

VAULT OF CODES

Cybersecurity project

Python internship

Cybersecurity project :- Develop a password manager with strong encryption.

Source Code:-

```
import tkinter as tk
from tkinter import messagebox
from cryptography.fernet import Fernet
import random
import string

class PasswordManager:
    def __init__(self, master):
        self.master = master
        self.master.title("Password Manager")
        self.master.configure(bg='#B5E5CF')

        font_style = ("Helvetica", 12, "bold")
        search_bar_font = ("Helvetica", 14)

        # Initialize variables
        self.email_var = tk.StringVar()
        self.password_var = tk.StringVar()
        self.website_var= tk.StringVar()
        self.saved_passwords = []

        # GUI elements
        tk.Label(master, text="PASSWORD MANAGER ", bg="#3498db",font=font_style).grid(row=12, column=3, sticky=tk.E)
        tk.Label(master, text="Website:- ", bg="#3498db",font=font_style ).grid(row=15, column=2, sticky=tk.E)
        tk.Entry(master, textvariable=self.website_var, font=search_bar_font, width=20 ).grid(row=15, column=3,pady=25)

        tk.Label(master, text="Email/Username:- ", bg="#3498db", font=font_style).grid(row=18, column=2, sticky=tk.E)
        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20 ).grid(row=18, column=3, pady=15)

        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=20,
column=3, pady=15)

        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22,
column=3, columnspan=2, pady=16)
        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db",
font=font_style).grid(row=24, column=3, columnspan=2, pady=16)

        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db",
font=font_style).grid(row=26, column=3, columnspan=2, pady=16)

        tk.Button(master, text="check password strength", command=self.check_password_strength , bg="#3498db",
font=font_style).grid(row=28, column=3, columnspan=2, pady=16)

        # Key for encryption (you should securely store this in a real-world scenario)
        self.key = Fernet.generate_key()
        self.cipher_suite = Fernet(self.key)
```

```

def save_password(self):
    website = self.website_var.get()
    email = self.email_var.get()
    password = self.password_var.get()

    if email and password:
        self.saved_passwords.append({"website": website, "email": email, "password": password})
        messagebox.showinfo("Success", "Password saved successfully!")
    else:
        messagebox.showwarning("Error", "Please enter both email and password.")

def show_passwords(self):
    if self.saved_passwords:
        passwords_text = "\n".join([f"website: {entry['website']}, Email: {entry['email']}, Password: {entry['password']}"] for entry in self.saved_passwords)
        messagebox.showinfo("Saved Passwords", passwords_text)
    else:
        messagebox.showinfo("Saved Passwords", "No passwords saved yet.")

def generate_password(self):
    length = 12
    characters = string.ascii_letters + string.digits + string.punctuation
    generated_password = ''.join(random.choice(characters) for i in range(length))

    self.password_var.set(generated_password)
    messagebox.showinfo("Generated Password", f"Your generated password is:\n{generated_password}")

def check_password_strength(self):
    password = self.password_var.get()
    if len(password) < 8:
        strength = "Weak"
    elif len(password) < 12:
        strength = "Moderate"
    else:
        strength = "Strong"
    messagebox.showinfo("Password Strength", f"Password Strength: {strength}")

def encrypt_password(self, password):
    return self.cipher_suite.encrypt(password.encode())

def decrypt_password(self, encrypted_password):
    return self.cipher_suite.decrypt(encrypted_password.encode()).decode()

if __name__ == "__main__":
    root = tk.Tk()
    Password_Manager= PasswordManager(root)
    root.mainloop()

```

OUTPUT:- we execute the above code we can see like this.....

The screenshot shows a Windows desktop environment. On the left is a code editor window titled "password_manager.py" containing Python code for a password manager. The code uses Tkinter for a graphical user interface with fields for Website, Email/Username, and Password, and buttons for Save Password, Show Saved Passwords, Generate Password, and Check Password Strength. On the right is a terminal window showing the command "python password_manager.py" being run in the directory "C:\Users\gagan". The system tray at the bottom indicates it's 27°C and shows other icons like weather and battery status.

```
C: > Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=1, column=1, sticky=tk.E)
24        tk.Label(master, text="Website:-", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var, font=search_bar_font, width=20).grid(row=20, column=3, pady=15)
26
27        tk.Label(master, text="Email/Username:-", bg="#3498db", font=font_style).grid(row=21, column=2, sticky=tk.E)
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20).grid(row=21, column=3, pady=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=20, column=3, pady=15)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.E)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
```

➤ After that , we can enter the website ,email, password and then click on the save password.

This screenshot is identical to the one above, showing the same code editor and terminal windows. The difference is in the terminal output, which now shows the command "python password_manager.py" being run, indicating the application has been executed.

```
C: > Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=1, column=1, sticky=tk.E)
24        tk.Label(master, text="Website:-", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var, font=search_bar_font, width=20).grid(row=20, column=3, pady=15)
26
27        tk.Label(master, text="Email/Username:-", bg="#3498db", font=font_style).grid(row=21, column=2, sticky=tk.E)
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20).grid(row=21, column=3, pady=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=20, column=3, pady=15)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.E)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
128
129
129
130
131
132
133
134
135
136
137
```

The screenshot shows a Windows desktop environment. On the left is a code editor window titled "password_manager.py" containing Python code for a password manager. The code uses Tkinter for a graphical user interface, including fields for Website, Email/Username, and Password, along with buttons for Save Password, Show Saved Passwords, Generate Password, and Check Password Strength. On the right is a terminal window showing the command "python password_manager.py" being run in the directory "C:\Users\gagan". The terminal output includes the Python code and some initial logs. The taskbar at the bottom has various pinned icons.

➤ After that, we can see like this.....and then click on the “ok”.

This screenshot is similar to the previous one but shows a "Success" message dialog box in the foreground. The dialog says "Password saved successfully!" with an "OK" button. The rest of the interface and terminal window are identical to the first screenshot.

➤ Same like this...we can add another details....after that click on save password.

A screenshot of a Windows desktop environment. On the left is a code editor window titled "password_manager.py" showing Python code for a password manager. The code imports tkinter, messagebox, and fernet, and defines a PasswordManager class with methods for saving, generating, and checking passwords. On the right is a terminal window with the command "python password_manager.py" running, and the output shows the application's window titled "PASSWORD MANAGER". The window has fields for Website, Email/Username, and Password, along with buttons for Save Password, Show Saved Passwords, Generate Password, and check password strength.

A screenshot of a Windows desktop environment, similar to the previous one. The code editor window shows the same "password_manager.py" code. The terminal window now shows the application's window with a "Success" message box overlaid. The message box contains the text "Password saved successfully!" and an "OK" button. The rest of the application interface remains the same, with fields for Website, Email/Username, and Password, and buttons for Save Password, Show Saved Passwords, Generate Password, and check password strength.

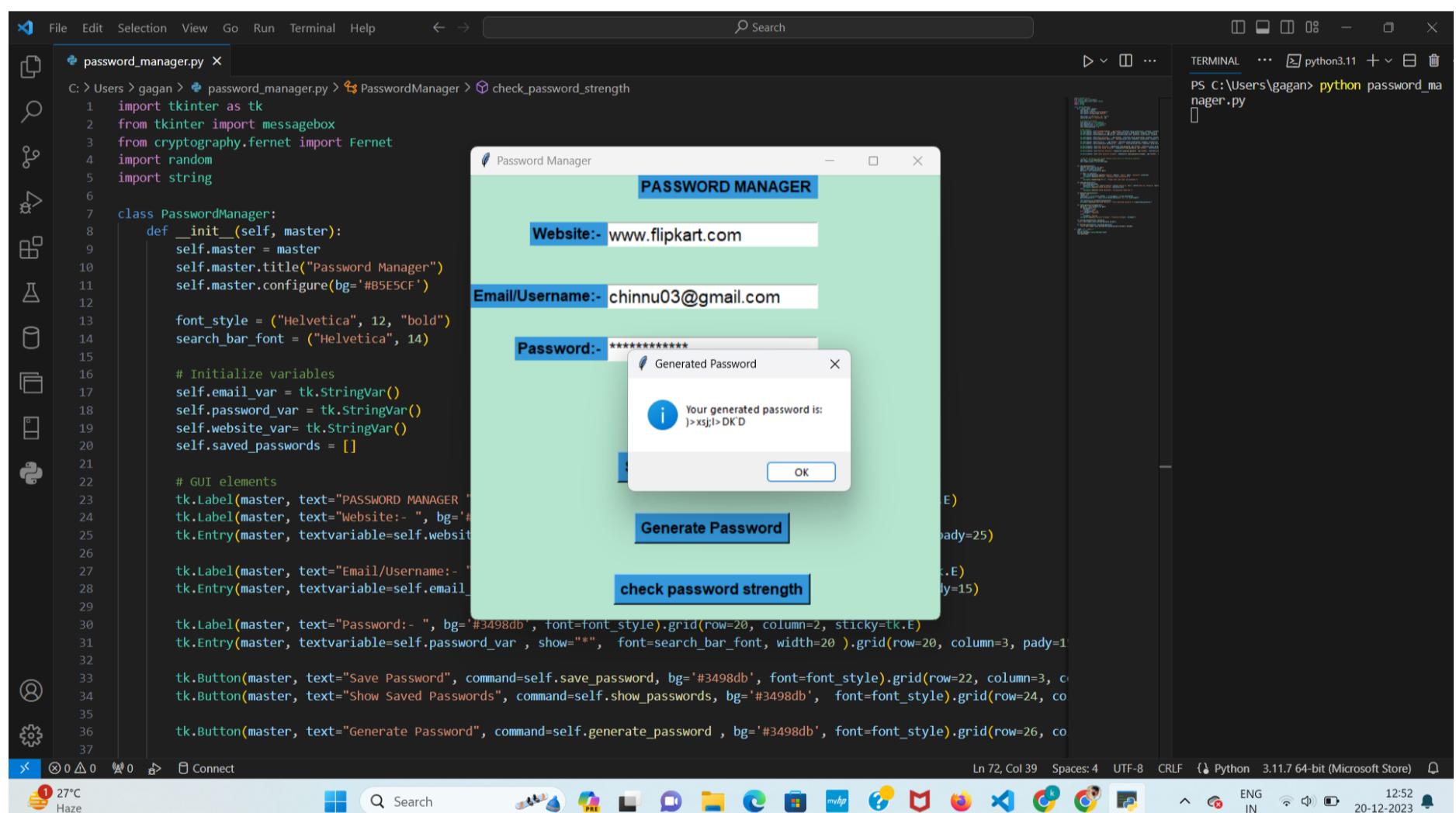
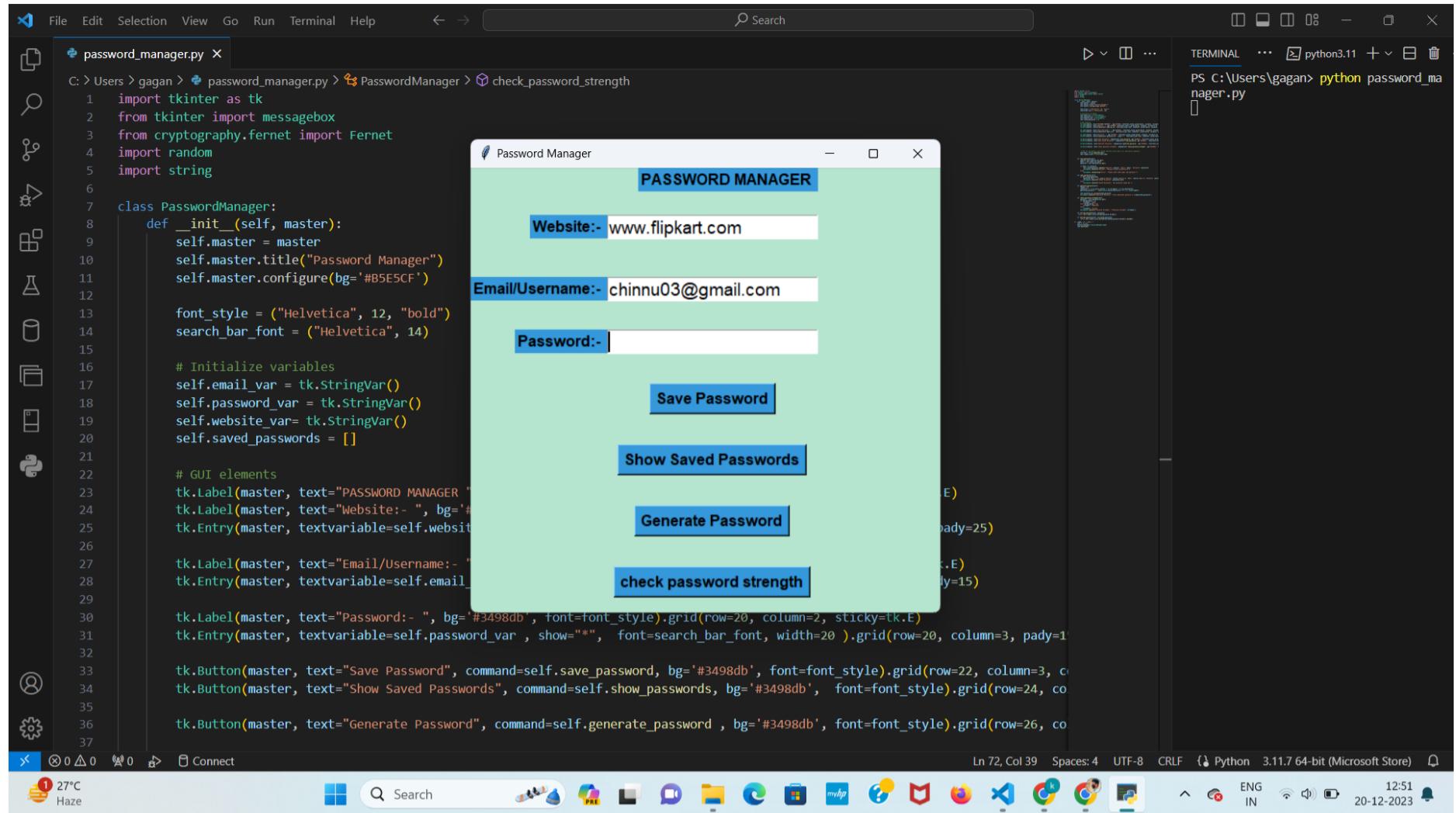
➤ NOW, we can click on the show passwords , then we can see the saved passwords.....

```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=0, column=0, sticky=tk.E)
24        tk.Label(master, text="Website:- ", bg="#3498db", font=font_style).grid(row=1, column=0, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var, font=search_bar_font).grid(row=1, column=1, pady=15)
26
27        tk.Label(master, text="Email/Username:- ", bg="#3498db", font=font_style).grid(row=2, column=0, sticky=tk.E)
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font).grid(row=2, column=1, pady=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=3, column=0, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=3, column=1, pady=15)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=4, column=0, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=5, column=0, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=6, column=0, columnspan=2, sticky=tk.E)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
```

➤ We can see like this.....

```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=0, column=0, sticky=tk.E)
24        tk.Label(master, text="Website:- ", bg="#3498db", font=font_style).grid(row=1, column=0, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var, font=search_bar_font).grid(row=1, column=1, pady=15)
26
27        tk.Label(master, text="Email/Username:- ", bg="#3498db", font=font_style).grid(row=2, column=0, sticky=tk.E)
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font).grid(row=2, column=1, pady=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=3, column=0, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=3, column=1, pady=15)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=4, column=0, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=5, column=0, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=6, column=0, columnspan=2, sticky=tk.E)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
```

➤ Now, we can generate the password, we can see like this... after that generated password and then click on "ok".



```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=0, column=0, sticky=tk.E)
24        tk.Label(master, text="Website:-", bg="#3498db", font=font_style).grid(row=1, column=0, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var).grid(row=1, column=1, padx=25)
26
27        tk.Label(master, text="Email/Username:-", bg="#3498db", font=font_style).grid(row=2, column=0, sticky=tk.E)
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20).grid(row=2, column=1, pady=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var, show="*", font=search_bar_font, width=20 ).grid(row=20, column=3, pady=15)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.E)
37
```

➤ We can see here, now check the strength of the password ...

```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=0, column=0, sticky=tk.E)
24        tk.Label(master, text="Website:-", bg="#3498db", font=font_style).grid(row=1, column=0, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var).grid(row=1, column=1, padx=25)
26
27        tk.Label(master, text="Email/Username:-", bg="#3498db", font=font_style).grid(row=2, column=0, sticky=tk.E)
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20).grid(row=2, column=1, pady=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var, show="*", font=search_bar_font, width=20 ).grid(row=20, column=3, pady=15)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.E)
37        tk.Button(master, text="check password strength", command=self.check_password_strength, bg="#3498db", font=font_style).grid(row=28, column=3, columnspan=2, sticky=tk.E)
```

➤ We can see here ,password is “strong”.

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface with the following details:

- File Explorer:** On the left, showing icons for files, folders, and other project components.
- Code Editor:** The main area displays the Python script `password_manager.py`. The code uses the `tkinter` library for a graphical user interface (GUI). It defines a `PasswordManager` class with methods for saving, generating, and checking passwords. It also includes a `check_password_strength` function that opens a separate window to evaluate the strength of the entered password.
- Terminal:** A right-hand panel titled "TERMINAL" shows the command `python password_manager.py` being run in the terminal, with the output of the password strength check visible.
- Preview:** A central preview window titled "Password Manager" shows the application's UI. It has fields for "Website:- www.flipkart.com", "Email/Username:- chinnu03@gmail.com", and "Password:-". A modal dialog box titled "Password Strength" displays the message "Password Strength: Strong".
- Status Bar:** At the bottom, it shows file status (0 changes), connection status, and system information like battery level (27°C Haze), language (ENG IN), volume, and date (20-12-2023).

➤ Same like this...we can check the password strength “strong or weak....”

The screenshot shows a code editor with a Python script titled `password_manager.py`. The script uses Tkinter to create a password manager application. The application has three input fields: "Website:-" with value `www.netflix.com`, "Email/Username:-" with value `dhana3@gmail.com`, and "Password:-" with value `*****`. It features four buttons: "Save Password", "Show Saved Passwords", "Generate Password", and "check password strength". The code editor's status bar indicates the file is 3.11.7 64-bit (Microsoft Store) and the terminal shows the command `python password_manager.py` was run.

```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
 1 import tkinter as tk
 2 from tkinter import messagebox
 3 from cryptography.fernet import Fernet
 4 import random
 5 import string

 6
 7 class PasswordManager:
 8     def __init__(self, master):
 9         self.master = master
10         self.master.title("Password Manager")
11         self.master.configure(bg="#B5E5CF")
12
13         font_style = ("Helvetica", 12, "bold")
14         search_bar_font = ("Helvetica", 14)
15
16         # Initialize variables
17         self.email_var = tk.StringVar()
18         self.password_var = tk.StringVar()
19         self.website_var= tk.StringVar()
20         self.saved_passwords = []
21
22         # GUI elements
23         tk.Label(master, text="PASSWORD MANAGER").grid(row=1, column=2, sticky=tk.E)
24         tk.Label(master, text="Website:- ", bg="#B5E5CF").grid(row=2, column=2, sticky=tk.E)
25         tk.Entry(master, textvariable=self.website_var).grid(row=2, column=3, pady=15)
26
27         tk.Label(master, text="Email/Username:- ")
28         tk.Entry(master, textvariable=self.email_var).grid(row=3, column=3, pady=15)
29
30         tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31         tk.Entry(master, textvariable=self.password_var, show="*", font=search_bar_font, width=20).grid(row=20, column=3, pady=15)
32
33         tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.W)
34         tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.W)
35
36         tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.W)
37
38         tk.Button(master, text="check password strength", command=self.check_password_strength, bg="#3498db", font=font_style).grid(row=28, column=3, columnspan=2, sticky=tk.W)
```

➤ Now, we can see the password “weak”.

```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=0, column=0, sticky=tk.E)
24        tk.Label(master, text="Website:- ", bg="#3498db").grid(row=1, column=0, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var).grid(row=1, column=1, sticky=tk.W, padx=25)
26
27        tk.Label(master, text="Email/Username:- ")
28        tk.Entry(master, textvariable=self.email_var).grid(row=2, column=0, sticky=tk.E, padx=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=20, column=3, pady=1)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.E)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
```

```
C:\> Users > gagan > password_manager.py > PasswordManager > check_password_strength
1 import tkinter as tk
2 from tkinter import messagebox
3 from cryptography.fernet import Fernet
4 import random
5 import string
6
7 class PasswordManager:
8     def __init__(self, master):
9         self.master = master
10        self.master.title("Password Manager")
11        self.master.configure(bg="#B5E5CF")
12
13        font_style = ("Helvetica", 12, "bold")
14        search_bar_font = ("Helvetica", 14)
15
16        # Initialize variables
17        self.email_var = tk.StringVar()
18        self.password_var = tk.StringVar()
19        self.website_var= tk.StringVar()
20        self.saved_passwords = []
21
22        # GUI elements
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=0, column=0, sticky=tk.E)
24        tk.Label(master, text="Website:- ", bg="#3498db").grid(row=1, column=0, sticky=tk.E)
25        tk.Entry(master, textvariable=self.website_var).grid(row=1, column=1, sticky=tk.W, padx=25)
26
27        tk.Label(master, text="Email/Username:- ")
28        tk.Entry(master, textvariable=self.email_var).grid(row=2, column=0, sticky=tk.E, padx=15)
29
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=20, column=2, sticky=tk.E)
31        tk.Entry(master, textvariable=self.password_var , show="*", font=search_bar_font, width=20 ).grid(row=20, column=3, pady=1)
32
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=22, column=3, columnspan=2, sticky=tk.E)
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=24, column=3, columnspan=2, sticky=tk.E)
35
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=26, column=3, columnspan=2, sticky=tk.E)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
```

➤ After that, click on the generate password then password will be generate then check the password strength.

The screenshot shows a Windows desktop environment. In the center is a Microsoft Visual Studio Code window displaying a Python script named `password_manager.py`. The code implements a password manager GUI using Tkinter. A terminal window to the right shows the command `python password_manager.py` being run. The terminal output indicates the script is executing successfully. The desktop taskbar at the bottom has several pinned icons, including File Explorer, Edge, and various Microsoft Office applications.

```
C:\> Users > gagan > password_manager.py > ...  
1 import tkinter as tk  
2 from tkinter import messagebox  
3 from cryptography.fernet import Fernet  
4 import random  
5 import string  
6  
7 class PasswordManager:  
8     def __init__(self, master):  
9         self.master = master  
10        self.master.title("Password Manager")  
11        self.master.configure(bg="#B5E5CF")  
12  
13        font_style = ("Helvetica", 12, "bold")  
14        search_bar_font = ("Helvetica", 14)  
15  
16        # Initialize variables  
17        self.email_var = tk.StringVar()  
18        self.password_var = tk.StringVar()  
19        self.website_var= tk.StringVar()  
20        self.saved_passwords = []  
21  
22        # GUI elements  
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=1, column=1, sticky=tk.E)  
24        tk.Label(master, text="Website:- ", bg="#3498db", font=font_style).grid(row=2, column=1, sticky=tk.E)  
25        tk.Entry(master, textvariable=self.website_var, font=search_bar_font, width=20).grid(row=2, column=2, sticky=tk.W, padx=10, pady=10)  
26  
27        tk.Label(master, text="Email/Username:- ", bg="#3498db", font=font_style).grid(row=3, column=1, sticky=tk.E)  
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20).grid(row=3, column=2, sticky=tk.W, padx=10, pady=10)  
29  
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=4, column=1, sticky=tk.E)  
31        tk.Entry(master, textvariable=self.password_var, show="*", font=search_bar_font, width=20).grid(row=4, column=2, sticky=tk.W, padx=10, pady=10)  
32  
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=5, column=1, columnspan=2, sticky=tk.E, ipady=15)  
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=6, column=1, columnspan=2, sticky=tk.E, ipady=15)  
35  
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=7, column=1, columnspan=2, sticky=tk.E, ipady=15)  
37  
Ln 72, Col 39 Spaces: 4 UTF-8 CRLF { Python 3.11.7 64-bit (Microsoft Store) 12:56 ENG IN 20-12-2023
```

This screenshot is nearly identical to the one above, showing the same `password_manager.py` code in VS Code and its execution in the terminal. The main difference is the state of the application window: it now displays a generated password in a modal dialog box. The dialog says "Your generated password is: kt=\$Q>ox6+<" and has an "OK" button. The rest of the application interface remains the same, with fields for website, email/username, and password, along with buttons for saving, generating, and checking password strength.

```
C:\> Users > gagan > password_manager.py > ...  
1 import tkinter as tk  
2 from tkinter import messagebox  
3 from cryptography.fernet import Fernet  
4 import random  
5 import string  
6  
7 class PasswordManager:  
8     def __init__(self, master):  
9         self.master = master  
10        self.master.title("Password Manager")  
11        self.master.configure(bg="#B5E5CF")  
12  
13        font_style = ("Helvetica", 12, "bold")  
14        search_bar_font = ("Helvetica", 14)  
15  
16        # Initialize variables  
17        self.email_var = tk.StringVar()  
18        self.password_var = tk.StringVar()  
19        self.website_var= tk.StringVar()  
20        self.saved_passwords = []  
21  
22        # GUI elements  
23        tk.Label(master, text="PASSWORD MANAGER").grid(row=1, column=1, sticky=tk.E)  
24        tk.Label(master, text="Website:- ", bg="#3498db", font=font_style).grid(row=2, column=1, sticky=tk.E)  
25        tk.Entry(master, textvariable=self.website_var, font=search_bar_font, width=20).grid(row=2, column=2, sticky=tk.W, padx=10, pady=10)  
26  
27        tk.Label(master, text="Email/Username:- ", bg="#3498db", font=font_style).grid(row=3, column=1, sticky=tk.E)  
28        tk.Entry(master, textvariable=self.email_var, font=search_bar_font, width=20).grid(row=3, column=2, sticky=tk.W, padx=10, pady=10)  
29  
30        tk.Label(master, text="Password:- ", bg="#3498db", font=font_style).grid(row=4, column=1, sticky=tk.E)  
31        tk.Entry(master, textvariable=self.password_var, show="*", font=search_bar_font, width=20).grid(row=4, column=2, sticky=tk.W, padx=10, pady=10)  
32  
33        tk.Button(master, text="Save Password", command=self.save_password, bg="#3498db", font=font_style).grid(row=5, column=1, columnspan=2, sticky=tk.E, ipady=15)  
34        tk.Button(master, text="Show Saved Passwords", command=self.show_passwords, bg="#3498db", font=font_style).grid(row=6, column=1, columnspan=2, sticky=tk.E, ipady=15)  
35  
36        tk.Button(master, text="Generate Password", command=self.generate_password , bg="#3498db", font=font_style).grid(row=7, column=1, columnspan=2, sticky=tk.E, ipady=15)  
37  
Ln 6, Col 1 Spaces: 4 UTF-8 CRLF { Python 3.11.7 64-bit (Microsoft Store) 13:03 ENG IN 20-12-2023
```

➤ Here, we can see the password is “strong”.

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface with the following details:

- File Explorer:** On the left, showing icons for files and folders.
- Code Editor:** The main area displays the `password_manager.py` file. The code uses the `tkinter` library to create a GUI for managing passwords. It includes functions for generating strong passwords and saving them to a list. The `Font` dropdown at the top is set to "Fira Code".
- Terminal:** On the right, showing the command `PS C:\Users\gagan> python password_manager.py` being run in a terminal window.
- Output:** Below the terminal, the output of the script is shown, including the creation of a `passwords.txt` file and the generation of a new password.
- >Password Manager Application:** A central window titled "PASSWORD MANAGER" is displayed. It has fields for "Website:- www.netflix.com", "Email/Username:- dhana3@gmail.com", and "Password:- *****". A "Password Strength" dialog box is open, stating "Password Strength: Strong".

THANK YOU

Submitted by

B.CHAITANYA