Proceso ETL (Extraer, Transformar, Cargar)

María Alejandra Marín Henríquez

SENA

ADSO : 2901817

09/08/2025

1 Instrucción

1.1 Investiga y define qué es un proceso ETL.

1.2 Explica la importancia del ETL en proyectos de análisis de datos

2 Tipos de herramientas ETL:

2.1 Investiga y describe al menos 3 herramientas ETL (pueden ser open source o comerciales).

2.2 Menciona ventajas y desventajas de cada una.

3 Actividad práctica:

3.1 Utiliza un archivo de datos de tu elección para realizar un proceso ETL básico:

- 3.1.1 **Extracción:** Lee los datos desde el archivo.
- 3.1.2 **Transformación:** Realiza al menos dos transformaciones (por ejemplo: limpieza de datos, cambio de formato, filtrado de columnas, etc.).
- 3.1.3 **Carga:** Exporta el resultado a un nuevo archivo (puede ser Excel, CSV o base de datos simple).

3.2 Puedes usar Python (pandas), Power BI, Talend, o la herramienta que prefieras.

4 Demostración:

4.1 Documenta el proceso realizado (puedes usar capturas de pantalla, código fuente, o un pequeño informe).

4.2 Explica los retos encontrados y cómo los resolviste.

5 Análisis solicitado:

5.1 Realiza un análisis simple sobre los datos transformados (por ejemplo: estadísticas descriptivas, gráficos, tendencias, etc.).

5.2 Expón tus conclusiones.

1. Introducción

Este informe se hace con el fin de desarrollar una actividad practica orientada a la compresión del proceso ETL (**Extracción, Transformación, y carga)** utilizando Python y la librería pandas.
El informe esta compuesto por la conceptualización del proceso, la descripción de la herramienta **ETL**, las practicas con un archivo **CSV** de ejemplo, un análisis de datos y por ultimo una breve conclusión.

1.1 Investiga y define qué es un proceso ETL.

ETL (extracción, trasformación y carga) es un proceso automatizado que se encarga de tomar datos que están robustos de varias fuentes, Los trasforma siguiendo sus tres pasos para una mejor organización de los datos. ETL es esencial para garantizar que los datos que se estas utilizando sean precisos, coherentes y relevantes.
Si no se llegase a realizar el proceso **ETL** los datos pueden terminar de manera incompleta, inexactos u obsoletos.

ETL cuentan con tres pasos para realizar una buena organización de los datos:

Extracción: Se encarga de obtener los datos de diversas fuentes como (Base de datos, Archivos, APIS etc…)

Trasformación: Una vez obtenido los datos se encarga de toda la parte de limpieza de estos modificándolos y estandarizándolos para una mejor calidad en los datos

Carga: Se encarga del almacenamiento de los datos procesados en un destino final como en (Data Waterhouse, Archivos, etc…)

1.2 Explica la importancia del ETL en proyectos de análisis de datos.

ETL es clave a la hora de realizar un proyecto de análisis de datos ya que nos permite la extracción de datos de múltiples fuentes, la trasformación de datos dependiendo de las necesidades del análisis y la carga de datos en un sistema almacenado adecuado.

ETL nos ayuda con la limpieza y normalización de los datos, eliminando toda duplicación, errores y unificando formatos.

Otras de las opciones que nos permite realizar ***ETL*** es la combinación de datos de diferentes fuentes, ***ETL*** es muy esencial para mantener una buena estructura de los datos siendo de alta calidad y confiable.

2. Tipos de herramientas ETL.

2.1 Investiga y describe al menos 3 herramientas ETL (pueden ser open source o comerciales)

2.2 Menciona ventajas y desventajas de cada una.

Herramientas	***Descripción***	***Ventajas***	***Desventajas***
Apache Airflow	Plataforma de código abierto que cuenta con una interfaz de usuario web y otra de línea de comando para gestionar activar flujos de trabajo.	Capacidad de escalar y gestionar flujos de trabajo complejos.	Aun que cuenta con una interfaz web su capacidad visualización y monitoreo puede llegar a hacer compleja.
Oracle Data Integrator	Ayuda al usuario a construir almacenes de datos complejos, cuenta con una serie de conectores listos para usar con muchas bases de datos	Ofrece todos los elementos de la integración de datos, desde el movimiento de datos hasta la sincronización, calidad y gestión.	Su licencia puede ser muy costosa para pequeñas y medianas empresas.

Pentaho Data Integration	Proyecto en apache para mover y transformar datos	Escalable, buen para el flujo en tiempo real.	Curva de aprendizaje alta.

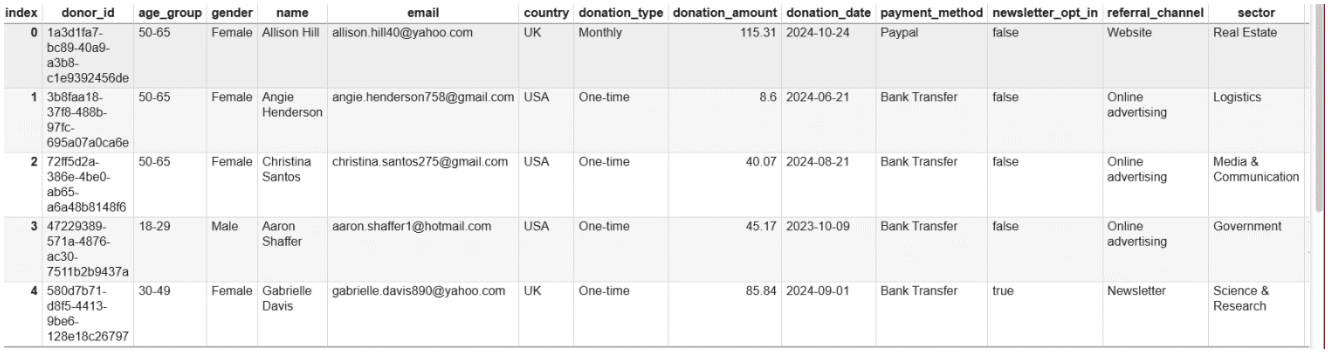
3. *Actividad práctica*

3.1 Utiliza un archivo de datos de tu elección para realizar un proceso ETL básico

Se utilizo un archivo llamado (***animal_charity_donation_records.csv***) con información que nos va ser útil para la actividad practica

3.2 Extracción: Lee los datos desde el archivo.

```
import pandas as pd
import matplotlib.pyplot as plt



df = pd.read_csv('/content/animal_charity_donation_records.csv')
display(df.head())
```



index	donor_id	age_group	gender	name	email	country	donation_type	donation_amount	donation_date	payment_method	newsletter_opt_in	referral_channel	sector
0	1a3d1fa7-bc89-40a9-a3b8-c1e9392456de	50-65	Female	Allison Hill	allison.hill40@yahoo.com	UK	Monthly	115.31	2024-10-24	Paypal	false	Website	Real Estate
1	3b8faa18-37f8-488b-97fc-695a07a0ca6e	50-65	Female	Angie Henderson	angie.henderson758@gmail.com	USA	One-time	8.6	2024-06-21	Bank Transfer	false	Online advertising	Logistics
2	72ff5d2a-386e-4be0-ab65-a6a48b8148f6	50-65	Female	Christina Santos	christina.santos275@gmail.com	USA	One-time	40.07	2024-08-21	Bank Transfer	false	Online advertising	Media & Communication
3	47229389-571a-4876-ac30-7511b2b9437a	18-29	Male	Aaron Shaffer	aaron.shaffer1@hotmail.com	USA	One-time	45.17	2023-10-09	Bank Transfer	false	Online advertising	Government
4	580d7b71-d8f5-4413-9be6-128e18c26797	30-49	Female	Gabrielle Davis	gabrielle.davis890@yahoo.com	UK	One-time	85.84	2024-09-01	Bank Transfer	true	Newsletter	Science & Research

3.3 Transformación: Realiza al menos dos transformaciones (por ejemplo: limpieza de datos, cambio de formato, filtrado de columnas, etc.).

```
# Contar el número de filas duplicadas en el DataFrame
#  devuelve una serie booleana indicando si cada fila está duplicada
# cuenta cuántos valores True hay, es decir, cuántas filas duplicadas existen
duplicate_rows = df.duplicated().sum()

# Imprimir el número total de filas duplicadas encontradas
print(f"Number of duplicate rows: {duplicate_rows}")
```

```
# Seleccionar las columnas numéricas del DataFrame
#  filtra solo las columnas cuyo tipo de dato sea numérico
# | obtiene únicamente los nombres de esas columnas
numerical_cols = df.select_dtypes(include=['number']).columns

# Imprimir un mensaje para indicar qué tipo de estadística se mostrará
print("Descriptive statistics for numerical columns:")

# Mostrar las estadísticas descriptivas (conteo, media, desviación estándar, mí
# sobre las columnas numéricas seleccionadas
display(df[numerical_cols].describe())
```

```
Descriptive statistics for numerical columns:
```

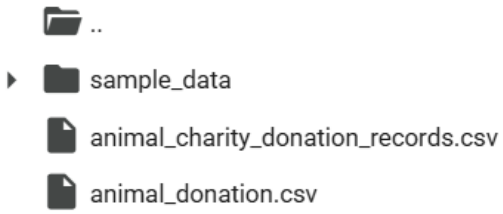
	donation_amount
count	10000.000000
mean	51.696998
std	56.745420
min	0.550000
25%	12.070000
50%	28.110000
75%	71.592500
max	702.930000

```python
# Convertir la columna 'donation_date' a formato de fecha y hora (datetime)
#  transforma los valores a objetos datetime, permitiendo realizar operaciones
df['donation_date'] = pd.to_datetime(df['donation_date'])

# Mostrar la información general del DataFrame
#  presenta el número de filas, columnas, nombres, tipo de datos y cantidad de
#   mostrar el resultado de forma más legible
display(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column             Non-Null Count  Dtype         
---  ------             --------------  -----         
 0   donor_id           10000 non-null  object        
 1   age_group          10000 non-null  object        
 2   gender             10000 non-null  object        
 3   name               10000 non-null  object        
 4   email              10000 non-null  object        
 5   country            10000 non-null  object        
 6   donation_type      10000 non-null  object        
 7   donation_amount    10000 non-null  float64       
 8   donation_date      10000 non-null  datetime64[ns]
 9   payment_method     10000 non-null  object        
 10  newsletter_opt_in  10000 non-null  bool          
 11  referral_channel   10000 non-null  object        
 12  sector             10000 non-null  object        
 13  campaign           10000 non-null  object        
dtypes: bool(1), datetime64[ns](1), float64(1), object(11)
memory usage: 1.0+ MB
None
```

3.4 Carga: Exporta el resultado a un nuevo archivo (puede ser Excel, CSV o base de datos simple).

```
display(df.head())

df.to_csv("animal_donation.csv", index=False)
```



4. Demostración

4.1 Explica los retos encontrados y cómo los resolviste.

En este punto se realizaron diferentes trasformaciones sobre el conjunto de datos para la preparación de la información antes de su análisis

1. ***Eliminación de duplicación:***
   En este punto nos encargamos de identificar y cuantificar filas repetidas en ***DataFrame***, utilizando el método ***duplicated()*** de la librería pandas junto con .sum() que nos permite contar cuantas filas duplicadas hay.

   Como resultado de la prueba identificamos que hay 0 filas duplicadas en el conjunto de datos.

2. ***Análisis de columnas numéricas:***
   Se realizo una selección de las columnas de tipo numérica para hacer el cálculo de las estadísticas descriptivas utilizando .***describe().*** Estos nos permiten obtener información como promedio, valores mínimos y máximos.

   Como resultado se detectó que la columna seleccionada en este caso ***donation_amount*** tiene :

Descriptive statistics for numerical columns:

	donation_amount
count	10000.000000
mean	51.696998
std	56.745420
min	0.550000
25%	12.070000
50%	28.110000
75%	71.592500
max	702.930000

3. ***Conversión de formatos de fechas :***
   Se convirtió la columna ***donation_date*** en el tipo de dato ***datetime[ns]*** utilizando ***pd.to_datetime().*** Eso nos facilita operaciones posteriores relacionadas con fechas, filtros por rango o años y mes.

   Como resultado la columna ***donation_date*** cambio a ***object*** a ***datetime[ns]*** lo que nos garantiza un majeño adecuado en este tipo de operaciones de análisis temporal.

5. *Análisis solicitado:*

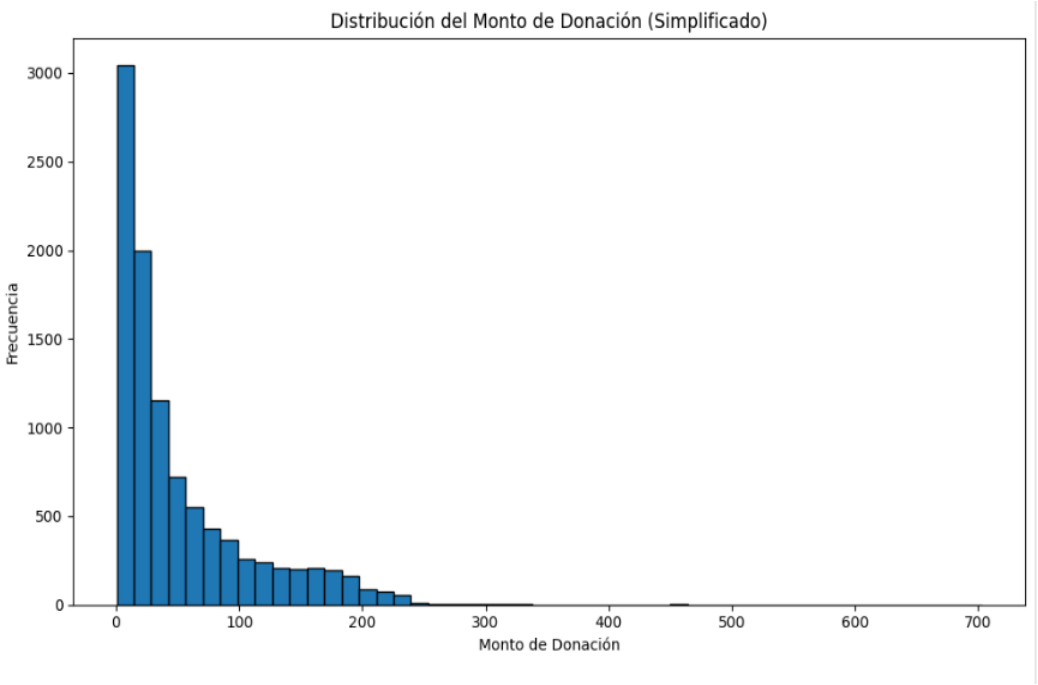
5.1 Realiza un análisis simple sobre los datos transformados (por ejemplo: estadísticas descriptivas, gráficos, tendencias, etc.).

- Estadística descriptiva:

	donation_amount
count	10000.000000
mean	51.696998
std	56.745420
min	0.550000
25%	12.070000
50%	28.110000
75%	71.592500
max	702.930000

- Grafico :

```
import pandas as pd
import matplotlib.pyplot as plt

# Cargar el conjunto de datos
df = pd.read_csv('/content/animal_donation.csv')

# Mostrar las primeras 5 filas del dataframe
display(df.head())

# Crear una figura y ejes únicos para el gráfico
fig, ax = plt.subplots(figsize=(10, 6))

# Crear un histograma de la columna 'donation_amount'
# bins=50: Especifica el número de contenedores para el histograma
# edgecolor='k': Establece el color del borde de las barras en negro
ax.hist(df['donation_amount'], bins=50, edgecolor='k')

# Establecer el título del histograma
ax.set_title('Distribución del Monto de Donación (Simplificado)')

# Establecer la etiqueta para el eje x
ax.set_xlabel('Monto de Donación')

# Establecer la etiqueta para el eje y
ax.set_ylabel('Frecuencia')

# Ajustar el diseño para evitar la superposición de etiquetas
plt.tight_layout()

# Mostrar el gráfico
plt.show()
```



5.2 Expón tus conclusiones.

- El proceso ETL nos permite obtener una mejor estructura y datos listos para el análisis ya que este mejora su calidad.
- Python con pandas es una buena herramienta para este tipo de casos básicos.
- La limpieza de datos es un paso muy importante para evitar conclusiones erróneas

6. Bibliografía

- ***www.datacamp.com***
- ***aws.amazon.com***
- ***www.linkedin.com***
- ***kompremos.com***

7. Anexos

- ***Archivo original animal_charity_donation_records.csv***
- ***Archivo prueba animal_donation.csv***
- ***Código Python usado***