Kevin Jonson

2311104021

15

Register

kevinjonson

••••••••

Register

Login

kevinjonson

••••••••

Login

Login berhasil!

```html
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8" />
  <title>Simple Auth App</title>
  <style>
    body { font-family: sans-serif; padding: 20px; max-width: 400px; margin: auto; }
    input, button { display: block; width: 100%; margin: 10px 0; padding: 8px; }
    .status { margin-top: 15px; color: red; font-weight: bold; }
  </style>
</head>
<body>
  <h2>Register</h2>
  <input type="text" id="reg-username" placeholder="Username" />
  <input type="password" id="reg-password" placeholder="Password" />
  <button onclick="register()">Register</button>

  <h2>Login</h2>
  <input type="text" id="log-username" placeholder="Username" />
  <input type="password" id="log-password" placeholder="Password" />
  <button onclick="login()">Login</button>

  <div class="status" id="status"></div>

  <script src="app.js"></script>
</body>
</html>
```

```javascript
function hashPassword(password) {
  return btoa(password); // NOT secure, hanya base64 (untuk demo)
}

function validate(username, password) {
  const asciiOnly = /^[A-Za-z]+$/;
  const specialChar = /[!@#$%^&*]/;
  const digit = /\d/;
  const errors = [];

  if (!asciiOnl any t(username)) errors.push("Username hanya huruf A-Z");
  if (username.length < 4 || username.length > 20) errors.push("Username harus 4-20 karakter");
  if (password.length < 8 || password.length > 20) errors.push("Password harus 8-20 karakter");
  if (!specialChar.test(password)) errors.push("Password perlu karakter unik (!@#$%^&*)");
  if (!digit.test(password)) errors.push("Password harus mengandung angka");
  if (password.toLowerCase().includes(username.toLowerCase())) errors.push("Password tidak boleh mengandung username"

  return errors;
}

function register() {
  const username = document.getElementById("reg-username").value.trim();
  const password = document.getElementById("reg-password").value.trim();
  const users = JSON.parse(localStorage.getItem("users") || "[]");

  const errors = validate(username, password);
  if (errors.length > 0) return showStatus(errors.join("\n"));

  if (users.find(u => u.username === username)) {
    return showStatus("Username sudah digunakan");
  }

  users.push({ username, password: hashPassword(password) });
  localStorage.setItem("users", JSON.stringify(users));
```