

# Graficar calificaciones alumnos



Tengo un archivo llamado "calificaciones\_alumnos.xlsx", el cual tiene un campo llamado "Nombre" y varias columnas de calificaciones llamadas Mat\_CalculoIntegral, Mat\_ProgramacionOOP y Mat\_EstructuraDatos. Ayúdame a crear un programa en Python para graficar las calificaciones para cada alumno pero no quiero que se encimen sus etiquetas en el eje X, al inicio del código agrega un comentario explicando lo que hace el programa y documenta las líneas importantes del código



Claro, aquí tienes un programa en Python que grafica las calificaciones de los alumnos de un archivo Excel llamado "calificaciones\_alumnos.xlsx". El programa evita que las etiquetas en el eje X se encimen. He añadido comentarios para explicar cada sección del código.

python

```
# Este programa lee un archivo Excel que contiene las calificaciones de los alumnos en
# varias materias,
# y grafica dichas calificaciones para cada alumno, evitando que las etiquetas del eje X
# se encimen.

import pandas as pd
import matplotlib.pyplot as plt

# Cargar el archivo Excel
archivo = 'calificaciones_alumnos.xlsx'
datos = pd.read_excel(archivo)

# Obtener los nombres de los alumnos y sus calificaciones
nombres = datos['Nombre']
calificaciones = datos[['Mat_CalculoIntegral', 'Mat_ProgramacionOOP',
'Mat_EstructuraDatos']]

# Crear un gráfico de barras apiladas para cada alumno
fig, ax = plt.subplots(figsize=(10, 6))

# Definir el ancho de las barras y la posición de cada grupo de barras
bar_width = 0.25
index = range(len(nombres))

# Graficar las barras para cada materia
for i, columna in enumerate(calificaciones.columns):
    ax.bar(index, calificaciones[columna], bar_width, label=columna)

# Ajustar las etiquetas del eje X para que no se encimen
ax.set_xticks(index)
ax.set_xticklabels(nombres, rotation=45, ha='right')

# Añadir etiquetas y título
ax.set_xlabel('Alumnos')
ax.set_ylabel('Calificaciones')
ax.set_title('Calificaciones de los alumnos en distintas materias')
ax.legend()

# Mostrar el gráfico
```

```
plt.tight_layout()
plt.show()
```

Este código realiza las siguientes tareas:

1. **Importar librerías necesarias:** Utiliza ``pandas`` para manejar el archivo Excel y ``matplotlib.pyplot`` para generar el gráfico.
2. **Cargar el archivo Excel:** Lee el archivo Excel que contiene los nombres de los alumnos y sus calificaciones.
3. **Obtener nombres y calificaciones:** Extrae los nombres de los alumnos y sus calificaciones en diferentes materias.
4. **Crear el gráfico de barras apiladas:** Define las barras para cada materia y ajusta las etiquetas del eje X para que no se encimen, rotándolas 45 grados.
5. **Añadir etiquetas y título:** Añade etiquetas a los ejes y un título al gráfico.
6. **Mostrar el gráfico:** Ajusta el diseño del gráfico y lo muestra en pantalla.

Este programa generará un gráfico de barras apiladas que permite visualizar claramente las calificaciones de cada alumno sin que las etiquetas del eje X se encimen.



como instalo lo siguiente para python "openpyxl"



Para instalar la biblioteca ``openpyxl`` para Python, puedes usar el administrador de paquetes ``pip``. Abre una terminal o símbolo del sistema y ejecuta el siguiente comando:

```
sh
```

```
pip install openpyxl
```

Este comando descargará e instalará la biblioteca ``openpyxl`` y sus dependencias necesarias. Una vez completada la instalación, podrás usar ``openpyxl`` para trabajar con archivos Excel en Python.

Si estás usando un entorno de desarrollo integrado (IDE) como PyCharm, también puedes instalar ``openpyxl`` a través de la interfaz de usuario del IDE, generalmente en la sección de administración de paquetes o dependencias.

¿Te gustaría saber algo más específico sobre ``openpyxl``?



Tengo un archivo llamado "calificaciones\_alumnos.xlsx", el cual tiene un campo llamado "Nombre" y varias columnas de calificaciones llamadas Mat\_CalculoIntegral, Mat\_ProgramacionOOP y Mat\_EstructuraDatos. Ayúdame a crear un programa en Python para agregar una columna llamada Mat\_Fisica que contenga valores aleatorios entre 0 y 100 con un decimal, al inicio del código agrega un comentario explicando lo que hace el programa y documenta las líneas importantes del código



Claro, aquí tienes un programa en Python que agrega una columna llamada `Mat\_Fisica` con valores aleatorios entre 0 y 100 (con un decimal) a tu archivo Excel "calificaciones\_alumnos.xlsx". El programa está documentado con comentarios para explicar cada parte del código.

python

```
# Este programa lee un archivo Excel que contiene las calificaciones de los alumnos en
# varias materias,
# agrega una nueva columna llamada "Mat_Fisica" con valores aleatorios entre 0 y 100 (con
# un decimal),
# y guarda el archivo actualizado.

import pandas as pd
import numpy as np

# Cargar el archivo Excel
archivo = 'calificaciones_alumnos.xlsx'
datos = pd.read_excel(archivo)

# Generar valores aleatorios para la columna "Mat_Fisica"
np.random.seed(0) # Para reproducibilidad
datos['Mat_Fisica'] = np.round(np.random.uniform(0, 100, datos.shape[0]), 1)

# Guardar el archivo Excel con la nueva columna
archivo_actualizado = 'calificaciones_alumnos_actualizado.xlsx'
datos.to_excel(archivo_actualizado, index=False)

print(f'Archivo actualizado guardado como {archivo_actualizado}')
```

Este código realiza las siguientes tareas:

1. **Importar librerías necesarias:** Utiliza `pandas` para manejar el archivo Excel y `numpy` para generar valores aleatorios.
2. **Cargar el archivo Excel:** Lee el archivo Excel que contiene los nombres de los alumnos y sus calificaciones.
3. **Generar valores aleatorios:** Utiliza `numpy` para generar valores aleatorios entre 0 y 100 (con un decimal) y los asigna a la nueva columna `Mat\_Fisica`.
4. **Guardar el archivo Excel actualizado:** Guarda el archivo Excel con la nueva columna `Mat\_Fisica`.

El archivo actualizado se guardará con el nombre "calificaciones\_alumnos\_actualizado.xlsx". Puedes cambiar este nombre según tus necesidades.



como instala numpy en python



Para instalar la biblioteca `numpy` en Python, puedes utilizar el administrador de paquetes `pip`. Abre una terminal o símbolo del sistema y ejecuta el siguiente comando:

```
sh
```

```
pip install numpy
```

Este comando descargará e instalará la biblioteca ``numpy`` y sus dependencias necesarias. Una vez completada la instalación, podrás usar ``numpy`` en tus scripts de Python.

Si estás usando un entorno de desarrollo integrado (IDE) como PyCharm, también puedes instalar ``numpy`` a través de la interfaz de usuario del IDE, generalmente en la sección de administración de paquetes o dependencias.

Para verificar que la instalación fue exitosa, puedes abrir una terminal o un entorno interactivo de Python (como Jupyter Notebook o IPython) e intentar importar ``numpy`` con el siguiente comando:

```
python
```

```
import numpy as np
print(np.__version__)
```

Si no hay errores y se imprime la versión de ``numpy``, la instalación fue exitosa.



Para el nuevo archivo generado quiero que me ordene la tabla por medio del Nombre, al inicio del código agrega un comentario explicando lo que hace el programa y documenta las líneas importantes del código



Claro, aquí tienes el programa modificado para que ordene la tabla por el campo "Nombre" antes de guardar el archivo actualizado:

```
python
```

```
# Este programa lee un archivo Excel que contiene las calificaciones de los alumnos en
# varias materias,
# agrega una nueva columna llamada "Mat_Fisica" con valores aleatorios entre 0 y 100 (con
# un decimal),
# ordena la tabla por el campo "Nombre", y guarda el archivo actualizado.

import pandas as pd
import numpy as np

# Cargar el archivo Excel
archivo = 'calificaciones_alumnos.xlsx'
datos = pd.read_excel(archivo)

# Generar valores aleatorios para la columna "Mat_Fisica"
np.random.seed(0) # Para reproducibilidad
datos['Mat_Fisica'] = np.round(np.random.uniform(0, 100, datos.shape[0]), 1)

# Ordenar la tabla por el campo "Nombre"
datos = datos.sort_values(by='Nombre')

# Guardar el archivo Excel con la nueva columna y la tabla ordenada
archivo_actualizado = 'calificaciones_alumnos_actualizado.xlsx'
datos.to_excel(archivo_actualizado, index=False)
```

```
print(f'Archivo actualizado guardado como {archivo_actualizado}')
```

Este código realiza las siguientes tareas:

1. **Importar librerías necesarias:** Utiliza `pandas` para manejar el archivo Excel y `numpy` para generar valores aleatorios.
2. **Cargar el archivo Excel:** Lee el archivo Excel que contiene los nombres de los alumnos y sus calificaciones.
3. **Generar valores aleatorios:** Utiliza `numpy` para generar valores aleatorios entre 0 y 100 (con un decimal) y los asigna a la nueva columna `Mat\_Fisica`.
4. **Ordenar la tabla:** Ordena la tabla por el campo "Nombre".
5. **Guardar el archivo Excel actualizado:** Guarda el archivo Excel con la nueva columna `Mat\_Fisica` y la tabla ordenada por "Nombre".

El archivo actualizado se guardará con el nombre "calificaciones\_alumnos\_actualizado.xlsx". Puedes cambiar este nombre según tus necesidades.



Quiero saber cuántos registros y cuántos campos tiene mi tabla, al inicio del código agrega un comentario explicando lo que hace el programa y documenta las líneas importantes del código



Claro, aquí tienes el programa modificado para que también imprima la cantidad de registros (filas) y campos (columnas) de la tabla antes de agregar la nueva columna, ordenar la tabla y guardar el archivo actualizado.

python

```
# Este programa lee un archivo Excel que contiene las calificaciones de los alumnos en
# varias materias,
# imprime la cantidad de registros (filas) y campos (columnas) en la tabla, agrega una
# nueva columna
# llamada "Mat_Fisica" con valores aleatorios entre 0 y 100 (con un decimal), ordena la
# tabla por el
# campo "Nombre", y guarda el archivo actualizado.

import pandas as pd
import numpy as np

# Cargar el archivo Excel
archivo = 'calificaciones_alumnos.xlsx'
datos = pd.read_excel(archivo)

# Imprimir la cantidad de registros y campos
num_registros = datos.shape[0]
num_campos = datos.shape[1]
print(f'Cantidad de registros (filas): {num_registros}')
print(f'Cantidad de campos (columnas): {num_campos}')

# Generar valores aleatorios para la columna "Mat_Fisica"
np.random.seed(0) # Para reproducibilidad
datos['Mat_Fisica'] = np.round(np.random.uniform(0, 100, datos.shape[0]), 1)

# Ordenar la tabla por el campo "Nombre"
datos = datos.sort_values(by='Nombre')
```

```
# Guardar el archivo Excel con la nueva columna y la tabla ordenada
archivo_actualizado = 'calificaciones_alumnos_actualizado.xlsx'
datos.to_excel(archivo_actualizado, index=False)

print(f'Archivo actualizado guardado como {archivo_actualizado}')
```

Este código realiza las siguientes tareas:

1. **Importar librerías necesarias:** Utiliza `pandas` para manejar el archivo Excel y `numpy` para generar valores aleatorios.
2. **Cargar el archivo Excel:** Lee el archivo Excel que contiene los nombres de los alumnos y sus calificaciones.
3. **Imprimir la cantidad de registros y campos:** Calcula y muestra el número de filas y columnas de la tabla.
4. **Generar valores aleatorios:** Utiliza `numpy` para generar valores aleatorios entre 0 y 100 (con un decimal) y los asigna a la nueva columna `Mat\_Fisica`.
5. **Ordenar la tabla:** Ordena la tabla por el campo "Nombre".
6. **Guardar el archivo Excel actualizado:** Guarda el archivo Excel con la nueva columna `Mat\_Fisica` y la tabla ordenada por "Nombre".

El archivo actualizado se guardará con el nombre "calificaciones\_alumnos\_actualizado.xlsx". Puedes cambiar este nombre según tus necesidades.



Quiero saber qué campos son numéricos, al inicio del código agrega un comentario explicando lo que hace el programa y documenta las líneas importantes del código



Aquí tienes el programa actualizado para identificar qué campos en la tabla son numéricos. También he añadido comentarios explicativos al inicio del código y en las líneas importantes.

python

```
# Este programa lee un archivo Excel que contiene las calificaciones de los alumnos en
# varias materias,
# imprime la cantidad de registros (filas) y campos (columnas), identifica y muestra qué
# campos son numéricos,
# agrega una nueva columna llamada "Mat_Fisica" con valores aleatorios entre 0 y 100 (con
# un decimal),
# ordena la tabla por el campo "Nombre", y guarda el archivo actualizado.

import pandas as pd
import numpy as np

# Cargar el archivo Excel
archivo = 'calificaciones_alumnos.xlsx'
datos = pd.read_excel(archivo)

# Imprimir la cantidad de registros y campos
num_registros = datos.shape[0]
num_campos = datos.shape[1]
print(f'Cantidad de registros (filas): {num_registros}')
print(f'Cantidad de campos (columnas): {num_campos}')
```

```
# Identificar y mostrar qué campos son numéricos
campos_numericos = datos.select_dtypes(include=[np.number]).columns
```

```
print('Campos numéricos:')
for campo in campos_numericos:
    print(f'- {campo}')

# Generar valores aleatorios para la columna "Mat_Fisica"
np.random.seed(0) # Para reproducibilidad
datos['Mat_Fisica'] = np.round(np.random.uniform(0, 100, datos.shape[0]), 1)

# Ordenar la tabla por el campo "Nombre"
datos = datos.sort_values(by='Nombre')

# Guardar el archivo Excel con la nueva columna y la tabla ordenada
archivo_actualizado = 'calificaciones_alumnos_actualizado.xlsx'
datos.to_excel(archivo_actualizado, index=False)

print(f'Archivo actualizado guardado como {archivo_actualizado}')
```

Este código realiza las siguientes tareas:

1. **Importar librerías necesarias:** Utiliza `pandas` para manejar el archivo Excel y `numpy` para la generación de números aleatorios.
2. **Cargar el archivo Excel:** Lee el archivo Excel que contiene los nombres de los alumnos y sus calificaciones.
3. **Imprimir la cantidad de registros y campos:** Calcula y muestra el número de filas y columnas de la tabla.
4. **Identificar campos numéricos:** Usa `select\_dtypes` para seleccionar columnas de tipo numérico y las imprime.
5. **Generar valores aleatorios:** Crea una nueva columna `Mat\_Fisica` con valores aleatorios entre 0 y 100 (con un decimal).
6. **Ordenar la tabla:** Ordena la tabla por el campo "Nombre".
7. **Guardar el archivo Excel actualizado:** Guarda el archivo Excel con la nueva columna `Mat\_Fisica` y la tabla ordenada por "Nombre".

El archivo actualizado se guardará con el nombre "calificaciones\_alumnos\_actualizado.xlsx". Puedes cambiar este nombre según tus necesidades.

ChatGPT puede cometer errores. Comprueba la información importante.