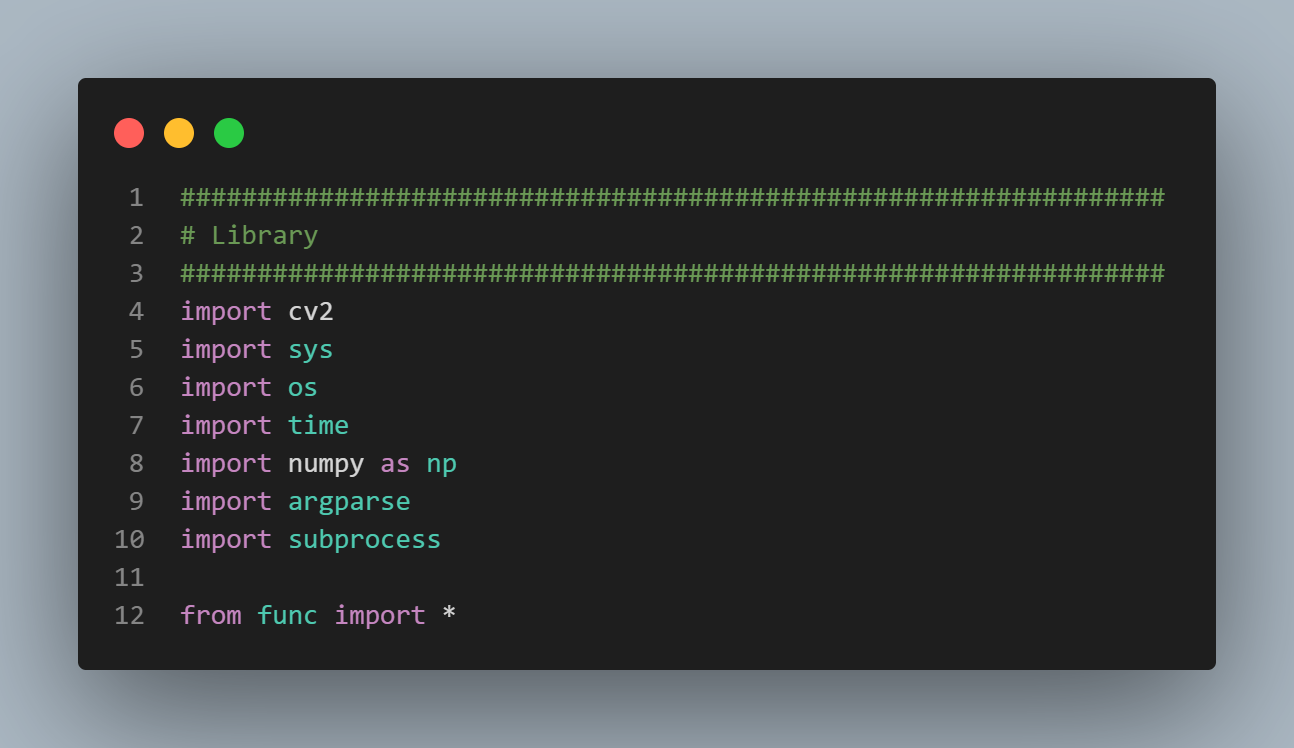
**“유니스트 적정기술 프로젝트 – Python – Digital”**

**[main.py]**

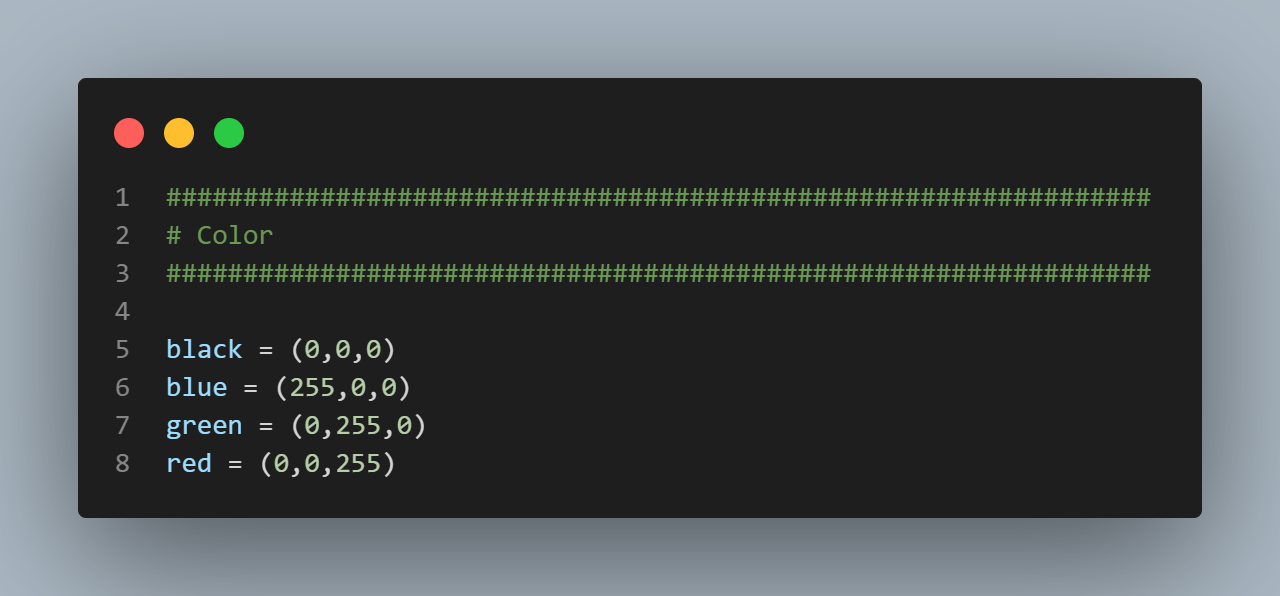
**7-segment를 읽는데 필요한 변수들인 ‘디바이스의 이름’, ‘화면의 각도(카메라를 처음 설치했을 때 기기의 배치에 따라 화면이 회전했을 때의 경우를 대비’, ‘박스의 개수’, ‘박스의 위치’, ‘박스의 크기’를 통해 값을 읽는다. 박스의 위치를 지정하여 학습된 숫자를 읽는다. Main.py에서는 변수를 받아오고 func 함수를 실행하는 역할을 한다.**

**1. Library 설치**



파이썬 라이브러리를 호출한다. func파일은 같이 첨부되어 있는 func.py 파일이다.

**2. Color 설정**



총 3가지 색깔, 파랑, 초록, 빨강색을 설정한다.

**3. Argument 설정**



Argparse 라이브러리에 있는 Add\_argument() method를 이용하여 프로그램에 필요한 인자들을 정의한다. 여기서 인자 앞에 ‘--‘는 optional 인자(선택형 인자)이다.

변수 이름에서 확인할 수 있듯이 1) ‘device’ = 기기 이름, 2) ‘angle’ = 화면의 회전 각도, 3) ‘bbox’ = 바운딩 박스(bounding box)의 수 4) ‘scale’ = 바운딩 박스(bounding box)의 눈금 5) ‘pos’ = ‘바운딩 박스(bounding box)의 좌표’

**4. Config 설정**



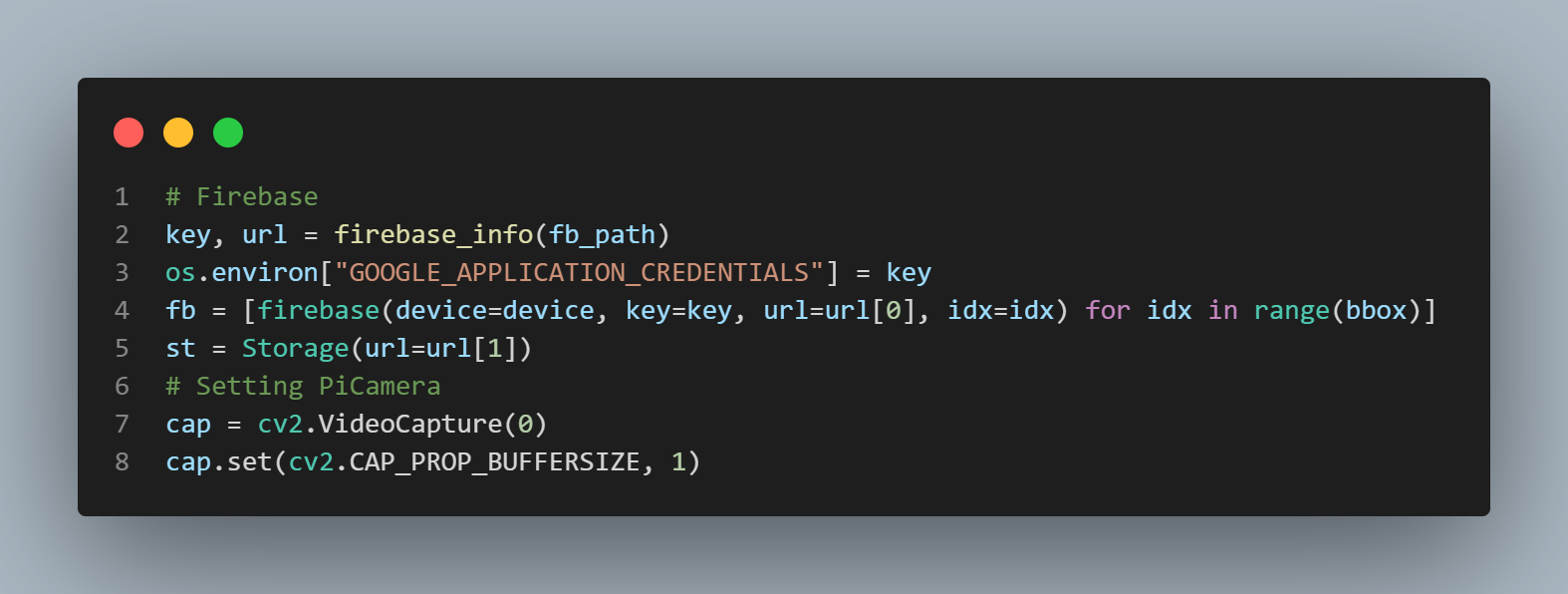
기본 설정된 값 및 파일 지정 경로를 불러온다.

**5. 변수 지정**



Arguments 받아온 값으로 코드에서 사용할 변수를 지정한다.

**6. Firebase 지정 및 파이카메라 선언**

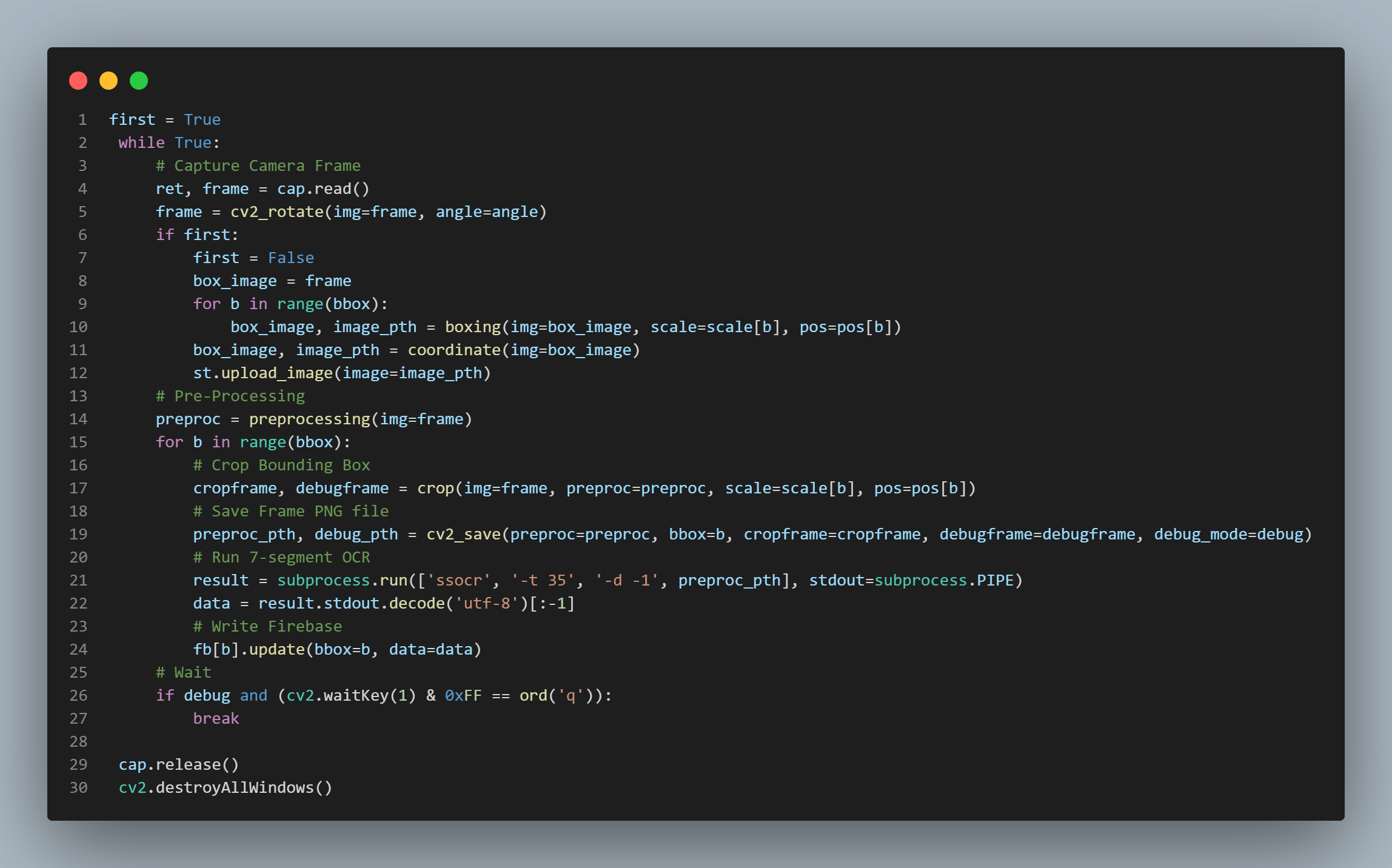


Firebase url의 경우 만들어진 key.json을 통해 url선언을 하였고, 파이카메라의 경우 이미지 생성을위해 사용되는 video capture기능을 사용하여 선언하였다.

\* Firebase 프로젝트 설정은 https://firebase.google.com/?hl=ko 에서 할 수 있다.

: Analog와 동일 내용

**7. 카메라 작동 알고리즘**



Line 4-12 : 디지털 센서 카메라의 프레임을 찾는 과정

Line 17 : bounding box를 자름

Line 19 : PNG 파일로 프레임을 저장

Line 21-22 : 표시 장치의 일종인 7세그먼트 표시 실행

Line 24 : Firebase로 데이터 전송

Line 26-27 : Break를 통해 제어 흐름을 중단하고 빠져나옴

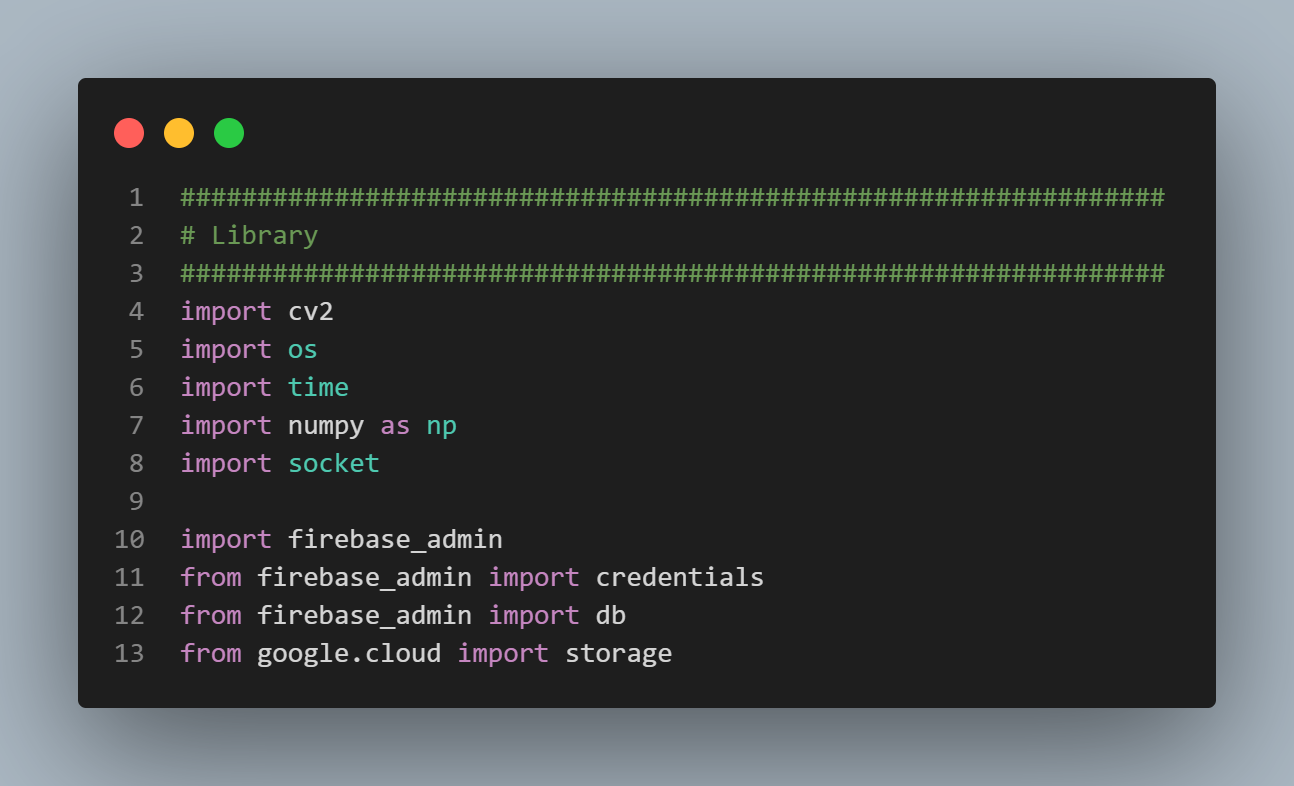
**8. Arguments 값 불러오고 main function 실행**

****

**[func.py]**

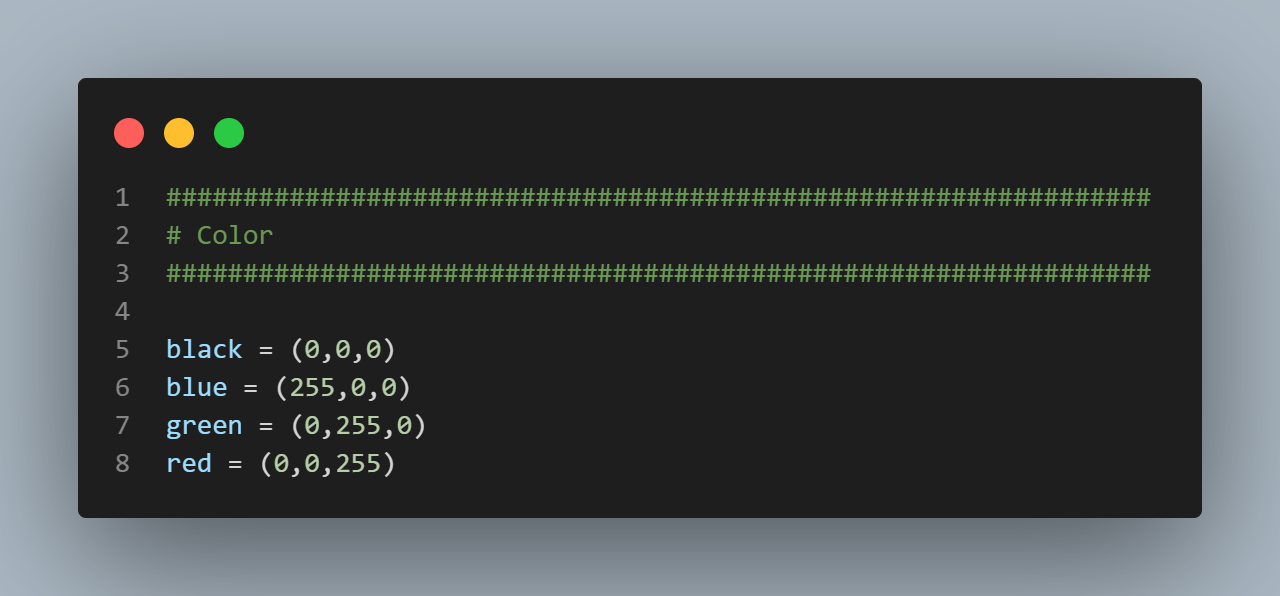
**이미지를 흑백화하여 숫자 정보와 배경 정보를 추출한 뒤 숫자 정보를 기존에 학습되었던 내용과 비교하여 숫자를 찾는 과정이다.**

**1. Library 설치**



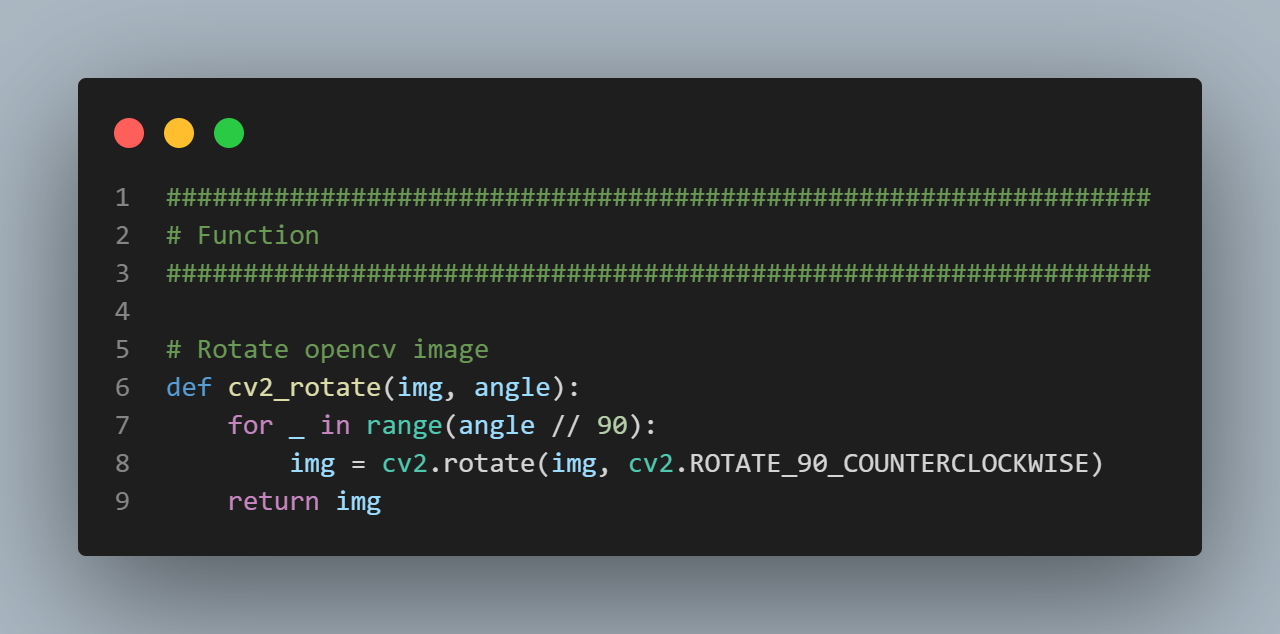
파이썬 라이브러리 호출한다.

**2. Color 설정**



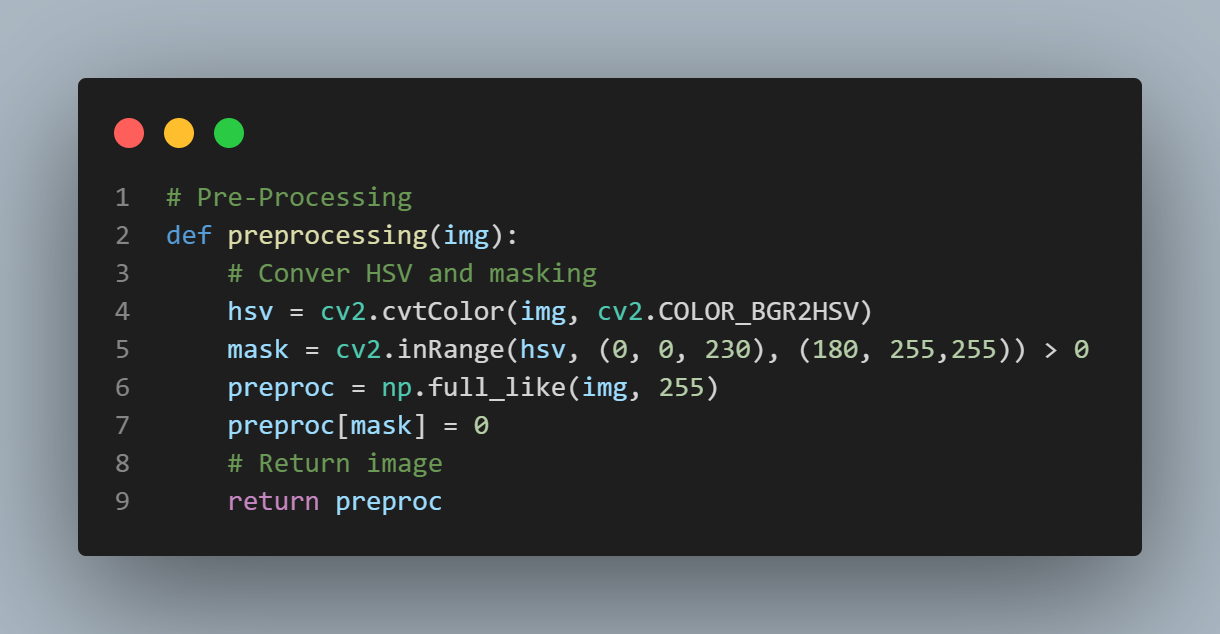
총 4가지 색깔, 검정, 파랑, 초록, 빨강색을 설정한다.

**3. 이미지 회전**



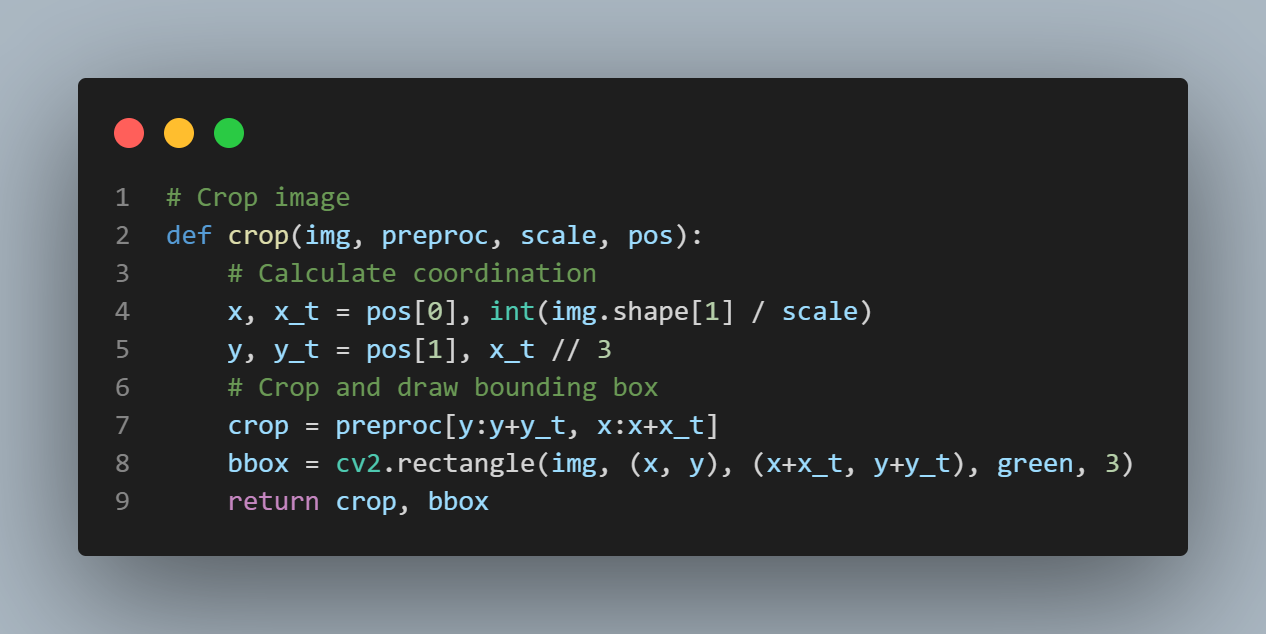
파이카메라가 종방향이 아닐 때 이미지 회전을 위한 코드이다.

**4. HSV 및 마스킹 변환**



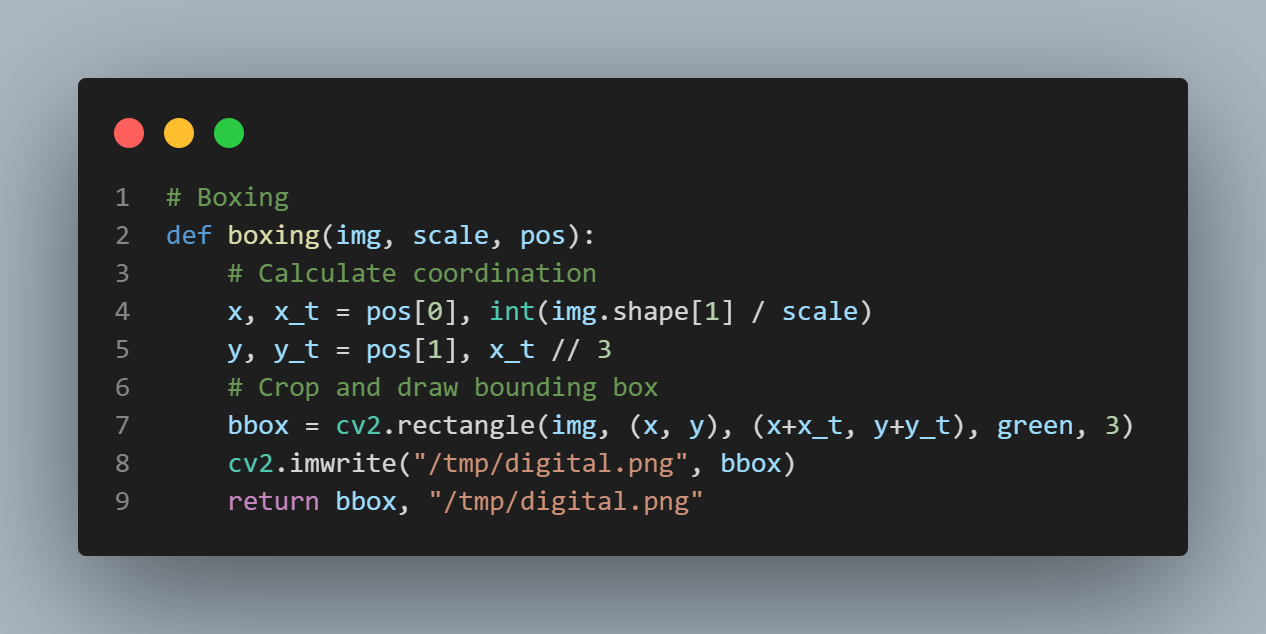
Cv2.cvtColor 기능 중, COLOR\_BGR2HSV를 사용하면 지배적인 파장의 색상을 강조하는 형태로 변형된다. 또한 csv2.inRange함수를 활용하여 특성 색상 영역을 추출한다.

**5. 이미지를 자름**



좌표 계산 후, 이미지를 자르고 bounding box를 그린다.

**6. 이미지 저장**



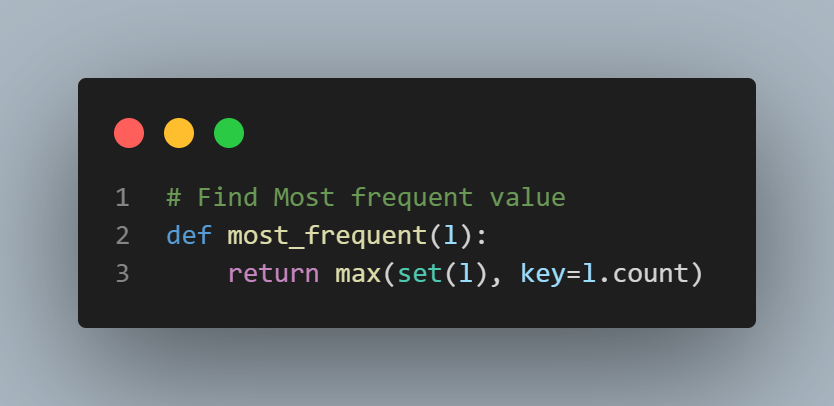
좌표를 계산한 후 이미지를 자르고 bounding box를 그려, Cv2.imwrite함수를 사용하여 ”/tmp/digital.png”로 이미지 파일을 저장한다.

**7. 좌표 그리기**



이미지에 좌표를 그린 후, “/tmp/digital.png” 로 저장한다.

**8. 가장 자주 나오는 값 추출**



**9. 전처리 과정 및 파일 저장**



Csv.imwrite 기능을 활용하여 각각의 과정을 담은 사진들을 png파일로 저장한다.

**10. 파이어베이스 정보 추출**



연결하는 소켓과 ip는 때에 따라서 수정해야 한다. 사용하는 기기의 IP 주소를 Line 120 칸에 있는 IP 대신 입력해주면 된다.

**11. Firebase 관련 클래스 코드**



Firebase 설정을 위해 기존의 같은 태그를 없애고 기기의 ip를 서버로 업데이트 하는 코드이다.



Firebase에 데이터를 업로드하고 저장하며 기기의 IP를 서버의 태그 하단에 저장한다.



Firebase 데이터를 클리어하는 코드와 이미지를 업로드하는 코드이다. 이 코드에서는 이미지를 ‘digital.png’로 업로드 하였는데, 파일 명을 변경할 수도 있다.