

Project Overview:

This project is a simple password generator application built using Python and the Tkinter library for the graphical user interface (GUI). The program generates a random, secure password based on the user's specified length and provides an option to copy the generated password to the clipboard using the `pyperclip` library.

Libraries Used:

1. **tkinter:** A standard Python library used for creating graphical user interfaces.
2. **random:** A standard Python library used for generating random values.
3. **string:** A standard Python library used for accessing string constants like uppercase letters, lowercase letters, digits, and punctuation.
4. **pyperclip:** A third-party Python library used for copying text to the clipboard.

Code Explanation:

```
from tkinter import *  
import random, string  
import pyperclip
```

- **tkinter:** Imported to create the GUI components.
- **random:** Imported to generate random characters for the password.
- **string:** Imported to access various character sets (uppercase, lowercase, digits, punctuation).
- **pyperclip:** Imported to copy the generated password to the clipboard

```
root = Tk()  
root.geometry("400x400")  
root.resizable(0,0)  
root.title("DataFlair - PASSWORD GENERATOR")
```

- **Tk():** Creates the main application window.
- **geometry("400x400"):** Sets the window size to 400x400 pixels.
- **resizable(0, 0):** Disables window resizing by the user.
- **title("DataFlair - PASSWORD GENERATOR"):** Sets the title of the window.

```
Label(root, text='PASSWORD GENERATOR', font='arial 15  
bold').pack()
```

```
Label(root, text='DataFlair', font='arial 15  
bold').pack(side=BOTTOM)
```

- **Label:** Creates text labels in the window.
 - The first label displays the title "PASSWORD GENERATOR" at the top.
 - The second label displays "DataFlair" at the bottom of the window.

```
pass_label = Label(root, text='PASSWORD LENGTH',  
font='arial 10 bold').pack()  
pass_len = IntVar()  
length = Spinbox(root, from_=8, to_=32,  
textvariable=pass_len, width=15).pack()  
pass_str = StringVar()
```

- **Label:** Creates a label prompting the user to select the password length.
- **IntVar():** A Tkinter variable class that holds integer values. It is used to store the selected password length.
- **Spinbox:** A widget that allows the user to select a password length between 8 and 32 characters. The selected length is stored in `pass_len`.
- **StringVar():** A Tkinter variable class that holds string values. It is used to store the generated password.

```
def Generator():  
    password = ''  
    for x in range(0, 4):  
        password +=  
random.choice(string.ascii_uppercase)  
        password +=  
random.choice(string.ascii_lowercase)  
        password += random.choice(string.digits)  
        password += random.choice(string.punctuation)  
  
    for y in range(pass_len.get() - 4):  
        password +=  
random.choice(string.ascii_uppercase +  
string.ascii_lowercase + string.digits +  
string.punctuation)  
  
    pass_str.set(password)
```

- **Generator():** This function generates a random password:
 - **password = '':** Initializes an empty string to store the password.

- **for x in range(0, 4):** Ensures the password has at least one character from each character set (uppercase, lowercase, digit, punctuation).
- **for y in range(pass_len.get() - 4):** Fills the rest of the password with a random selection of characters based on the user-defined length.
- **pass_str.set(password):** Updates the `pass_str` variable with the generated password, which is displayed in the Entry widget.

```
def Copy_password() :
    pyperclip.copy(pass_str.get())
```

- **Copy_password():** Copies the generated password to the clipboard using the `pyperclip.copy()` function.

```
Button(root, text='GENERATE PASSWORD',
command=Generator).pack(pady=5)
Entry(root, textvariable=pass_str).pack()
Button(root, text='COPY TO CLIPBOARD',
command=Copy_password).pack(pady=5)
```

- **Button:**
 - The first button calls the `Generator` function to generate a password when clicked.
 - The second button calls the `Copy_password` function to copy the generated password to the clipboard.
- **Entry:** A widget that displays the generated password, which is bound to the `pass_str` variable.

```
root.mainloop()
```

- **root.mainloop():** Starts the Tkinter event loop, which keeps the application running and responsive to user interactions.

Conclusion:

This project provides a simple yet effective tool for generating random passwords, with the option to copy them to the clipboard for easy use. It's an excellent example of how to combine Python's standard libraries with third-party modules to create a functional application with a graphical user interface.