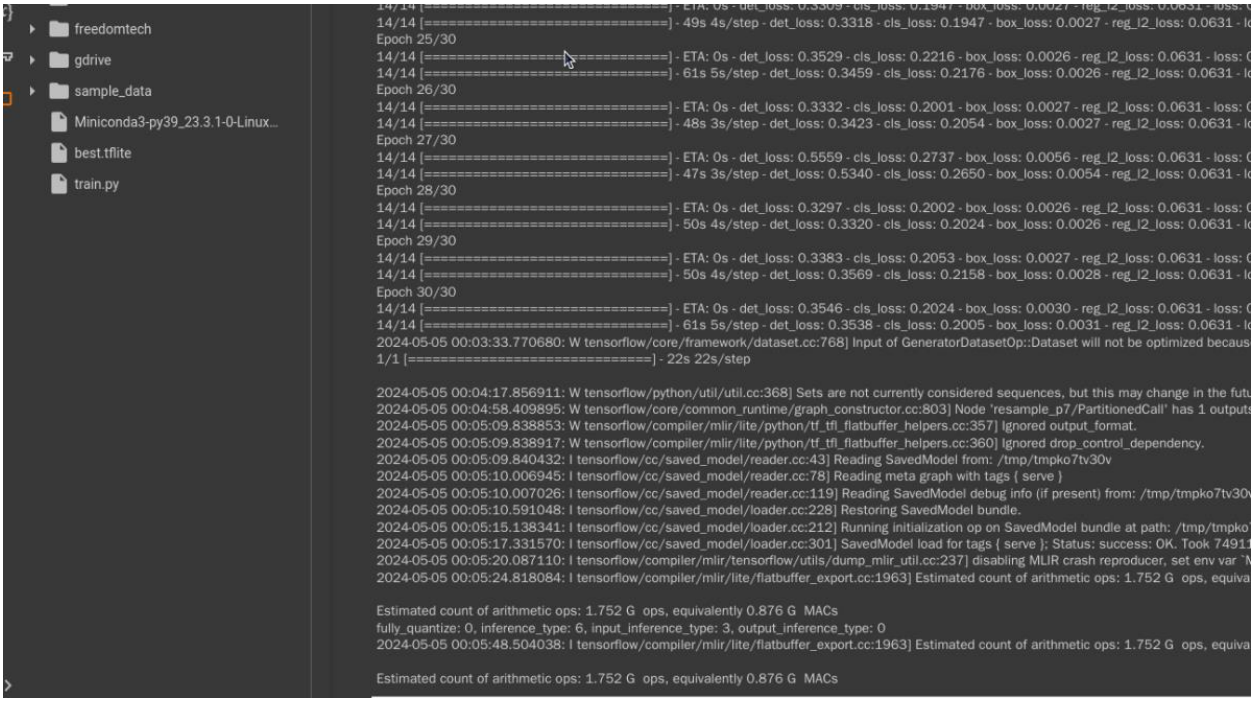
epochs=30 (개인적으로 실행 시 80 으로 변경. 현재는 30 을 유지. 100 으로 하면 90 에서 중단할 수 있음)으로 변경후 저장확인(파일이름 좌측 상단의 '*'기호가 사라지면(조금 기다리면) 저장된 것임) 후 'x'하여 닫음.

. 학습

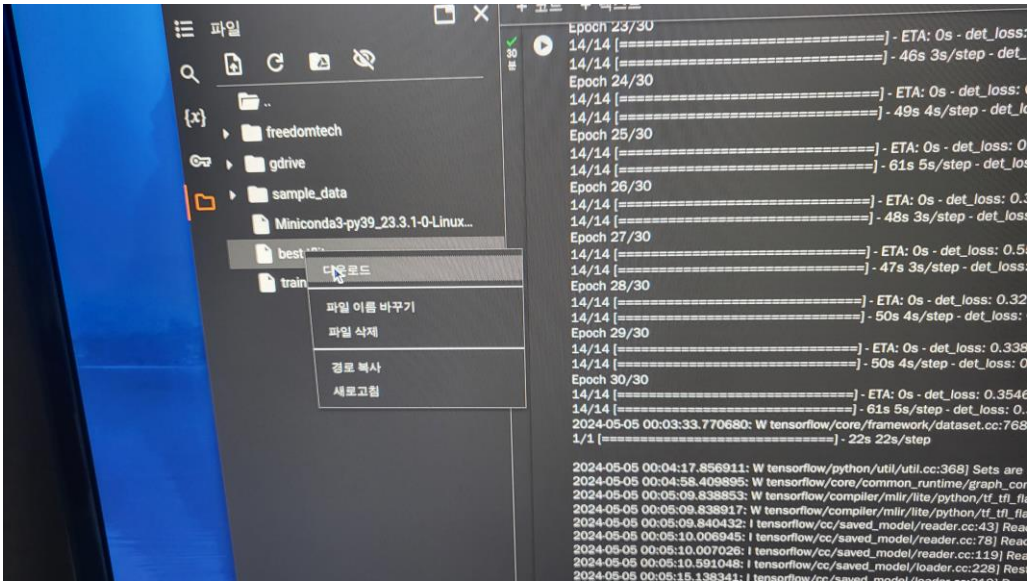
```
%%shell
eval "$(conda shell.bash hook)"
conda activate myenv
python /content/train.py
```

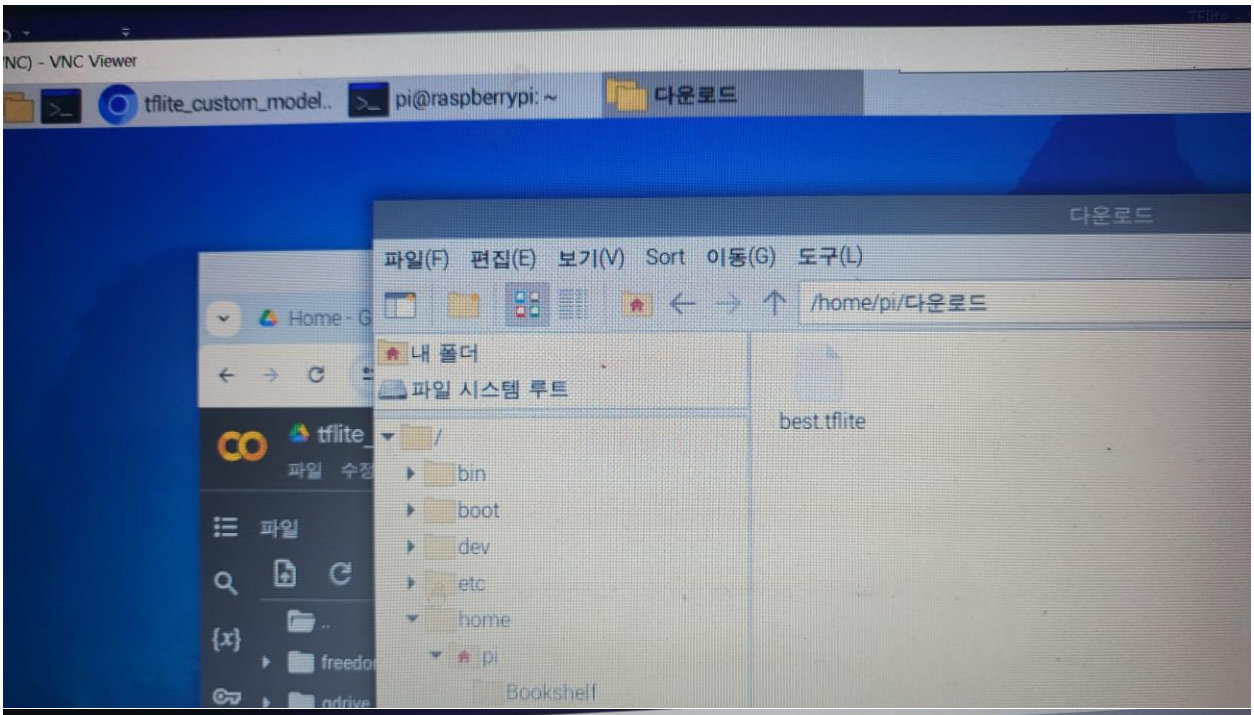


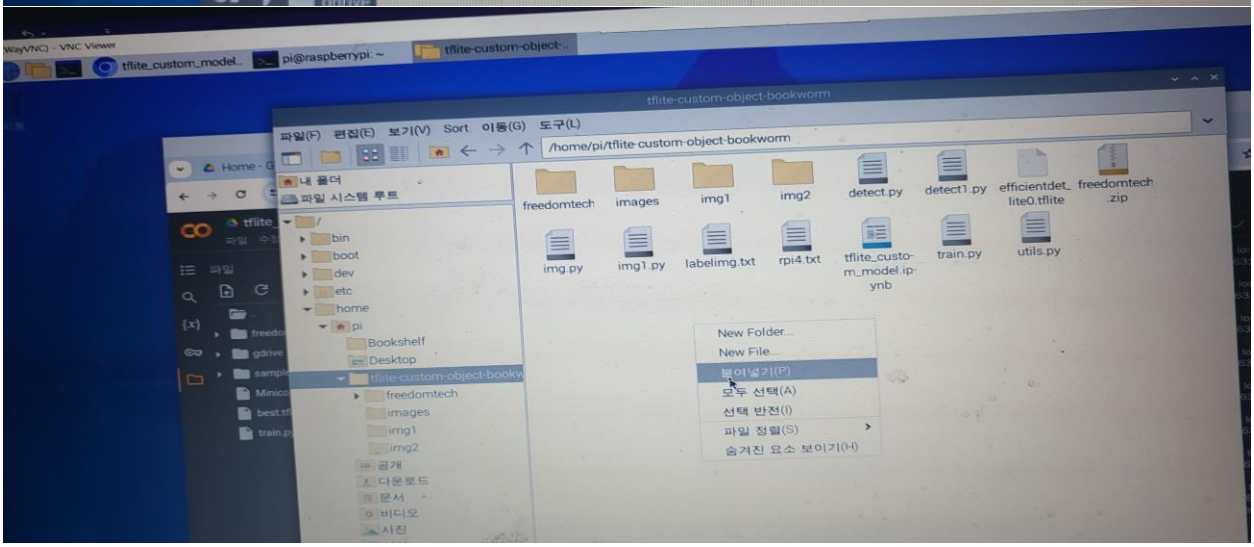
(중략)



. 좌측에 'best.tflite' 가 만들어지면 우측 버튼 '다운로드'하여 home/downloads 로 옮긴 후에 다시 repository 로 옮기고 (update), detect.py, detect1.py 에서 이들로 수정한 후 detect 실행.







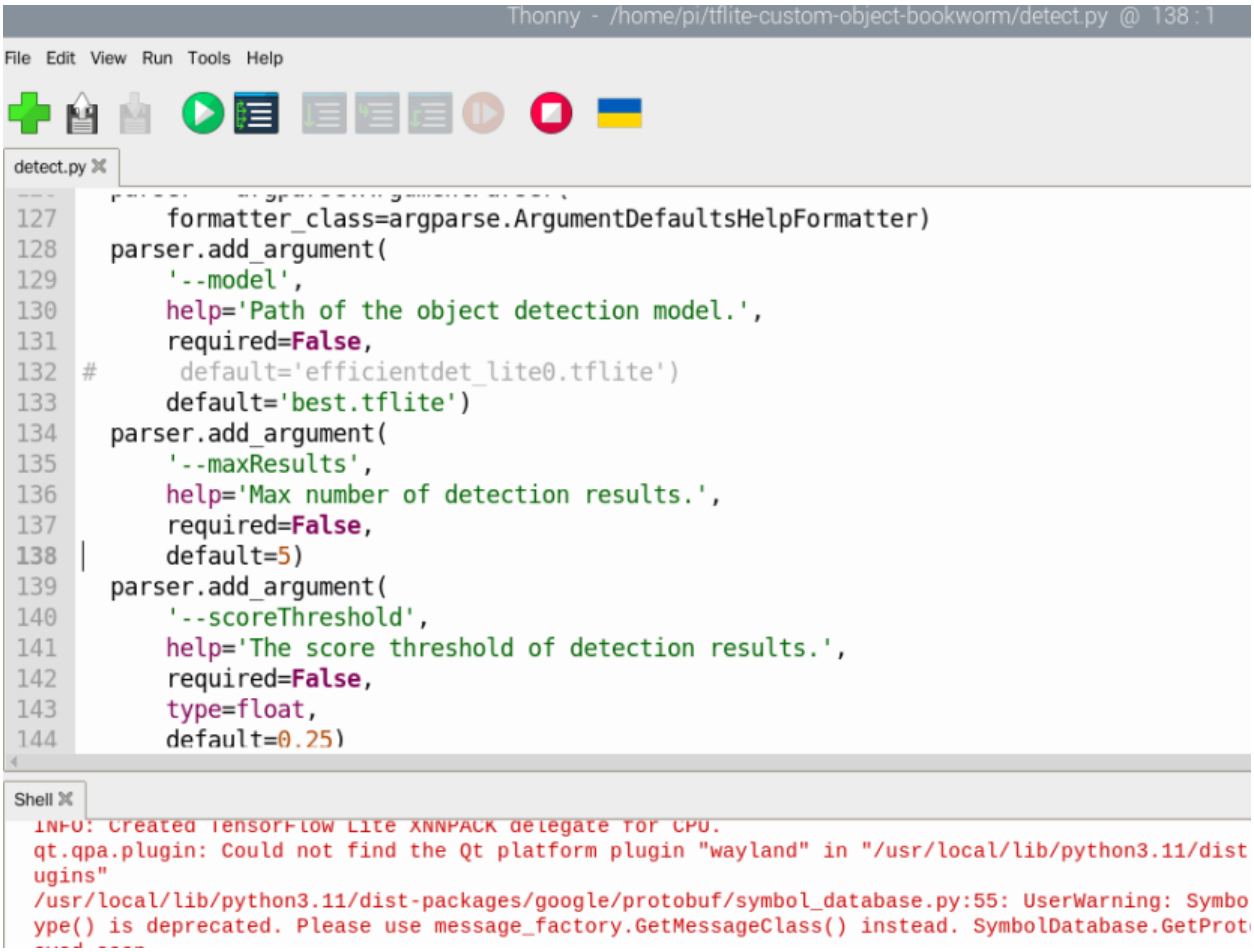
. repository 에 있는 detect.py 를 Thonny 로 염

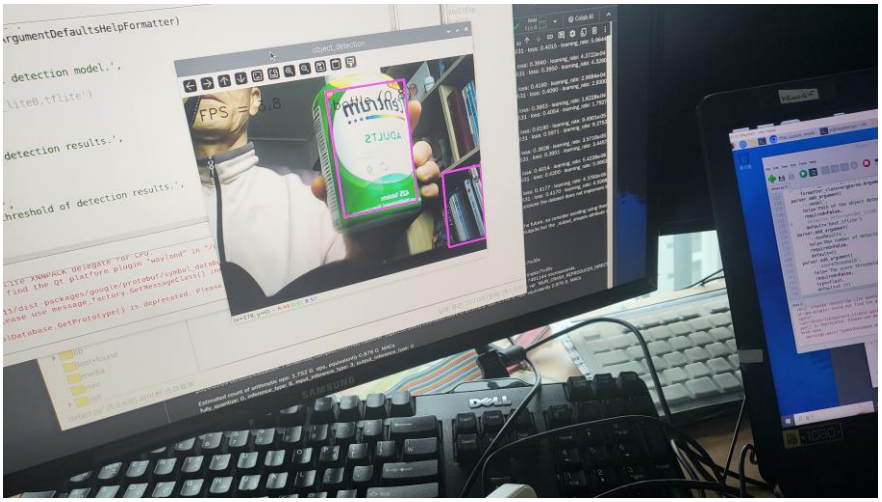
.

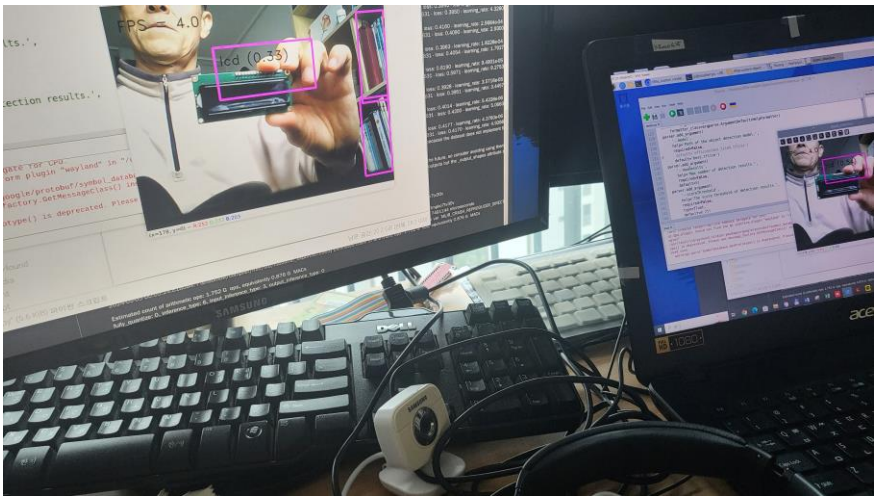
. 파일 수정 (#132 는 comment, #133 은 uncomment)

```
130        help='Path of the object detection model.',
131        required=False,
132  #      default='efficientdet_lite0.tflite')
133        default='best.tflite')
134    parser.add_argument(
```

. 실행 (상단 녹색의 우측 화살표 클릭)




object_detection
FPS = 3.8


FPS = 4.0
Hcd (0.33)