

[Data트랙] 월말평가1 – Sample



| Background

- ✓ 자료구조에 대한 이해와 사용
- ✓ 알고리즘 문제 해결 능력 향상
- ✓ 파이썬 기초 문법에 대한 이해

| Goal

- ✓ 파이썬을 사용하여 알고리즘 문제를 해결 할 수 있다.

| 환경 설정

1) 파일 이름 및 제출 방법

- 1, 2번 문제에 대한 소스 파일은 python문제번호_지역_반_이름.py로 만든다.
- 3번은 텍스트 파일로 작성한다.

python1_대전_4반_김싸피.py

python2_대전_4반_김싸피.md

- 위 3개의 파일만 지역_반_이름.zip으로 압축하여 제출한다.

대전_4반_김싸피.zip

(탐색기에서 파일 선택 후 오른쪽 클릭 – 보내기 – 압축(zip)폴더 선택)

3) 채점

- 주석이 없는 경우, 주석이 코드 내용과 맞지 않는 경우, 지정된 출력 형식을 만족하지 않는 경우 해당 문제는 0점 처리될 수 있다.
- import를 사용한 경우 해당 문제는 0점 처리될 수 있다. (import sys도 예외 없음)

4) 테스트케이스는 부분적으로 제공되며, 전체가 공개되지는 않는다.

5) 각 문제의 배점이 다르므로 표기된 배점을 반드시 확인한다.

- 샘플 평가이므로 배점이 표기되는 위치만 공유드립니다.
- 실제 평가에서는 배점 반드시 확인할 것
- ## 성실과 신뢰로 테스트에 임할 것 (부정 행위시 강력 조치 및 근거가 남음)

※ 소스코드 유사도 판단 프로그램 기준 부정 행위로 판단될 시,

0점 처리 및 학사 기준에 의거 조치 실시 예정



[Data트랙] 월말평가1 – Sample

| 문제 1 : 텍스트 전처리기 (배점)

고객 정보가 담긴 데이터베이스(딕셔너리 리스트)를 정제하려고 합니다. 각 함수는 아래 설명된 기능을 정확히 수행해야 하며, 최종적으로 `process_customers()` 함수를 실행하면 유효한 고객 정보만 정제된 형태로 리스트에 담아 반환되어야 합니다.

스켈레톤 코드 및 요구 조건

```
def clean_name(name):
    """
    이름의 앞뒤 공백을 제거하고, 첫 글자만 대문자로 바꿉니다.
    (예: "  heLLo " -> "Hello")
    """
    pass

def make_greeting(name):
    """
    정리된 이름을 받아 "안녕하세요, [이름]님!" 형식의 문자열로 만듭니다.
    (예: "홍길동" -> "안녕하세요, 홍길동님!")
    """
    pass

def process_namelist(name_list):
    """
    전체 이름 리스트를 받아, 비어있지 않은 이름만 골라 인사말로 만들어
    리스트로 반환합니다.
    (이름이 공백으로만 이뤄진 경우는 무시합니다.)
    위에 작성한 함수들을 적절히 활용해야 합니다.
    """
    pass

# -----
# 아래 코드는 절대 수정하지 마시오.
# -----
raw_names = [
    "  홍길동",
    "김싸피 ",
    "  ",
    "lee sunsin"
]

result = process_namelist(raw_names)
print(result)
```

출력 예시

```
['안녕하세요, 홍길동님!', '안녕하세요, 김싸피님!', '안녕하세요, Lee sunsin님!']
```

[Data트랙] 월말평가1 – Sample



| 문제 2 : 변수의 스코프 (배점)

다음 코드의 출력 결과를 예상하고, 그 이유를 '변수의 스코프 (Scope)' 개념을 중심으로 설명하시오.

```
x = 100

def my_function():
    x = 200
    print(f"함수 안: {x}")

my_function()
print(f"함수 밖: {x}")
```