

```
In [1]: from __future__ import division, print_function
import matplotlib.pyplot as plt
import numpy as np
import urllib2
```

```
In [2]: %matplotlib inline
```

시험에서 사용하는 모듈 импорт 하기

```
import urllib2
import numpy as np
```

파이썬 기본 자료형

문제1.

float(부동소수점)를 하나 입력받아, 그 숫자를 한 변의 길이로 하는 정삼각형의 면적과 둘레의 길이를 튜플로 리턴하는 함수 eq_triangle_length를 구현하는 코드를 작성하라.

전분량안:

```
import math
def eq_triangle_length(r):
    area = math.sqrt(3) * 1/4 * float(r)**2
    circumference = 3 * float(r)
    return (area, circumference)
```

.

words 변수가 아래와 같이 선언되어 있다.

```
words = ' When you are smiling, the whole world smiles with yo
u '
```

문자열 자료형

문제2.

문자열 양 끝에 있는 스페이스를 삭제하고, 모든 문자를 대문자로 변화시키며, 스페이스를 기준으로 문자열을 쪼개어 다음과 같은 결과를 출력하는 코드를 작성하라.

```
[ 'WHEN' ,  
'YOU' ,  
'ARE' ,  
'SMILING' ,  
'THE' ,  
'WHOLE' ,  
'WORLD' ,  
'SMILES' ,  
'WITH' ,  
'YOU' ]
```

전문감안:

```
words-separated = [ ]  
for word in words.split():  
    word_new = word.strip().upper()  
    words-separated.append(word_new)
```

.

리스트 조건 제시법

아래의 코드는 0부터 1000사이의 홀수들의 제곱의 리스트를 조건제시법으로 생성한다.

```
odd_1000 = [ x**2 for x in range(0, 1000) if x%2 == 1]
```

문제 3.

0부터 500 사이의 숫자들 중에서 홀수이면서 3의 배수인 숫자들을 제공한 값들의 리스트를 조건 제시법으로 생성하는 코드를 작성하라.

조건문만:

$\text{odd_too_3} = [x^2 \text{ for } x \text{ in range(too) if } x \% 2 == 1 \text{ and } x \% 3 == 0]$

.

넘파이 활용 기초

넘파이 어레이를 생성하는 방법은 몇 개의 기본적인 함수를 이용하면 된다.

```
np.arange()  
np.zeros()  
np.ones()  
np.diag()
```

문제 4.

아래의 모양의 어레이를 생성한느 코드를 작성하라. 단, 언급된 네 개의 함수들 중 필요한 함수만을 사용해야 하며, 수동으로 생성된 리스트나 어레이는 허용되지 않는다.

$$\begin{bmatrix} 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 7 \end{bmatrix}$$

조건문만:

$\text{np.diag}(\text{np.arange}(3, 8, 2))$

.

오류 및 예외 처리

문제 5.

아래 코드가 하는 일을 설명하라.

```
number_to_square = raw_input("A number to divide 100: ")

try:
    number = float(number_to_square)
    print("100을 입력한 값으로 나눈 결과는", 100/number, "입니다.")
except ZeroDivisionError:
    raise ZeroDivisionError('0이 아닌 숫자를 입력하세요.')
except ValueError:
    raise ValueError('숫자를 입력하세요.')
```

전문 강안:

입력 받은 숫자로 100을 나눈 값을 계산하는 코드이다.
하지만, 0을 입력하면 '0이 아닌 숫자를 입력하세요.'
란 경고창과 함께 코드실행이 멈춘다.
또한 숫자가 아닌 값이 입력된 경우
'숫자를 입력하세요.'란 경고창과 함께
코드실행이 멈춘다.

.

문자열 자료형

아래 사이트는 커피 콩의 현재 시세를 보여준다.

```
http://beans-r-us.appspot.com/prices.html
```

위 사이트의 내용을 html 소스코드로 보면 다음과 같으며, 검색된 시간의 커피콩의 가격은 Current price of coffee beans 문장이 담겨 있는 줄에 명시되어 있다.

```
<html><head><title>Welcome to the Beans'R'Us Pricing Page</title>
<link rel="stylesheet" type="text/css" href="beansrus.css" />
</head><body>
<h2>Welcome to the Beans'R'Us Pricing Page</h2>
<p>Current price of coffee beans = <strong>$5.94</strong></p>
<p>Price valid for 15 minutes from Sun Sep 10 12:21:58 2017.</p>
</body></html>
```

위의 소스코드를 웹페이지 주소만 알고 있을 때 아래와 같이 읽어드릴 수 있다.

```
page = urllib.urlopen("http://beans-r-us.appspot.com/prices.html")
text = page.read().decode("utf8")
```

위의 코드를 실행하면 text 변수에 웹페이지의 전체 내용이 하나의 문자열로 저장된다.

문제 6.

이 때 text에 저장된 문자열에서 가격 정보를 확인하는 방법을 코드를 이용하여 설명하라.

전문답안 :

```
price_index = text.find('>$')+2
price = text[price_index, price_index+4]
```

.

사전

record_list.txt 파일은 여덟 명의 수영 선수의 50m 기록을 담고 있다.

```
player1 21.09
player2 20.32
player3 21.81
player4 22.97
player5 23.29
player6 22.09
player7 21.20
player8 22.16
```

문제 7.

아래코드가 하는 일을 설명하고, 출력된 결과를 답하여라.

```
from __future__ import print_function

record_f = open("record_list.txt", 'r')
record = record_f.read().decode('utf8').split('\n')

record_dict = {}

for line in record:
    (player, p_record) = line.split()
    record_dict[p_record] = player

record_f.close()

record_list = record_dict.keys()
record_list.sort()

print(record_list)
```

전문강사님:
인용된 파일로부터 각각의 줄에
포함된 선수이름과 기록을 추출하여,
기록을 키로, 선수이름을 키값으로
하여 record_list 사전에
저장한다. 이후 기록 순서를
확정하기 위해 키들만의
리스트를 정렬한다.

.

넘파이 활용 기초

population.txt 파일은 1900년부터 1920년까지 캐나다 북부지역에서 서식한 산토끼(hare)와 스라소니(lynx)의 숫자, 그리고 채소인 당근(carrot)의 재배숫자를 아래 내용으로 순수 텍스트 데이터로 담고 있다.

#	year	hare	lynx	carrot
	1900	30e3	4e3	48300
	1901	47.2e3	6.1e3	48200
	1902	70.2e3	9.8e3	41500
	1903	77.4e3	35.2e3	38200
	1904	36.3e3	59.4e3	40600
	1905	20.6e3	41.7e3	39800
	1906	18.1e3	19e3	38600
	1907	21.4e3	13e3	42300
	1908	22e3	8.3e3	44500
	1909	25.4e3	9.1e3	42100
	1910	27.1e3	7.4e3	46000
	1911	40.3e3	8e3	46800
	1912	57e3	12.3e3	43800
	1913	76.6e3	19.5e3	40900
	1914	52.3e3	45.7e3	39400
	1915	19.5e3	51.1e3	39000
	1916	11.2e3	29.7e3	36700
	1917	7.6e3	15.8e3	41800
	1918	14.6e3	9.7e3	43300
	1919	16.2e3	10.1e3	41300
	1920	24.7e3	8.6e3	47300

아래 코드는 연도, 토끼 개체수, 스라소리 개체수, 당근 개체수를 따로따로 떼어 내어 각각 어레이로 변환하여 year, hares, lynxes, carrots 변수에 저장하는 코드이다.

```
In [16]: data = np.loadtxt('populations.txt')
year, hares, lynxes, carrots = data.T
```

문제 8.

위 코드에서 `data.T`에 대해 간단히 설명하라.

전분담안:

`data.T`는 `data` 행렬의 전치행렬을 나타낸다.

따라서 4 * 2 행렬 모양이 되며, 각 행은

차례대로 년도, 산토끼 개체수, 스라소니 개체수,

당근개체수를 연도별로 담고 있다.

또한 각 행을 차례대로 `year`, `hares`, `lynxes`, `carrots`

- 변수에 할당한다.