

## 1, 목차

1) 이상기온 정의

2) 이상기온 출력 구현도

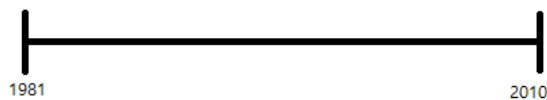
3) 이상기온 출력 구현

### 1) 이상기온이란?

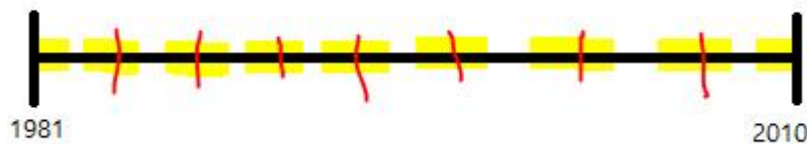
기상학적 측면에서 이상기후의 정의는 '기온, 강수량 등의 기후요소가 평년값(1981~2010년)에 비해 현저히 높거나 낮은 수치를 나타내는 극한현상'으로 세계기상기구(WMO)에서는 90퍼센타일 또는 10퍼센타일 미만의 범위를 나타냅니다.

\*퍼센타일 : 평년  $\pm 30$ 일 기간에 발생한 자료를 크기순으로 작은 쪽에서부터 세어 몇 번째인지를 나타내는 백분위수

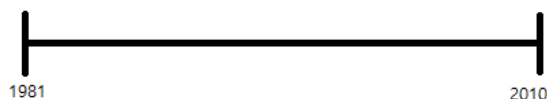
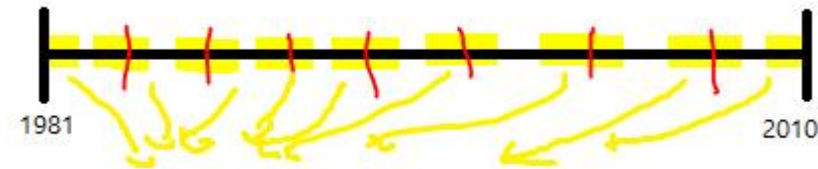
예를들어 1981년부터 2010년까지의 범위를 아래와 같이 놓았을때,



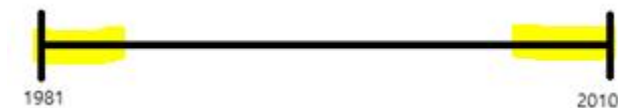
1월 1일이 이상기온인지 알고 싶으면,



빨간 선이 1981~2010년까지 1월 1일이라고 보았을 때, 형관색으로 칠해진 값들(1월 1일에서  $\pm 30$ 일 떨어진 값들)을 다 한꺼번에 저장하고 알고자 하는 년도의 1월 1일에 값을 넣었을 때,



아래에 형관 칠해진 부분 (하위 상위 10퍼센트)에 들어가면 이상기온으로 보여진다



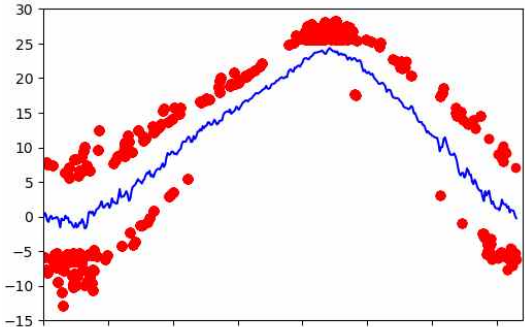
### 2) 이상기온 출력 구현도

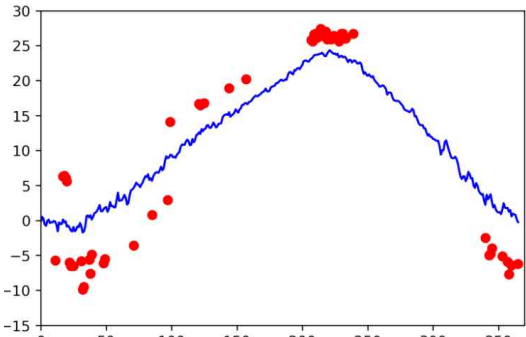
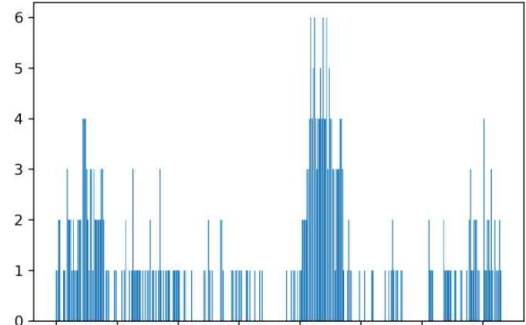
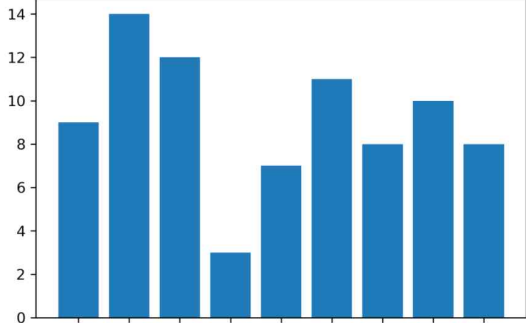
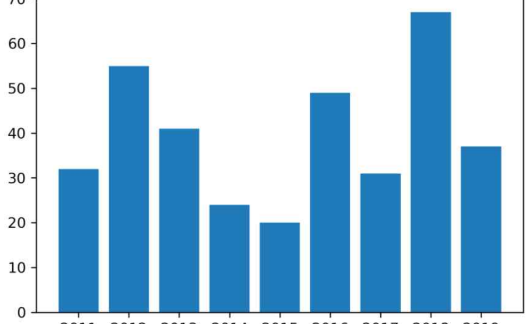
1981년도부터 2010년까지의 값들을 호출하고 일 수(1~366(윤년 포함))에 따라서 값을 저장한다. 각 일수마다  $\pm 30$ 일 값과 함께 리스트로 저장한다. 알고자 하는 일수와 같은 index 리스트에 온도를 추가하고 index를 반환, 리스트의 10퍼센트, 90퍼센트 이상인지 확인 후 맞으면 이상기온 리스트에 날짜와 온도를 추가한다. 마친후 추가한 값을 remove시켜준다.

### 3) 이상기온 출력 구현

<pre>for row in data :     if 1980==int(row[0][:4]):         day[row[0][5:10]] = num         temperature.append( [])         num+=1         standard.append(0)         if row[0][5:10]=='12-31':             break</pre>	<p>1980년 윤년인 해를 가지고 데이터의 기본 프레임을 만듭니다.</p> <p>day={} # 월일을 일수로 바꿔주는 딕션러리 temperature는 일수에 따른 온도 저장 standard는 일수에 따른 온도 저장</p>
<pre>for row in data :     if 1980&lt;int(row[0][:4])&lt;2011:         temperature[day[row[0][5:10]]].append(float(row[3]))         if row[0][5:10]!='02-29':             standard[day[row[0][5:10]]]+=float(row[3])/30         else :             standard[day[row[0][5:10]]]+=float(row[3])/6     if row[0]=='2010-12-31':         break</pre>	<p>평년 데이터를 리스트에다 저장하고 기준이 되는 평년의 평균 데이터를 저장</p>
<pre>for num in range(366):     temp=[]     for i in range(61):         temp+=temperature[(num+i-31)%366]     group.append(sorted(temp))</pre>	<p>일수마다 평년의 같은 일수를 가운데 놓고 모은 일정구간 리스트를 만들어줍니다.</p>
<pre>for row in data :     if row!=[]:         temp=copy(group[day[row[0][5:10]])         temp.append(float(row[3]))         temp.sort()         location=temp.index(float(row[3]))         if location&lt;183 or location&gt;1647:             # ubnormal.append([day[row[0][5:10]],row[0],float(row[3]),location])             year_ub.append([day[row[0][5:10]],float(row[3])])         temp.remove(float(row[3]))         if row[0][5:10]=='12-31':             output.append(sorted(year_ub))             year_ub=[]</pre>	<p>평년 다음 10년동안 년도마다 발생한 이상 기온을 검출하고, 년도마다의 리스트에 저장 시켜 줍니다.</p>

## 개별 plot

	<pre>plt.figure( dpi = 100) for i in output:     output_x=[]     output_y=[]     for j in i:         output_x.append(j[0])         output_y.append(j[1])     plt.plot(output_x,output_y,'ro')     output_year.append([output_x,output_y])</pre> <p>10년간 이상기온</p>
---	--

	<pre> want_know_year=int(input('알고 싶은 해의 이상기후(2011-2019) : ')) want_know_year-=2011  output_x=[] output_y=[] for j in output[want_know_year]:     output_x.append(j[0])     output_y.append(j[1]) </pre>	<p>지정 연도 이상기온</p>
	<pre> num_ub_day={} for i in output_year:     for j in i[0]:         temp=num_ub_day.get(j,0)         num_ub_day[j]=temp+1  x=list(num_ub_day.keys()) y=num_ub_day.values() </pre>	<p>일수에 따른 이상기온 횟수</p>
	<pre> for i in output_year:     temp=[]     for j in i[0]:         second=j         if second!=1 and first==1:             first=j         elif second!=third-1 and first!=1:             if first&gt;j-2:                 first=-1             else:                 temp.append([first,j-first])                 first=-1         elif second!=1 and first!=1:             third=second     when_ub_day.append(len(temp)) </pre>	<p>연도에 따른 이상기온 일수</p>
	<pre> for i in output_year:  num_ub_year.append(len(i[0] ])) print(num_ub_year) </pre>	<p>연도에 따른 이상기온 일수</p>