

# 월말평가

## 평가 안내

- 제공된 파이썬 파일에 답안 작성 이후, 본인 이름으로 폴더 이름을 변경 후 압축하여 제출합니다.
  - 예) 이름이 홍길동인 경우 `홍길동.zip` 입니다.
  - 압축을 풀었을 때, 폴더 구조는 아래와 같습니다.
    - 홍길동/
      - 01.py
      - 02.py
      - 03\_01.py
      - 03\_02.py
      - 04.py

## 문제 1. 나만의 단어장 (20점) - 01.py

파이썬 클래스를 활용하여 나만의 단어장을 만드시오.

- `Word` 클래스에 대한 명세는 다음과 같습니다.

인스턴스가 생성 될 때, `wordbook` 이라는 비어있는 dictionary가 인스턴스 변수로 만들어집니다.

인스턴스변수	타입
wordbook	Dict

메소드 명	인 자	반환값 (return)	비고
add	영문, 한글	없음	영문, 한글을 wordbook 딕셔너리에 각각 키(key)와 값(value)로 추가됩니다.
delete	영문	True/False	1) wordbook 딕셔너리 key에 해당 영문이 없으면, False가 반환됩니다. 2) 해당 영문이 있으면, wordbook 딕셔너리에서 삭제하고 True가 반환된다.
print	없음	없음	wordbook 딕셔너리를 아래 출력 예시에 맞춰 출력합니다.

# 아래의 코드를 실행하였을 때 아래와 같이 출력되어야 합니다.

```
my_book = Word()
my_book.add('apple', '사과')
my_book.add('banana', '바나나')
my_book.add('cherry', '체리')
my_book.add('durian', '두리안')
my_book.print()
print(my_book.delete('cherry'))
print(my_book.delete('durian'))
print(my_book.delete('egg'))
my_book.print()
```

```
apple: 사과
banana: 바나나
cherry: 체리
durian: 두리안
True
True
False
apple: 사과
banana: 바나나
```

## 문제 2. 도형 만들기 (30점) - 02.py

파이썬 클래스를 활용하여 꼭지점과 사각형을 표현하시오.

- `Point` 클래스에 대한 명세는 다음과 같습니다.

인스턴스가 생성될 때 x좌표 값과 y좌표 값이 생성됩니다. 각각 인스턴스 변수명은 `x`, `y` 입니다.

인스턴스변수	타입
x	int
y	int

```
# 예) 좌표 (3, 5)의 점
p1 = Point(3, 5)
```

생성자를 제외한 메소드는 없습니다.

- `Rectangle` 클래스에 대한 명세는 다음과 같습니다.

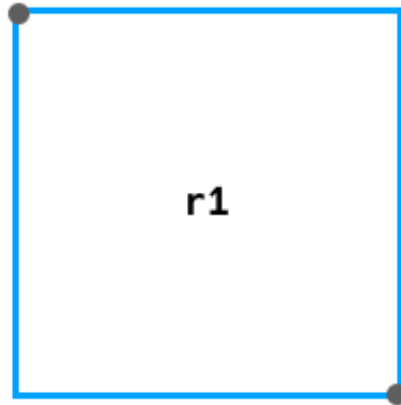
인스턴스가 생성시 좌측상단좌표와 우측하단좌표가 생성됩니다. 각각 인스턴스 변수명은 `p1`, `p2` 입니다.

인스턴스변수	타입
p1	Point 클래스 인스턴스
p2	Point 클래스 인스턴스

메소드명	인 자	반환값 (return)	비고
get_area	없 음	면적	면적을 계산하여 반환합니다.
get_perimeter	없 음	둘레값	둘레를 계산하여 반환합니다.
is_square	없 음	True/False	정사각형인지 아닌지를 판단하여 True/False를 반환합니 다.

아래의 예시의 r1은 다음과 같이 구성됩니다. (다음장)

**p1 (1, 3)**



**p2 (3, 1)**

# 아래의 코드를 실행하였을 때 아래와 같이 출력되어야 합니다.

```
p1 = Point(1, 3)
p2 = Point(3, 1)
r1 = Rectangle(p1, p2)
print(r1.get_area())
print(r1.get_perimeter())
print(r1.is_square())
p3 = Point(2, 5)
p4 = Point(8, 3)
r2 = Rectangle(p3, p4)
print(r2.get_area())
print(r2.get_perimeter())
print(r2.is_square())
```

```
4
8
True
12
16
False
```

## 문제 3. 알파벳 카운트 (20점) - 03\_01.py, 03\_02.py

Q1. 문자열에서 나온 알파벳과 알파벳 갯수를 딕셔너리로 반환하는 `alphabet_count` 함수를 만드시오. (10점)

```
def alphabet_count(word):  
    # 아래에 코드를 작성하시오.  
    # word는 소문자로만 구성 되어있습니다.  
    # 딕셔너리를 반환합니다.
```

```
# 예시 입력 및 출력  
print(alphabet_count('hello'))  
print(alphabet_count('internationalization'))  
print(alphabet_count('haha'))
```

```
{'h': 1, 'e': 1, 'l': 2, 'o': 1}  
{ 'i': 4, 'n': 4, 't': 3, 'e': 1, 'r': 1, 'a': 3, 'o': 2, 'l': 1, 'z': 1}  
{ 'h': 2, 'a': 2}
```

Q2. 문자열에서 가장 많이 나오는 알파벳을 반환하는 `alphabet_mode` 함수를 만드시오. (10점)

문자열 갯수가 동일할 때에는 가장 먼저 나타난 문자를 반환합니다.

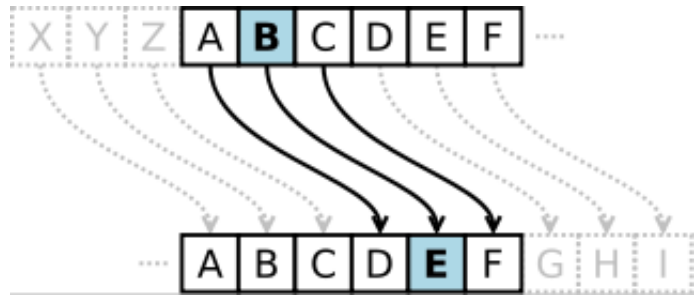
```
def alphabet_mode(word):  
    # 아래에 코드를 작성하시오.  
    # word는 소문자로만 구성 되어있습니다.  
    # word에서 가장 많이 발생한 알파벳 하나를 반환합니다.
```

```
# 예시 입력 및 출력  
print(alphabet_count('hello'))  
print(alphabet_count('internationalization'))  
print(alphabet_count('haha'))
```

```
l  
i  
h
```

# 문제 4. 시저 암호(30점) - 04.py

시저 암호(Caesar Cipher)는 암호학에서 가장 기본이 되는 치환 암호입니다. 암호화하고자 하는 내용을 알파벳별로 일정한 거리(n)만큼 밀어서 다른 알파벳으로 바꾸는 방식입니다.



예를 들어, `apple` 은  $n=1$ 일 경우, 암호화 결과는 `bqqmf` 가 됩니다.

`yes` 는  $n=2$ 일 경우, 암호화 결과는 `agu` 가 됩니다.

**TIP)** 아스키코드는 미국 ANSI에서 표준화한 정보교환용 부호체계입니다.

총 128개의 부호가 사용되며, 아래의 표는 알파벳 소문자가 포함된 93~126까지의 아스키 코드의 일부를 나타낸 것입니다.

10진수	Symbol	10진수	Symbol
93	]	110	n
94	^	111	o
95	_	112	p
96	`	113	q
97	a	114	r
98	b	115	s
99	c	116	t
100	d	117	u
101	e	118	v
102	f	119	w
103	g	120	x
104	h	121	y
105	i	122	z
106	j	123	{
107	k	124	
108	l	125	}
109	m	126	~

파이썬에서는 `ord(char)` 을 통하면 쉽게 해당 문자의 대응되는 10진수 값을 확인할 수 있습니다.

반대로 integer 값을 문자로 바꾸기 위해서는 `chr(int)` 을 사용합니다.

```
ord('a') #=> 97
ord('z') #=> 122
chr(97) #=> 'a'
chr(122) #=> 'z'
```

위의 아스키 코드 테이블을 활용하여, 소문자로 구성된 문자열과 양의 정수  $n$ 이 주어진다면 암호화된 결과를 반환하는 함수 `cipher` 를 만드시오.

```
def cipher(word, n):  
    # 아래에 코드를 작성하시오.  
    # word는 모두 소문자로만 구성되어 있습니다.  
    # n은 양의 정수입니다.  
    # 암호화된 문자열을 반환합니다.
```

```
# 예시 입력 및 출력  
print(cipher('apple', 1))  
print(cipher('apple', 27))  
print(cipher('zoo', 2))
```

```
bqqmf  
bqqmf  
bqq
```