



Esa tabla web, la transformamos en la siguiente base de datos.

4]:

	Horas	Temperatura	V_viento(km_h)	Date
0	22	19	16	2022-11-17
1	23	18	13	2022-11-17
0	0	18	9	2022-11-18
1	1	17	7	2022-11-18
2	2	16	9	2022-11-18
3	3	16	12	2022-11-18
4	4	15	14	2022-11-18
5	5	14	15	2022-11-18
6	6	14	16	2022-11-18
7	7	14	16	2022-11-18
8	8	14	15	2022-11-18

4. Contenido del Dataset

Para este dataset del tiempo, tenemos los siguientes datos:

- **Horas:** Columna de tipo int que nos da información de la hora en la cual se recogen cada uno de los datos de la tabla.
- **Temperatura:** Columna tipo int que nos devuelve la información de los grados que alcanzamos en cada hora del día.
- **Viento:** Columna tipo int con la información de la velocidad en Km/h del viento en Barcelona.
- **Date:** Columna tipo object con la fecha de la información en formato YYYY-MM-DD.

5. Propietario

No he encontrado un análisis previo del tiempo.es con web scrapping, lo que si que he intentado es simular un análisis de algunos parámetros meteorológicos que vi en el siguiente enlace: <https://levelup.gitconnected.com/weather-data-analysis-and-visualization-with-pandas-dec9c6078065> . Mi idea, era plasmar algo parecido utilizando herramientas de web scrapping, en este caso me decanté por Selenium.

6. Inspiración

Este conjunto de datos es interesante porque te permite monitorizar los principales datos meteorológicos existentes. Lo que te permite responder preguntas sobre todo de carácter predictivo.

¿Por ejemplo, la temperatura está relacionada con el consumo de ciertos productos?

¿Está relacionado el tiempo con la subida de casos de Covid u otras enfermedades?

O algunas preguntas de carácter prescriptivo:

¿Vemos una tendencia cambiante de temperaturas en relación con años anteriores? ¿A qué se debe?

Estás son algunas de las preguntas que se podrían responder con este dataset, el cual sirve como complemento en muchos tipos de análisis.

7. Licencia

La licencia escogida para la publicación de este conjunto de datos ha sido CC BY-SA 4.0 License. Los motivos para la elección de esta licencia han sido:

La persona es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato.

Adaptar — remezclar, transformar y crear a partir del material.

Y más detalladamente de:

- La persona que utilice los datos debe dar crédito de donde los está utilizando, dar un enlace a la licencia, e indicar si se han realizado cambios. Reconociendo de este modo el trabajo ajeno y viendo en que puntos han existido modificaciones.
- Si un tercero mezcla, transforma o crea nuevo material a partir de estos datos, éste podrá distribuir su contribución siempre que utilice la misma licencia que la obra original.
- Se permite un uso comercial. Esto haría que incrementen las probabilidades de que una empresa utilice los datos generados y realicen trabajos de calidad que reporten cierto reconocimiento al autor original.

8. Código Utilizado

Muestro ejemplo del código utilizado (la totalidad del mismo estará en el repositorio de GitHub).

```

# Inicializamos el navegador
driver.get('https://eltiempo.es')

WebDriverWait(driver, 5)\
.until(EC.element_to_be_clickable((By.CSS_SELECTOR,
'button.didomi-components-button didomi-button didomi-dismiss-button didomi-components-button--color didomi-
.click() #En este punto vamos a ir a buscar el botón de aceptar las cookies, añadiendo el enlace que nos proporciona el código html.

WebDriverWait(driver, 5)\
.until(EC.element_to_be_clickable((By.CSS_SELECTOR,
'input#term')))\
.send_keys('Barcelona') #En este punto escribimos Barcelona en el buscador

WebDriverWait(driver, 5)\
.until(EC.element_to_be_clickable((By.CSS_SELECTOR,
'i.icon.icon-search')))\
.click() # Una vez añadido Barcelona en el buscador, pulsamos en el icono de la lupa.

WebDriverWait(driver, 5)\
.until(EC.element_to_be_clickable((By.CSS_SELECTOR,
'i.icon.icon-sm.icon-city')))\
.click() # Pulsamos en el icono de ciudad, que nos dará la información relativa a Barcelona

WebDriverWait(driver, 5)\
.until(EC.element_to_be_clickable((By.XPATH,
'/html/body/div[5]/div[1]/div[4]/div/main/section[4]/section/div/article/section/ul[1]/li[2]/h2/a')))\
.click() # Pulsamos en la casilla de horas. Para ello debemos añadir el full Xpath del enlace.

# Por último, buscamos la tabla a descargar.
WebDriverWait(driver, 5)\
.until(EC.element_to_be_clickable((By.XPATH,
'/html/body/div[5]/div[1]/div[4]/div/section[4]/section/div[1]/ul'))

```

La principal dificultad que me he encontrado ha sido la de entender el proceso de funcionamiento de Selenium y conocer cada una de las funcionalidades que posee.

Otra de las dificultades que me he encontrado ha sido la de tratar los datos una vez los he extraído de la fuente principal, ya que he tenido que realizar diferentes operaciones para realizar la limpieza y obtener los datos finales.

9. Dataset

El dataset está disponible en el repositorio de Github y en Zenodo.

10. Github y Zenodo

Los enlaces a la información son:

<https://zenodo.org/record/7343486#.Y3vqDHbMJPY>

<https://github.com/jlagol/PEC2>

11. Video

El enlace a Google Drive es el siguiente:

https://drive.google.com/file/d/1L9ljL5jtYwtJ7xH2LwhKXOPY04mLVpp9/view?usp=share_link