

✓ Tello Drone Python 개발환경 구축

✓ 준비물

- PC용 와이파이 모듈
- DJI_SDK - <https://github.com/dji-sdk/Tello-Python>

◦ 개발자에게 다른 프로그램에 추가하거나 연결할 수 있는 커스텀 앱을 제작할 수 있는 기능을 제공하는 도구 모음

✓ DJI 로보마스터 TT 텔로 텔런트 파이썬 제어

✓ 실습 전 준비 - 파이썬 텔로 드론 제어 프로그램 설치

- 1. 파이썬 과학툴 패키지 Anaconda 설치 - 아나콘다 설치 (https://www.anaconda.com/products/individual-d_)
- 2. 파이썬 통합개발 환경 Pycharm 설치 - 파이참 설치 (<https://www.jetbrains.com/ko-kr/pycharm/download/#section=windows>)
- 3. 텔로/텔로 에듀/텔로 텔런트 로보마스터 준비

- 텔로는 DJI의 기술이 들어간 RYZE ROBOTICS 사의 미니드론
- 기본적으로 스마트폰으로 조종을 하며, 조종기로도 조종할 수 있음
- 텔로드론은 스크래치만 사용가능
- 텔로 에듀 드론은 스크래치 / 파이썬의 두가지 언어가 모두 사용가능

 텔로	기체 구분	 텔로에듀
80g (프로펠러 및 배터리 포함)	무게	87g (프로펠러 및 배터리 포함)
Scratch	호환 코딩 프로그램	Scratch Python Swift
안됨	군집 비행	지원
Tello DroneBlocks	사용 앱	Tello Edu DroneBlocks
기체 x 1 배터리 x 1 프로펠러 2대 분량 프로펠러 가드 1대 분량	구성품	기체 x 1 배터리 x 1 프로펠러 2대 분량 프로펠러 가드 1대 분량 usb 충전 케이블 x 1 패드 x 4

✓ 드론 연결 순서

- 1. 드론의 전원을 작동시킨다
- 2. Tello 드론을 와이파이에 연결한다
- 3. 드론을 작동시킨다

✓ PC 설정 및 코딩

- PC에서 WiFi 접근이 가능해야 함
 - Tello의 전원을 넣고 PC의 WiFi를 Tello와 연결시킴(PC에서 와이파이로 Tello 드론을 접속).
- 파이썬 IDE(Pycharm/vscode등)을 이용하여 코딩후 실행

```
1 # tello_control_test_1.py 파일
2
3 import socket
4 from time import sleep
5
6 if __name__ == "__main__":
7     local_ip = ''
```

```
8     local_port = 8890 # read port
9
10    socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM) # socket for sending cmd
11    socket.bind((local_ip, local_port))
12
13    tello_ip = '192.168.10.1' # Tello는 AP(Access Point)이며 주소는 ip 192.168.10.1을 사용
14
15    #Send Command & Receive Response
16    #명령어는 UDP Port 8889를 통하여 보내줄 수 있으며 첫번째는 'command' 명령을 보내어 SDK 모드가 되도록 함
17    tello_port = 8889 # write port
18    tello_adderss = (tello_ip, tello_port)
19
20    socket.sendto('command'.encode('utf-8'), tello_adderss) # Enter SDK Mode
21
22    try:
23        index = 0
24        while True:
25            outStr=""
26            response, ip = socket.recvfrom(1024)
27            if response == 'ok':
28                continue
29            outStr = 'Tello State:' + str(response)
30            print(outStr)
31            sleep(0.2)
32    except KeyboardInterrupt:
33        pass
```

```

IDLE Shell 3.13.1
File Edit Shell Debug Options Window Help
6.00;agz:-1000.00;WrWn'
Tello State:b'mid:-1;x:-100;y:-100;z:-100;mpry:0,0,0;pitch:0;roll:0;yaw:0;vgx:0;
vgy:0;vgz:0;templ:50;temph:53;tof:10;h:0;bat:88;baro:26.87;time:0;agx:8.00;agy:-
6.00;agz:-1001.00;WrWn'
Tello State:b'mid:-1;x:-100;y:-100;z:-100;mpry:0,0,0;pitch:0;roll:0;yaw:0;vgx:0;
vgy:0;vgz:0;templ:50;temph:53;tof:10;h:0;bat:88;baro:26.94;time:0;agx:8.00;agy:-
4.00;agz:-998.00;WrWn'
Tello State:b'mid:-1;x:-100;y:-100;z:-100;mpry:0,0,0;pitch:0;roll:0;yaw:0;vgx:0;
vgy:0;vgz:0;templ:50;temph:53;tof:10;h:0;bat:88;baro:26.88;time:0;agx:8.00;agy:-
6.00;agz:-999.00;WrWn'
Tello State:b'mid:-1;x:-100;y:-100;z:-100;mpry:0,0,0;pitch:0;roll:0;yaw:0;vgx:0;
vgy:0;vgz:0;templ:50;temph:53;tof:10;h:0;bat:88;baro:26.91;time:0;agx:9.00;agy:-

```

- Send Command & Receive Response

- Tello는 AP(Access Point)이며 주소는 ip 192.168.10.1을 사용
- 명령어는 UDP Port 8889를 통하여 보내줄 수 있으며 첫번째는 'command' 명령을 보내어 SDK 모드가 되도록 함

- Receive Tello State

- PC에 UDP Server 0.0.0.0 UDP port: 8890를 설치하여 Tello로 부터 오는 메시지를 받을 수 있음

- Receive Tello Video Stream

- PC에 UDP Server 0.0.0.0 UDP port: 11111를 설치하여 Tello로 부터 오는 비디오를 받을 수 있음
- 비디오 스트림을 받기 위해서는 "streamon" 명령어를 UDP Port 8889로 보내주면 됨

Control Commands

Command	Description	Possible Response
Command	Enter SDK mode.	ok / error
takeoff	Auto takeoff.	
land	Auto landing.	
streamon	Enable video stream.	
streamoff	Disable video stream.	
emergency	Stop motors immediately.	
up x	Ascend to "x" cm. x = 20-500	
down x	down "x" Descend to "x" cm. x = 20-500	
left x	Fly left for "x" cm. "x" = 20-500	
right x	Fly right for "x" cm. "x" = 20-500	
forward x	Fly forward for "x" cm. "x" = 20-500	
back x	Fly backward for "x" cm. "x" = 20-500	
cw x	Rotate "x" degrees clockwise. "x" = 1-360	


✎ SDK 라이브러리 사용

- 소켓 프로그램이 아닌 SDK에서 제공하는 Tello() 클래스를 사용하면 더욱 쉽게 프로그램 개발이 가능

- 다운 받은 SDK에서 Tello() 클래스를 제공
- 소켓프로그램을 클래스 안에 내장한 것으로 좀 더 빠르고 편하게 개발할 수 있음

```
1 # tello_control_test_2.py파일
2 # 반드시 드론을 멀리 떨어져서 실행할 것
3 # 두 개의 파일 - 폴더 Single_Tello_Test에 있는 stats와 tello를 같은 폴더에 위치하고 실행할 것
4 # command 파일 작성할 것
5
6 from tello import Tello
7 from datetime import datetime
8 import time
9
10 start_time = str(datetime.now())
11 file_name = "command.txt"
12 f = open(file_name, "r")
13 commands = f.readlines()
14 tello = Tello()
15
16 for command in commands:
17     if command != '' and command != '\n':
18         command = command.rstrip()
19         if command.find('delay') != -1:
20             sec = float(command.partition('delay')[2])
21             print('delay %s' % sec)
22             time.sleep(sec)
23             pass
24         else:
25             tello.send_command(command)
26
27 log = tello.get_log()
28 outFile = open('log.txt', 'w+')
29 for stat in log:
30     stat.print_stats()
```

```
31     str = stat.return_stats()  
32     outFile.write(str)
```

 command - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

command

takeoff

delay 5

land

✓ [DJITelloPy API Reference](#)

✓ [기본동작제어](#)

- takeoff/land - takeoff(), land()
 - move up/down - move_up(), move_down()
 - move left/right - move_left(), move_right()
 - rotate_CW_CCW - rotate_clockwise(), rotate_counter_clockwise()
 - send_rc_control_async - send_rc_control(self, left_right_velocity, foward_backward_velocity, up_down_velocity, yaw_velocity)
-

✓ 실습(1)

- takeoff -> 3회 up(40) -> down(40) 반복 -> landing
-

✓ 실습(2)

- takeoff -> fwd(40) -> fwd(40) -> fwd(40) -> cw(180) -> fwd(40) -> fwd(40) -> fwd(40) -> land
-

✓ 참고사항

```
1 #기본적인 동작 제어
2 from djitellopy import Tello
3 tello = Tello()
4 tello.connect()
5 tello.takeoff()
6 tello.move_left(100)
7 tello.rotate_clockwise(90)
8 tello.move_forward(100)
9 tello.land()
```

```
1 #반복문 이용하기 : for in range()
2 from djitellopy import Tello
3 myTello = Tello()
4 myTello.connect()
5 myTello.takeoff()
```

```
6 myTello.move_up(30)
7 myTello.move_down(30)
8 myTello.move_up(30)
9 myTello.move_down(30)
10 myTello.move_up(30)
11 myTello.move_down(30)
12 myTello.land()
13
14 '''from djitellopy import Tello
15 myTello = Tello()
16 myTello.connect()
17 myTello.takeoff()
18 for i in range(0, 3):
19     myTello.move_up(30)
20 myTello.rotate_counter_clockwise(90)
21 myTello.move_down(30)
22 myTello.land()'''
```

```
1 #파이썬 함수 이용하기 : def
2 from djitellopy import Tello
3 myTello = Tello()
4 myTello.connect()
5 myTello.takeoff()
6 myTello.move_up(30)
7 myTello.move_down(30)
8 myTello.move_up(30)
9 myTello.move_down(30)
10 myTello.move_up(30)
11 myTello.move_down(30)
12 myTello.land()
13
14 '''def move_up_down(t):
15 myTello.move_up(t)
16 myTello.move_down(t)
17 for i in range(3):
18     t = 30
19     move_down(t)'''
```

✓ 실습(3)- for문 이용

- takeoff -> fwd(40) -> fwd(40) -> fwd(40) -> cw(180) -> fwd(40) -> fwd(40) -> fwd(40) -> cw(180) -> land

```
1 #input() 함수를 사용한 제어
2 from djitellopy import Tello
3 myTello = Tello()
4 myTello.connect()
5 battery_level = tello.get_battery()
6 print(battery_level)
7
8 while True:
```

```
9     command = int(input("Enter Command!"))
10     print(command, end="\n")
11     if (command == 1):
12         myTello.takeoff()
13     elif (command == 2):
14         myTello.move_up(30)
15     elif (command == 3):
16         myTello.move_down(30)
17     elif (command == 4):
18         myTello.land()
19     else :
20         break
21 print("Drone mission completed!")
```

✓ 실습(4)

- input() 함수를 이용한 드론 조종기 만들기

- takeoff()
- move_up(20)
- move_down(20)
- move_left(20)
- move_right(20)
- move_forward(20)
- move_backward(20)
- rotate_clockwise(90)
- rotate_counter_clockwise(90)
- flip_back()
- flip_forward()

- flip_left()
 - flip_right()
 - land()
-

□ □ □ □ □