**TASK3-PASSWORD GENERATOR APPLICATION**

**PROJECT TITLE:**

**Password Generator GUI App  
  
DESCRIPTION:  
 A graphical user interface application built using CustomTkinter that allows users to generate strong, random passwords of user-specified length using letters, digits, and punctuation characters.  
  
FEATURES:  
1. CustomTkinter-based dark-themed UI.  
2.Input for password length.  
3.Validates for positive integers.  
4.Generates secure passwords using random and string modules.  
  
TECHNOLOGIES USED:  
 1.Python  
 2.CustomTkinter  
 3.random, string modules  
  
TARGET USERS:  
1.General users seeking strong password generation.  
2.Security and privacy-conscious individuals.**

**APPENDIX:**

**import customtkinter as ctk**

**import random**

**import string**

**#Creating the class for Password Generation**

**class PasswordGeneratorApp(ctk.CTk):**

**def \_\_init\_\_(self):**

**super().\_\_init\_\_()**

**self.title("PassWord Generator")**

**self.geometry("800x500")**

**self.resizable(False,False)**

**ctk.set\_appearance\_mode("dark")**

**ctk.set\_default\_color\_theme("dark-blue")**

**self.create\_widgets()**

**#Labeling all the elements in the page**

**def create\_widgets(self):**

**self.title\_label = ctk.CTkLabel(self,text="PASSWORD GENERATOR",font=("Arial",28,"bold"),text\_color="#5e35b1")**

**self.title\_label.pack(pady=20)**

**self.lenght\_frame = ctk.CTkFrame(self,fg\_color="transparent")**

**self.lenght\_frame.pack(pady=20)**

**self.lenght\_label=ctk.CTkLabel(self.lenght\_frame,text="Enter Password Length",font=("Arial",18),text\_color="#5e35b1")**

**self.lenght\_label.grid(row=0,column=0,padx=10)**

**self.lenght\_entry = ctk.CTkEntry(self.lenght\_frame,width=100,font=("Arial",18))**

**self.lenght\_entry.grid(row=1,column=0)**

**self.gen\_button= ctk.CTkButton(self,text="Generate Password",fg\_color="#5e35b1",command=self.generate\_password,width=200,)**

**self.gen\_button.pack(pady=20)**

**self.password\_out = ctk.CTkEntry(self,width=400,font=("Arial",18),justify="center")**

**self.password\_out.pack(pady=10)**

**#Function for Password generation using random module.**

**def generate\_password(self):**

**try:**

**length = int(self.lenght\_entry.get())**

**if length <=0:**

**self.password\_out.delete(0,'end')**

**self.password\_out.insert(0,"Enter positive number!")**

**return**

**chars = string.ascii\_letters + string.digits + string.punctuation**

**password = ''.join(random.choice(chars) for \_ in range(length))**

**self.password\_out.delete(0,'end')**

**self.password\_out.insert(0,password)**

**except ValueError:**

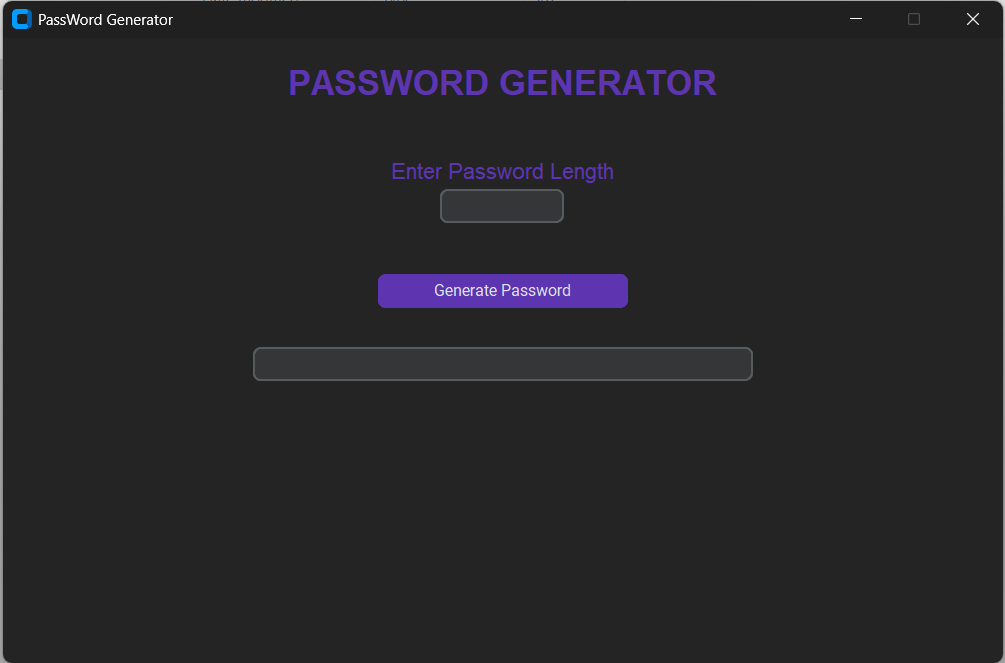
**self.password\_out.delete(0,'end')**

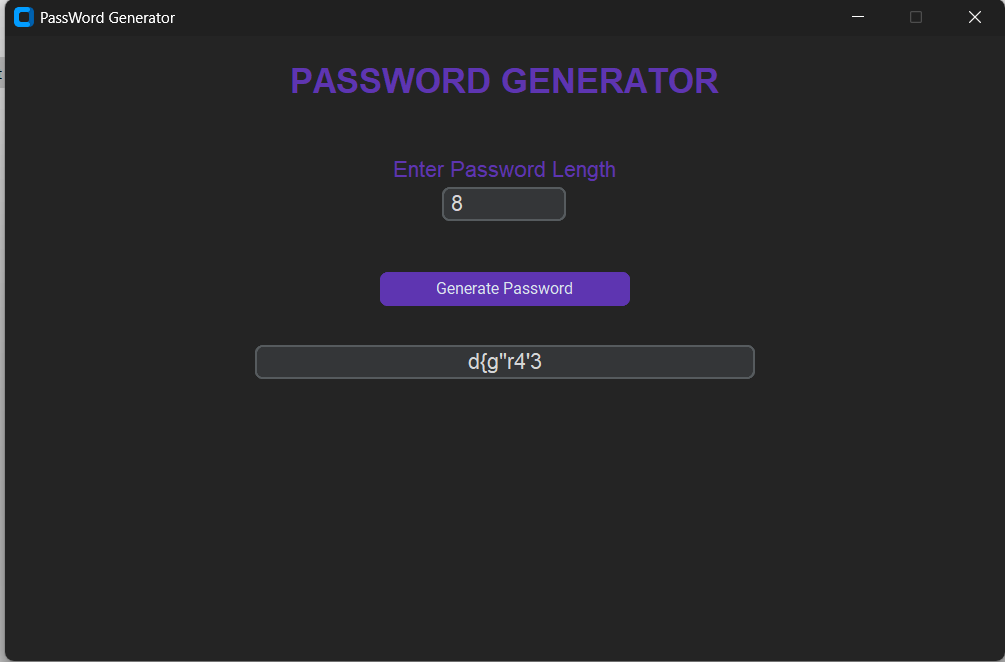
**self.password\_out.insert(0,"Please Enter a Valid Number!!")**

**app=PasswordGeneratorApp()**

**app.mainloop()**

**OUTPUTS:**

****

****