



Python Study Week5 Solve

문제



ProMIS 상사에 인턴으로 첫 출근을 한 상호는 1층에서 엘리베이터를 탔다.
(*엘리베이터 안에서 상호의 자리는 랜덤으로 설정한다.)

현재 엘리베이터에는 인턴 6명이 이미 탑승해 있는 상황이다.

오늘은 인턴들에게 따로 정해진 사무실이 없기에, 총 8층까지 있는 이 건물에 2층부터 1명씩 내려서 근무를 하기로 결정했다.

엘리베이터 문쪽과 가까운 사람부터 순서대로 내리게 되는데, 만약 상호가 내리게 된다면 시스템을 종료한다!

진행 방식 및 조건

1. `List` 자료구조를 사용합니다.

- 아래의 리스트를 사용하며, **오른쪽부터 차례대로 하차**합니다. (현재 기준 2층에서 후동 하차)

```
elevator = ['시운', '다현', '현제', '유승', '혜원', '후동']
```

2. `random` 으로 숫자 1개를 배정하여 상호를 해당하는 엘리베이터 위치에 넣어줍니다.

- 아래의 변수를 사용합니다.
- 만약 location이 1이 나온다면,

`elevator = ['시운', '상호', '다현', '현제', '유승', '혜원', '후동']` 이런식으로 되겠죠 !

```
location = random.randint(0,7)
```

3. `for` 반복문 or `while` 반복문을 사용하여 한 명씩 하차합니다.

- 하차할 때마다, **누가 몇층에서 하차했는지**를 출력하고, **현재 엘리베이터에 남아 있는 사람**도 출력해줍니다.
 - Ex) xxx님이 x층에서 하차하셨습니다.

현재 엘리베이터: ['시운', '상호', '다현', '현제', '유승', '혜원']

- 만약 **상호가 하차를 했다면**,

상호님이 무사히 출근을 완료하였습니다! 를 출력하고 시스템을 종료합니다.



개인적으로는 for 반복문을 사용해서, 총수를 자동으로 받도록 코드를 짜보시는 걸 추천드립니다! 😊

학습 목표

1. `random` 모듈
2. `List` 자료구조
3. `for` or `while` 반복문
4. `insert` , `pop` 메소드
5. `if` 조건문
6. `break` 문



코드를 작성하는 여러 가지 방법이 있겠지만, `List` 와 `for` 반복문, 그리고 `pop` 메소드는 아주 친한 사이니까 이번 기회로 연습해보시면 좋을 것 같아요 😊

출력 예시

```
후동님이 2층에서 하차하셨습니다.
현재 엘리베이터: ['시운', '상호', '다현', '현제', '유승', '혜원']
혜원님이 3층에서 하차하셨습니다.
현재 엘리베이터: ['시운', '상호', '다현', '현제', '유승']
유승님이 4층에서 하차하셨습니다.
현재 엘리베이터: ['시운', '상호', '다현', '현제']
현제님이 5층에서 하차하셨습니다.
현재 엘리베이터: ['시운', '상호', '다현']
다현님이 6층에서 하차하셨습니다.
현재 엘리베이터: ['시운', '상호']
상호님이 7층에서 하차하셨습니다.
현재 엘리베이터: ['시운']
상호님이 무사히 출근을 완료하였습니다!
```

해설

```
import random

elevator = ['시운', '다현', '현제', '유승', '혜원', '후동'] # 이미 탑승해 있는 인턴 6명

location = random.randint(0,7) # 들어갈 위치를 설정
elevator.insert(location, '상호') # 해당 위치에 넣어줌

for floor in range(2, len(elevator)+2): # 2층부터 8층까지 반복 설정
    people = elevator.pop() # pop()을 사용하여 하차할 사람을 리스트에서 꺼냄
    print(f'{people}님이 {floor}층에서 하차하셨습니다.')
    print(f'현재 엘리베이터: {elevator}')

    if people == '상호': # 상호가 하차했을 경우
        print('상호님이 무사히 출근을 완료하였습니다!')
        break
```

1. `insert()` 메소드를 사용하여, `location`에 할당된 숫자의 `index`에 `'상호'` 를 넣어줍니다.
2. `for` 반복문의 범위를 `range(2, len(elevator)+2)` 로 설정하여 2층부터 8층까지 차례대로 받을 수 있도록 합니다.
 - a. * 현재 엘리베이터의 길이가 7이므로, +2를 해주어야 2부터 8까지 받을 수 있음

```
for i in range(len(elevator)):
    people = elevator.pop()
    print(f'{people}님이 {i+2}층에서 하차하셨습니다.')
    print(f'현재 엘리베이터: {elevator}')
```



위처럼 짜주어도 상관은 없겠죠!

3. `pop()` 메소드를 사용해서 `List` 에서 한 사람을 꺼냄과 동시에 `people` 변수에 넣어줍니다.
4. 만약 `people` 이 `'상호'` 일 경우에는, 시스템을 종료합니다.