



IIC2115 – Programación como Herramienta para la Ingeniería (II/2025)

Ejercicio Formativo 1 Capítulo 7

Aspectos generales

- **Objetivos:** Practicar el uso de APIs públicas y técnicas básicas de web scraping para obtener y procesar datos de fuentes externas utilizando Python.
- **Entrega:** lunes 17 de noviembre a las 17:30 hrs. en el repositorio privado y respondiendo el ticket de salida.
- **Formato de entrega:** archivo Python Notebook (**E1.ipynb**) con el avance logrado para el ejercicio. El archivo debe estar ubicado en la carpeta **C7**. Utilice múltiples celdas de texto y código para facilitar el trabajo del cuerpo docente.
- **ULTRA IMPORTANTE:** todas las celdas utilizadas deben estar ejecutadas al momento de entregar el ejercicio, de modo que las salidas generadas sean visibles. En caso de no cumplir con esto, su entrega no será considerada como validación del ticket de salida.

Introducción

Con el fin de practicar el uso de APIs y la extracción de datos de la web, en esta actividad deberá realizar una serie de consultas en tiempo real a la API de OpenWeatherMap y *scrapear* una pagina web para obtener datos climáticos de distintas ciudades.

Descripción de los datos

En este ejercicio, trabajará con información climática proveniente tanto de la API de OpenWeatherMap como de una página web pública. También contará con los datos complementarios, utilizados en el material

del curso, que contienen información geográfica y administrativa para enriquecer su análisis:

- **contries.csv**: Contiene información general por país, incluyendo nombre, códigos ISO, capital, población, área, región y continente.
- **cities.csv**: Contiene información detallada de ciudades, incluyendo sus coordenadas, país, estado y códigos ISO correspondientes.

Misión 1

Cree una función `get_weather(ciudad, api_key)` que reciba el nombre de una ciudad y una API key, y retorne un diccionario con la siguiente información del clima actual:

- Temperatura (en grados Celsius)
- Sensación térmica (**feels_like**)
- Descripción del clima
- Humedad (%)
- Velocidad del viento (m/s)

Utilice los parámetros `units=metric` y `lang=es` en la URL de consulta. Luego, construya una segunda función `get_weather_multi(ciudades, api_key)` que reciba una lista de nombres de ciudades y devuelva un **DataFrame** con la información climática para cada una.

Misión 2

Utilice su función para consultar el clima actual en al menos diez ciudades de cinco países distintos. Se deben tomar estas ciudades y países desde los conjuntos de datos complementarios, priorizando capitales o ciudades grandes de diferentes continentes.

El **DataFrame** resultante debe incluir las siguientes columnas:

[Ciudad, País, Temperatura, Sensación, Diferencia, Clima, Humedad, Viento]

La columna **Diferencia** debe representar la diferencia entre la temperatura real y la de sensación. Ordene la tabla de mayor a menor según esta diferencia y comente brevemente los resultados.

Misión 3

Enriquezca su análisis cruzando los datos del clima con la información contenida en `countries.csv`. Para cada país presente en la tabla construida en la misión anterior, agregue las columnas:

- `population` (población)
- `area` (superficie)
- `continent`
- `region`

Luego, agrupe o visualice los resultados por continente, y comente si observa diferencias relevantes en las condiciones climáticas entre regiones del mundo.

Misión 4

Desde Wikipedia u otra fuente confiable, realice web scraping de una tabla que contenga las temperaturas máximas históricas por país. Puede utilizar, por ejemplo:

https://es.wikipedia.org/wiki/Anexo:R%C3%A9cords_meteorol%C3%B3gicos_mundiales

Extraiga la tabla utilizando `pandas` y construya un `DataFrame` con las columnas relevantes.

Luego:

- Relacione los países presentes en su análisis con los del `countries.csv` para asegurar consistencia de nombres.
- Agregue una columna llamada `Máxima histórica` al `DataFrame` climático.
- Compare la temperatura actual de cada ciudad con la máxima histórica de su país.

Comente si los valores parecen razonables y qué dificultades presenta esta comparación (por ejemplo: diferencias de escala, fuente de los datos, ubicación específica de la medición, etc.).