**[ 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. 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**](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”**

**[ 실습 6 ]**

**1. dicLab2.py 라는 소스를 생성한다.**

**2. 다음 내용으로 구성되는 딕셔너리를 하나 생성한다.**

**키 : 한 글자 요일명**

**값 : 아래 URL 을 참고하여 오늘부터 일요일까지의 날씨정보를 추출하고**

**최저온도와 최고온도를 튜플로 생성하여 사용한다.**

**3. 사용자에게 “요일명을 영문으로 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&oq=%EC%9D%BC%EA%B8%B0+%EC%98%88%EB%B3%B4&aqs=chrome..69i57j0l7.10632j1j7&sourceid=chrome&ie=UTF-8**](https://www.google.com/search?q=%EC%9D%BC%EA%B8%B0+%EC%98%88%EB%B3%B4&oq=%EC%9D%BC%EA%B8%B0+%EC%98%88%EB%B3%B4&aqs=chrome..69i57j0l7.10632j1j7&sourceid=chrome&ie=UTF-8)

**[ 실습 7 ]**

**1. dicLab3.py 라는 소스를 생성한다.**

**2. 다음 내용으로 구성되는 딕셔너리를 하나 생성한다.**

**키 : 우리 팀원 이름**

**값 : 다음 내용으로 구성되는 딕셔너리**

**키 : 나이, 전공, 거주지, 메일주소, 취미**

**값 : 실제 (^^)**

**3. 모든 팀원들의 이름과 정보를 출력해 본다.**

**[ 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 연산을 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 }**

**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 }**

**[ 실습 9 ]**

**1. setLab2.py 이라는 소스를 생성한다.**

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

**당연히 여기서도 동일한 숫자가 중복하여 저장되지 않게 한다.**

**3. 수행 결과는 다음과 같다.**

**행운의 로또번호 : X, X, X, X, X, X**