# Project Title

Password Strength Checker

# Objective

To build a Python program that checks the strength of a user's password and suggests a stronger one if needed using built-in libraries.

# Technologies Used

* + Python 3
  + Libraries: re (regex), string, random
  + CLI environment
  + (Optional) GUI using Tkinter

# How It Works

The program evaluates password strength by checking:

* + Length of the password (at least 8 characters)
  + Inclusion of uppercase and lowercase letters
  + Use of digits and special symbols

It rates the password as Weak, Medium, or Strong and suggests a randomly generated strong password if the entered one is weak or medium.

# Python Code

import re, string, random

def check\_strength(password): length\_error = len(password) < 8

digit\_error = re.search(r"\d", password) is None uppercase\_error = re.search(r"[A-Z]", password) is None lowercase\_error = re.search(r"[a-z]", password) is None

symbol\_error = re.search(r"[!@#$%^&\*(),.?\":{}|<>]", password) is None

score = 5 - sum([length\_error, digit\_error, uppercase\_error, lowercase\_error, symbol\_error]) return "Weak" if score <= 2 else "Medium" if score == 3 or score == 4 else "Strong"

def suggest\_password(length=12):

characters = string.ascii\_letters + string.digits + string.punctuation return ''.join(random.choice(characters) for \_ in range(length))

if name == " main ":

pwd = input("Enter your password: ") strength = check\_strength(pwd) print(f"Password Strength: {strength}") if strength != "Strong":

print("Suggested Strong Password: ", suggest\_password())

# Sample Output

Input: abc123 Output: Weak

Suggested: R@7!xP#eLd12

Input: Hello123 Output: Medium

Suggested: A2$xGq!BvT90

Input: Hello@2025 Output: Strong

# How to Run This Project

1. Install Python from https://python.org
2. Save the Python code as password\_strength\_checker.py
3. Open Command Prompt and run: cd path/to/your/project/folder

python password\_strength\_checker.py

1. Enter your password when prompted.

# Conclusion

This project helps users identify weak passwords and promotes the use of secure ones. It demonstrates basic Python skills and awareness of cyber hygiene.