# ACTIVIDAD EN CLASE - SEMANA 2

## PROCESAMIENTO Y LIMPIEZA DE DATOS DE CLIENTES EN ARCHIVOS TXT, CSV, JSON Y XML UTILIZANDO PYTHON

## ****Objetivo General****

Aplicar estructuras de control, manipulación de cadenas, lectura y escritura de archivos, y manejo de formatos estructurados (CSV, JSON, XML), con el fin de desarrollar un flujo de procesamiento de datos en Python orientado a la limpieza, transformación y análisis básico de información de clientes.

## ****Descripción de la práctica****

El postgraduante desarrollará un script en Python que procese datos de entrada distribuidos en distintos tipos de archivos. A través del uso de condicionales, bucles, funciones integradas de cadenas, y bibliotecas estándar como json, xml.etree.ElementTree y pandas, se espera que el alumno sea capaz de normalizar datos, generar registros limpios y exportar resultados estructurados, simulando un escenario común de ingeniería de datos.

**Archivos proporcionados (incluidos en el archivo ZIP):**

|  |  |
| --- | --- |
| Nombre del archivo | Descripción |
| clientes.txt | Lista de nombres con errores comunes: espacios innecesarios, letras desordenadas |
| ventas.csv | Registros de ventas por cliente (Nombre, Monto, Fecha) |
| config.json | Parámetros de configuración del procesamiento (umbral, ordenamiento) |
| clientes.xml | Información adicional de clientes (nombre, edad, ciudad) |
| log.txt | Archivo para registrar las operaciones del script |

## ****Instrucciones de desarrollo****

### 1. ****Lectura y limpieza de**** clientes.txt

* Leer el archivo línea por línea ignorando líneas vacías o con espacios.
* Limpiar cada nombre utilizando strip() y title().
* Almacenar los nombres limpios en una lista y registrar en el log cuántos nombres únicos se obtuvieron.

### 2. ****Lectura de parámetros desde**** config.json

* Leer los parámetros "limite\_ventas" y "ordenar\_por" desde el archivo JSON.
* Imprimir estos valores como parte del log del script.

### 3. ****Procesamiento de**** ventas.csv ****con**** pandas

* Leer el archivo CSV con pandas.
* Filtrar ventas mayores o iguales al valor definido en "limite\_ventas".
* Agrupar por cliente y calcular el total de ventas por nombre.
* Ordenar según el campo "ordenar\_por".

### 4. ****Lectura de**** clientes.xml

* Utilizar xml.etree.ElementTree para extraer:
  + Nombre
  + Edad
  + Ciudad
* Vincular esta información con los nombres ya normalizados del paso 1.

### 5. ****Unificación de la información****

* Para cada cliente presente en los archivos, combinar la información:
  + Nombre limpio
  + Ciudad (desde XML)
  + Edad (desde XML)
  + Total de ventas (desde CSV)
  + Indicador es\_mayor\_edad (True si edad >= 18)
* Almacenar los datos combinados en una nueva estructura.

### 6. ****Exportación y generación de resultados****

* Exportar los datos finales a un archivo clientes\_final.csv.
* Escribir en log.txt el mensaje:
  + “Procesamiento finalizado con éxito.”
  + Fecha y hora de ejecución
  + Cantidad de registros procesados

## ****Resultados esperados****

* Archivo clientes\_final.csv limpio, consolidado y estructurado.
* Registro detallado de pasos realizados en log.txt.
* Manejo correcto de estructuras de control, tipos de datos, archivos y funciones de texto