무선 센서네트워크
11주차
(라즈베리파이기반 실습-2)

인하공업전문대학 컴퓨터 정보과

김한결 강사

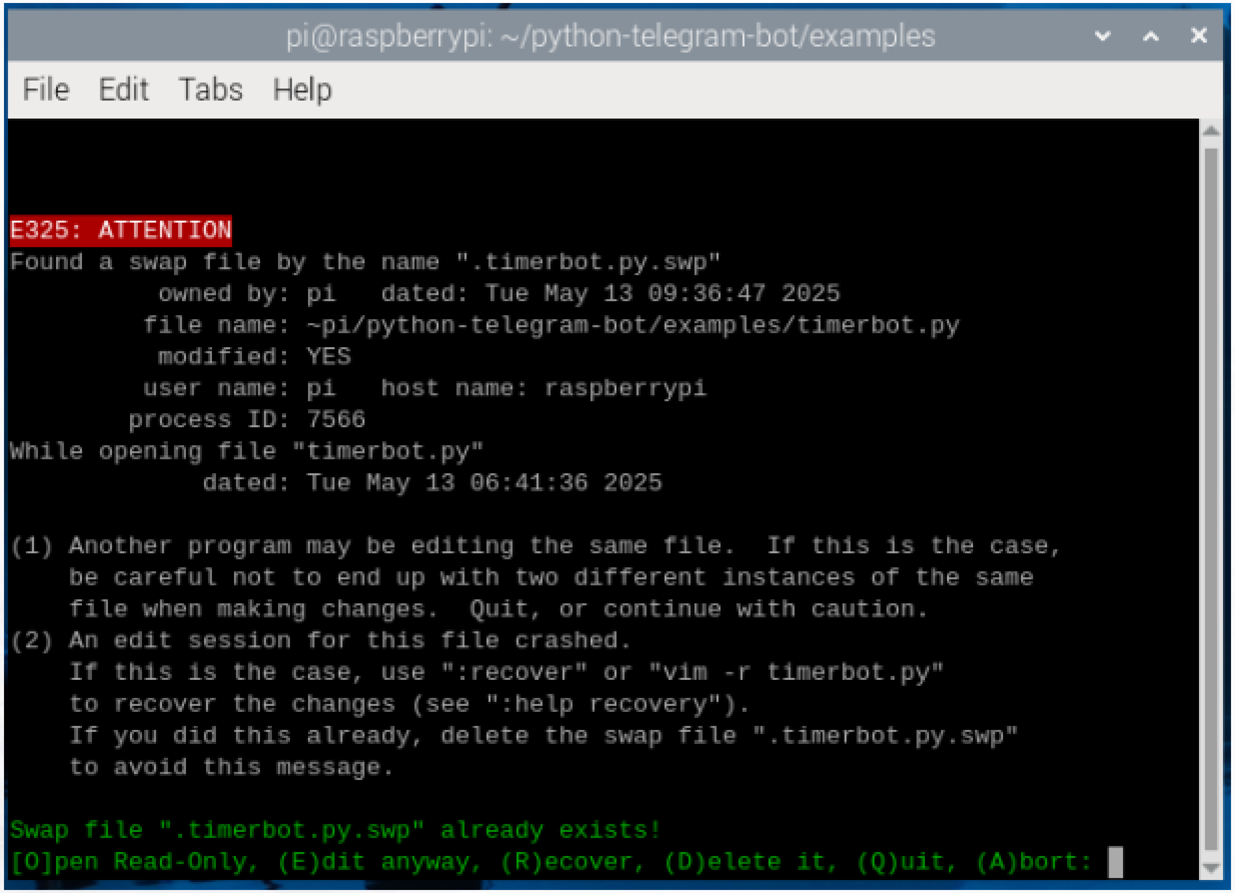
목차

- 라즈베리파이 OpenCV && USB Camera &&Telegram
  1. Python 라이브러리 OpenCV 설치
  2. USB Camera 테스트 (Python Code)
  3. Telegram 설치 및 이미지 전송

Python 라이브러리 OpenCV 설치

Cygwin vim swp 경고

- E 키 입력

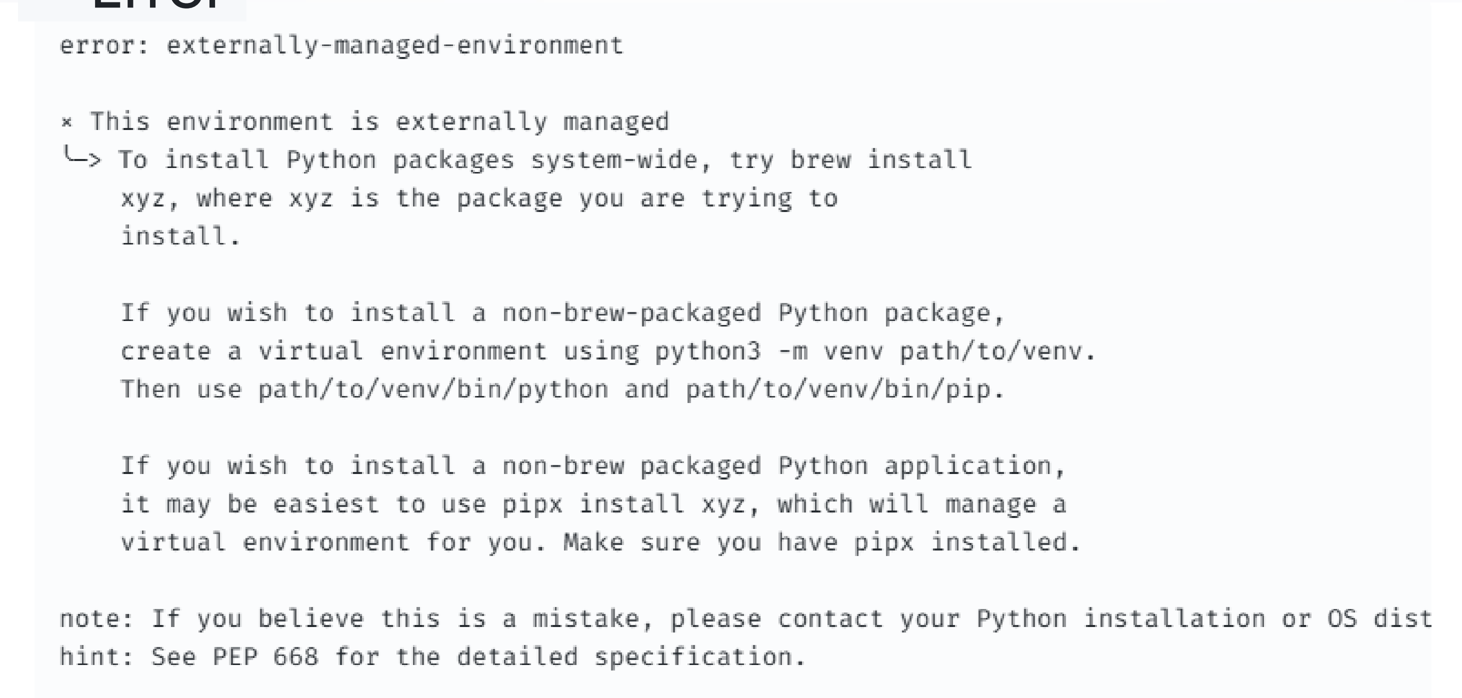


```
E325: ATTENTION
Found a swap file by the name ".timerbot.py.swp"
          owned by: pi   dated: Tue May 13 09:36:47 2025
         file name: ~pi/python-telegram-bot/examples/timerbot.py
          modified: YES
         user name: pi   host name: raspberrypi
        process ID: 7566
While opening file "timerbot.py"
             dated: Tue May 13 06:41:36 2025

(1) Another program may be editing the same file.  If this is the case,
    be careful not to end up with two different instances of the same
    file when making changes.  Quit, or continue with caution.
(2) An edit session for this file crashed.
    If this is the case, use ":recover" or "vim -r timerbot.py"
    to recover the changes (see ":help recovery").
    If you did this already, delete the swap file ".timerbot.py.swp"
    to avoid this message.

Swap file ".timerbot.py.swp" already exists!
[O]pen Read-Only, (E)dit anyway, (R)ecover, (D)elete it, (Q)uit, (A)bort:
```

Python 라이브러리 OpenCV 설치

pip 설치 전 확인 사항

- Error



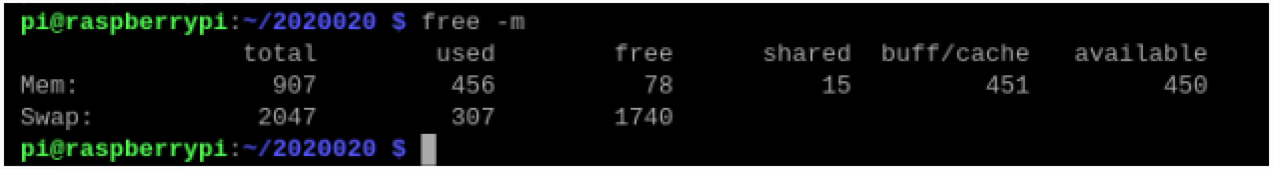
```
error: externally-managed-environment

× This environment is externally managed
╰─> To install Python packages system-wide, try brew install
    xyz, where xyz is the package you are trying to
    install.

    If you wish to install a non-brew-packaged Python package,
    create a virtual environment using python3 -m venv path/to/venv.
    Then use path/to/venv/bin/python and path/to/venv/bin/pip.

    If you wish to install a non-brew packaged Python application,
    it may be easiest to use pipx install xyz, which will manage a
    virtual environment for you. Make sure you have pipx installed.

note: If you believe this is a mistake, please contact your Python installation or OS dist
hint: See PEP 668 for the detailed specification.
```

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

Python 라이브러리 OpenCV 설치

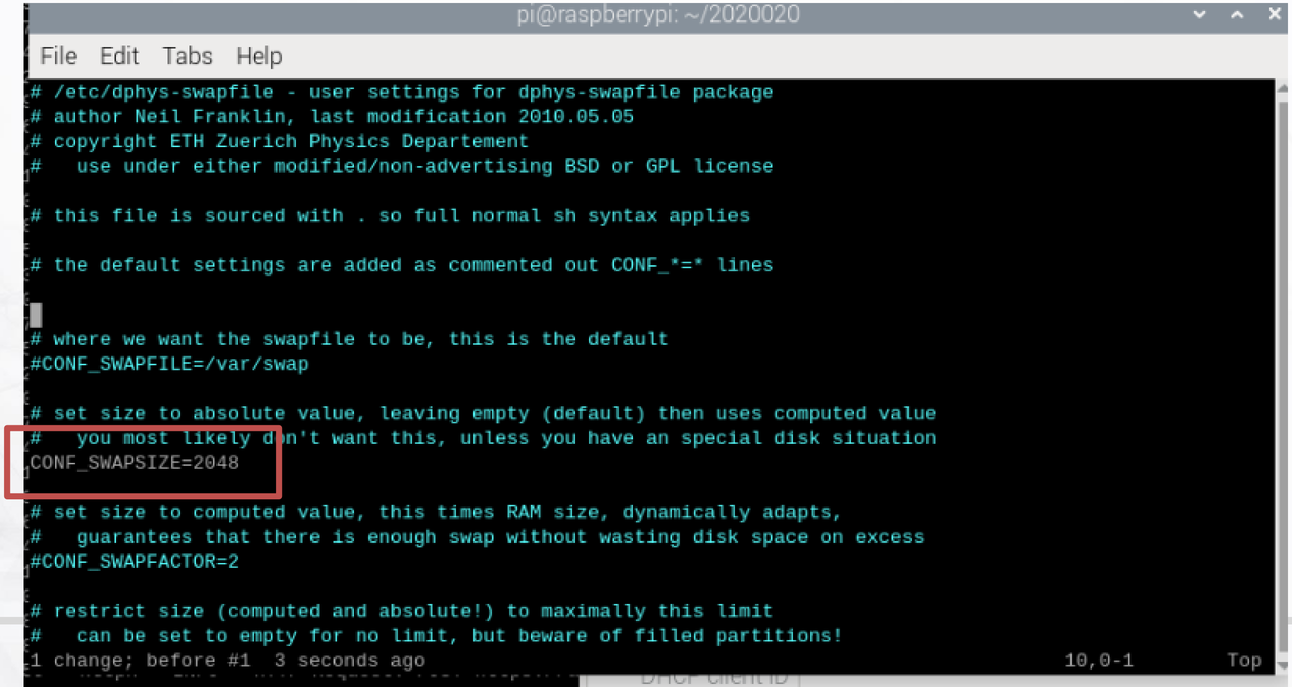
pip OpenCV 설치 전 확인 사항

- swap 메모리 확인
  - ✓ free –m



```
pi@raspberrypi:~/2020020 $ free -m
              total        used        free      shared  buff/cache   available
Mem:            907         456          78          15         451         450
Swap:          2047         307        1740
pi@raspberrypi:~/2020020 $ 
```

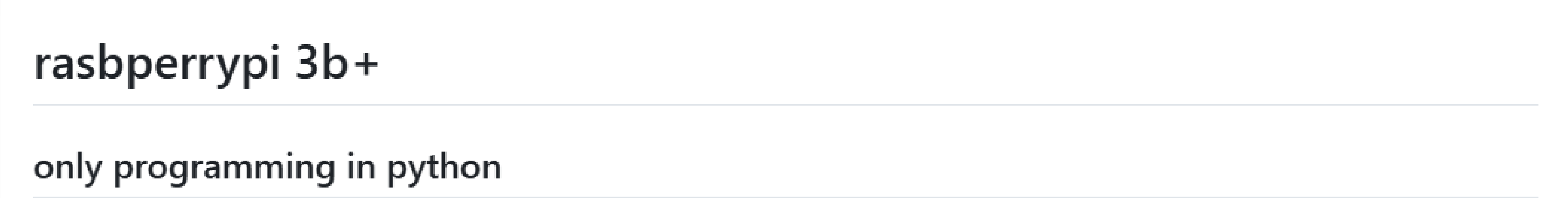
$ sudo vim /etc/dphys-swapfile
 - CONF_SWAPSIZE = 200 -> 2048로 변경



```
# /etc/dphys-swapfile - user settings for dphys-swapfile package
# author Neil Franklin, last modification 2010.05.05
# copyright ETH Zuerich Physics Departement
#   use under either modified/non-advertising BSD or GPL license

# this file is sourced with . so full normal sh syntax applies

# the default settings are added as commented out CONF_*=* lines


# where we want the swapfile to be, this is the default
#CONF_SWAPFILE=/var/swap

# set size to absolute value, leaving empty (default) then uses computed value
#   you most likely don't want this, unless you have an special disk situation
CONF_SWAPSIZE=2048

# set size to computed value, this times RAM size, dynamically adapts,
#   guarantees that there is enough swap without wasting disk space on excess
#CONF_SWAPFACTOR=2

# restrict size (computed and absolute!) to maximally this limit
#   can be set to empty for no limit, but beware of filled partitions!
1 change; before #1  3 seconds ago                            10,0-1        Top
```

Python 라이브러리 OpenCV 설치

pip OpenCV 설치



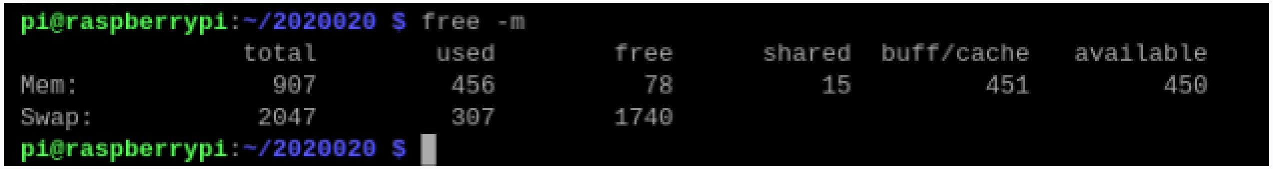
rasbperrypi 3b+

only programming in python

```
$ pip install opencv-contrib-python
$ sudo apt-get install python3-opencv
```

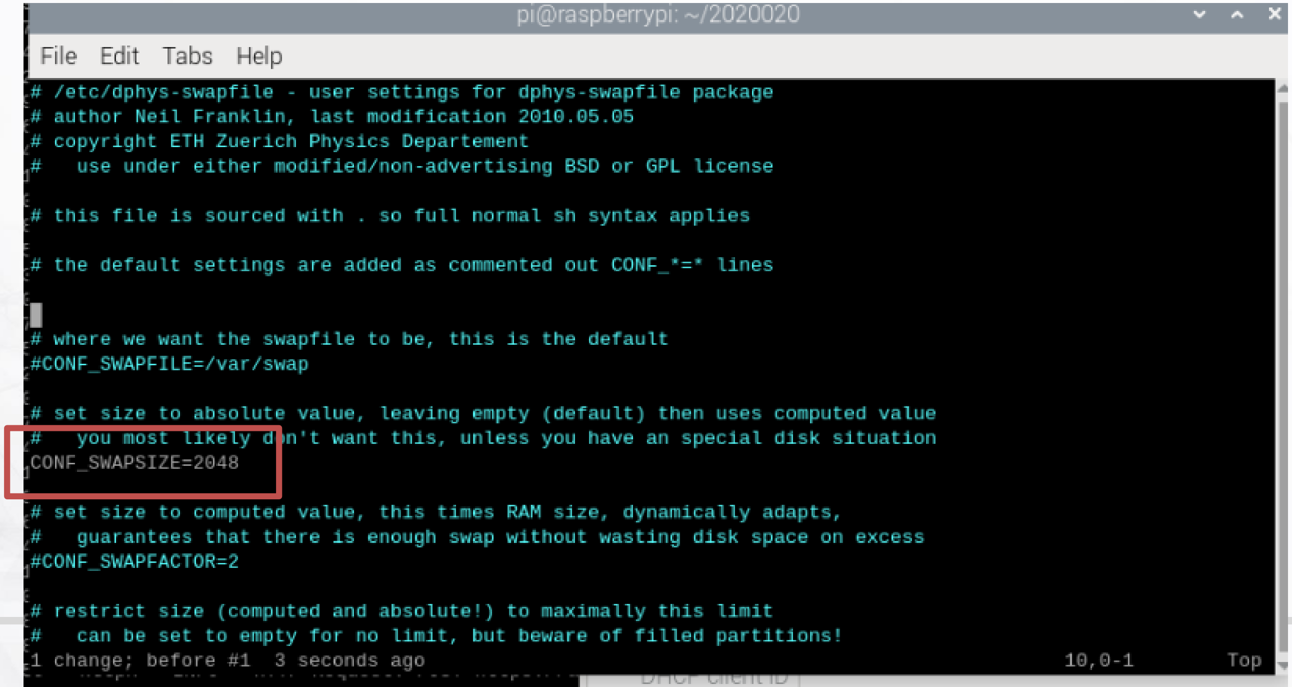
pip OpenCV 설치 후 변경 사항

- swap 메모리 확인
  - ✓ free –m



```
pi@raspberrypi:~/2020020 $ free -m
              total        used        free      shared  buff/cache   available
Mem:            907         456          78          15         451         450
Swap:          2047         307        1740
pi@raspberrypi:~/2020020 $ 
```

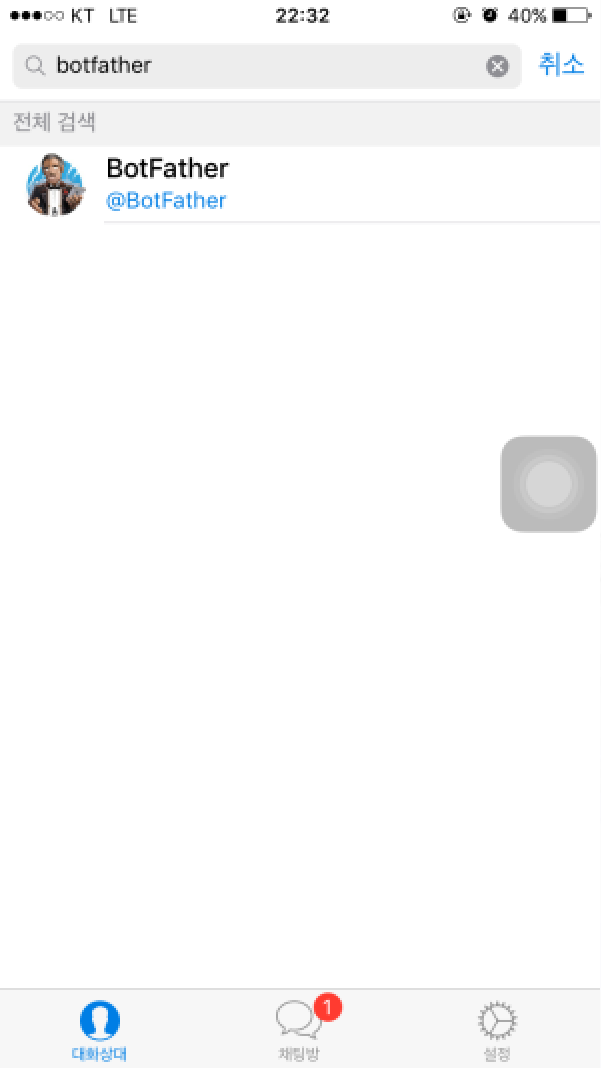
$ sudo vim /etc/dphys-swapfile
 - CONF_SWAPSIZE = 2048 -> 200로 변경



```
# /etc/dphys-swapfile - user settings for dphys-swapfile package
# author Neil Franklin, last modification 2010.05.05
# copyright ETH Zuerich Physics Departement
#   use under either modified/non-advertising BSD or GPL license

# this file is sourced with . so full normal sh syntax applies

# the default settings are added as commented out CONF_*=* lines


# where we want the swapfile to be, this is the default
#CONF_SWAPFILE=/var/swap

# set size to absolute value, leaving empty (default) then uses computed value
#   you most likely don't want this, unless you have an special disk situation
CONF_SWAPSIZE=2048

# set size to computed value, this times RAM size, dynamically adapts,
#   guarantees that there is enough swap without wasting disk space on excess
#CONF_SWAPFACTOR=2

# restrict size (computed and absolute!) to maximally this limit
#   can be set to empty for no limit, but beware of filled partitions!
1 change; before #1  3 seconds ago                                10,0-1        Top
```

라즈베리파이 + Telegram 연동 하기

- 텔레그램 봇 검색
  - botfather



●●●○○ KT LTE 22:32 40%

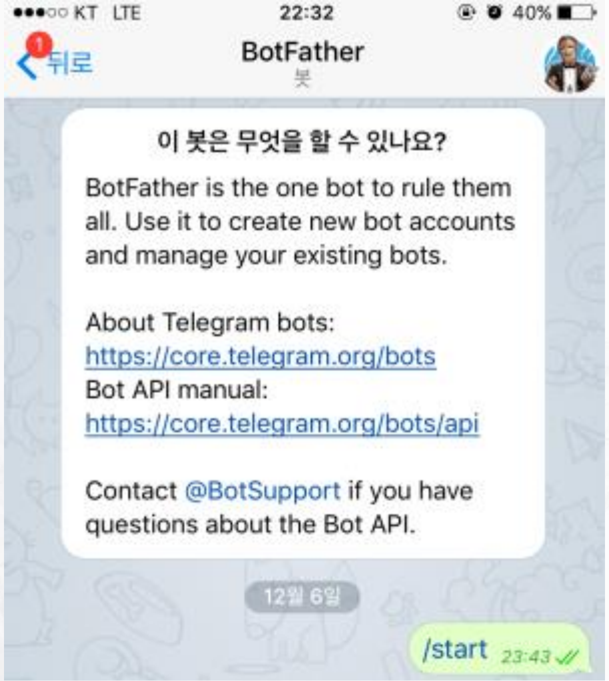
botfather 취소

전체 검색

BotFather
@BotFather

라즈베리파이 + Telegram 연동 하기

- 텔레그램 봇 만들기
  - /start



이 봇은 무엇을 할 수 있나요?

BotFather is the one bot to rule them all. Use it to create new bot accounts and manage your existing bots.

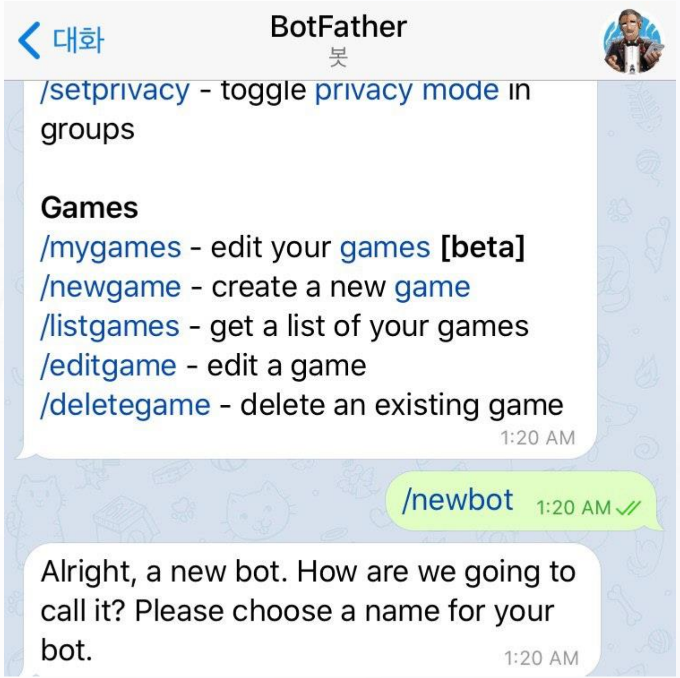
About Telegram bots:
https://core.telegram.org/bots
Bot API manual:
https://core.telegram.org/bots/api

Contact @BotSupport if you have questions about the Bot API.

12월 6일

/start 23:43

- 텔레그램 봇 만들기
  - /newbot



/setprivacy - toggle privacy mode in groups

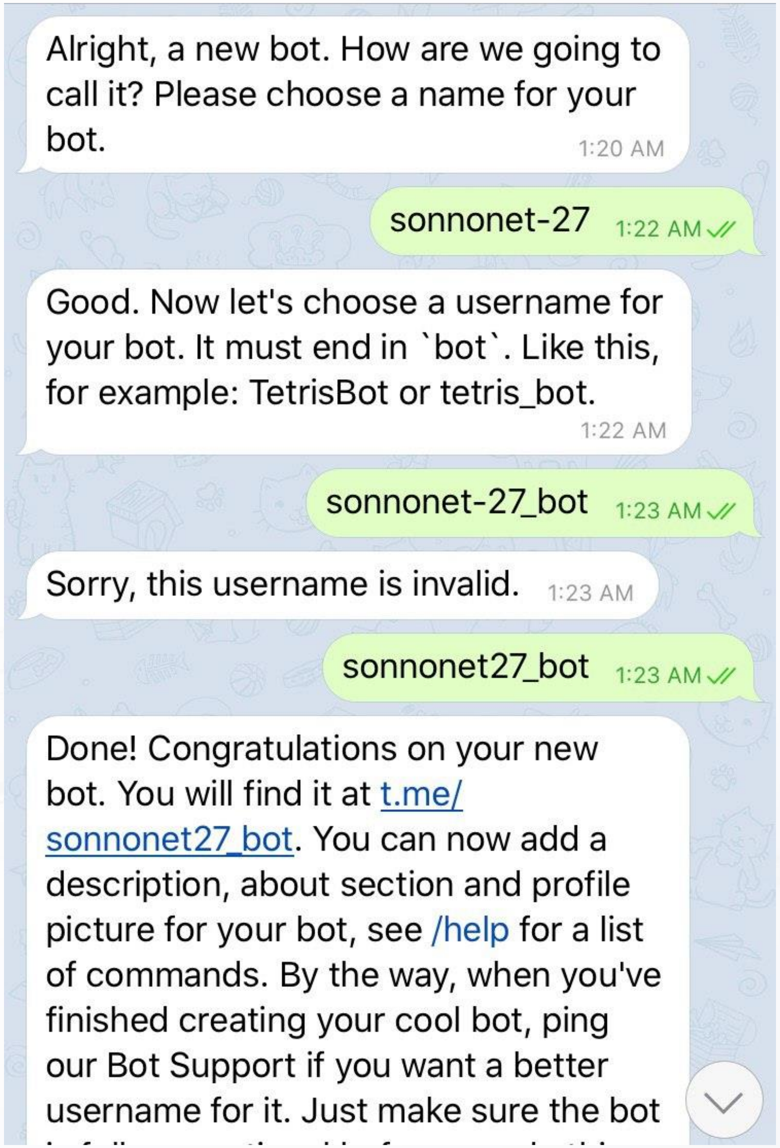
Games
/mygames - edit your games [beta]
/newgame - create a new game
/listgames - get a list of your games
/editgame - edit a game
/deletegame - delete an existing game
1:20 AM

/newbot 1:20 AM

Alright, a new bot. How are we going to call it? Please choose a name for your bot. 1:20 AM

라즈베리파이 + Telegram 연동 하기

- 텔레그램 봇 이름 짓기
  - <원하는 이름>_bot



Alright, a new bot. How are we going to call it? Please choose a name for your bot. 1:20 AM

sonnonet-27 1:22 AM

Good. Now let's choose a username for your bot. It must end in `bot`. Like this, for example: TetrisBot or tetris_bot. 1:22 AM

sonnonet-27_bot 1:23 AM

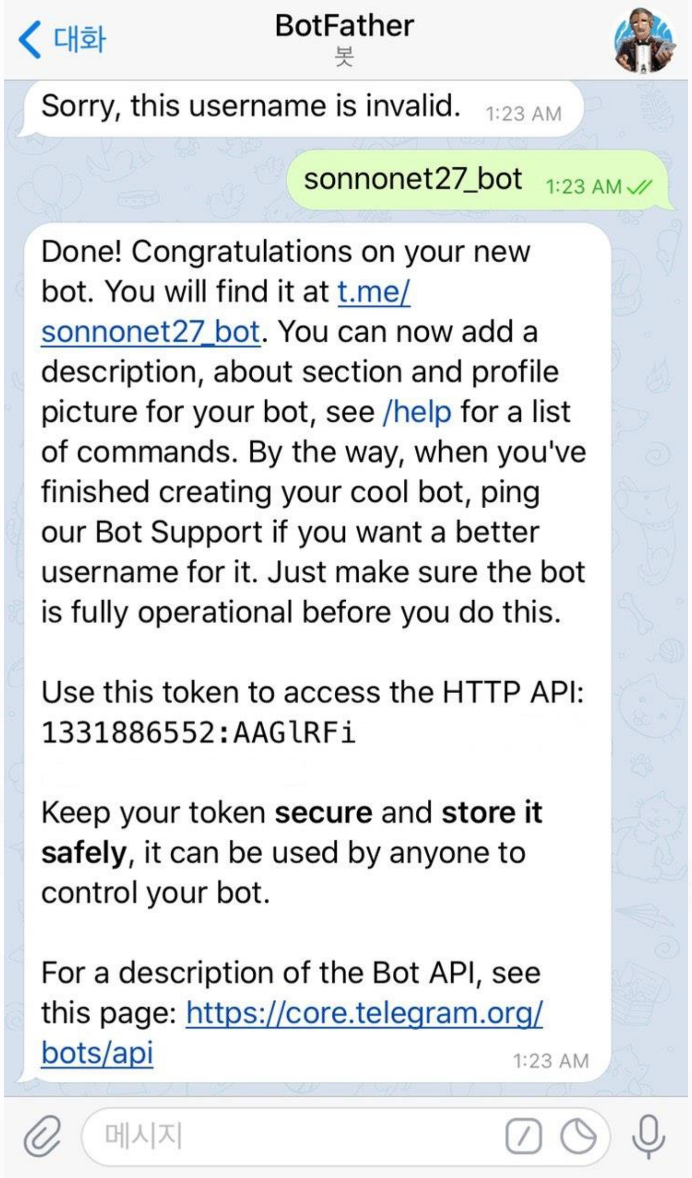
Sorry, this username is invalid. 1:23 AM

sonnonet27_bot 1:23 AM

Done! Congratulations on your new bot. You will find it at t.me/sonnonet27_bot. You can now add a description, about section and profile picture for your bot, see /help for a list of commands. By the way, when you've finished creating your cool bot, ping our Bot Support if you want a better username for it. Just make sure the bot

라즈베리파이 + Telegram 연동 하기

- 사용자 텔레그램 봇 token 부여 확인



Sorry, this username is invalid. 1:23 AM

sonnonet27_bot 1:23 AM

Done! Congratulations on your new bot. You will find it at t.me/sonnonet27_bot. You can now add a description, about section and profile picture for your bot, see /help for a list of commands. By the way, when you've finished creating your cool bot, ping our Bot Support if you want a better username for it. Just make sure the bot is fully operational before you do this.

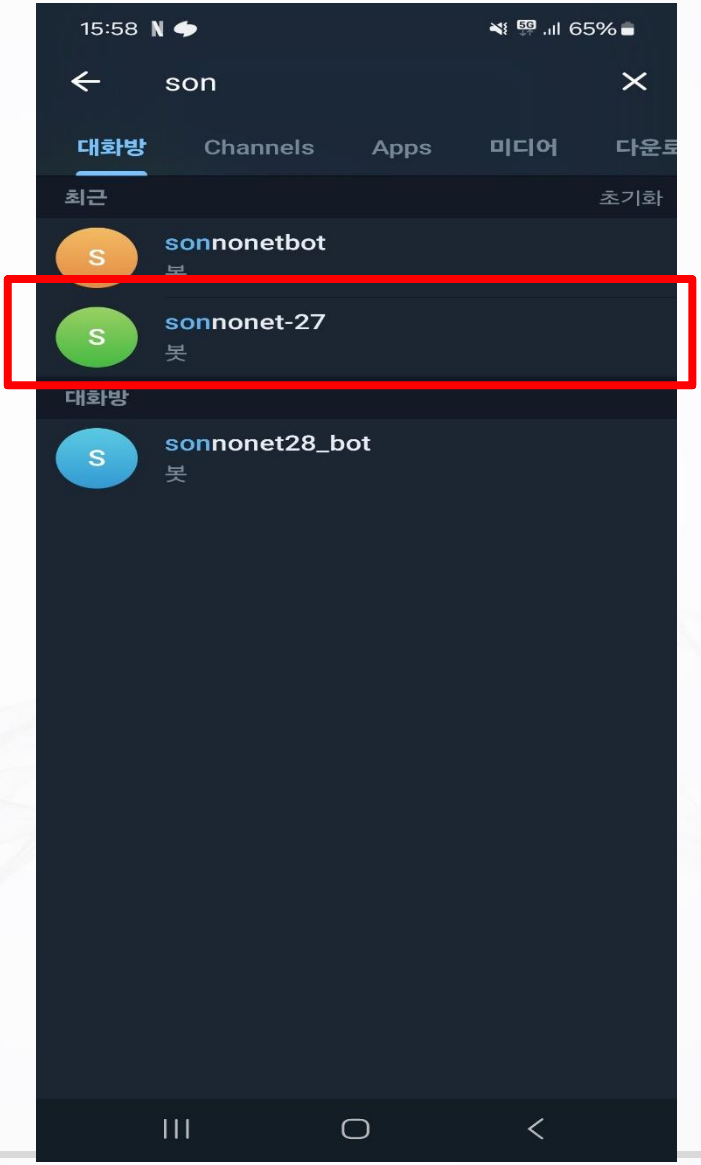
Use this token to access the HTTP API:
`1331886552:AAGlRFi`

Keep your token **secure** and **store it safely**, it can be used by anyone to control your bot.

For a description of the Bot API, see this page: https://core.telegram.org/bots/api 1:23 AM

라즈베리파이 + Telegram 연동 하기

- 사용자 텔레그램 봇 대화 시작



15:58 N 65%

son

최근 초기화

sonnonetbot
봇

sonnonet-27
봇

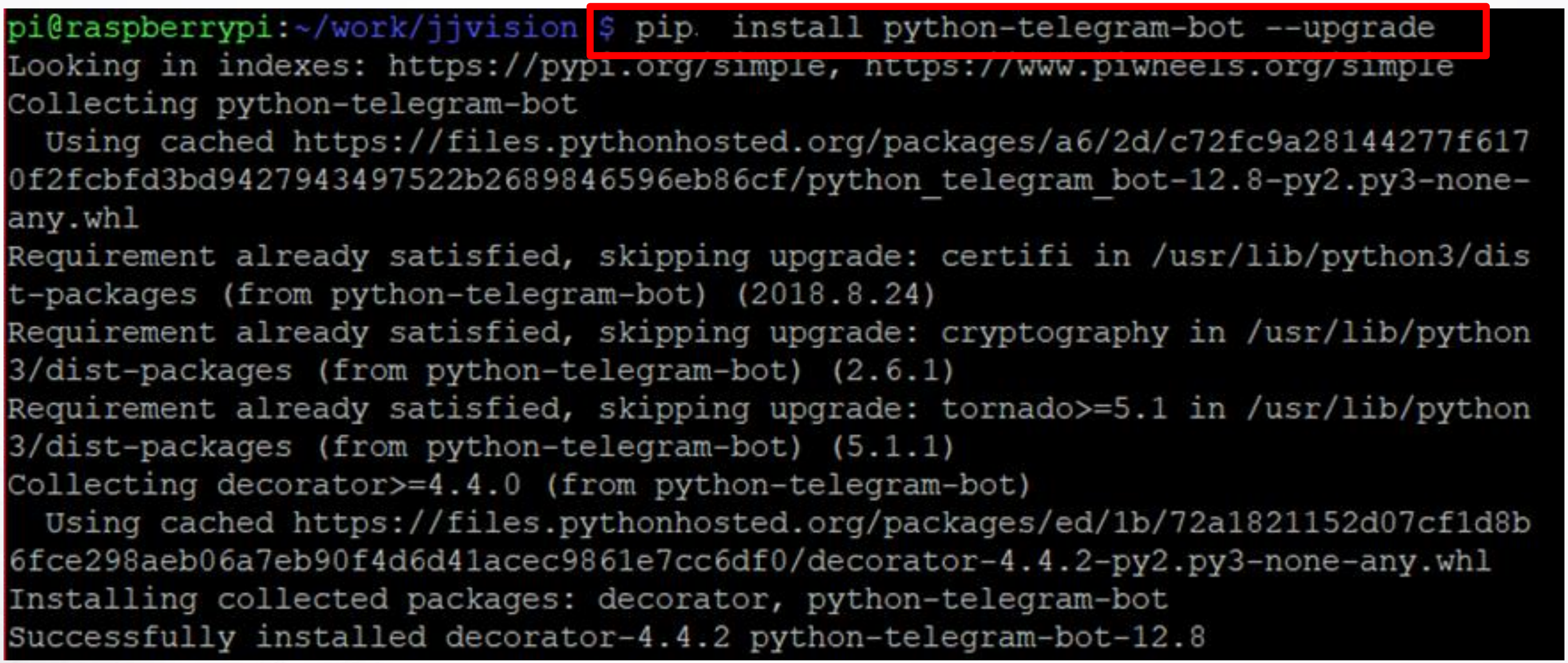
대화방

sonnonet28_bot
봇

라즈베리파이 + Telegram 연동 하기

python Telegram API 설치

```
pi@raspberrypi: ~ $ pip install python-telegram-bot --upgrade
```

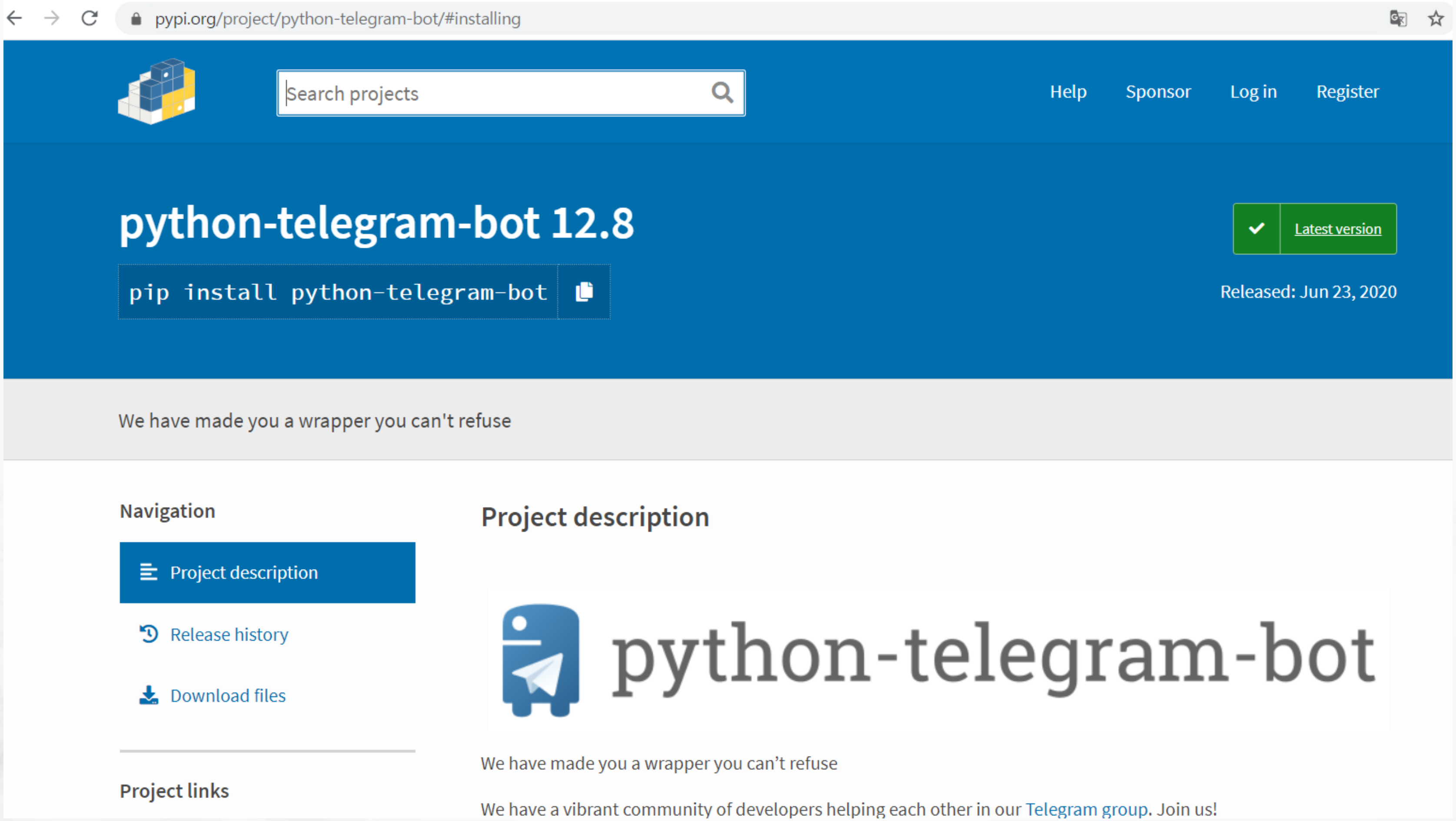


```
pi@raspberrypi:~/work/jjvision $ pip  install python-telegram-bot --upgrade
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting python-telegram-bot
  Using cached https://files.pythonhosted.org/packages/a6/2d/c72fc9a28144277f617
0f2fcbfd3bd9427943497522b2689846596eb86cf/python_telegram_bot-12.8-py2.py3-none-
any.whl
Requirement already satisfied, skipping upgrade: certifi in /usr/lib/python3/dis
t-packages (from python-telegram-bot) (2018.8.24)
Requirement already satisfied, skipping upgrade: cryptography in /usr/lib/python
3/dist-packages (from python-telegram-bot) (2.6.1)
Requirement already satisfied, skipping upgrade: tornado>=5.1 in /usr/lib/python
3/dist-packages (from python-telegram-bot) (5.1.1)
Collecting decorator>=4.4.0 (from python-telegram-bot)
  Using cached https://files.pythonhosted.org/packages/ed/1b/72a1821152d07cf1d8b
6fce298aeb06a7eb90f4d6d41acec9861e7cc6df0/decorator-4.4.2-py2.py3-none-any.whl
Installing collected packages: decorator, python-telegram-bot
Successfully installed decorator-4.4.2 python-telegram-bot-12.8
```

라즈베리파이 + Telegram 연동 하기

python Telegram API 사이트

https://pypi.org/project/python-telegram-bot/



python-telegram-bot 12.8

```
pip install python-telegram-bot
```

Released: Jun 23, 2020

We have made you a wrapper you can't refuse

Project description

We have made you a wrapper you can't refuse

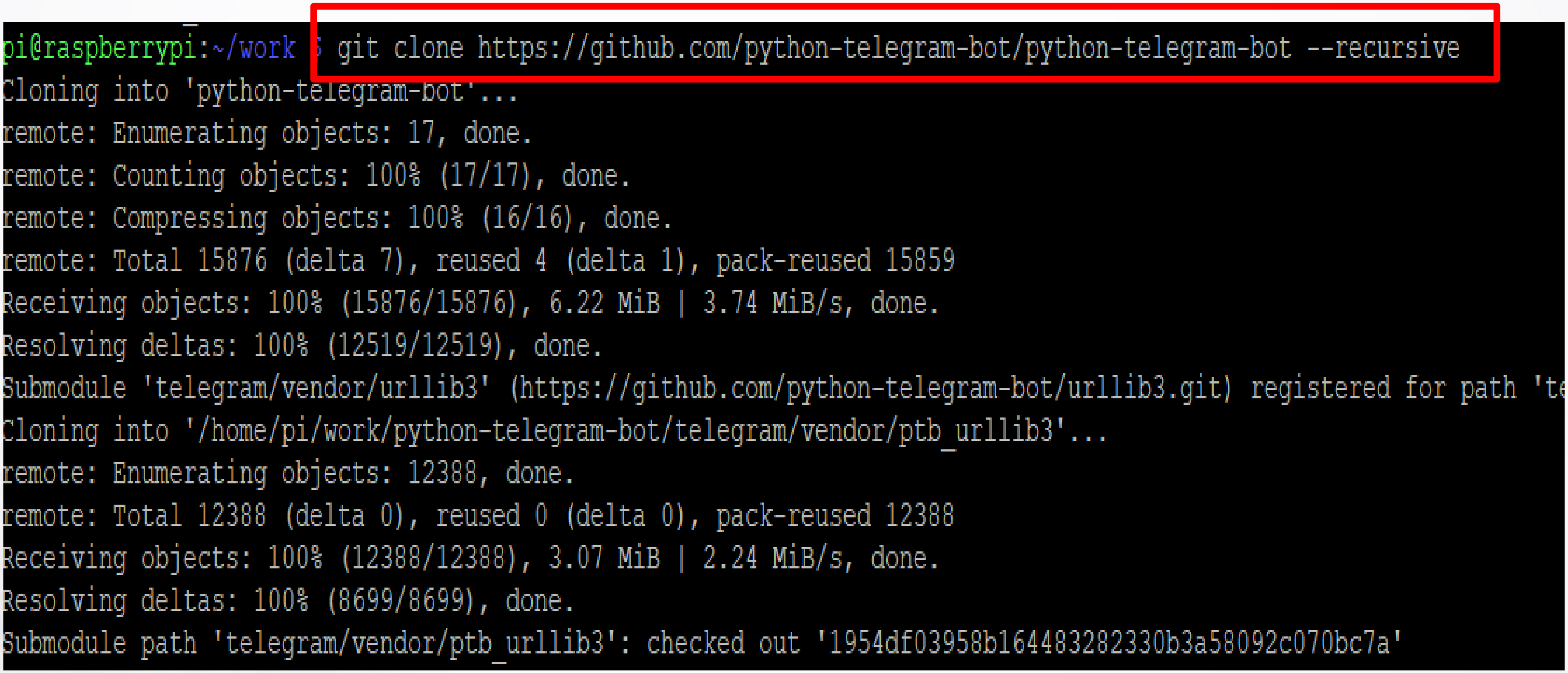
We have a vibrant community of developers helping each other in our Telegram group. Join us!

라즈베리파이 + Telegram 연동 하기

라즈베리파이 Telegram API 사용

```
pi@raspberrypi: ~ $
```

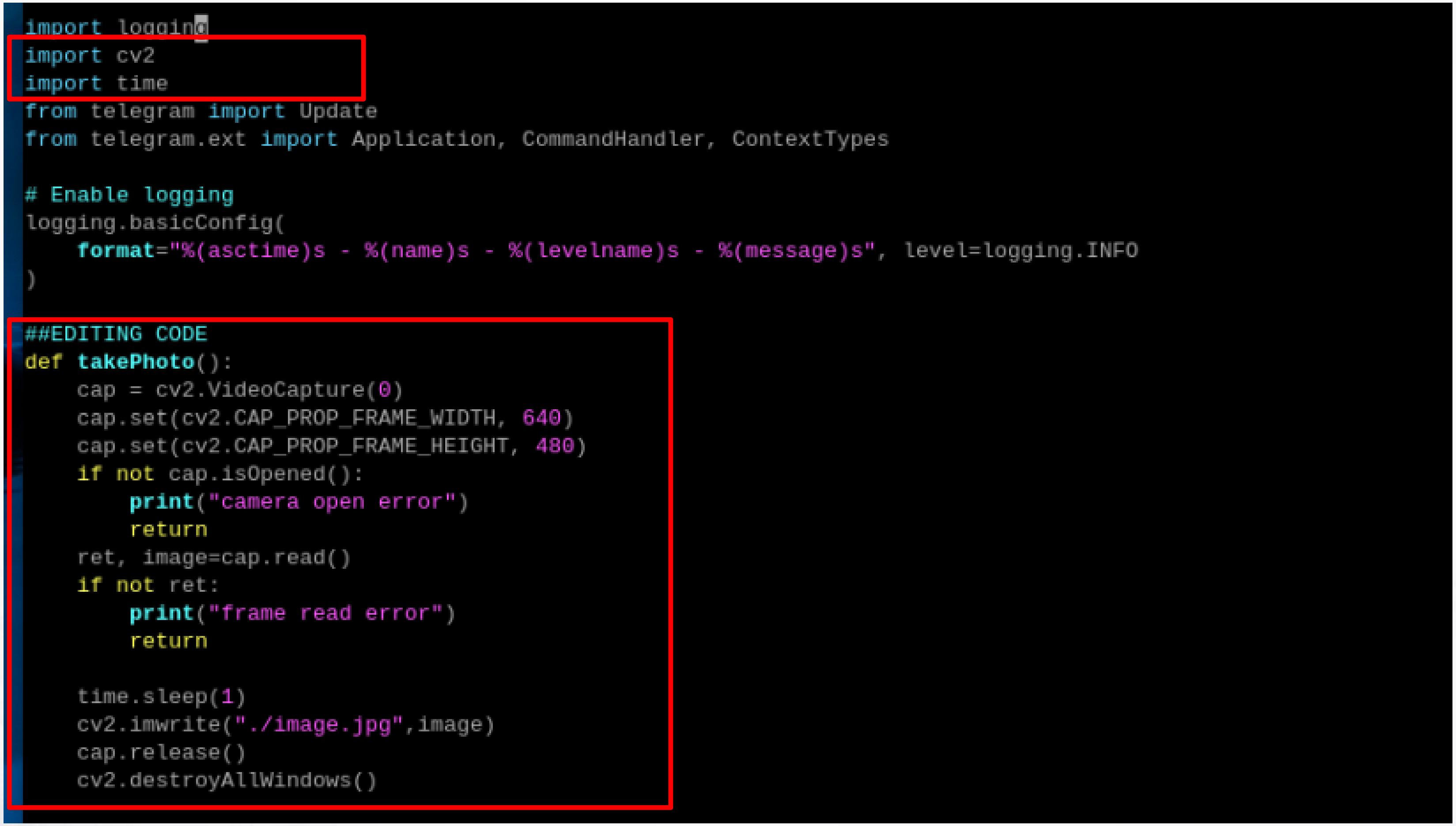
git clone https://github.com/python-telegram-bot/python-telegram-bot --recursive



```
pi@raspberrypi:~/work $ git clone https://github.com/python-telegram-bot/python-telegram-bot --recursive
Cloning into 'python-telegram-bot'...
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 15876 (delta 7), reused 4 (delta 1), pack-reused 15859
Receiving objects: 100% (15876/15876), 6.22 MiB | 3.74 MiB/s, done.
Resolving deltas: 100% (12519/12519), done.
Submodule 'telegram/vendor/urllib3' (https://github.com/python-telegram-bot/urllib3.git) registered for path 'te
Cloning into '/home/pi/work/python-telegram-bot/telegram/vendor/ptb_urllib3'...
remote: Enumerating objects: 12388, done.
remote: Total 12388 (delta 0), reused 0 (delta 0), pack-reused 12388
Receiving objects: 100% (12388/12388), 3.07 MiB | 2.24 MiB/s, done.
Resolving deltas: 100% (8699/8699), done.
Submodule path 'telegram/vendor/ptb_urllib3': checked out '1954df03958b164483282330b3a58092c070bc7a'
```

라즈베리파이 카메라 이용

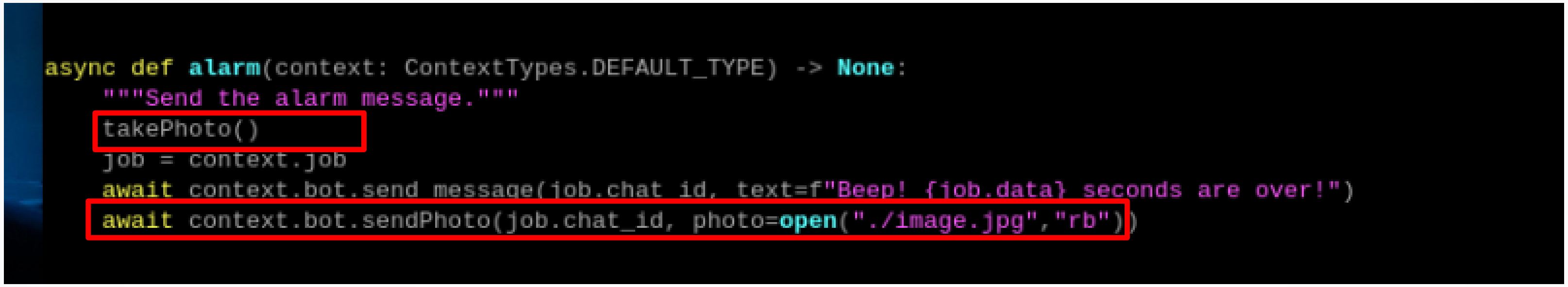
TimerBot.py 수정



```
import logging
import cv2
import time
from telegram import Update
from telegram.ext import Application, CommandHandler, ContextTypes

# Enable logging
logging.basicConfig(
    format="%(asctime)s - %(name)s - %(levelname)s - %(message)s", level=logging.INFO
)

##EDITING CODE
def takePhoto():
    cap = cv2.VideoCapture(0)
    cap.set(cv2.CAP_PROP_FRAME_WIDTH, 640)
    cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 480)
    if not cap.isOpened():
        print("camera open error")
        return
    ret, image=cap.read()
    if not ret:
        print("frame read error")
        return

    time.sleep(1)
    cv2.imwrite("./image.jpg",image)
    cap.release()
    cv2.destroyAllWindows()
```

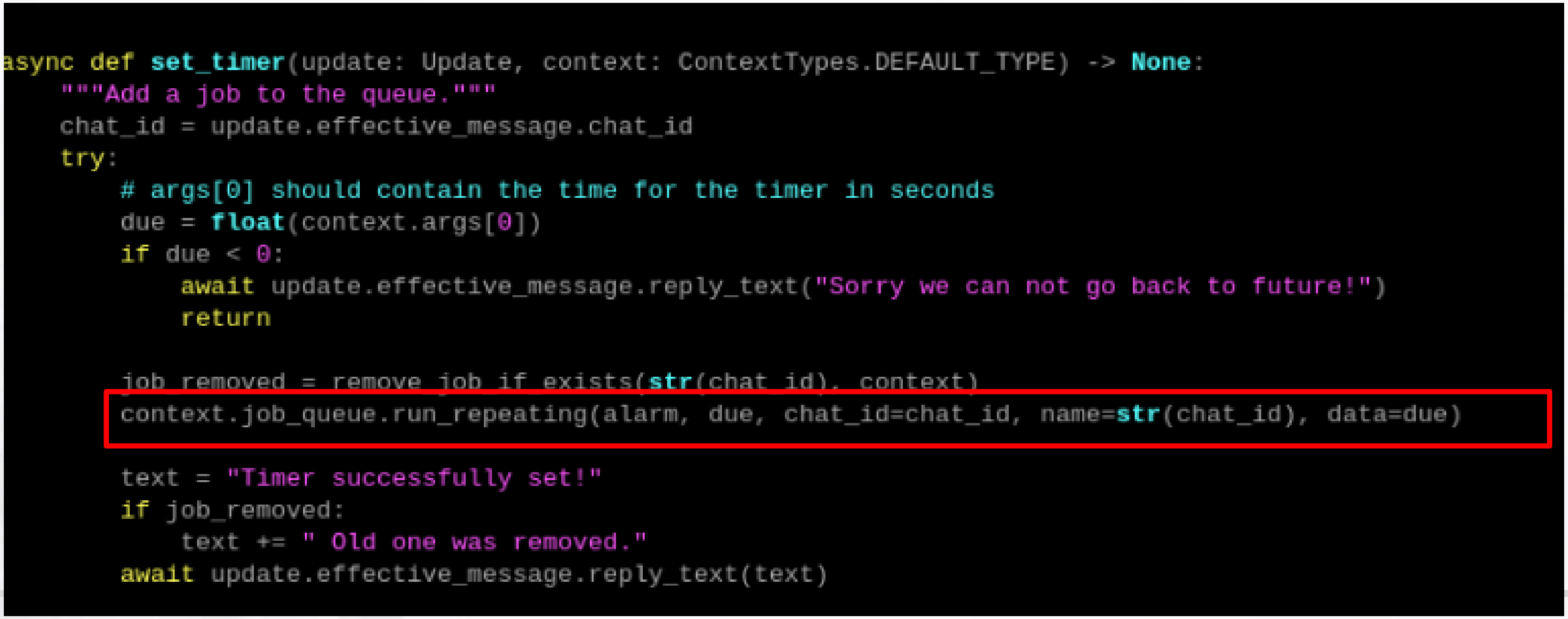
라즈베리파이 카메라 이용

TimerBot.py 수정 - 1



```
async def alarm(context: ContextTypes.DEFAULT_TYPE) -> None:
    """Send the alarm message."""
    takePhoto()
    job = context.job
    await context.bot.send_message(job.chat_id, text=f"Beep! {job.data} seconds are over!")
    await context.bot.sendPhoto(job.chat_id, photo=open("./image.jpg","rb"))
```

- context.job_queue.run.once() or repeating()



```
async def set_timer(update: Update, context: ContextTypes.DEFAULT_TYPE) -> None:
    """Add a job to the queue."""
    chat_id = update.effective_message.chat_id
    try:
        # args[0] should contain the time for the timer in seconds
        due = float(context.args[0])
        if due < 0:
            await update.effective_message.reply_text("Sorry we can not go back to future!")
            return

        job_removed = remove_job_if_exists(str(chat_id), context)
        context.job_queue.run_repeating(alarm, due, chat_id=chat_id, name=str(chat_id), data=due)

        text = "Timer successfully set!"
        if job_removed:
            text += " Old one was removed."
        await update.effective_message.reply_text(text)
```

11주차 수업이 끝났습니다

고생하셨습니다.