

Computing Foundations for Data Science

HW 5

제출기한: 2021/09/28 10:59AM

주의사항

- 코드를 Jupyter Notebook에서 작성하였더라도 python 파일(.py)로 제출할 것.
- 함수가 의도한 값을 return 하는지를 확인할 것. (print 와 혼동하지 말 것)
- 파일명은 P1.py ~ P5.py 를 유지하고, 해당 파일들을 HW5_학번_이름.zip 으로 압축하여 제출할 것. 예를 들면 학번이 2020-12345 이고, 이름이 Keondo Park 이라면 HW5_2020_12345_KeondoPark.zip 으로 압축하여 제출.
- 예시로 제시한 입력 값 외에도 조교가 임의로 생성한 입력 값으로도 코드가 잘 실행되는지 테스트할 예정.
- 뼈대 코드의 함수 이름 및 매개변수(parameter)는 변경하지 말 것.

문제 1.

alkaline_metals.txt 파일은 다음과 같이 구성되어 있다. 각 행은 이름, 원자번호, 원자량으로 구성되어 있다.

```
beryllium 4 9.012
magnesium 12 24.305
calcium 20 20.078
strontium 38 87.62
barium 56 137.327
radium 88 226
```

alkaline_metals.txt의 파일을 불러오고, **list of lists** 형태로 파일의 내용을 **return**하는 함수 P1을 작성하여라. 단, 아래 조건을 만족한다.

(1) 각 Inner list는 각 행을 [이름, 원자번호, 원자량] 형태로 저장한다.

(2) (1)에서 만든 모든 리스트를 원소로 갖는 list of lists가 P1 함수의 최종 **return**값이 된다. (아래 그림 참고)

```
[['beryllium', '4', '9.012'],
 ['magnesium', '12', '24.305'],
 ['calcium', '20', '20.078'],
 ['strontium', '38', '87.62'],
 ['barium', '56', '137.327'],
 ['radium', '88', '226']]
```

문제 2.

alkaline_metals.txt 파일의 내용을 다음과 같은 형태로 변형하여 새로운 텍스트 파일인 **output.txt**에 저장하는 P2 함수를 작성하여라.

```
beryllium's atomic number is 4 and atomic weight is 9.012
magnesium's atomic number is 12 and atomic weight is 24.305
calcium's atomic number is 20 and atomic weight is 20.078
strontium's atomic number is 38 and atomic weight is 87.62
barium's atomic number is 56 and atomic weight is 137.327
radium's atomic number is 88 and atomic weight is 226
```

※ P2 함수를 통해 새로운 텍스트 파일이 생성되어야 한다. (alkaline_metals.txt를 변경하는 것이 아님)

※ output.txt는 alkaline_metals.txt와 같은 경로에 저장할 것.

(채점 시, parameter로 'alkaline_metals.txt', 'output.txt'를 지정할 예정이며, 경로 문제로 제대로 실행 혹은 저장되지 않을 시 감점 사유임)

※ 출력 파일에 작성되는 내용을 print하거나 return하지 않는다.

※ 띄어쓰기 및 특수문자 사용에 유의할 것

문제 3.

임의의 txt파일이 주어질 때, 파일의 마지막 줄부터 첫번째 줄까지 차례로 읽어 리스트의 원소로 저장 후 해당 리스트를 **return**하는 P3함수를 작성하여라. 즉, **return**되는 리스트에는 각 행이 파일과 반대 순서로 저장된다.

만약 문제 1의 alkaline_metals.txt 파일이 주어진다면, **P3 함수의 return**은 다음과 같다.

```
['radium 88 226', 'barium 56 137.327\n', 'strontium 38 87.62\n', 'calcium 20 20.078\n', 'magnesium 12 24.305\n', 'beryllium 4 9.012\n']
```

문제 4.

임의의 txt파일이 주어질 때, 파일을 불러온 뒤 다음 조건에 맞는 list를 **return**하는 P4 함수를 작성하여야.

- (1) 각 행을 읽은 순서대로 하나의 list에 추가한다.
- (2) 단, '#'나 '/'로 시작하는 행은 주석으로 취급하여 추가하지 않는다.
- (3) P3 함수는 최종 list를 **return**한다.

만약 파일이 다음과 같이 주어진다면,

```
## alkaline
// alkaline
alkaline
// alkaline
alkaline
```

P4 함수의 return은 다음과 같다.

```
['alkaline\n', 'alkaline']
```

문제 5.

임의의 txt 파일이 주어질 때, 해당 파일에서 **가장 긴 단어의 길이(int)**를 **return**하는 P5 함수를 작성하여야.

만약 다음과 같은 파일이 주어질 경우, P5 함수의 return은 9이다. (beryllium, magnesium, strontium이 모두 9자로 가장 길다.)

```
beryllium magnesium
calcium strontium barium
radium calcium
calcium
```

※ 단어는 공백(' ') 또는 개행문자(\n)로 구분된다.

※ 파일에는 적어도 한 개의 단어가 있으며, 숫자는 존재하지 않는다. (한글 혹은 영어만 존재함)