시스템 설계

*Open Source SW Contribution*

모바일시스템공학과 : 32157629 한송이

32161620 박산희

소프트웨어학과 : 32157451 이제영

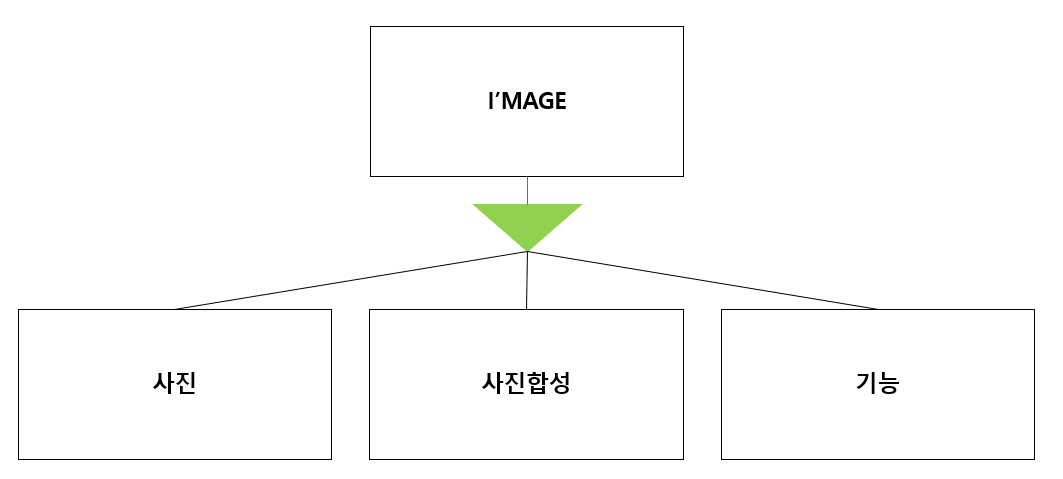
32157022 고수열

32157204 박상준

**I’MAGE**

**1. 시스템 아키텍쳐**

* **모듈 설명**



1-1 사진 모듈

-사진 모듈 안에는 사진을 준비시키는 준비과정과 관련된 모듈이다. 변환시킬 사진과 스타일로 쓸 사진들을 등록하고, 앨범에서 불러오는 기능과 있는 모듈이다. 또한 사진을 촬영할 수 있는 기능도 넣어 사진을 직접 찍으면서 준비시킬 수도 있는 기능도 담고 있다.

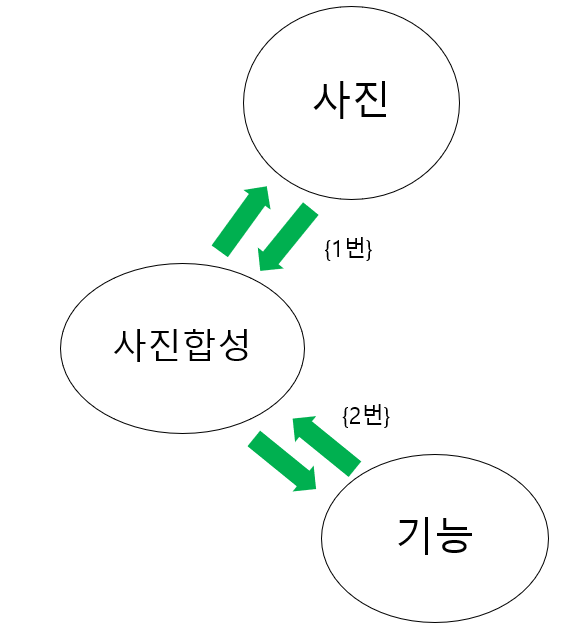
1-2 사진합성 모듈

-사진을 불러와 편집, 합성하는 과정과 관련된 모듈이다. 사진을 부분별로 나누어 바꾸고자 하는 부분만 바꿀 수 있도록 하거나 전체 사진을 바꾸는 등 편집, 합성에 관련된 기능들을 담고 있다.

1-3 기능 모듈(사진 편집 모듈)

-사진 편집과 관련된 모듈이다. 채도, 대비, RGB의 평균을 통해서 추천을 할 수도 있고 사용자가 원하는 색으로 바꾸는 등에 관련된 기능을 담고 있다.

* **모듈 간 인터페이스**



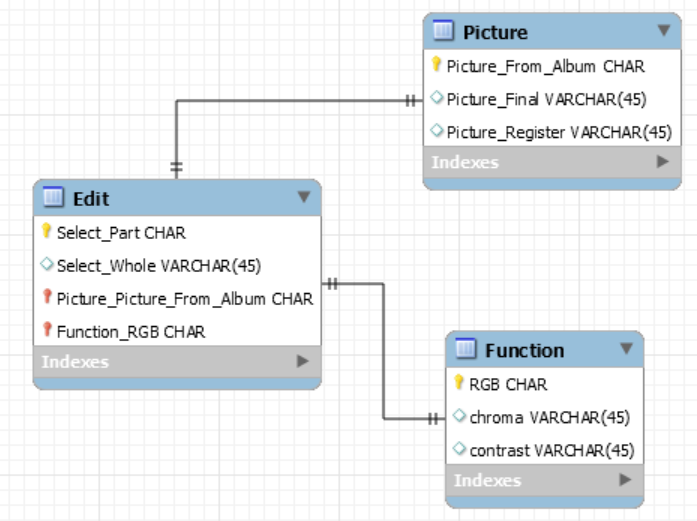
각 모듈간 인터페이스에는 크게 1번과정과 2번과정이 있다.

1번과정에서는 사진 모듈과 사진합성 모듈간의 인터페이스이다. 사진 모듈에서 찍거나 불러와 사진을 준비시키고 사진합성 모듈에서 그 준비시킨 사진을 이용하여 합성하고 편집하는 인터페이스를 제공한다.

2번과정에서는 사진합성 과정에서 채도, 대비를 바꾸고 RGB를 추천하는 등 사용자 정의대로 사진합성의 기능을 더할 수 있도록 인터페이스를 제공한다.

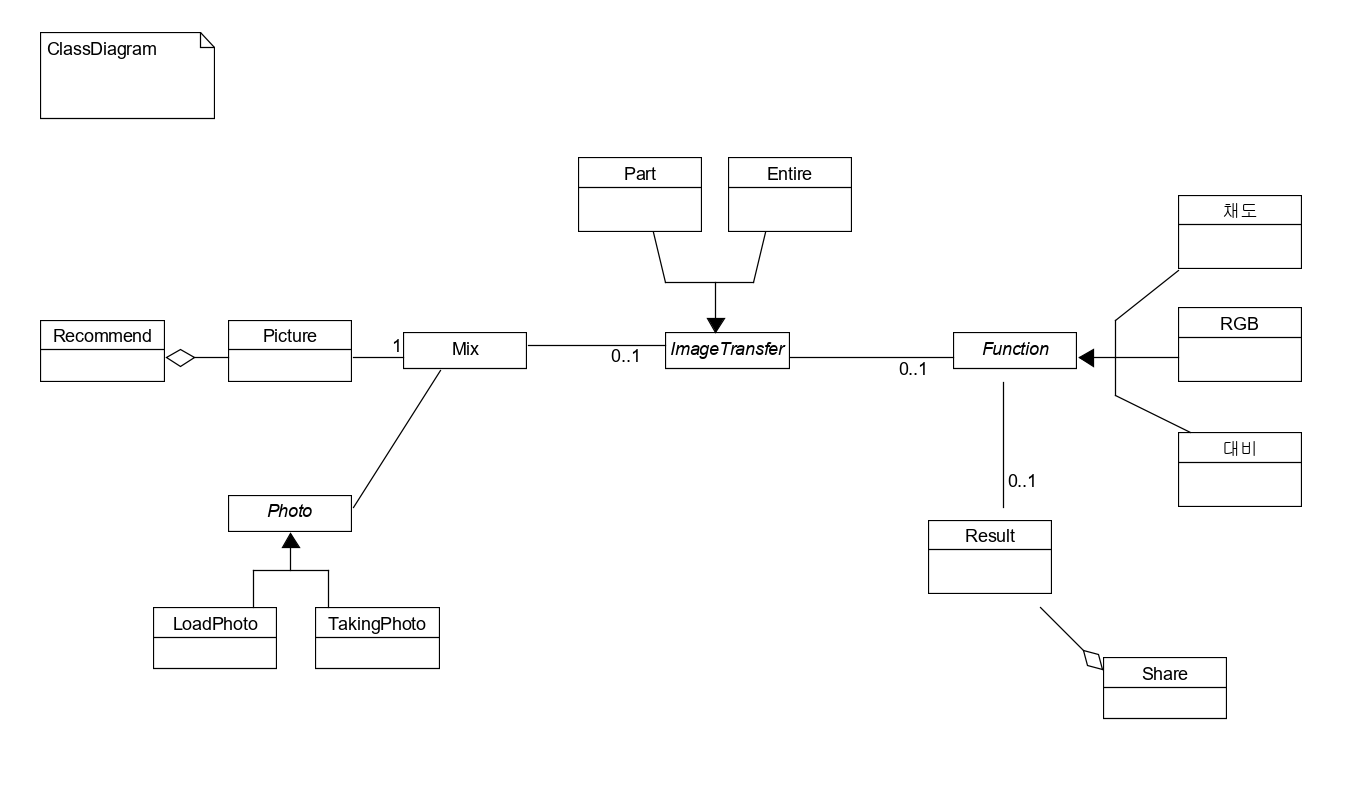
결과적으로 기능 모듈과 사진합성 모듈의 결과물로서 완성된 사진을 다시 사진 모듈로 보내 앨범에 저장한다.

* **데이터 베이스 스키마**

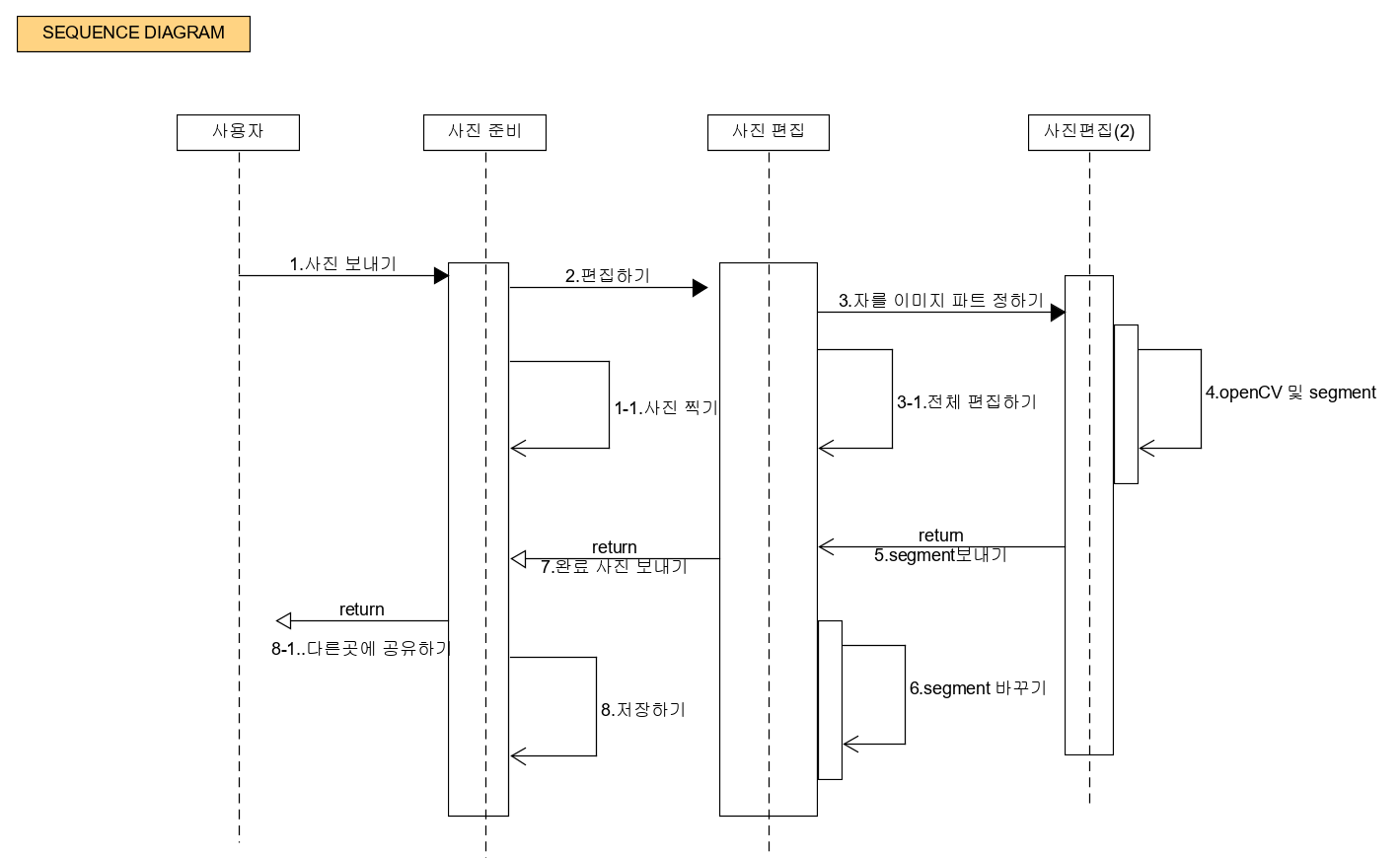


* **Class diagram 및 Sequence diagram**

-class diagram



-sequence diagram



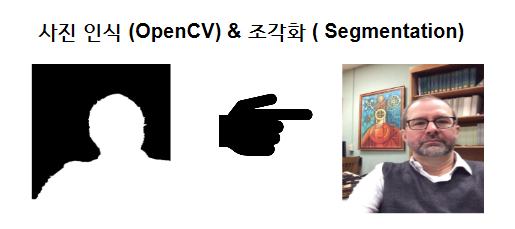
**2. 요소 기술**

**- 핵심 기술 (외부 API 포함) 설명**

**[Style transfer]**



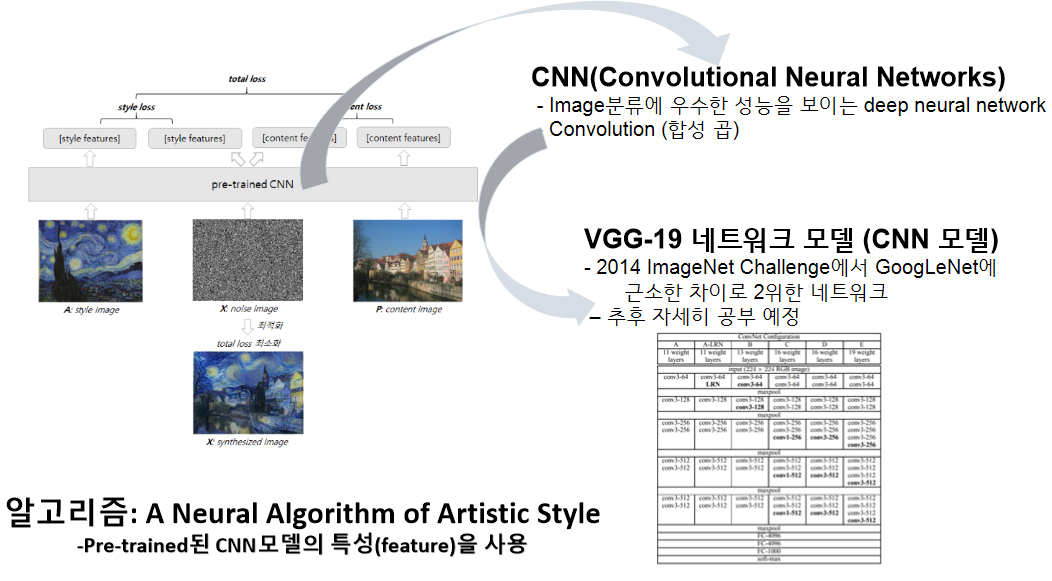
\*Image transfer : 일반인에게 익숙하지 않은 예술분야를 IT기술과 융합하여 일상생활에 적용 할 수 있게 한다. ex) SNS에 활용, 기념품의 역할



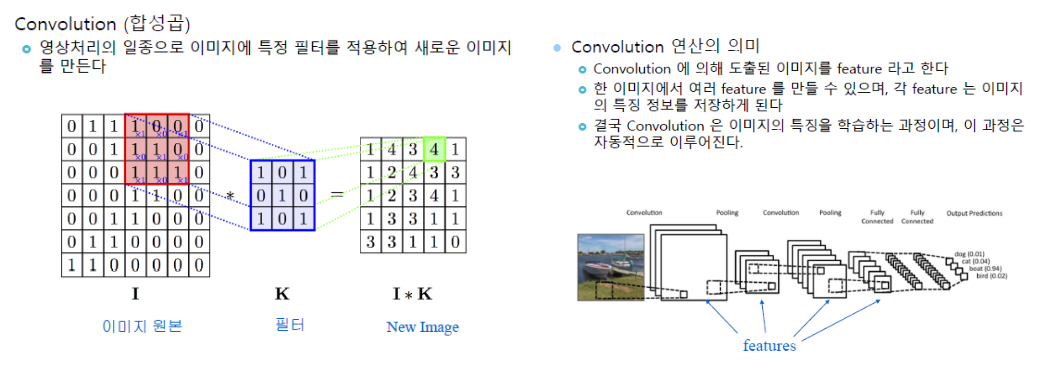
\*OpenCV(Open Source Computer Vision) : 주로 실시간 컴퓨터 비전을 목적으로 한 프로그래밍 라이브러리이다.

\*Segmentation : 디지털 영상을 여러 개의 픽셀 집합으로 나누는 과정을 말한다. 영상 분할은 특히 영상에서 물체와 경계(선, 곡선)를 찾는데 사용된다. 일반적으로 일반 영상에서의 대표 영상분할 알고리즘은 임계값 기법, 에지 검출 기법, 영역 성장법, 텍스처 특징 값을 이용하는 기법 등이 사용되고 있다.

**[Neural style transfer]**

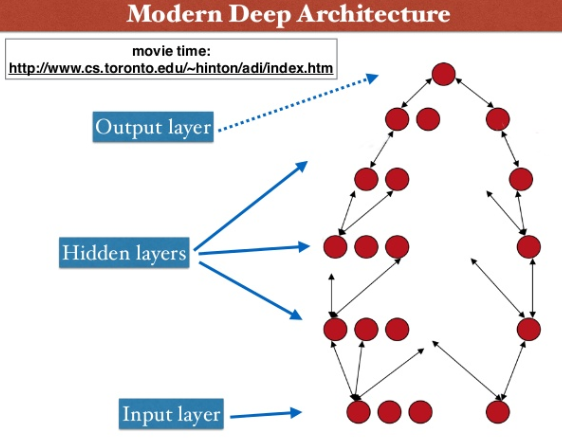


\*Convolution(합성곱) 참고



\*Neural style transfer : pre-trained 된 모델을 기반으로 content image 와 style image를 이용해 이미지 변환 네트워크를 학습하는 방법. 재학습이 필요없고 실시간 변환이 가능.

**[Deep Learning]**

****

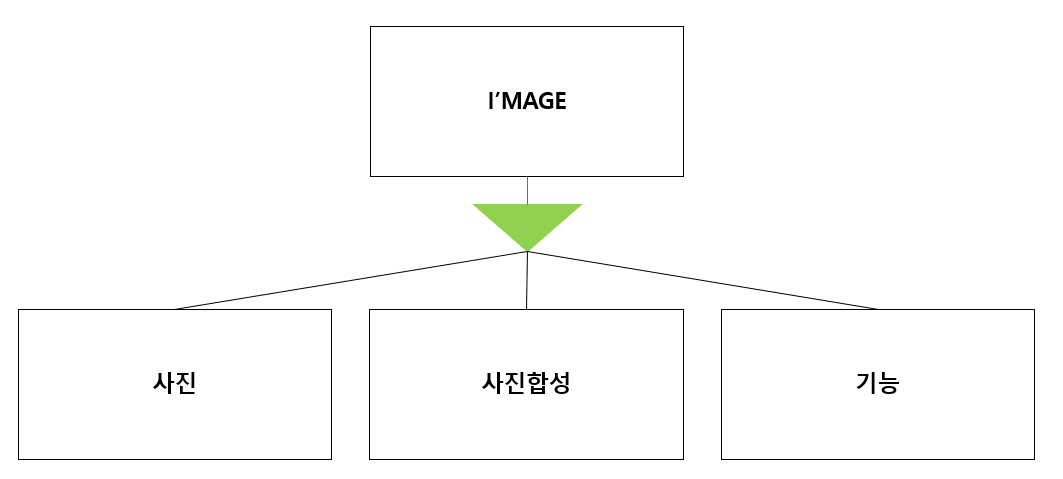
딥러닝이란, 다층구조 형태의 신경망을 기반으로 하는 머신 러닝의 한 분야로, 다량의 데이터로부터 높은 수준의 추상화 모델을 구축하고자 하는 기법이다. **얼굴이나 표정을 인식하는 등의 특정 학습 목표에 대해,딥 러닝은 학습을 위한 더 나은 표현 방법과 효율적인 모델 구축에 초점을 맞춘다.**

**\***좋아하는 미술품을 고르고, 딥러닝 네트워크를 이용하여 선, 색 및 음영을 학습시킨다. 네트워크의 새로운 이미지를 입력시키면, 네트워크는 이미지를 당신이 좋아하는 화풍과 유사하게 만들어 낸다.

****

**3. 테스트 계획**

- 단위 및 시스템 테스트



**<사진 모듈>**

-사진이 잘 찍히나 테스트

-사진이 앨범에서 잘 불러와지는지 테스트

-사진을 등록 시 앱을 종료하였다가 켜도 앱 내에 남아있는지 테스트

-완성된 사진을 저장 시 핸드폰 앨범에 잘 저장되는지 테스트

**<사진합성>**

-사진내에 인물과 물체를 잘 구별하는지 테스트

-사진 합성 시 잘 합성되었는지 테스트

-부분 합성 시 부분이 잘 나뉘어 합성되었는지 테스트

**<기능>**

-채도 변경 시 잘 변경되는지 테스트

-대비 변경 시 잘 변경되는지 테스트

-RGB추천 시스템 적절히 변경되는지 테스트

**<버그 트래킹>**

-mantis

mantis란? 무료 오픈 소스 웹 기반 버그 추적 시스템이다. 설치가 비교적 간단하며, 환경 설정을 위한 접근 방식이 2가지만 있기에 접근 계정 등 여러 설정 값을 정의하기가 쉽다. 또한 문제 추적 시스템, 프로젝트 관리 도구로 편리하게 사용이 가능하다.

**4. 문서화 계획**

Read Me에 사용자가이드와 개발자가이드를 작성한다.

예시 )

|  |
| --- |
| **Read Me**  **I’mage = 예술적 스타일의 이미지 재창조 구현 (제목)**  이것은 “예술적 스타일의 이미지 재창조” 를 구현 한 것이다. image transfer 오픈소스에 Open CV를 도입하고 수정 및 개작하였다. 사진 촬영이나 저장 된 사진을 불러와서 OpenCV를 통해 사진을 조각화 하여 다른 사진과 합성하거나 내부적으로 변환시켜 새로운 이미지로 재창조한다. (어플 기본 설명)  (원본 이미지와 재창조된 이미지를 캡쳐하여 이 위치에 넣는다.)  **Authors**  (제작 정보 안내 : 실행 하는데 필수적으로 사용 되거나 특수한 변수와 클래스를 기재하고 설명한다.)  **method of use** (단계마다 어플 캡쳐하여 사진을 넣는다.)  ( 어플 완성 후 전/후 사진과 동영상을 넣는다.   1. 조각화 된 구역 중 한 구역 선택하는 사진 2. 변환 선택 내역 중 선택하는 사진 3. 변환 된 사진 4. 1~3 까지 반복한 사진 5. 과정이 실행되는 동영상 )   I’Mage 앱에 접속  사진촬영 후 저장 과 사진 불러오기 중 선택  선택한 사진에서 조각화 된 부분 중 변경 할 영역 선택 (자율 선택 가능)  변경 할 영역의 스타일 및 색상 선택  변환 버튼 클릭  변환 된 사진 저장  **Bulid**  (빌드 안내 : 빌드하는데 필요한 설치 환경과 버전을 명시한다.)  **Copying / License**  (저작권 및 사용권 정보를 명시한다) |

**5. 코딩 스타일 가이드 라인**

**표기법 정리**

파스칼 표기법 : 첫 단어를 대문자로 시작. (ImageTransfer)

카멜 표기법 : 맨 처음 문자를 소문자로 표기함. 각 단어의 첫 문자는 대문자로 표기. (imageTransfer)

스네이크 표기법 : 단어를 밑줄문자로 표기. (Image\_transfer)

**변수**

스네이크 표기법을 사용한다. 모든 문자는 소문자로 표기한다.

|  |  |
| --- | --- |
| 좋은 예 | image\_transfer |
| 나쁜 예 | Image\_Transfer |

변수의 이름은 해당 변수의 기능에 대해서 사용한다.

a, I, x 같은 변수는 사용하지 않는다.

기능이 비슷한 변수들은 이름이 정확하게 구분되게 짓는다.

|  |  |
| --- | --- |
| 좋은 예 | test\_print  test\_loop |
| 나쁜 예 | test1  test2 |

단, 다음과 같은 단순한 loop에 사용되는 변수는 제외한다.

for i in test\_list:

print(i)

**모듈**

전체를 소문자로 표기한다.

모듈은 콤마로 연결하지 않는다.

|  |  |
| --- | --- |
| 좋은 예 | import scipy.io  import os |
| 나쁜 예 | import Scipy.io, os |

모듈 상수는 대문자로 표기한다.

**클래스**

여러개의 단어가 있다면 두 번째 이후부터는 단어의 첫 글자만 대문자로 표기한다.

첫 번째 단어의 첫 번째 글자는 소문자이다. (카멜 표기법)

|  |  |
| --- | --- |
| 좋은 예 | class imageTransfer(): |
| 나쁜 예 | class Imagetransfer():  class Image\_Transfer(): |

클래스의 이름은 그 기능을 명확하게 알 수 있는 용어로 정한다.

클래스의 protected instance attribute는 하나의 밑줄로 시작한다.

클래스의 private instance attribute는 2개의 밑줄로 시작한다.

클래스의 public attribute는 밑줄로 시작하지 말아야 한다.

**함수**

함수의 이름은 그 기능을 명확하게 알 수 있는 용어로 정한다.

함수가 시작되기 전에 “”“ ”“” (주석)을 사용하여 간략하게 함수의 기능을 기술한다.

단, 들여쓰기 레벨에 맞아야 함.

함수명은 스네이크 표기법으로 단어의 사이는 \_를 사용한다.

|  |  |
| --- | --- |
| 좋은 예 | def calc\_volume(){  “”“  부피를 계산하는 함수  반환 값 : 부피  “”“  ...  } |
| 나쁜 예 | def calc\_volume(){  ...  } |

**if, for, while 블럭 문장**

블록이 2문장 이상일 때 하나의 줄로 작성하지 않는다.

**들여쓰기**

공백(space)으로 한다. tab 과 혼용해서 사용하지 않는다.

문법적으로 들여 쓸 때는 4번으로 한다.

함수나 클래스에서는 2번을 들여 쓴다.

변수 할당시 할당자 앞뒤로 하나의 공백만 넣는다.

|  |  |
| --- | --- |
| 좋은 예 | test = test + 100 |
| 나쁜 예 | test=test+100 |