

Trabajar con openpyxl en Python

openpyxl es una biblioteca de Python para leer y escribir archivos Excel (xlsx). A continuación algunos ejemplos básicos de cómo usar openpyxl para realizar tareas comunes con archivos Excel.

1. Crear un archivo Excel y escribir datos

```
from openpyxl import Workbook

# Crear un nuevo libro de trabajo
wb = Workbook()

# Seleccionar la hoja activa
ws = wb.active

# Escribir datos en la celda A1
ws['A1'] = 'Hello'
ws['B1'] = 'World'

# Guardar el archivo
wb.save('example.xlsx')
```

2. Leer datos de un archivo Excel

```
from openpyxl import load_workbook

# Cargar el libro de trabajo existente
wb = load_workbook('example.xlsx')

# Seleccionar la hoja activa
```

```
ws = wb.active
```

```
# Leer datos de la celda A1  
data = ws['A1'].value  
print(data)  # Salida: Hello
```

```
# Leer datos de todas las celdas en la primera fila  
for cell in ws[1]:  
    print(cell.value)
```

3. Modificar celdas en un archivo Excel existente

```
from openpyxl import load_workbook
```

```
# Cargar el libro de trabajo existente  
wb = load_workbook('example.xlsx')
```

```
# Seleccionar la hoja activa  
ws = wb.active
```

```
# Modificar la celda A1  
ws['A1'] = 'Updated'
```

```
# Guardar los cambios en el mismo archivo  
wb.save('example.xlsx')
```



Untitled14.ipynb ☆

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda

Archivos



{x}



..

sample_data



example.xlsx



+ Código + Texto



0 s



```
from openpyxl import Workbook

# Crear un nuevo libro de trabajo
wb = Workbook()

# Seleccionar la hoja activa
ws = wb.active

# Escribir datos en la celda A1
ws['A1'] = 'Hello'
ws['B1'] = 'World'

# Guardar el archivo
wb.save('example.xlsx')

# Cargar el libro de trabajo
wb = load_workbook("example.xlsx")
ws = wb.active

# Leer datos
for row in ws.iter_rows(values_only=True):
    print(row)
```



('Hello', 'World')