

## ■ 몸 풀기 문제 1

- 괄호 안의 인자인 `number1`, `number2`를 입력값으로 받아 `result=(num1+num2)`을 수행하고, 그의 결과값 `result`를 반환한 후 종료하는 이름이 `average`라는 함수를 정의해보자.

## ■ 몸 풀기 문제 2

- 다음의 리스트를 입력으로 활용하여 0일 때만 출력이 없고, 그 이외에 각 값을 출력하는 함수를 작성하라.
- 호출 `print_list([1, 2, 3, 0])`

### ■ Exercise 1.

- 사용자로부터 N단을 입력받아서 다음과 같은 결과가 출력되도록 gugudan 함수를 작성하라.

```
9 x 1 = 9
9 x 2 = 18
9 x 3 = 27
9 x 4 = 36
9 x 5 = 45
9 x 6 = 54
9 x 7 = 63
9 x 8 = 72
9 x 9 = 81
```

### ■ Exercise 2.

- 6명의 학생이 시험을 치렀다. 각 시험 점수를 score라는 리스트에 넣고 이에 대해 최고점을 구하는 함수를 for문을 이용해서 작성하라.
- Score = [55, 82, 34, 98, 76, 62]

### ■ Exercise 3.

- TV 음량의 현재 음량은 3이고, 증가시킬 만큼의 음량을 입력하면 음량이 증가하는 프로그램을 작성하라

```
현재 음량은 3 입니다.  
증가시킬 만큼의 음량을 입력: 6  
증가 후의 음량은 9 입니다.
```

### ■ Exercise 4.

- 회사의 부서에서 출근 여부를 입력하는 프로그램을 만들려고 한다. 출근한 사람이 본인의 이름을 입력하면 출근이 완료되었다고 출력하고, 없는 이름을 입력하면 출근이 불가하다고 출력하는 함수를 작성해보자.

- list=['yujin', 'suzi', 'jisu', 'jennie', 'rose', 'lisa']

```
이름을 입력하세요 : jisu  
출근 완료되었습니다.
```

```
이름을 입력하세요 : jennie  
출근 완료되었습니다.
```

```
이름을 입력하세요 : 양민  
출근이 불가합니다.
```

```
이름을 입력하세요 : |
```

## ■ Exercise 5.

- 사칙 연산 계산기를 만들고자 한다. 입력은 숫자 2개와 연산자(+, -, \*, /)이다. (입력 및 출력 형태는 그림 참고)
  - 출력을 담당하는 disp() 함수 작성
  - 부호를 확인한 후 특정 연산을 호출할 Calc() 함수 작성: 부호로 각 램다 함수 호출
  - 사칙 연산을 담당하는 Lambda 함수 4개 작성
  - End 또는 end를 입력할 때까지 무한 루프가 작동하도록 구현
  - 입력 시 문자열을 구분하여 부호 추출(방법은 자유)

```
계산할 수와 연산기호를 입력
5 - 6
-1
계산할 수와 연산기호를 입력
7 * 8
56
계산할 수와 연산기호를 입력
end
```