구현된 모든 소스들은 메일로 제출하며 첨부파일로 추가한다.

이 때 메일 제목은 “1월26일실습1-XXX” 로 작성하여 전송한다.

[ 실습 1 ]

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”

웹 크롤링을 학습하면 이것도 프로그래밍화 할 수 있지만 지금은 학습 전이니 눈으로 보고 직접 작성한다,

[ 실습 2 ]

1. dicLab2.py 라는 소스를 생성한다.

2. 다음 내용으로 구성되는 딕셔너리를 하나 생성한다.

키 : 한 글자 요일명(‘월”, ‘화’ …’일’)

값 : 아래 URL 을 참고하여 오늘부터 다음주 화요일까지의 날씨정보를 읽어서

최저온도와 최고온도 정보를 읽고 튜플로 생성하여 사용한다.(하드코딩)

3. 사용자에게 “요일명을 한글로 입력하세요 :”를 출력하면서 요일명 한 개를 입력받고

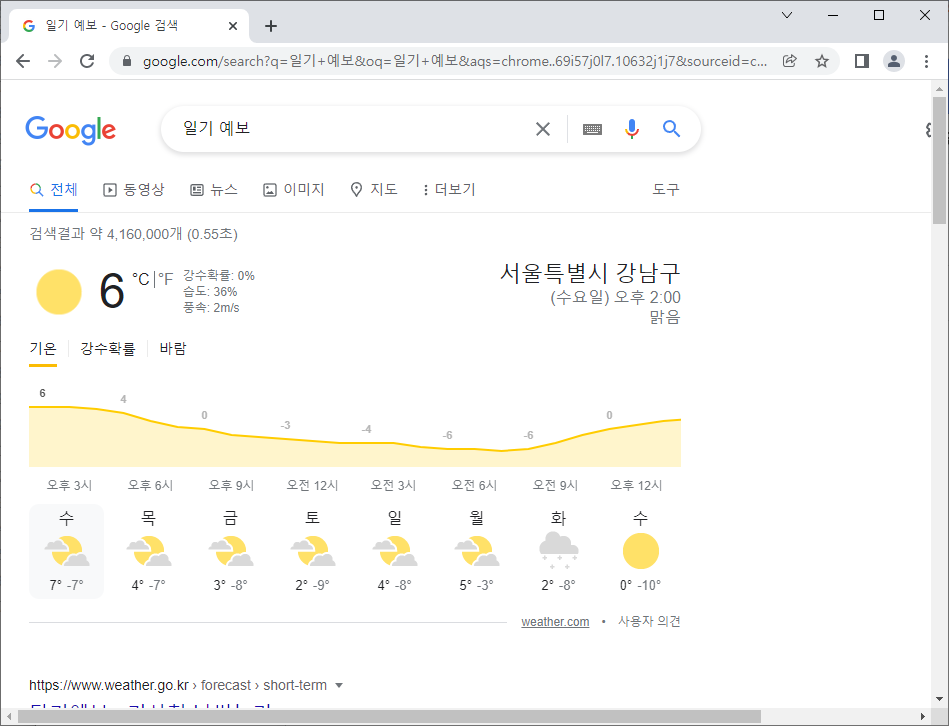
미리 생성한 딕셔너리에서 그 요일에 해당하는 최저온도와 최고온도의 튜플을 추출해서

“x요일의 최저온도는 x 이고 최고 온도는 x입니다” 를 출력하며 딕셔너리에 없는 요일명이

입력된 경우에는 “x요일의 정보를 찾을 수 없습니다”

[ 주간 날씨를 채크할 수 있는 웹 사이트의 URL ]

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 3 ]

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 4 ]

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

[ 실습 5 ]

파일명 : packunpacLab.py

다음 리스트를 생성하고

listv3 = [ 'p', 'y', 't', 'h', 'o', 'n' ]

(1) v1, v2, v3, v4, v5,v6 에 언 패킹해서 저장한 후에 각 변수의 값을 행 단위로 화면에 출력한

다.

(2) listv3 를 언패킹하여 print() 함수에 전달하여 출력한다. 

(3) listv3 를 그냥 print() 함수에 전달하여 출력한다. 