Documentación del Punto 2: Manipulación de Datos

Resumen

Este documento detalla el proceso seguido para resolver el Punto 2 de la prueba técnica, que consiste en cargar, procesar y fusionar datasets, y generar un resultado final basado en condiciones específicas.

Requerimientos

1. **Cargar datos maestros** desde un archivo Excel.
2. **Filtrar registros específicos**:
    - Agente: "EMGESA" o "EMGESA S.A.".
    - Tipo de central: "H" o "T".
3. **Cargar datos de producción por hora** desde un archivo de texto delimitado (dDEC1204.TXT).
4. **Realizar el merge** entre ambos datasets usando la columna central como clave.
5. **Calcular la suma horizontal** de las columnas de horas para cada registro.
6. **Filtrar registros con suma mayor que cero**.
7. **Guardar el resultado** en un archivo CSV.

Estructura del Código

El desarrollo se divide en dos archivos:

1. **main.py**: Controlador principal que gestiona la ejecución del pipeline.
2. **core.py**: Contiene funciones modulares que implementan cada paso del pipeline.

Descripción de las Funciones

1. load_master_data

Descripción: Carga los datos maestros desde un archivo Excel, normaliza los nombres de columnas eliminando caracteres especiales y aplica un mapeo para renombrarlas.

Entrada:

- file_path (str): Ruta del archivo Excel.

Salida:

- master_data (DataFrame): DataFrame con nombres de columnas normalizados y renombrados.

Código:

```
master_data = pd.read_excel(file_path)

master_data.columns = (

  master_data.columns.str.strip()

  .str.lower()

  .str.replace(' ', '_')

  .str.replace(r'[^\w]', '', regex=True)

  .map(remove_accents)

)

return master_data.rename(columns=RENAME_MAP)
```

---

2. filter_master_data

Descripción: Filtra el DataFrame maestro para incluir solo registros relevantes.

Lógica de Filtrado:

- nombre_visible_agente debe contener "EMGESA" o "EMGESA S.A.".
- tipo_de_central_hidro_termo_filo_menor debe ser "H" o "T".

Entrada:

- master_data (DataFrame): DataFrame procesado del archivo Excel.

Salida:

- filtered_data (DataFrame): DataFrame filtrado según los criterios especificados.

---

3. load_ddec_data

Descripción: Carga los datos de producción por hora desde un archivo delimitado por comas.

Detalles Adicionales:

- La codificación latin1 maneja caracteres especiales en el archivo de texto.
- Se asignan nombres a las columnas dinámicamente.

Código:

```
ddec_data = pd.read_csv(file_path, delimiter=',', header=None, encoding="latin1")

ddec_data.columns = ["central"] + [f"Hora_{i}" for i in range(1, 25)]
```

---

4. merge_datasets

Descripción: Realiza un merge entre los datos maestros filtrados y los datos de producción usando la columna central como clave.

Entrada:

- master_data (DataFrame): DataFrame filtrado de datos maestros.
- ddec_data (DataFrame): DataFrame con datos de producción por hora.

Salida:

- merged_data (DataFrame): DataFrame resultante de la unión de los datasets.

---

5. calculate_horizontal_sum

Descripción: Calcula la suma horizontal de las columnas de horas y filtra los registros con suma mayor a cero.

Código:

```
hour_columns = [col for col in data.columns if col.startswith('hora_')]
```

```
data['suma_horizontal'] = data[hour_columns].sum(axis=1)

filtered_data = data[data['suma_horizontal'] > 0]
```

---

6. save_results

Descripción: Guarda los resultados filtrados en un archivo CSV.

Entrada:

- data (DataFrame): DataFrame con los datos finales procesados.
- output_file (str): Ruta donde se guardará el archivo CSV.

Ejecución

Pasos para ejecutar el script:

1. Colocar los archivos de entrada en la carpeta raw_data.
2. Ejecutar el script main.py desde el entorno configurado.
3. Verificar los resultados en processed_data/filtered_results.csv.

Comando:

```
python main.py
```

Resultados Finales

El archivo final contiene los registros procesados y filtrados con las siguientes características:

- Columnas principales:
    - central, nombre_visible_agente, tipo_de_central_hidro_termo_filo_menor.
    - Columnas de horas (hora_1 a hora_24).
    - suma_horizontal para cada registro.
- Registros cuya suma horizontal de horas es mayor a cero.

Conclusión

El proceso implementado sigue un enfoque modular y asegura integridad en los datos manipulados. Cada paso está documentado y puede adaptarse fácilmente a nuevos requerimientos.