

Python 3.12.6 (tags/v3.12.6:a4a2d2b, Sep 6 2024, 20:11:23) [MSC v.1940 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

```
import tkinter as tk
import csv
import tkinter.messagebox as messagebox
import os

# Función para guardar los datos en el archivo CSV
def save_data():
    name = entry_name.get()
    grade = entry_grade.get()

    # Verificar que los campos no estén vacíos
    if name == "" or grade == "":
        messagebox.showerror("Error", "Los campos no pueden estar vacíos.")
        return

    # Validar que la nota sea un número
    try:
        grade = float(grade)
    except ValueError:
        messagebox.showerror("Error", "La nota debe ser un número.")
        return

    # Guardar los datos en el archivo CSV
    with open('notas.csv', 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow([name, grade])

    # Limpiar los campos de entrada
    entry_name.delete(0, tk.END)
    entry_grade.delete(0, tk.END)

    # Mostrar mensaje de éxito
    messagebox.showinfo("Éxito", "Datos guardados correctamente.")

# Inicializar el archivo CSV con encabezados si no existe
if not os.path.exists('notas.csv'):
    with open('notas.csv', 'w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(["Nombre", "Nota"])

# Configuración de la interfaz gráfica
root = tk.Tk()
root.title("Gestor de Notas")
```

```

except ValueError:
    messagebox.showerror("Error", "La nota debe ser un número.")
    return

# Guardar los datos en el archivo CSV
with open('notas.csv', 'a', newline='') as file:
    writer = csv.writer(file)
    writer.writerow([name, grade])

# Limpiar los campos de entrada
entry_name.delete(0, tk.END)
entry_grade.delete(0, tk.END)

# Mostrar mensaje de éxito
messagebox.showinfo("Éxito", "Datos guardados correctamente.")

# Inicializar el archivo CSV con encabezados si no existe
if not os.path.exists('notas.csv'):
    with open('notas.csv', 'w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(["Nombre", "Nota"])

# Configuración de la interfaz gráfica
root = tk.Tk()
root.title("Gestor de Notas")

# Etiquetas y campos de entrada
label_name = tk.Label(root, text="Nombre:")
label_name.pack()
entry_name = tk.Entry(root)
entry_name.pack()

label_grade = tk.Label(root, text="Nota:")
label_grade.pack()
entry_grade = tk.Entry(root)
entry_grade.pack()

# Botón para guardar los datos
save_button = tk.Button(root, text="Guardar", command=save_data)
save_button.pack(pady=10)

# Dimensiones de la ventana
root.geometry("300x200")

# Ejecutar la interfaz gráfica
root.mainloop()
|

```