

1월11일실습_1

[실습 1]

1. listLab5.py 라는 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

```
import random

list = []
while True:
    x = random.randint(1,45)
    if len(list) == 6:
        break
    if x in list:
        continue
    else:
        list.append(x)

print('행운의 로또번호 :', list[0], list[1], list[2], list[3], list[4], list[5])
```

행운의 로또번호 : 33 8 1 20 7 9

```
import random
lotto_list = []
while True :
    num = random.randint(1,45)
    if num not in lotto_list:
        lotto_list.append(num)
    if len(lotto_list) == 6 :
        break
print("행운의 로또번호 : ", lotto_list)
```

행운의 로또번호 : [10, 26, 5, 40, 38, 21]

[실습 2]

1. listLab6.py 라는 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

```
list = [[10, 12, 14, 16],
        [18, 20, 22, 24],
        [26, 28, 30, 32],
        [34, 36, 38, 40]]

print('1행 1열의 데이터 :', list[0][0])
print('3행 4열의 데이터 :', list[2][3])
print('행의 갯수 :', len(list))
print('열의 갯수 :', len(list[0]))
print('3행의 데이터들 :', list[2][0], list[2][1], list[2][2], list[2][3])
print('2행의 데이터들 :', list[1][0], list[1][1], list[1][2], list[1][3])
print('2행의 데이터들 :', list[1])
print('왼쪽 대각선 데이터들 :', list[0][0], list[1][1], list[2][2], list[3][3])
print('오른쪽 대각선 데이터들 :', list[0][3], list[1][2], list[2][1], list[3][0])
```

1행 1열의 데이터 : 10
3행 4열의 데이터 : 32
행의 갯수 : 4
열의 갯수 : 4
3행의 데이터들 : 26 28 30 32
2행의 데이터들 : 18 20 22 24
2행의 데이터들 : [18, 20, 22, 24]
왼쪽 대각선 데이터들 : 10 20 30 40
오른쪽 대각선 데이터들 : 16 22 28 34

```
list = [ [10, 12, 14, 16], [18, 20, 22, 24], [26, 28, 30, 32], [34, 36, 38, 40]]

print("1행 1열의 데이터 :", list[0][0])
print("3행 4열의 데이터 :", list[2][3])

print("행의 갯수 :", len(list))
print("열의 갯수 :", len(list[0]))

print("3행의 데이터들 :", list[2][0], list[2][1], list[2][2], list[2][3])
print("2열의 데이터들 :", list[0][1], list[1][1], list[2][1], list[3][1])

print("왼쪽 대각선 데이터들 :", end = "")
for i in range(len(list)):
    print(list[i][i], end=" ")
print()
print("오른쪽 대각선 데이터들 :", end = "")
j = len(list[0])
for i in range(len(list)):
    print(list[i][j-1], end=" ")
    j = j - 1
```

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 28
왼쪽 대각선 데이터들 : 10 20 30 40
오른쪽 대각선 데이터들 : 16 22 28 34

[실습 3]

1. listLab7.py 라는 소스를 생성한다.
2. 다음과 같은 내용으로 구성되는 **이차원 리스트**를 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 합**을 구하여 다음과 같이 출력한다.
1행의 합은 x 입니다.
2행의 합은 x 입니다.
3행의 합은 x 입니다.
4행의 합은 x 입니다.

```
list = [[10, 20, 30, 40, 50], [5, 10, 15],  
        [11, 22, 33, 44], [9, 8, 7, 6, 5, 4, 3, 2, 13]]  
  
for i in range(4):  
    print(i+1, '행의 합은 ', sum(list[i]), ' 입니다.', sep='')
```

1행의 합은 150 입니다.
2행의 합은 30 입니다.
3행의 합은 110 입니다.
4행의 합은 57 입니다.

[실습 4]

1. listLab8.py 라는 소스를 생성한다.
2. 다음과 같은 내용으로 구성되는 **이차원 리스트**를 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.

```
list1=[['B', 'C', 'A', 'A'], ['C', 'C', 'B', 'B'], ['D', 'A', 'A', 'D']]

count1, count2, count3, count4 = 0,0,0,0
for l in list1:
    for e in l:
        if e == 'A':
            count1+=1
        elif e == 'B':
            count2 +=1
        elif e == 'C':
            count3 +=1
        elif e == 'D':
            count4 +=1

list2 = [count1, count2, count3, count4]

print("A 는 ", list2[0], "개 입니다.", sep="")
print("B 는 ", list2[1], "개 입니다.", sep="")
print("C 는 ", list2[2], "개 입니다.", sep="")
print("D 는 ", list2[3], "개 입니다.", sep="")
# 혹은
for i in range(4) :
    print(chr(65+i), " 는 ", list2[i], "개 입니다.", sep="")
```

A 는 4개 입니다.
 B 는 3개 입니다.
 C 는 3개 입니다.
 D 는 2개 입니다.
 A 는 4개 입니다.
 B 는 3개 입니다.
 C 는 3개 입니다.
 D 는 2개 입니다.

[실습 5]

1. dicLab1.py 라는 소스를 생성한다.
2. 다음 내용으로 구성되는 딕셔너리를 하나 생성한다.
 키 : **red, blue, green, yellow, orange, black, white, violet, pink, lime**
 값 : https://www.w3schools.com/colors/colors_picker.asp 사이트에 가서 이 칼라들의 #으로 시작하는 RGB 값을 찾아서 사용한다.
3. 사용자에게 “**칼라명을 영문으로 입력하세요 .:**”를 출력하면서 **칼라명 한 개를 입력받고**
 미리 생성한 딕셔너리에서 그 칼라에 해당하는 RGB 값을 추출해서 “**xx 칼라의 RGB 값은 xxx** 입니다”를 출력하며

딕셔너리에 없는 칼라명이 입력된 경우에는 “**xx 칼라의 RGB 값을 찾을 수 없습니다**”

```
dic = { 'red': '#ff0000', 'blue': '#0000ff', 'green': '#008000',
        'yellow': '#ffff00', 'orange': '#ffa500', 'black': '#000000',
        'white': '#ffffff', 'violet': '#ee82ee', 'pink': '#ffc0cb',
        'lime': '#00ff00' }

colorname= input("칼라명을 영문으로 입력하세요 :")

if colorname in dic:
    print(colorname, "칼라의 RGB 값은", dic[colorname], "입니다")
else:
    print(colorname, "칼라의 RGB 값을 찾을 수 없습니다")
```

칼라명을 영문으로 입력하세요 :red
red 칼라의 RGB 값은 #ff0000 입니다

[실습 6]

1. dicLab2.py 라는 소스를 생성한다.
2. 다음 내용으로 구성되는 **딕셔너리**를 하나 생성한다.
키 : 한 글자 요일명('월', '화' ... '일')
값 : 아래 URL 을 참고하여 오늘부터 일요일까지의 **날씨정보를 추출**하고
최저온도와 최고온도를 튜플로 생성하여 사용한다.
3. 사용자에게 **"요일명을 한글로 입력하세요 :"**를 출력하면서 칼라명 한 개를 입력받고
미리 생성한 딕셔너리에서 그 칼라에 해당하는 최저온도와 최고온도의 튜플을 추출해서
"x요일의 최저온도는 x 이고 최고 온도는 x입니다" 를 출력하며
딕셔너리에 없는 요일명이 입력된 경우에는 **"x요일의 정보를 찾을 수 없습니다"**
<https://www.google.com/search?q=%EC%9D%BC%EA%B8%B0+%EC%98%88%EB%B3%B4&og=%EC%9D%BC%EA%B8%B0+%EC%98%88%EB%B3%B4&aqs=chrome..69i57j0l7.10632j1j7&sourceid=chrome&ie=UTF-8>

```
dic = { '월': (-13, -3), '화': (-6, 2), '수': (-8, 8), '목': (-4, 6), '금': (-7, 7), '토': (-14, -3), '일': (-7, -2) }

Day_of_the_week = input("요일명을 한글로 입력하세요(한글자):")

if Day_of_the_week in dic :
    print(Day_of_the_week, "요일의 최저온도는", dic[Day_of_the_week][0], "이고 최고 온도는", dic[Day_of_the_week][1], "입니다", sep="")
else:
    print(Day_of_the_week, "요일의 정보를 찾을 수 없습니다", sep="")
```

요일명을 한글로 입력하세요(한글자):월
월요일의 최저온도는-13이고 최고 온도는-3입니다

[실습 8]

1. setLab1.py 이라는 소스를 생성한다.
2. 비어있는 셋을 2개 만들고 각각 1~20 사이의 숫자 10개를 추출하여 저장한다.
3. 생성된 2 개의 셋에 대하여 **집합 연산**을 수행하고 결과를 다음과 같이 출력한다.
집합 1 : {x, x, x, x, x, x, x, x, x, x}
집합 2 : {x, x, x, x, x, x, x, x, x, x}
두 집합에 모두 있는 데이터 : {x, x, x, x, x, x, x, x}

집합1 또는 집합2 에 있는 데이터 : {x, x, x, x, x, x, x }

집합1에는 있고 집합2에는 없는 데이터 : {x, x, x, x, x, x, x }

집합2에는 있고 집합1에는 없는 데이터 : {x, x, x, x, x, x, x }

집합1과 집합 2가 각자 가지고 있는 데이터 : {x, x, x, x, x, x, x }

```
set_1 = set()
set_2 = set()

import random
for x in range(10):
    i = random.randint(1,20)
    set_1.add(i)
for y in range(10):
    j = random.randint(1,20)
    set_2.add(j)

print("집합 1 :", set_1)
print("집합 2 :", set_2)
print("두 집합에 모두 있는 데이터 :", set_1 & set_2)
print("집합1 또는 집합2 에 있는 데이터 :", set_1 | set_2)
print("집합1에는 있고 집합2에는 없는 데이터 :", set_1 - set_2)
print("집합2에는 있고 집합1에는 없는 데이터 :", set_2 - set_1)
print("집합1과 집합 2가 각자 가지고 있는 데이터 :", set_1 ^ set_2)
```

집합 1 : {4, 5, 10, 11, 14, 15, 16, 17}

집합 2 : {3, 8, 10, 12, 13, 16, 20}

두 집합에 모두 있는 데이터 : {16, 10}

집합1 또는 집합2 에 있는 데이터 : {3, 4, 5, 8, 10, 11, 12, 13, 14, 15, 16, 17, 20}

집합1에는 있고 집합2에는 없는 데이터 : {4, 5, 11, 14, 15, 17}

집합2에는 있고 집합1에는 없는 데이터 : {3, 8, 12, 13, 20}

집합1과 집합 2가 각자 가지고 있는 데이터 : {3, 4, 5, 8, 11, 12, 13, 14, 15, 17, 20}

```
import random

set1 = set()
set2 = set()

while len(set1) <= 10:
    num = random.randint(1, 20)
    set1.add(num)

while len(set2) <= 10:
    num = random.randint(1, 20)
    set2.add(num)

print("집합 1:", set1)
print("집합 2:", set2)
print("두 집합에 모두 있는 데이터 :", set1 & set2)
print("집합1 또는 집합2 에 있는 데이터:", set1|set2)
print("집합1에는 있고 집합2에는 없는 데이터:", set1-set2)
print("집합2에는 있고 집합1에는 없는 데이터:", set2-set1)
print("집합1과 집합2가 각자 가지고 있는 데이터:", set1^set2)
```

집합 1 : {2, 4, 11, 13, 14, 16, 19, 20}

집합 2 : {2, 4, 7, 9, 10, 11, 12, 14, 19, 20}

두 집합에 모두 있는 데이터 : {2, 4, 11, 14, 19, 20}

집합1 또는 집합2 에 있는 데이터 : {2, 4, 7, 9, 10, 11, 12, 13, 14, 16, 19, 20}

집합1에는 있고 집합2에는 없는 데이터 : {16, 13}

집합2에는 있고 집합1에는 없는 데이터 : {9, 10, 12, 7}

집합1과 집합 2가 각자 가지고 있는 데이터 : {7, 9, 10, 12, 13, 16}

[실습 9]

1. setLab2.py 이라는 소스를 생성한다.
2. 비어있는 셋을 하나 만들고 이 안에 **1~45 사이의 난수를 추출하여 6개**를 저장하는데 당연히 여기서도 **동일한 숫자가 중복하여 저장되지 않게** 한다.
3. 수행 결과는 다음과 같다.
행운의 로또번호 : X, X, X, X, X, X

```
import random

set1 = set()

while True :
    set1.add(random.randint(1, 45))
    if len(set1) == 6 :
        break

print( "행운의 로또번호 :", ", ".join( repr(e) for e in set1) )
```

행운의 로또번호 : 32, 35, 42, 12, 20, 23