

MÓDULO INNOVACIÓN EN GESTIÓN DE DATOS

EVIDENCIA DE APRENDIZAJE N°3

En el mismo repositorio GitHub del grupo crear una carpeta llamada **evidencia3** dejar las resoluciones de las siguientes consignas:

Programación I:

Ampliar la aplicación Python de la evidencia 2, agregando las siguientes consignas:

1. Opciones de **ordenamiento**, permitir elegir por una de las siguientes dos formas, guardando en el archivo **usuarios.ispc** los usuarios ordenados solo si se elige esta opción (no es necesario mantener el orden en el crud, es solo al usar esta opción que debe ordenarse):
 - **técnica propia**: permitir ordenar los usuarios por sus username utilizando una de las 4 técnicas vistas (burbuja, selección, inserción o quick sort. A elección de ustedes) y
 - **ordenar por python**: usando el método `sort()` de python (usando key por username)
2. Buscar un usuario por username debe además agregar un mensaje diciendo que fue realizado por la técnica de búsqueda secuencial (si no se había ordenado previamente) o por técnica de búsqueda binaria (si se eligió ordenarlos)
3. **Carga de registros pluviales de un año**. Si se elige esta opción, se debe especificar un año y,
 - Si el archivo csv correspondiente a registro pluvial de dicho año se encuentra, entonces hacer uso del mismo, luego el usuario, debe elegir un mes y se debe mostrar en pantalla los registros de dicho mes.
 - Si el archivo de ese año no se encuentra registrado, entonces **cargar de manera aleatoria** (pueden usar la librería **random**) los registros pluviales de un año en una lista de listas, donde logré representar los 365 días del año. La lista principal debe ser una lista de 12 elementos, donde cada elemento es a su vez, una lista de 30 o 31 elementos (números que expresan los milímetros llovidos en ese día).

- Pasar esa lista a un archivo csv (llamar a los archivos como **registroPluvial2023.csv** por ejemplo).

Nota: Si bien en el segundo elemento de la lista principal se encontrará la lista de las lluvias en febrero, lista donde se deben cargar solo 28 o 29 datos tener en cuenta que **no** es necesario controles de fecha de año bisiesto, si desean que cada elemento de la lista principal sea de 31 elementos, es válido, siempre y cuando no ingresen un valor en el día 31 de aquellos meses que no tienen esa cantidad de días.

Opcional: Si desean utilizar **Pandas** pueden obviar el uso de listas de listas. Crear un **DataFrame**, en lugar de utilizar listas de listas, pueden crear un DataFrame de Pandas. Cada columna podría representar un mes y cada fila un día. Esto facilitará el manejo y análisis de los datos.

4. Hacer con **matplotlib** algunos gráficos. Deben ser al menos los siguientes:
 - Un gráfico de lluvias anuales usando un gráfico de **barras**,
 - otro de **dispersión** donde tenga en el eje x los meses (enero a diciembre) y en el eje y los días (1 a 31)
 - y el otro que sea uno **circular** que abarque todos los meses.

Base de Datos II:

- Presentar la configuración y conexión a base de datos desde el backend.
- Incorporar al backend las consultas y filtrados (CRUD y consultas complejas) que han sido presentadas y evaluadas en la Evidencia nro 2 según los objetivos y temática de su Proyecto.
- Contemplar las sugerencias detalladas en la Evidencia nro 2.

Fecha de Entrega: 23/10/24

Fecha de Entrega del recuperatorio: 30/10/24