

# stegano.md

과제08, 컴퓨터의 개념 및 실습 수업용 파일입니다. 언어학과, 2021-12659 박유나가 작성하였습니다.

## 코드 설명

```
image = Image.open("mandrill_stegano2.png")
width, height = image.size

pixels = image.load()
```

mandrill\_stegano.png파일을 봤을 때, 하단의 빨간 부분이 의심스러워 그부분만 잘라서 사용했습니다. 옳지 않은 방법인 줄 알면서도 pil 모듈에 있는 사진 자르는 기능을 사용할 줄을 몰라, 사진편집프로그램을 이용하여 사진을 잘라내었습니다.

```
list = []
for h in range(0, height, 16):
    for w in range(0, width, 16):
        list.append(chr(pixels[w, h][1]))
''.join(list)
```

처음에 pixels[w, h]를 그냥 print 했을 때 green과 blue가 16개씩 반복되어 16step으로 잘라내었습니다. red는 계속 바뀌지만 green과 blue는 비슷하기에 chr을 이용하여 유니코드로 변환한 값을 리스트에 추가하고 리스트를 문자열로 합쳤습니다. 그랬더니,

"#look at the red channel #collect lsb #convert binary to unicode #try chr(0b0000000001000001)" 라는 문장이 보였습니다.

검색해서 찾아보니, red channel은 pixel의 rgb 값 중 r을 의미하는 것 같았으며, lsb는 최하위 비트를 의미하였습니다. 그래서 이를 토대로 나름대로 해결 방안을 세워보았습니다. 우선, 빨간 상자의 r 값을 binary로 바꾸고 lsb를 수집합니다. 그리고 이를 chr(0b0000000001000001)처럼 16개로 나누고 chr을 통해 유니코드로 변환합니다. 이 방안에 따라 아래와 같은 코드를 작성하였습니다.

```
# red channeel 및 lsb 추출
list_red = []
for h in range(height):
    for w in range(width):
        list_red.append(bin(pixels[w, h][0])[-1])

# 16개씩 분할
list_red2 = [list_red[i:i + 16] for i in range(0, len(list_red), 16)]

# binary를 유니코드로 변환
for i in list_red2:
    s = ''.join(i)
```

```
s = '\0b' + s
umm = int(s, 2)
chr(umm)
```

chr(umm)의 결과로는 한자, 잘 안쓰이는 한글, 기호 그리고 '\x\*\*' or '\u\*\*'만 출력되었습니다. 그래서 과제는 수행은 실패하였다고 생각합니다. 나름대로 시간도 투자하고 고민도 많이 했지만 끝내 답을 알아내지 못하여 아쉽습니다. 점수는 기대하지 않지만 피드백을 통해 올바른 풀이를 보고 싶어서 파일을 제출합니다.