

```

import tkinter as tk
import random
import string

def generate_password():
    try:
        length = int(entry_length.get())
        if length < 4:
            show_error("The length must be at least 4 characters.")
            return

        characters = string.ascii_letters + string.digits #
Alphanumérique par défaut
        if special_chars_var.get():
            characters += string.punctuation # Ajoute les caractères
spéciaux

        password = ''.join(random.choice(characters) for _ in
range(length))

        entry_password.delete(0, tk.END) # Efface le champ de mot de
passe
        entry_password.insert(0, password) # Insère le mot de passe
généré
        error_label.config(text="") # Réinitialise le message
d'erreur
    except ValueError:
        show_error("Error, please enter a valid number.") # Gère les
erreurs de saisie

def toggle_special_chars():
    special_chars_var.set(not special_chars_var.get()) # Change
l'état des caractères spéciaux
    if special_chars_var.get():
        special_chars_button.config(text="Exclude the special
characters", bg="#90EE90") # Light green
    else:
        special_chars_button.config(text="Include the special
characters", bg="#32CD32") # Lime green

def show_error(message):
    error_label.config(text=message) # Affiche le message d'erreur
    blink_error(7) # Démarre le clignotement du message d'erreur

def blink_error(times):
    if times > 0:
        current_color = error_label.cget("fg")
        new_color = "#00FF00" if current_color == "#000000" else "#
000000" # Alterne entre vert et noir
        error_label.config(fg=new_color)

        # Continue le clignotement tous les 500 millisecondes
        root.after(500, lambda: blink_error(times - 1)) # Réduit le
compteur

```

```

        else:
            error_label.config(text="") # Efface le message d'erreur
            après le clignotement

def copy_to_clipboard():
    root.clipboard_clear() # Vide le presse-papiers
    root.clipboard_append(entry_password.get()) # Copie le mot de
    passe
    show_error("Password copied to clipboard!") # Affiche le message
    de confirmation

    # Efface les champs de saisie
    entry_length.delete(0, tk.END) # Efface le champ de longueur
    entry_password.delete(0, tk.END) # Efface le champ de mot de
    passe

# Création de la fenêtre principale
root = tk.Tk()
root.title("Bull PassWord Builder")
root.iconbitmap("cadenas.ico") # Remplacez "icone.ico" par le chemin
vers votre ico
root.geometry("400x400") # Taille de la fenêtre
root.configure(bg="#000000") # Fond noir

# Création de la barre de titre personnalisée
title_bar = tk.Frame(root, bg="#0e690c", relief="raised", bd=2) #
Vert foncé
title_bar.pack(fill=tk.X)

# Ajout du label de titre avec une police personnalisée
title_label = tk.Label(title_bar, text="Bull PassWord Builder", bg="#
0e690c", fg="#08f903",
                        font=("Agency FB", 14)) # Police
personnalisée
title_label.pack(pady=5)

# Label pour la saisie de la longueur du mot de passe
tk.Label(root, text="Length of the password:", bg="#000000", fg="#
00FF00", font=("Agency FB", 12)).pack(pady=5)

entry_length = tk.Entry(root, width=10, justify="center", bg="#
90EE90", fg="#000000", font=("Arial", 12)) # Fond vert clair,
largeur réduite
entry_length.pack(pady=5)

# Bouton pour inclure/exclure les caractères spéciaux
special_chars_var = tk.BooleanVar()
special_chars_button = tk.Button(root, text="Include the special
characters", bg="#32CD32", fg="#000000",
                                command=toggle_special_chars) #
Vert lime
special_chars_button.pack(pady=5)

# Bouton pour générer le mot de passe

```

```

tk.Button(root, text="Build Password", bg="#32CD32", fg="#000000",
command=generate_password).pack(
    pady=10) # Vert lime

# Champ pour afficher le mot de passe généré
password_label = tk.Label(root, text="Password generated:", bg="#
000000", fg="#00FF00", font=("Agency FB", 12),
                        anchor="center") # Texte vert
password_label.pack(pady=5)

# Champ d'entrée pour afficher le mot de passe généré
entry_password = tk.Entry(root, width=20, bg="#000000", fg="#00FF00",
font=("14 SegmentLED", 20),
                        justify="center") # Fond noir et texte
vert
entry_password.pack(pady=5)

# Bouton pour copier le mot de passe dans le presse-papiers
tk.Button(root, text="Copy to Clipboard", bg="#32CD32", fg="#000000",
command=copy_to_clipboard).pack(
    pady=5) # Vert lime

# Champ pour afficher les erreurs et les messages
error_label = tk.Label(root, text="", bg="#000000", fg="#00FF00",
font=("Agency FB", 12)) # Texte vert
error_label.pack(pady=5)

# Lancement de la boucle principale
root.mainloop()

```