

Python Registration and Login System Project

Project Description: Developed a strong consumer authentication machine in Python offering registration, login, and password recuperation using file coping with techniques.

Key Features:

Registration Validation: Implemented strict electronic mail/username validation, making sure proper format with rules along with no main special characters or numbers, right "@" and "." placement.

Password Security: Enforced steady password standards, consisting of period constraints (6-16 characters) and necessities for at least one unique character, digit, uppercase, and lowercase letter.

Data Storage: Utilized Python document dealing with to soundly shop person credentials for continual information control.

Login Authentication: Verified person credentials towards stored records with options for registration, password retrieval, and password update if present credentials matched.

Technical Skills: Python, record coping with, string manipulation, validation, conditional common sense.

Stage 1: Registration

→ Username Validation:

◆ Pattern:

- Must contain "@" and at least one "." after the "@".
- No special characters or numbers at the beginning.
- No "." immediately following the "@".
- Example of valid usernames: `sherlock@gmail.com`, `nothing@yahoo.in`.
- Example of invalid usernames: `@gmail.com`, `my@.in`, `123#@gmail.com`.

→ Password Validation:

◆ Criteria:

- Length should be between 6 and 16 characters.
- Must include at least one uppercase letter, one lowercase letter, one digit, and one special character.

→ Implementation:

- ◆ Prompt the user to enter a username and password.
- ◆ Validate the username and password based on the criteria above.
- ◆ If valid, proceed to Stage 2; if invalid, display an appropriate error message and re-prompt.

Stage 2: Storing Data

- Once the username and password are validated, store them in a file.
- Store data in a structured format, like **username:password**, so it can be easily retrieved.

Stage 3: Login

- User Login:
 - ◆ Prompt the user to enter their username and password.
 - ◆ Check if the credentials match any entry in the file.
- If Credentials Match:
 - ◆ Grant access and complete the login process.
- If Credentials Do Not Match:
 - ◆ Ask the user if they would like to:
 - Register if they don't have an account.
 - Forget Password if they have an account but forgot their password.
- Forgot Password Option:
 - ◆ Prompt for their username.
 - ◆ If the username exists, retrieve and display the original password or allow them to set a new password.
- If Username Not Found:
 - ◆ Prompt the user to register since they don't have an account.