DOCUMENTATION FOR PYTHON CODE

Name of the Intern : N.V.S.S HARSHAVARDHAN

Intern ID : SMI65278

Name of the Internship : Python

Level of the task done : Basic

Task Name : Password Generator

Software Used : Python IDLE 3.11 64 bit

Password Generator

This Python code implements generation of password for our accounts.

Features:

- **Secure Password Generation:** Creates strong passwords containing a mix of uppercase and lowercase letters, digits, and special characters.
- **Customizable Length:** Allows users to specify the desired length for each password, ensuring passwords meet complexity requirements.
- **Minimum Length Enforcement:** Enforces a minimum password length of 3 characters to prevent weak passwords.
- **User-Friendly Input:** Prompts users for the number of passwords and their individual lengths in a clear and concise manner.
- **Multiple Password Generation:** Generates multiple passwords in a single run, saving time for users who need a collection of strong passwords.

Code Structure:

1. **Import Libraries :** Imports random for random number generation and string for character sets.
2. **generate_password Function:**

    - Takes a desired password length (pw_length) as input.
    - Creates a string containing all character types.
    - Initializes an empty password list.

- Introduces variables to track minimum occurrences of character types (min_upper, min_lower, min_digit).
- Defines lists for character types (char_types) and their minimum counts (char_counts).
- Iterates until the desired password length is reached:
- Randomly chooses a character type.
- Ensures there's space for all required characters.
- Appends a random character from the chosen type to the password.
- Increments the count for that character type.
- Shuffles the password for enhanced security.
- Returns the generated password.

3. **main Function:**

- Prompts the user for the number of passwords to generate.
- Iterates for the specified number of passwords:
- Prompts for the length of each password, enforcing a minimum of 3.
- Appends the desired length to a list.
- Generates passwords based on individual lengths and stores them in a list.
- Prints each password with a corresponding number.

4. **if __name__ == "__main__": Block:**

- Ensures main runs only when the script is executed directly, not when imported as a module.

Running the Application:

1. Save the code as a Python file (e.g., todo.py).
2. Open a terminal or command prompt and navigate to the directory where you saved the file.
3. Run the script using python todo.py.

Python Code:

```
import random

import string
```

```
def generate_password(pw_length):
    # Generate a password with a mix of letters, digits, and special characters
    password_characters = string.ascii_letters + string.digits + string.punctuation
    password = []
    for i in range(pw_length):
        # Randomly choose a character type: letter, digit, or special character
        character_type = random.choice([string.ascii_letters, string.digits, string.punctuation])
        # Randomly choose a character from the chosen character type
        password.append(random.choice(character_type))
    # Ensure at least one uppercase letter, one lowercase letter, and one digit
    while (not any(c.isupper() for c in password) or
           not any(c.islower() for c in password) or
           not any(c.isdigit() for c in password)):
        password[random.randrange(len(password))] = random.choice(password_characters)
    return ''.join(password)
def main():
    num_passwords = int(input("How many passwords do you want to generate? "))
    print("Generating " + str(num_passwords) + " passwords")
    password_lengths = []
    for i in range(num_passwords):
        length = int(input("Enter the length of Password #" + str(i+1) + " "))
```

```
        if length < 3:
            print("Minimum length of password should be 3")
            length = 3
        password_lengths.append(length)
    passwords = [generate_password(length) for length in password_lengths]
    for i in range(num_passwords):
        print("Password #" + str(i+1) + " = " + passwords[i])
main()
```

Example Usage:

How many passwords do you want to generate? 2
Generating 2 passwords
Enter the length of Password #1: 8
Enter the length of Password #2: 12
Password #1 = XT2#q$1P
Password #2 = 7u4bzN!fJ9K*

This generates two passwords, one with a length of 8 and another with a length of 12.