
Parte 1. Lectura y estructuración de datos

Se le proporciona un archivo `registros.txt` con el siguiente formato (cada línea representa un registro):

```
Carlos,Math,85
Ana,Science,91
Carlos,Science,78
Luis,Math,66
Ana,Math,72
Luis,Science,88
```

- a. Escriba un programa `read_grades()` que lea el archivo y almacene los datos en una estructura adecuada (lista de listas o lista de diccionarios).

Ejemplo:

```
[
    ["Calos", "Math", 85],
    ["Ana", "Science", 91],
    ...
]
```

- b. Haga un programa `print_average()` que imprima el promedio de cada estudiante con formato `Ana:81.5`
- c. Haga un programa `write_average()` que escriba en un nuevo archivo `promedios.txt` los promedios de cada estudiante con el siguiente formato:

```
Ana: 81.5
Carlos: 81.5
Luis: 77.0
```

Extracto para copiar y pegar y conformar el archivo de ejemplo:

```
Carlos,Math,85
Ana,Science,91
Carlos,Science,78
Luis,Math,66
Ana,Math,72
Luis,Science,88
```

Parte 2. Bitácora de accesos

Escriba un programa `process_log()` que lea un archivo `bitacora.log` con líneas tipo:

```
[2024-06-01 10:15:22] Usuario Ana accedió al sistema
[2024-06-01 11:02:03] Usuario Carlos cerró sesión
[2024-06-01 11:30:45] Usuario Ana subió un archivo
```

Su programa debe llamarse `process_log()` y debe:

- Mostrar al usuario más activo (el que más registros tiene). Esto con un `print` del nombre en consola.
- Contar cuántas acciones realizó cada usuario y guardar el resultado en `user_activity.txt`. Por ejemplo:

```
Usuario Ana tiene 2 actividades
```

```
Usuario Carlos tiene 1 actividades
```

Extracto para copiar pegar en archivo `bitacora.log` como ejemplo:

```
[2024-06-01 10:15:22] Usuario Ana accedió al sistema
[2024-06-01 11:02:03] Usuario Carlos cerró sesión
[2024-06-01 11:30:45] Usuario Ana subió un archivo
```

Parte 3. Archivos como Matriz de Datos

El archivo `mapa.txt` contiene una matriz de caracteres que representa un mapa con obstáculos (#), camino libre (.), y puntos de interés (X):

```
.#...X.  
..##..  
X...#.  
.#X...
```

Haga un programa `read_map()` que:

- Lee el archivo y almacena la matriz en una lista de listas en memoria.
- Recorre la matriz e imprime las coordenadas de todos los puntos de interés (X). Hace un print de los puntos donde se ubica una X, en el ejemplo: (0,4) (2,0) (3,2)

Extracto para copiar y pegar para conformar el archivo de ejemplo:

```
.#...X.  
..##..  
X...#.  
.#X...
```