Documentación Prueba Técnica DataKnow

# Sebastian Mariño

# 1. Carga de Información

La realización de este ejercicio fue desarrollada mediante el lenguaje de programación Python. En un primer acercamiento al problema, se utilizó la función open para abrir el archivo OFEI1204.txt ubicado en la carpeta '../Datos3'. El modo 'r' especifica que el archivo se abrirá en modo lectura. El contenido del archivo se leyó y almacenó en la variable txt\_data. Posteriormente, se imprimió el contenido del archivo para identificar las particularidades de la Data, en donde se notó que cada tabla de información venía delimitado por el titulo agente y líneas vacías.

Luego, se procedió a la extracción de información relevante del texto. Se dividió el contenido del archivo en secciones utilizando el método split con el delimitador "AGENTE:". Cada sección resultante contenía información sobre un agente específico. Se recorrieron estas secciones, extrayendo el nombre del agente y la tabla de datos asociada. La información de cada agente se agregó a la lista agentes, y las tablas de datos se procesaron y almacenaron en la lista dfs.

Después de completar la extracción de información, se utilizó la biblioteca pandas para manipular y analizar los datos de manera eficiente. Se concatenaron todas las tablas de datos en un único DataFrame llamado df\_final. Se filtraron las filas del DataFrame donde la columna "AGENTE" era igual a "EEPPM" para verificar la correcta lectura del archivo ya que se comparó esta salida con el archivo .txt original. Luego, se filtraron las filas donde la columna con índice 1 era igual a "D" y se reorganizaron las columnas. Se eliminó la columna 'TIPO' y se renombraron las columnas de acuerdo con los requisitos del problema.

Finalmente, se realizaron ajustes en el formato de algunas columnas, convirtiendo las columnas que representan horas a tipo de dato float. El DataFrame resultante se guardó en un archivo CSV llamado 'tabla\_punto\_1.csv' en la carpeta '../resultados'.

En resumen, el código realiza la extracción, procesamiento y manipulación de datos de un archivo de texto, utilizando Python y pandas, para obtener un DataFrame final que satisface los requisitos específicos del problema planteado.

# 2. Manipulación de datos

La implementación de este código en Python se realizó con el objetivo de manipular y analizar datos contenidos en un archivo Excel ('Datos Maestros VF.xlsx') y un archivo de texto ('dDEC1204.TXT'). Aquí se detalla paso a paso el procedimiento:

En primer lugar, se seleccionaron las columnas relevantes del archivo Excel utilizando la biblioteca pandas. Se leyó el archivo 'Datos Maestros VF.xlsx' y se extrajeron las columnas especificadas en la lista columnas\_seleccionadas. El DataFrame resultante se llamó df\_master\_data.

Luego, se filtraron las filas de df\_master\_data para obtener información específica sobre la empresa "EMGESA" y sobre centrales de tipo hidroeléctrico ("H") o térmico ("T"). El DataFrame resultante se denominó df\_master\_data\_emgesa.

A continuación, se leyó el archivo de texto 'dDEC1204.TXT' con la función pd.read\_csv, especificando el delimitador y la codificación adecuados. Se realizaron ajustes en el DataFrame resultante (df\_dec\_1204), eliminando la última fila y renombrando las columnas con nombres descriptivos.

Se llevó a cabo una fusión (pd.merge) entre los DataFrames df\_master\_data\_emgesa y df\_dec\_1204 utilizando la columna 'CENTRAL (dDEC, dSEGDES, dPRU…)' como clave de unión, y se especificó un tipo de fusión de tipo 'left'. El DataFrame resultante se llamó df\_merged\_emgesa.

A continuación, se seleccionaron las columnas que representan las horas ('Hora\_1', 'Hora\_2', ..., 'Hora\_24') y se calculó la suma de estas columnas para obtener la columna 'Total\_Horas'. Se filtraron las filas donde 'Total\_Horas' era mayor que 0, y el DataFrame resultante se llamó df\_tabla\_punto\_2.

Finalmente, se guardó el DataFrame df\_tabla\_punto\_2 en un archivo CSV llamado 'tabla\_punto\_2.csv' en la carpeta '../resultados'.

En resumen, este código realiza una serie de manipulaciones de datos utilizando pandas, fusionando información de un archivo Excel y un archivo de texto, y generando un DataFrame final que satisface los requisitos establecidos en el problema.

# 3. Prueba de SQL

Como primer paso se corrió la Query otorgada, la cual crea las tablas EMPLEADO,VACACIONES y se lleno con la información especificada.

Texto

Descripción generada automáticamente

Figura . Creación Tablas

A continuación, se muestran todas las Querys realizadas para obtener los datos solicitados y los respectivos resultados

1. **Seleccionar nombre, apellido y salario de todos los empleados:**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

Figura . Nombre apellido y salario

1. **Seleccionar nombre, apellido y salario de todos los empleados que ganen más de 4 millones:**

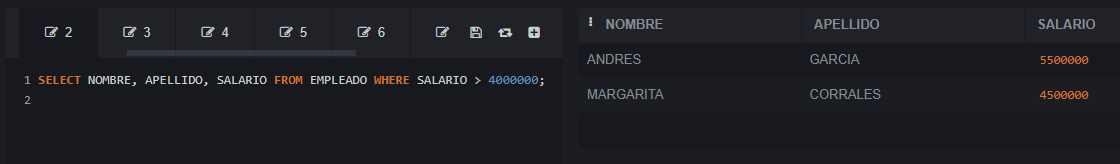
****

Figura . Empleados que ganas más de 4M

1. **Contar los empleados por sexo:**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

Figura . Conteo por sexo

1. **Seleccionar los empleados que no han hecho solicitud de vacaciones:**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

Figura . Empleados que no han solicitado vacaciones

1. **Seleccionar los empleados que tengan más de una solicitud de vacaciones y mostrar cuántas solicitudes tienen:**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

Figura . Empleados que han solicitado vacaciones más de una vez

1. **Determinar el salario promedio de los empleados:**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

Figura . Salario promedio

1. **Determinar la cantidad de días promedio solicitados de vacaciones por cada empleado:**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

Figura . Dias poromedio solicitados por empleado

1. **Seleccionar el empleado que ha solicitado la mayor cantidad de días de vacaciones:**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

Figura . Empleado que más ha solicitado vacaciones en total

1. **Consultar la cantidad de días aprobados y rechazados por cada empleado, mostrar 0 en caso de no tener solicitudes:**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura . Tabla de días aprobados y rechazados de vacaciones

# 4. Prueba de AWS