

## 파이썬 실습문제 2장-1

### Sec 21. 문자열 연산

1. 문자열 연산으로 \*표를 50개 출력함
2. str1에 '내 이름은 ' 저장, str2에 '홍길동입니다.' 저장
3. str1과 str2와 ' 학번은 12345678입니다.'을 덧셈연산으로 연결하여 print 구문으로 출력
4. 다시 \*표 50개 출력
5. 출력형식이 다음과 같게 코딩함 (이름과 학번은 본인으로 대체 가능)

\*\*\*\*\*

내 이름은 홍길동입니다. 학번은 12345678 입니다.

\*\*\*\*\*

### Sec 22. 문자열 인덱싱

1. 위 문제의 str1에서 인덱스 0, 2, 4 항목을 프린트함.
2. 인덱스 7번 항목을 프린트 하여 에러메시지를 확인함.
3. str2에서 인덱스 -1, -2, -3, -7 항목을 프린트함.
4. 결과는 다음과 같음

내  
이  
은

Traceback (most recent call last):

```
File "/Users/jwbae/{...중간생략 ...} /ch2/section_021~024.py", line 16, in <module>
    print(str1[7])
```

IndexError: string index out of range

.  
다  
니  
홍

### Sec 23~24. 문자열 슬라이싱

1. str1에서 '이름'만 추출하여 프린트 하도록 문자열을 슬라이싱함
2. str1에서 '내 이름'이 추출되도록 슬라이싱 하여 출력함. 슬라이싱을 맨처음부터 시작하게 지시함.
3. 위의 내용들을 역방향 인덱싱으로 다시 해보도록 함.
4. str2에서 '입니다.'을 추출하도록 역방향 슬라이싱을 실행함. 맨끝까지 슬라이싱이 되도록 지시함.
5. str1의 0번 인덱스 위치의 값을 "네"로 저장하는 시도를 했을 때 오류 내용을 확인함.
6. 결과출력은 다음과 같음.

이름  
내 이름  
이름

내 이름  
입니다.

Traceback (most recent call last):

```
File "/Users/jwbae/{...중간생략 ...} /ch2/section_021~024.py", line 36, in <module>  
    str1[0] = "네"
```

TypeError: 'str' object does not support item assignment

## Sec 25. 문자열 관련 메서드

1. str1에 "Hello, my princess. Are you hungry?" 라고 저장함. 문자열 메서드를 써서 다 음을 실행함.
2. str1을 모두 대문자로 바꾼 문자열을 출력함
3. str1을 모두 소문자로 바꾼 문자열을 출력함
4. str1을 각 단어 첫글자만 대문자로 바꾼 문자열을 출력함
5. str1에서 'r'의 개수를 출력함
6. str1이 '?'로 끝나는 것이 맞는지 bool 타입으로 출력함
7. str1이 'h'로 시작하는 것이 맞는지 bool 타입으로 출력함
8. str1을 모두 소문자로 바꾸고 그것이 'h'로 시작하는지 bool 타입으로 출력함
9. str1을 split()해서 str2에 저장하고 프린트함.
10. str1을 ','(쉼표) 기준으로 스플릿해서 str2에 저장하고 출력함.
11. str1을 '.'(마침표) 기준으로 스플릿해서 str2에 저장하고 출력함.
12. str1의 길이를 출력함
13. str1을 50개로 늘이면서 앞부분을 0으로 채운 내용을 출력함
14. 출력내용은 다음과 같음

HELLO, MY PRINCESS. ARE YOU HUNGRY?

hello, my princess. are you hungry?

Hello, My Princess. Are You Hungry?

3

True

False

True

['Hello,', 'my', 'princess.', 'Are', 'you', 'hungry?']

['Hello', ' my princess. Are you hungry?']

['Hello, my princess', ' Are you hungry?']

35

0000000000000000Hello, my princess. Are you hungry?