



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

Ejercicio Formativo 2 Capítulo 7

Aspectos generales

- **Objetivos:** Aplicar Dash para construir una página web interactiva que permita visualizar datos y presentar información de manera dinámica y ordenada.
- **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 (**E2.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.

Descripción del problema

En este ejercicio desarrollará una aplicación interactiva utilizando la librería **Dash**, que permitirá consultar, visualizar y comparar condiciones climáticas actuales de distintas ciudades del mundo.

La aplicación debe construir su lógica de consulta utilizando las funciones creadas en el Ejercicio Formativo 1, e incorporar elementos visuales claros y funcionales.

Misión 1

Desarrolle una aplicación de Dash que permita al usuario ingresar el nombre de una ciudad (o seleccionarla desde una lista) y, al hacer clic en un botón, mostrar su información climática actual.

La app debe incluir:

- Un campo de entrada de ciudad (`dcc.Input` o `dcc Dropdown`).
- Un botón de actualización.
- Una tarjeta o recuadro con la información del clima: temperatura, sensación térmica, humedad, viento, descripción.

Se recomienda el uso de componentes de `dash-bootstrap-components` para organizar y estilizar la interfaz.

Misión 2

Extienda la aplicación para permitir la comparación entre múltiples ciudades. El usuario debe poder ingresar varias ciudades separadas por coma, o seleccionarlas desde un componente multiselección.

La app debe mostrar:

- Una tarjeta para cada ciudad, organizada de manera ordenada.
- Un gráfico de barras que compare la temperatura actual entre las ciudades seleccionadas.

Puede utilizar `plotly.express` para construir el gráfico de manera rápida y personalizable.

Misión 3

Incorpore al menos una de las siguientes funcionalidades adicionales, utilizando los datos complementarios sobre países y ciudades o la tabla obtenida por web scraping):

- Mostrar el continente o región del país asociado a cada ciudad.
- Comparar la temperatura actual con la temperatura máxima histórica del país.
- Ordenar automáticamente las tarjetas de ciudad según temperatura o diferencia térmica.
- Resaltar visualmente si la temperatura actual supera un umbral respecto a la histórica.

La implementación debe ser clara, funcional y estar documentada con comentarios en el código cuando sea necesario.