

S2E1: Parser de datos I.

Un fabricante de instrumentación meteorológica realiza estaciones de medida que envían datos como cadenas de texto a un servidor con el siguiente formato:

```
var1:val1-var2:val2-(...)-varN:valN
```

Siendo "varX" el nombre de una variable y "valX" el valor de la misma.

De esta forma si por ejemplo una estación de medida recoge datos de temperatura, humedad y precipitación (y la hora) enviará un mensaje como este:

```
hora:3-temp:275.4-hum:56.3-prec:47
```

1. Elabora una función que pasando como argumento la variable que se quiere extraer, devuelva el valor de la misma. Por ejemplo:

```
meteo_str = hora:3-temp:275.4-hum:56.3-prec:47
hora = extraer(meteo, "hora")
temp = int(extraer(meteo, "temp")) - 273.2 # A celsius
print(hora, "-", temp, "[°C]") # -> 3 - 2.2 [°C]
```

2. Usa la función del apartado 1 para extraer la media, máximo y mínimo de temperatura, la humedad relativa máxima y mínima y la precipitación acumulada en los datos recogidos por la estación en un día. Puedes encontrar un ejemplo de datos en el archivo "ejemploMeteo.txt" de la carpeta src del repositorio.
3. **[Extra - No oblig.]** Mejora la función del apartado 1 para que el valor que se retorne tenga el tipo y formato que se desee. Por ejemplo:

```
meteo_str = hora:4-lluvia:si-viento:8.4
# Los 3 puntos quieren decir cosas extra que puede llevar la
funcion
extraer_plus(meteo_str, "lluvia", ...) -> True sin conversión
extraer_plus(meteo_str, "viento", ...) -> 8.4 (float) sin
conversión.
```