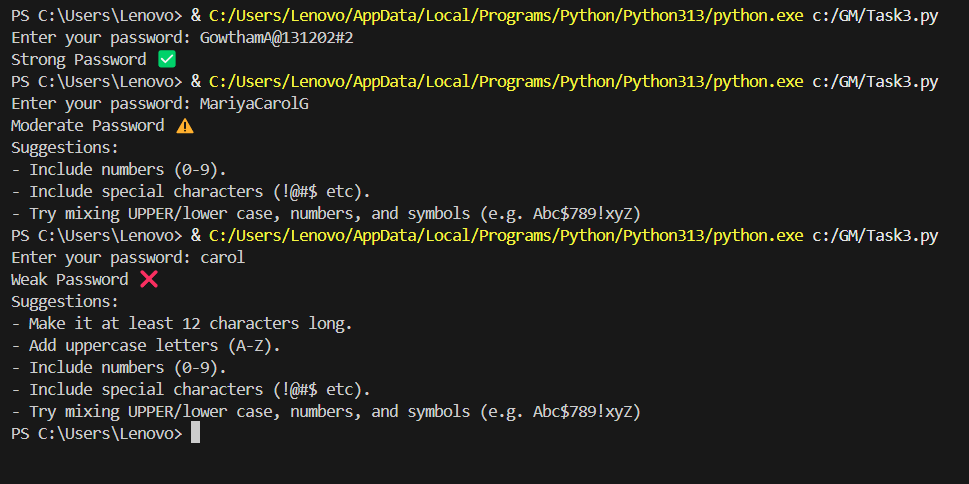
Task 3: Build a tool that assesses the strength of a password based on criteria such as length,presence of uppercase and lowercase letters, numbers and special characters.

Input code:



Output :



The working :

1. The code checks how strong a password is and gives helpful suggestions.
2. It starts with two sets of bad passwords — common ones and simple dictionary words.
3. It checks if the password:

* Is at least 12 characters long
* Has uppercase letters
* Has lowercase letters
* Has numbers
* Has special characters (like !@#)

1. Each of these checks adds 1 point to a strength score if passed.
2. If any check fails, a tip is added to the suggestions list.
3. Extra checks look for:

* Repeated characters (like aaa, 111)
* Common passwords (like 123456)
* Dictionary words (like apple, dragon)

1. If the password gets a full score and has no issues, it's marked **Strong .**
2. If it passes 3 or 4 checks, it's **Moderate**  with tips to improve.
3. If it passes less than 3 checks, it's **Weak**  with detailed suggestions.
4. The program asks for your password and prints the result with tips if needed.