

[파이썬 트랙] 1회차 월말평가 - Python



| Background

- ✓ Python

| Goal

- ✓ 프로그래밍 기본 문법에 대한 이해

| 환경 설정

- 1) 개발환경: visual studio code(이하 vscode)를 이용하여 코드를 작성한다.
- 2) 코드실행: vscode의 터미널창에서 python명령어를 통해 코드를 실행한다.
- 3) 코드제출: 다음 항목들을 참고하여 제출한다.

| 유의 사항

- 1) 성실하게 테스트에 임할 것 (부정 행위시 강력 조치 및 근거가 남음)
- 2) 소스코드 유사도 판단 프로그램 기준 부정행위로 판단될 시, 0점 처리 및 학사 기준에 의거 조치 실시 예정
- 3) 채점을 위한 별도의 데이터가 있음 (즉, 테스트 케이스와는 별도로 채점 케이스가 존재)

| 코드 작성

최종 제출 코드가 다음 항목에 해당하는 경우, 감점 혹은 0점 처리 될 수 있음

- 1) Syntax Error로 인한 코드 실행이 불가능한 경우
- 2) 주석 설명이 미흡한 경우
- 3) 출력 결과에 정답과 무관하거나 불필요한 출력이 있는 경우
- 4) 코드 수정 및 추가가 금지된 영역에 코드가 작성된 경우

[파이썬 트랙] 1회차 월말평가 - Python



| 코드 실행

vscode의 터미널창에서 python 실행 명령어로 결과를 확인.

```
$ python problem01_01.py
```

| 제출 안내

제출 안내 미 준수 시, 감점 혹은 0점 처리 될 수 있음

1) 압축 및 제출 파일 이름

- 지역0반_홍길동

ex) 서울1반_홍길동 / 부울경2반_김싸피

2) 압축 폴더 구조

- 시험을 진행했던 폴더 구조 그대로 압축하여 제출을 진행

```
서울1반_홍길동/  
    problem01_01.py  
    problem01_02.py  
    problem01_data.json  
    ...
```

제출 마감시간에 서버 요청이 집중될 수 있으므로, 미리 제출하는 것을 권장.

(마감 시간 이후 제출 불가)

[파이썬 트랙] 1회차 월말평가 - Python



| Problem 01 & 02

- ❖ 김싸피는 시험점수들의 평균을 계산하는 코드를 작성하려고 한다.
미리 수집한 과목별 점수는 파이썬의 리스트(list) 자료형으로 함수에 전달된다.
다음의 예시를 참고하여, 아래의 문제를 해결하시오.

- 예시

index	정보
0	python 점수 (ex. 80)
1	html 점수 (ex. 90)
2	javascript 점수 (ex. 85)
3	project 점수 (ex. 75)

- **problem01.py (15점)**

전체 점수의 합을 계산하는 함수 total을 완성하시오.

- **problem02.py (15점)**

전체 점수의 평균을 계산하는 함수 average를 완성하시오.

[파이썬 트랙] 1회차 월말평가 - Python



| Problem 03 & 04 & 05

- ❖ 정싸피는 이번 프로젝트에서 회원가입 유효성 검사팀에 들어가게 되었다. 유효성 검사팀은 회원가입 과정에서 사용자가 입력한 정보를 검사하는 프로그램을 만드는 팀이다. 팀 회의를 통해 정싸피가 맡아서 개발해야 하는 기능들이 정해졌다고 할 때, 아래의 문제들을 해결하여 정싸피를 도와주도록 하자.

- **problem03.py (10점)**

아이디는 회원가입 과정에서 필수로 입력해야 하는 정보이다.

get_user_id함수에는 파이썬의 딕셔너리(dictionary) 형태로 신규 생성을 목표하는 정보(user_data)가 전달인자로 주어진다.

입력된 정보에서 아이디(id)를 문자열형태로 반환하는 get_user_id 함수를 완성하시오

.

- **problem04.py (10점)**

새롭게 생성하려는 아이디는 기존 유저들의 아이디와 중복되면 안된다.

check_duplicate_id 함수에는 신규 생성을 목표하는 아이디와 기존 유저들의 아이디 정보가 들어있는 1차원 리스트가 전달인자로 주어진다.

만약, 신규 생성 아이디가 기존 아이디와 중복된다면 True, 아니라면 False를 반환하는 check_duplicate_id 함수를 완성하시오.

- **problem05.py (10점)**

비밀번호는 최소 8자이며, 최대 32자이다.

check_password_length 함수에는 신규 생성을 목표하는 계정의 비밀번호가 전달인자로 주어진다.

입력된 정보가 해당 범위를 만족한다면 True, 아니라면 False를 반환하는 check_password_length 함수를 완성하시오.

[파이썬 트랙] 1회차 월말평가 - Python



| Problem 06

- ❖ 아스키코드는 미국 ANSI에서 표준화한 정보교환용 부호체계이다. 총 128개의 부호가 사용되며, 다음 페이지의 표는 알파벳이 포함된 65~122까지의 아스키 코드의 일부를 나타낸 것이다. python 에서는 `ord(char)`을 이용하여 해당 문자에 대응하는 10진수 값을 확인할 수 있다. 반대의 경우 `chr(int)`를 사용한다.

```
ord('A') # => 65
ord('z') # => 122
chr(65)   # => 'A'
chr(122)  # => 'z'
```

- **problem06.py (5점)**

문장의 각 알파벳을 일정한 양의 정수 n 만큼 밀어서 다른 알파벳으로 바꾸는 암호화 방식을 시저 암호라고 한다. 문자열 `word`를 양의정수 n 만큼 밀어 완성된 시저 암호를 반환하는 함수 `caesar`를 제공된 아스키 코드표를 참고하여 완성하시오.
(단, 소문자는 소문자로 대문자는 대문자로 암호화한다.)

[파이썬 트랙] 1회차 월말평가 - Python



| Problem 06

10진수	Symbol	10진수	Symbol
65	A	97	a
66	B	98	b
67	C	99	c
68	D	100	d
69	E	101	e
70	F	102	f
71	G	103	g
72	H	104	h
73	I	105	i
74	J	106	j
75	K	107	k
76	L	108	l
77	M	109	m
78	N	110	n
79	O	111	o
80	P	112	p
81	Q	113	q
82	R	114	r
83	S	115	s
84	T	116	t
85	U	117	u
86	V	118	v
87	W	119	w
88	X	120	x
89	Y	121	y
90	Z	122	z

[파이썬 트랙] 1회차 월말평가 - Python



| Problem 07

- ❖ 이진수는 0,1로 모든 수를 표현하는 방식이다. 일상생활에서 사용하는 십진수 숫자를 이진수로 변경하기 위해서는 2로 나눈 몫을 2로 나누는 과정을 반복하며 나오는 나머지를 사용한다.

$$\begin{array}{r}
 2 \overline{) 55} \\
 2 \overline{) 27 \cdots 1} \\
 2 \overline{) 13 \cdots 1} \\
 2 \overline{) 6 \cdots 1} \\
 2 \overline{) 3 \cdots 0} \\
 1 \cdots 1
 \end{array}$$

$55 = 110111_2$

- problem07.py (5점)

십진수 숫자 n 을 받아 이진수로 변환하여 문자열로 반환하는 함수 `dec_to_bin`을 완성하시오.
(단, 반드시 재귀를 이용하여 구현한다.)

[파이썬 트랙] 1회차 월말평가 - Python



| Problem 08

- ❖ 강싸피는 엄청난 노력 끝에 어느 게임 개발 회사의 인턴으로 합격하여 근무하게 되었다. 첫 출근날 팀장은 새로 들어온 인턴들에게 과제를 몇 가지 부여했는데, 강싸피가 부여받은 과제들은 아래와 같다. 각 파일의 입출력 예시를 참고하여 요구사항에 맞게 각 문제를 해결하시오.

- **problem08.py (10점)**

캐릭터는 게임상에서 아이템을 인벤토리라는 공간에 저장할 수 있다. 인벤토리는 파이썬 프로그램의 리스트 형태로 관리된다.

인벤토리에서 특정 종류의 아이템의 목록만 반환하는 필터 함수를 만들려고 할 때, 다음과 같은 조건을 만족하도록 함수를 완성하시오.

1. 함수의 첫번째 인자는 찾고자 하는 아이템의 종류(type)가 문자열 데이터로 주어진다.
2. 함수의 두번째 인자는 캐릭터의 현재 인벤토리 정보가 리스트로 주어진다.
3. 인벤토리 리스트에는 아이템들의 정보들이 각각 딕셔너리 데이터(하단의 표 참조)로 저장되어 있다.
4. 찾고자 하는 종류(type)에 해당하는 아이템(딕셔너리)들로만 구성된 새로운 리스트를 반환한다.

key	value
id	아이템의 고유 번호
name	아이템의 이름
type	아이템의 종류
grade	아이템의 등급

개별 아이템 정보(딕셔너리)

- 입력 예시

```
item_type='weapon'
inventory=[
    {'id': 1, 'name': 'Elder Wand', 'type': 'weapon', 'grade': 'legend'},
    {'id': 2, 'name': 'Healing Potion', 'type': 'potion', 'grade': 'normal'},
    {'id': 3, 'name': 'Steel Helmet', 'type': 'armor', 'grade': 'rare'},
    {'id': 4, 'name': 'Long Sword', 'type': 'weapon', 'grade': 'normal'},
]
```

- 출력 예시

```
[
    {'id': 1, 'name': 'Elder Wand', 'type': 'weapon', 'grade': 'legend'},
    {'id': 4, 'name': 'Long Sword', 'type': 'weapon', 'grade': 'normal'},
]
```


[파이썬 트랙] 1회차 월말평가 - Python



| Problem 09

• problem09.py (10점)

게임 캐릭터가 움직일 수 있는 범위가 있으며, 이 제한된 구역을 넘어가지 않도록 검사하는 함수를 만들려고 한다. 캐릭터는 2차원 평면 ($N * N$)에서 이동하며, 사용자의 키 입력에 따라 위, 아래, 왼쪽, 오른쪽으로 움직일 수 있다.

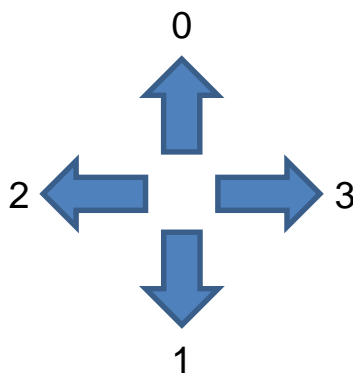
캐릭터의 현재 위치는 튜플(x, y) 형태로 주어지며, x와 y는 각각 2차원 평면에서의 행과 열을 의미한다. ($0 \leq x < 100$, $0 \leq y < 100$)

최대 범위는 숫자 N으로 주어진다. ($0 < N \leq 100$)

키 입력은 0부터 3까지의 숫자 M으로 주어지며, 각각 위, 아래, 왼쪽, 오른쪽을 의미한다. 만약, 키 입력의 결과로 최대 범위를 벗어난다면 False, 그렇지 않으면 True를 반환하는 함수 is_position_safe를 완성하시오.

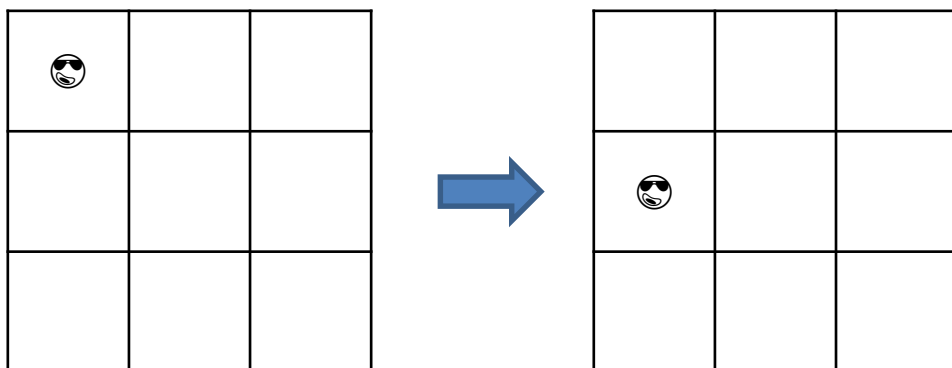
(0, 0)	(0, 1)	(0, 2)
(1, 0)	(1, 1)	(1, 2)
(2, 0)	(2, 1)	(2, 2)

좌표 예시



방향 예시

- 입력 예시 : N=3, M=1, position=(0, 0)
- 결과 예시 : True



[파이썬 트랙] 1회차 월말평가 - Python



| Problem 10

• problem10.py (10점)

캐릭터는 2차원 평면($N \times N$)에서 이동하며, 사용자의 키 입력에 따라 위, 아래, 왼쪽, 오른쪽으로 움직일 수 있다.

현재 2차원 평면은 0과 1로 이루어진 2차원 리스트(mat)로 주어진다.

캐릭터의 현재 위치는 숫자 1로 표현되며 필드에서의 캐릭터는 단 하나이다.

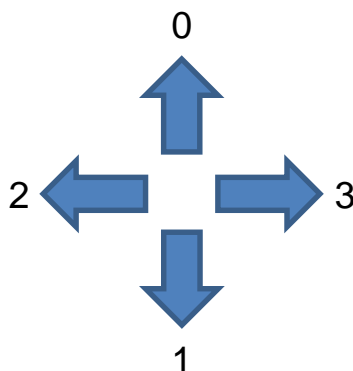
또한 캐릭터는 0으로 표시된 모든 영역으로 이동 가능하다. 키 입력은 0부터 3까지의 숫자로 이루어진 일차원 리스트(moves)로 주어지며, 숫자 0부터 3까지 각각 위, 아래, 왼쪽, 오른쪽을 의미한다.

사용자의 키 입력이 주어진다고 할 때, 캐릭터의 최종 위치인 x, y 를 리스트 형태로 출력하는 `get_final_position`을 완성하시오.

(단, x 와 y 는 각각 행과 열을 의미하며, 캐릭터가 범위 밖을 넘어간다면 `None`을 리턴한다.)

(0, 0)	(0, 1)	(0, 2)
(1, 0)	(1, 1)	(1, 2)
(2, 0)	(2, 1)	(2, 2)

좌표 예시



방향 예시

- 입력 예시 : $N=3$, `mat=[[1, 0, 0], [0, 0, 0], [0, 0, 0]]`, `moves=[1, 1, 3]`
- 결과 예시 : `[2, 1]`

