구현된 모든 소스들은 메일로 제출하며 첨부파일로 추가한다.

이 때 메일 제목은 “1월25일실습1-XXX” 로 작성하여 전송한다.

[ 실습 1 ]

1. listLab5.py 이라는 소스를 생성한다.

2. 비어있는 리스트를 하나 만들고 이 안에 1~45 사이의 난수를 추출하여 6개를 저장하는데

동일한 숫자가 중복하여 저장되지 않게 한다.

3. 수행 결과는 다음과 같다.

행운의 로또번호 : X, X, X, X, X, X

[ 실습 2 ]

1. listLab6.py 이라는 소스를 생성한다.

2. 다음과 같은 내용으로 구성되는 이차원 리스트를 생성한다.

10, 12, 14, 16

18, 20, 22, 24

26, 28, 30, 32

34, 36, 38, 40

3. 다음 결과를 출력한다.

1행 1열의 데이터 : 10

3행 4열의 데이터 : 32

행의 갯수 : 4

열의 갯수 : 4

3행의 데이터들 : 26 28 30 32

2열의 데이터들 : 12 20 28 36

왼쪽 대각선 데이터들 : 10 20 30 40

오른쪽 대각선 데이터들 : 16 22 28 34

[ 실습 3 ]

1. listLab7.py 이라는 소스를 생성한다.

2. 다음과 같은 내용으로 구성되는 이차원 리스트를 생성한다.

1행 10, 20, 30, 40, 50

2행 5, 10, 15

3행 11, 22, 33, 44

4행 9, 8, 7, 6, 5, 4, 3, 2, 13

3. 행단위 합을 구하여 다음과 같이 출력한다.

1행의 합은 x 입니다.

2행의 합은 x 입니다.

3행의 합은 x 입니다.

4행의 합은 x 입니다.

[ 실습 4 ]

1. listLab8.py 이라는 소스를 생성한다.

2. 다음과 같은 내용으로 구성되는 이차원 리스트를 생성한다.

'B', 'C', 'A', 'A'

'C', 'C', 'B', 'B'

'D', 'A', 'A', 'D'

3. 다음 내용으로 구성되는 리스트를 하나 생성한다.

첫 번째 원소에는 'A' 문자의 개수

두 번째 원소에는 'B' 문자의 개수

세 번째 원소에는 'C' 문자의 개수

네 번째 원소에는 'D' 문자의 개수

4. 다음과 형식으로 출력한다.

A 는 x개 입니다.

B 는 x개 입니다.

C 는 x개 입니다.

D 는 x개 입니다.

[ 실습 5 ]

1. 파일명 : funcLab10.py

2. 구현해야 하는 함수 사양

함수명 : sumeven

매개변수 : 가변형(전달받을 수 있는 아규먼트 개수에 제한이 없다.)

리턴값 : 1개

기능 : 아규먼트가 몇 개가 전달되든 처리해야 한다.

아규먼트는 1 이상의 숫자만 온다고 정한다.

전달된 아규먼트들에서 짝수에 해당하는 숫자들만 합을 계산해서 리턴한다.

전달된 아규먼트들 중에 짝수가 없으면 0을 리턴한다.

아규먼트가 전달되지 않으면 -1을 리턴한다.

3. 숫자를 다양하게 지정해서 sumEven() 함수를 호출해 본다.

print(sumeven(1, 2, 3, 4, 5))

print(sumeven(11, 22, 33, 44, 55))

print(sumeven(1, 3, 5))

print(sumeven())

6

66

0

-1

[ 실습 6 ]

1. 파일명 : funcLab11.py

2. 구현해야 하는 함수 사양

함수명 : number\_all\_sum

매개변수 : 가변형(전달받을 수 있는 아규먼트 개수에 제한이 없다.)

리턴값 : 1개

기능 : 아규먼트가 몇 개가 전달되든 처리해야 한다.

호출시 전달되는 아규먼트의 데이터 타입에는 제한이 없다. 그러므로 전달된 아규먼트의

타입을 채크하여 숫자만 처리하고 숫자가 아닌 데이터는 무시한다.

아규먼트가 전달되지 않았거나 전달되었다 하더라도 숫자가 없으면 None 을

리턴한다.

3. 숫자를 다양하게 지정해서 number\_all\_sum () 함수를 호출해 본다.

print(number\_all\_sum(1, 2, 3, 4, 5))

print(number\_all\_sum(1, 2, 'a', 3, 4, 'b', 5))

print(number\_all\_sum(10, 20, True))

print(number\_all\_sum())

print(number\_all\_sum('a', True, '가'))

15

15

30

None

None