

## 1. 수강과목 점수 평균 구하기

표준입력창에서 점수를 하나씩 차례로 입력받아 평균을 계산하여 내주는 함수

score\_average()를 만들어보자. score\_average() 프로시저를 호출하면 표준입출력창을 통하여 사용자와 대화식으로 프로그램이 아래와 같이 작동한다. 보통 폰트로 된 부분은 컴퓨터가 사용자에게 보여주는 부분이고, 파란색으로 볼드체로 진하게 보이는 부분이 사용자가 키보드로 입력한 부분이다.

```
Enter your scores. I will calculate your average score.
score : -15
score again: 120
score again: 92
score : 만점
score again: 78
score : 54
score : 88
score : 64
score : 95
score : 0
Your average score of 6 subject(s) is 78.5
```

자가 키보드로 입력한 부분이다. 허용하는 키보드 입력은 0 과 100 사이의 정수이다. 위의 -15, 120, 만점은 이 요구사항을 만족하지 못 하므로 재입력을 요구하였다. 실제 점수는 1 과 100사이의 정수만 허용하고, 0은 점수 입력이 끝났다는 표시로만 사용한다. 위의 사례에서 92, 78, 54, 88, 64, 95 는 모두 유효한 입력이다. 0을 입력 받으면 더 이상 입력을 받지 않고 그때까지 입력 받은 점수의 평균을 계산하여, 몇 과목이며 평균은 몇 점인지 표준 창에 위와 같이 프린트한다. 평균 점수는 소수점 첫째 미만은 반올림한다. 만약 다음과 같이 유효한 점수가 하나도 없으면 평균을 계산하는 대신 점수가 없다는 메시지를 프린트한다.

```
Enter your scores. I will calculate your average score.
score : 110
score again: 120
score again: 200
score again: 0
There is no score.
```

사실 평균을 구하기 위해서 0으로 나누면 ZeroDivisionError 오류가 발생하므로 어차피 별도로 처리해 주어야 한다. 즉, 0으로 나누는 대신 메시지를 프린트 한 것이다.

**알고리즘**

평균을 구하려면 입력받은 점수를 모두 더한 다음, 과목의 개수로 나누면 된다.

1. Enter your scores. I will calculate your average score. 메시지를 프린트한다.
2. count 변수를 0으로 초기화한다. 이 변수는 수강과목 개수가 몇 개인지 기억하는 변수이다.
3. total 변수를 0으로 초기화한다. 이 변수는 수강과목 점수의 합을 기억하는 변수이다.
4. 수강과목 점수를 차례로 하나씩 입력받는다. (각 점수 입력확인 필수 - 0이상 100이하 정수면 통과)
5. 입력의 종료는 0을 입력하여 표시한다. 즉, 입력이 0이면 입력 받는 작업을 중지한다.
6. 입력이 0이 아니면 정수로 변환한 뒤 total 변수의 값에 합하여 누적한다.
7. count 변수의 값을 1 증가한다.
8. count 변수의 값이 0이 아닌 경우에만 평균을 계산하여 소수점 첫짜리 미만은 반올림하고, Your average score of \_ subject(s) is \_ 형식으로 프린트 한다. 앞 부분은 과목의 수가 채워지고, 뒷 부분은 평균 점수가 채워진다.
9. count 변수의 값이 0인 경우에는 나눗셈을 하면 ZeroDivisionError 오류가 발생하므로 나누지 않고, There is no score. 라는 메시지를 프린트한다.

**완성해야 할 함수**

```
1 def score_average():
2     print("Enter your scores. I will calculate your average score.")
3     count = 0
4     total = 0
5     while True:
6
7
8
9
10
```

## 2. 수강과목 점수 평균 구하기 (과락은 빼고 평균 계산)

앞에서 완성한 프로시저 `score_average()`를 다음 추가 요구사항을 적용, 수정한 `score_average2()`를 만들어보자.

### 추가요구사항

- 입력한 과목 점수가 60점 미만인 점수를 제외하고 평균을 구하도록 수정한다.
- 그리고 60점 미만이어서 과락한 과목은 몇 과목인가?

`score_average2()` 프로시저를 호출하면 표준입출력창을 통하여 사용자와 대화식으로 프로그램이 아래와 같이 작동한다.

보통 폰트로 된 부분은 컴퓨터가 사용자에게 보여주는 부분이고, 파란색으로 볼드체로 진하게 보이는 부분이 사용자가 키보드로 입력한 부분이다.

```
Enter your scores. I will calculate your average score.
score : -15
score again: 120
score again: 92
score : 만점
score again: 78
score : 54
score : 88
score : 64
score : 95
score : 0
Your average score of 5 subject(s) is 83.4
You failed in 1 subject(s).
```

`score_average()`프로시저와 동일하지만 1~59 점 사이의 키보드 입력은 과락이므로 평균 계산에 포함하지 말아야 한다. 위의 사례에서는 54점이 유일하게 과락에 해당하는 점수이므로 평균 계산에 포함시키지 않았다. 과락한 점수는 몇 개인지 기억해 두었다가 몇 과목이 과락했다고 알려주어야 한다. 위에서는 1과목이므로 마지막 줄에서 그렇게 알려주었다.

평균을 구할 점수가 없을 때에도, 과락한 과목이 있으면 과락 과목의 개수를 알려주어야 한다.

```
Enter your scores. I will calculate your average score.  
score : 43  
score : 59  
score : 0  
There is no score.  
You failed in 2 subject(s).
```

### 3. 윤년인지 알아보는 함수 만들기

2월은 평년인지 윤년인지에 따라 각각 28일 또는 29일이 있다. 기본적으로 윤년은 4년마다 한번씩 돌아 오는데 오차를 줄이기 위해서 예외를 둔다. 윤년을 결정하는 규칙은 “4의 배수이면 윤년인데, 그 중에서 400의 배수를 제외한 100의 배수는 윤년이 아니다”이다. 따라서 2008, 2012, 2016, 1600, 2000, 2400 년은 모두 윤년이고, 2013, 2014, 2015, 2100, 2200, 2300은 모두 평년이다. 0 또는 양의 정수를 받아서 윤년이면 True, 평년이면 False를 내주는 함수 isleapyear를 작성하시오. 밑줄 친 부분에 논리식 하나를 만들어 채우면 된다. (음수 인수의 경우에는 0을 내주어야 한다).

```
def isleapyear(year):  
  
# test case  
for y in range(5):  
    print(y,isleapyear(y))  
for y in range(2010,2017):  
    print(y,isleapyear(y))  
for y in range(1900, 2600, 100):  
    print(y,isleapyear(y))
```