Taller: Python para Excel Lovers

Francisco Alfaro

06 de Abril del 2024

¡Visita el siguiente link para avanzar en la sesión de hoy!

Documentación: Python para Excel Lovers

Tabla de Contenidos

¿ Qué es Excel?

Los archivos XLSX son un formato de archivo de hoja de cálculo desarrollado por Microsoft para su programa Microsoft Excel. Son una versión mejorada del formato de archivo XLS más antiguo.

Ventajas de Excel 

- Interoperabilidad
- Facilidad de uso para usuarios no técnicos
- Herramientas de análisis integradas

Desventajas de Excel

- Limitaciones de escalabilidad
- Dificultad para automatizar tareas
- Limitaciones de personalización

Solución

Utilizar herramientas Python (Pandas) para manipular archivos excel sin problemas!

Objetivos 🏆

- Comprender tipos de datos: `.xlsx`, `.csv`
- Conocer la librería de `Pandas`(`Python`)
- Relacionar `Pandas` con `Excel`

> El material de esta presentación lo puede encontrar en el siguiente Link: Python para Excel Lovers.

Tabla de Contenidos

CSV y Excel 

- **CSV (Comma-Separated Values)**:
  - CSV es un formato de archivo que se utiliza para almacenar datos tabulares, donde cada línea del archivo representa una fila de datos y los valores de cada fila están separados por comas u otros delimitadores.

- **Excel (XLSX/XLS)**:
  - Excel es una aplicación de hojas de cálculo desarrollada por Microsoft. Los archivos de Excel pueden contener múltiples hojas de cálculo, gráficos, fórmulas y datos formateados.

Comparación

Cuadro 1: Comparación entre archivos CSV y Excel

CSV	Excel
Los datos se almacenan como texto plano.	Los datos se organizan en hojas de cálculo.
Cada fila representa un registro, con campos separados por comas u otro delimitador.	Las hojas de cálculo pueden contener múltiples filas y columnas.
Carecen de capacidad de formateo.	Pueden contener formatos de celda, estilos, gráficos y más.
Ampliamente compatible con muchas aplicaciones y lenguajes de programación.	Principalmente compatible con aplicaciones de la suite Microsoft Office.

Tabla de Contenidos

Qué es Python 

- Lenguaje de programación de alto nivel
- Interpretado y de propósito general
- Fácil de aprender y de sintaxis clara
- Ampliamente utilizado en desarrollo de software, análisis de datos, inteligencia artificial, entre otros.

Qué es Pandas

- Biblioteca de Python para manipulación y análisis de datos
- Ofrece estructuras de datos flexibles y potentes, como DataFrames y Series
- Proporciona herramientas para leer y escribir datos en varios formatos, incluidos archivos CSV y Excel
- Facilita operaciones comunes en análisis de datos, como filtrado, agrupación y pivoteado

Qué es Openpyxl

- Librería de Python para leer y escribir archivos de Excel (XLSX/XLS)
- Permite manipular hojas de cálculo, filas, columnas y celdas en archivos de Excel
- Proporciona una interfaz sencilla y potente para trabajar con datos en formato Excel en Python
- Útil para automatizar tareas relacionadas con la manipulación de datos en hojas de cálculo

Google Colab

- **Google Colab** permite escribir y ejecutar código de Python en el navegador. Es adecuado para tareas de aprendizaje automático, análisis de datos y educación.
- No requiere configuración y que ofrece acceso sin coste adicional a recursos informáticos, como GPUs.

Tabla de Contenidos

Caso de Estudio

Trabajaremos con los datos de puntuaciones IMDB disponibles en Kaggle. El archivo contiene tres hojas: ’1900’, ’2000’ y ’2010’, cada una con datos de películas de los años correspondientes.

	A	B	C	D	E	F	G	H	I
1	**Title**	**Year**	**Genres**	**Language**	**Country**	**Content R**	**Duration**	**Aspect Ra**	**Budget**
2	127 Hours	2010	Adventure|Biography|Drama|Thril	English	USA	R	94	1.85	18000000
3	3 Backyards	2010	Drama	English	USA	R	88		300000
4	3	2010	Comedy|Drama|Romance	German	Germany	Unrated	119	2.35	
5	8: The Mormon Proposition	2010	Documentary	English	USA	R	80	1.78	2500000
6	A Turtle's Tale: Sammy's Adventures	2010	Adventure|Animation|Family	English	France	PG	88	2.35	
7	Alice in Wonderland	2010	Adventure|Family|Fantasy	English	USA	PG	108	1.85	200000000
8	Alice in Wonderland	2010	Adventure|Family|Fantasy	English	USA	PG	108	1.85	200000000
9	All Good Things	2010	Crime|Drama|Mystery|Romance|T	English	USA	R	101	1.85	
10	Alpha and Omega	2010	Adventure|Animation|Comedy|Fa	English	USA	PG	90	1.85	20000000
11	Amigo	2010	Drama|War	English	USA	R	124		1700000
12	Anderson's Cross	2010	Comedy|Drama|Romance	English	USA	R	98		300000
13	Animals United	2010	Animation|Comedy|Family	German	Germany	PG	93	2.39	

1900s 2000s 2010s

Paso 01: Intalar e Importar librerías

- **Instalar librerías**:
  - `pip install pandas openpyxl xlrd`
- **Importar librerías**

```
import pandas as pd
```

Paso 02: Leer las hojas del excel

```
# Especificar la ruta del archivo Excel
excel_file = 'data/movies.xls'

# Lee los nombres de las hojas
hojas = pd.ExcelFile(excel_file).sheet_names

# Imprimir la lista con el nombre de las hojas en excel
hojas
```

```
['1900s', '2000s', '2010s']
```

Paso 03: Leer los archivos individualmente

```
# Leer archivos
df_1900 = pd.read_excel(excel_file, sheet_name=hojas[0])
df_2000 = pd.read_excel(excel_file, sheet_name=hojas[1])
df_2010 = pd.read_excel(excel_file, sheet_name=hojas[2])
```

Paso 04: Juntar los archivos en un solo Dataframe

```
# Juntar los DataFrames en uno solo
df = pd.concat(
        [df_1900, df_2000, df_2010],
        ignore_index=True
)

# Mostrar las primeras filas del DataFrame
df.head()
```

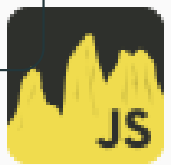
Paso 05: Guardar resultados en Excel

```
# Especifica la ruta donde deseas guardar el Excel
ruta_archivo_excel = 'data/movies2.xlsx'

# Crea un objeto ExcelWriter
with pd.ExcelWriter(ruta_archivo_excel) as writer:
  # Guarda el DataFrame en una hoja llamada '1900'
  df_1900.to_excel(writer,sheet_name='1900',index=False)

  # Guarda otro DataFrame en otra hoja llamada '2000'
  df_2000.to_excel(writer,sheet_name='2000',index=False)

  # Guarda otro DataFrame en otra hoja llamada '2010'
  df_2010.to_excel(writer,sheet_name='2010',index=False)
```



Funciones básicas: Explorar y visualizar los datos

- `df.head()` para mostrar las primeras filas del DataFrame.
- `df.tail()` para mostrar las últimas filas del DataFrame.
- `df.info()` para obtener información sobre el DataFrame.
- `df.describe()` para obtener estadísticas descriptivas del DataFrame.

Funciones básicas: Seleccionar y filtrar datos

- `df[columna]` para seleccionar una columna específica.
- `df.loc[fila, columna]` para seleccionar datos por etiquetas de fila y columna.
- `df.iloc[fila, columna]` para seleccionar datos por índices de fila y columna.
- `df[df['columna'] > valor]` para filtrar datos basados en una condición.

Funciones básicas: Operaciones de agrupación

- `df.groupby(columna)` para agrupar datos por una columna.
- `df.groupby(columna).sum()` para sumar los datos agrupados.
- `df.groupby(columna).mean()` para calcular la media de los datos agrupados.
- `df.groupby(columna).count()` para contar los datos agrupados.

Funciones básicas: Transformación de datos

- `pd.pivot_table()` para pivotar una tabla.
- `pd.concat()` para unir DataFrames.
- `pd.merge()` para combinar DataFrames.
- `df.apply()` para aplicar una función a cada elemento del DataFrame.

Funciones básicas: Manejar datos faltantes o duplicados

- `df.dropna()` para eliminar filas con datos faltantes.
- `df.fillna(valor)` para rellenar valores faltantes con un valor específico.
- `df.drop_duplicates()` para eliminar filas duplicadas.

Tabla de Contenidos

Resultados   

- Conocer conceptos básicos de archivos `.xlsx` y `.csv`.
- Conocer la librería de `Pandas`.
- Manipulación de Datos con `Python` y `Excel`.

Referencias

- Introducción a Python - Curso Github
- Análisis Exploratorio de Datos - Curso Github
- IMDB Movies Dataset - Kaggle

Taller: Python para Excel Lovers

Francisco Alfaro

06 de Abril del 2024