






File Edit Selection View Go Run ...


← →

Naresh

 v



—





app.py 1 X

12th, 14th -Inferential stats > Password Generator project, Tkinter project > app.py > ...


```
1  from tkinter import *
2  import random, string
3  import pyperclip
4
5  root = Tk()
6  root.geometry("400x400")
7  root.resizable(0,0)
8  root.title("PYTHON PROJECT  - PASSWORD GENERATOR")
9
10 Label(root, text='PASSWORD GENERATOR', font='arial 15 bold').pack()
11 Label(root, text='Python', font='arial 15 bold').pack(side=BOTTOM)
12
13 pass_label = Label(root, text='PASSWORD LENGTH', font='arial 10 bold').pack()
14 pass_len = IntVar()
15 length = Spinbox(root, from_=8, to=32, textvariable=pass_len, width=15).pack()
16 pass_str = StringVar()
17
18 def Generator():
19     password = []
20
21     # Ensuring at least one character from each type (Uppercase, Lowercase, Digits, Punctuation)
22     if pass_len.get() >= 4:
23         password.append(random.choice(string.ascii_uppercase))
24         password.append(random.choice(string.ascii_lowercase))
25         password.append(random.choice(string.digits))
26         password.append(random.choice(string.punctuation))
27
28     # Fill the rest with random choices until the specified length
29     for _ in range(pass_len.get() - 4):
30         password.append(random.choice(string.ascii_uppercase + string.ascii_lowercase + string.digits + string.punctuation))
```


Ln 4, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.1 64-bit Go Live

 37°C Sunny






Q Search





ENG IN



11:51  
14-04-2025

File Edit Selection View Go Run ...

← →

Naresh

app.py 1 x

12th, 14th -Inferential stats > Password Generator project, Tkinter project > app.py > ...

```
18 def Generator():
19
20     # Fill the rest with random choices until the specified length
21     for _ in range(pass_len.get() - 4):
22         password.append(random.choice(string.ascii_uppercase + string.ascii_lowercase + string.digits + string.punctuation))
23
24     # Shuffle to ensure randomness
25     random.shuffle(password)
26
27     else:
28         # If length is less than 4, just fill the required length with random choices
29         for _ in range(pass_len.get()):
30             password.append(random.choice(string.ascii_uppercase + string.ascii_lowercase + string.digits + string.punctuation))
31
32     # Convert list to string and set it to the variable
33     pass_str.set(''.join(password))
34
35
36 def Copy_password():
37     pyperclip.copy(pass_str.get())
38
39
40 Button(root, text='GENERATE PASSWORD', command=Generator).pack(pady=5)
41 Entry(root, textvariable=pass_str).pack()
42 Button(root, text='COPY TO CLIPBOARD', command=Copy_password).pack(pady=5)
43
44 root.mainloop()
```

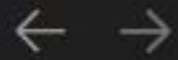
Ln 4, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.1 64-bit Go Live

37°C Sunny

Search

ENG IN 11:51 14-04-2025

Edit Selection View Go Run ...



PYTHON PROJECT - PASSWORD GENERA..



## PASSWORD GENERATOR

PASSWORD LENGTH

GENERATE PASSWORD

E%S5nW

COPY TO CLIPBOARD

Python

```
def Generator():
```

e Edit Selection View Go Run ... ← →

PYTHON PROJECT - PASSWORD GENERA.. — □ ×

# PASSWORD GENERATOR

PASSWORD LENGTH

9

GENERATE PASSWORD

h7E[+Aptj

COPY TO CLIPBOARD

Python

```
def Generator():
```