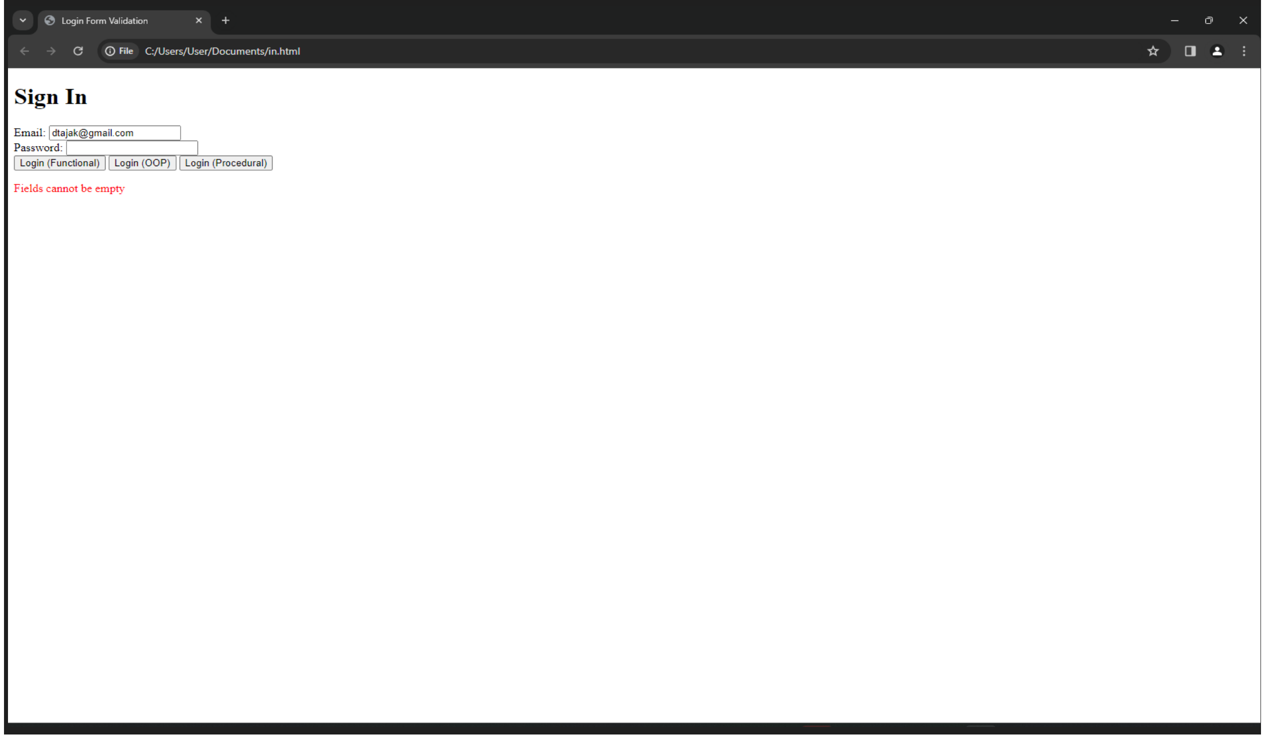
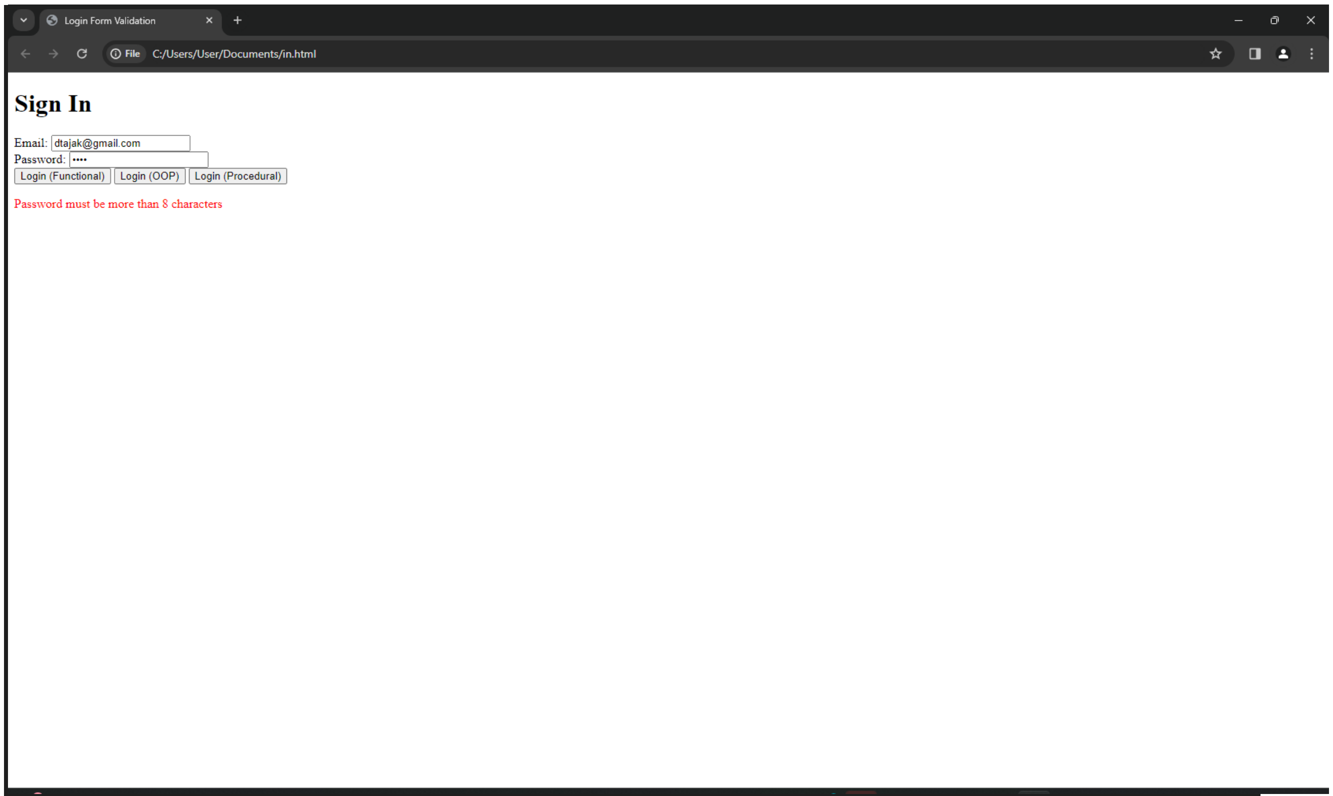
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Form Validation</title>
    <style>
        .error { color: red; }
    </style>
</head>
<body>
    <h1>Sign In </h1>
    <form id="loginForm">
        <label for="email">Email:</label>
        <input type="text" id="email" name="email"><br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"><br>
        <input type="button" value="Login (Functional)"
onclick="validateFunctional()">
        <input type="button" value="Login (OOP)" onclick="validateOOP()">
        <input type="button" value="Login (Procedural)"
onclick="validateProcedural()">
    </form>
    <p id="error" class="error"></p>

    <script>
        // Functional validation
        function validateEmailFunctional(email) {
            const pattern = /^[\w\.-]+@[\w\.-]+\.\w+$/;
            return pattern.test(email);
        }

        function validatePasswordFunctional(password) {
            return password.length > 8;
        }

        function validateFunctional() {
            const email = document.getElementById('email').value;
            const password = document.getElementById('password').value;
            const errorElement = document.getElementById('error');

            if (!email || !password) {
                errorElement.textContent = "Fields cannot be empty";
                return;
            }

            if (!validateEmailFunctional(email)) {
                errorElement.textContent = "Invalid email address";
                return;
            }

            if (!validatePasswordFunctional(password)) {
                errorElement.textContent = "Password must be more than 8
characters";
                return;
            }

            errorElement.textContent = "Validation successful";
        }

        // OOP validation
        class Validator {
            static validateEmail(email) {
                const pattern = /^[\w\.-]+@[\w\.-]+\.\w+$/;
                return pattern.test(email);
            }

            static validatePassword(password) {
                return password.length > 8;
            }
        }

        class Login {
            constructor(email, password) {
                this.email = email;
                this.password = password;
            }

            validate() {
                const errorElement = document.getElementById('error');

                if (!this.email || !this.password) {
                    errorElement.textContent = "Fields cannot be empty";
                    return;
                }

                if (!Validator.validateEmail(this.email)) {
                    errorElement.textContent = "Invalid email address";
                    return;
                }

                if (!Validator.validatePassword(this.password)) {
                    errorElement.textContent = "Password must be more than 8
characters";
                    return;
                }

                errorElement.textContent = "Validation successful";
            }
        }

        function validateOOP() {
            const email = document.getElementById('email').value;
            const password = document.getElementById('password').value;
            const login = new Login(email, password);
            login.validate();
        }

        // Procedural validation
        function validateEmailProcedural(email) {
            const pattern = /^[\w\.-]+@[\w\.-]+\.\w+$/;
            return pattern.test(email);
        }

        function validatePasswordProcedural(password) {
            return password.length > 8;
        }

        function validateProcedural() {
            const email = document.getElementById('email').value;
            const password = document.getElementById('password').value;
            const errorElement = document.getElementById('error');

            if (!email || !password) {
                errorElement.textContent = "Fields cannot be empty";
                return;
            }

            if (!validateEmailProcedural(email)) {
                errorElement.textContent = "Invalid email address";
                return;
            }

            if (!validatePasswordProcedural(password)) {
                errorElement.textContent = "Password must be more than 8
characters";
                return;
            }

            errorElement.textContent = "Validation successful";
        }
    </script>
</body>
</html>
```



Sign In

Email: dtajak@gmail.com
Password:
Login (Functional) Login (OOP) Login (Procedural)

Fields cannot be empty



Login Form Validation

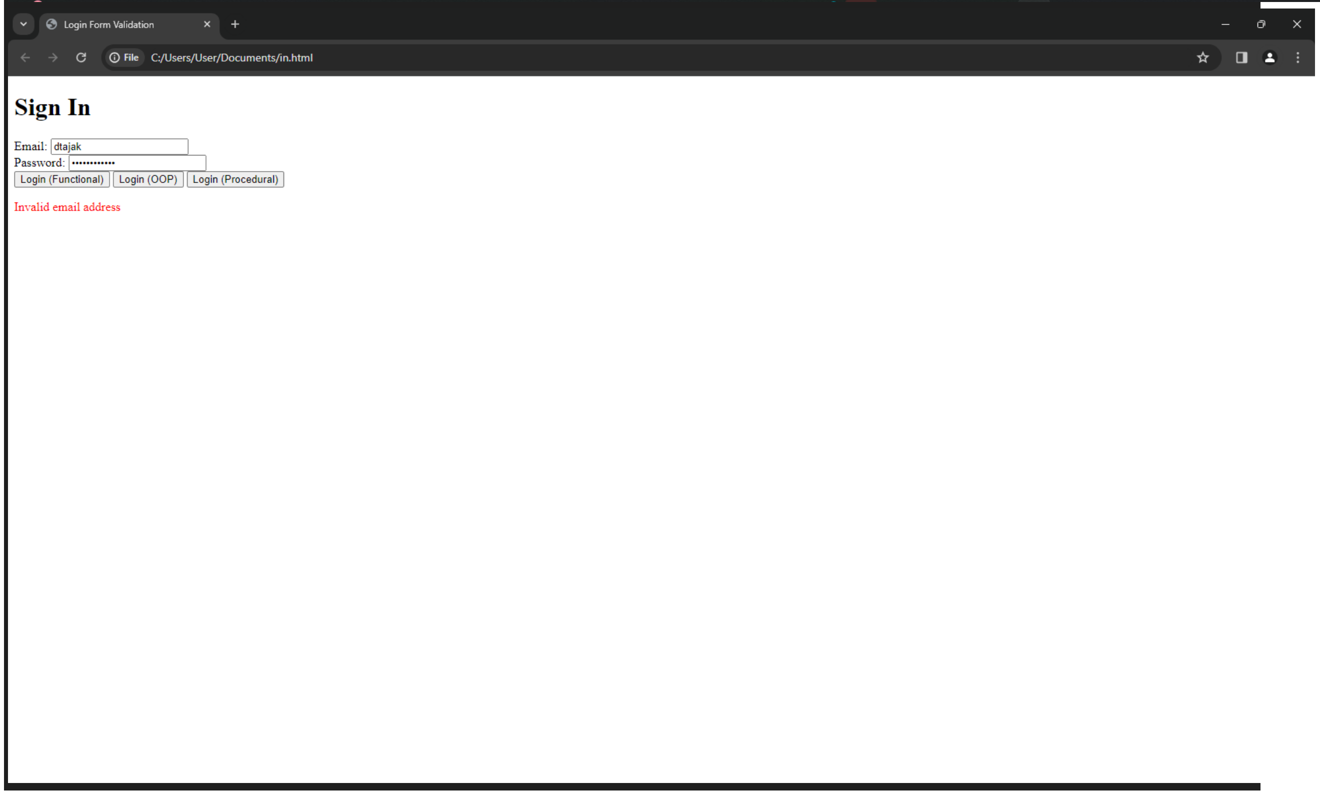
C:/Users/User/Documents/in.html

Sign In

Email: dtajak@gmail.com
Password: ••••

Login (Functional) Login (OOP) Login (Procedural)

Password must be more than 8 characters



Login Form Validation

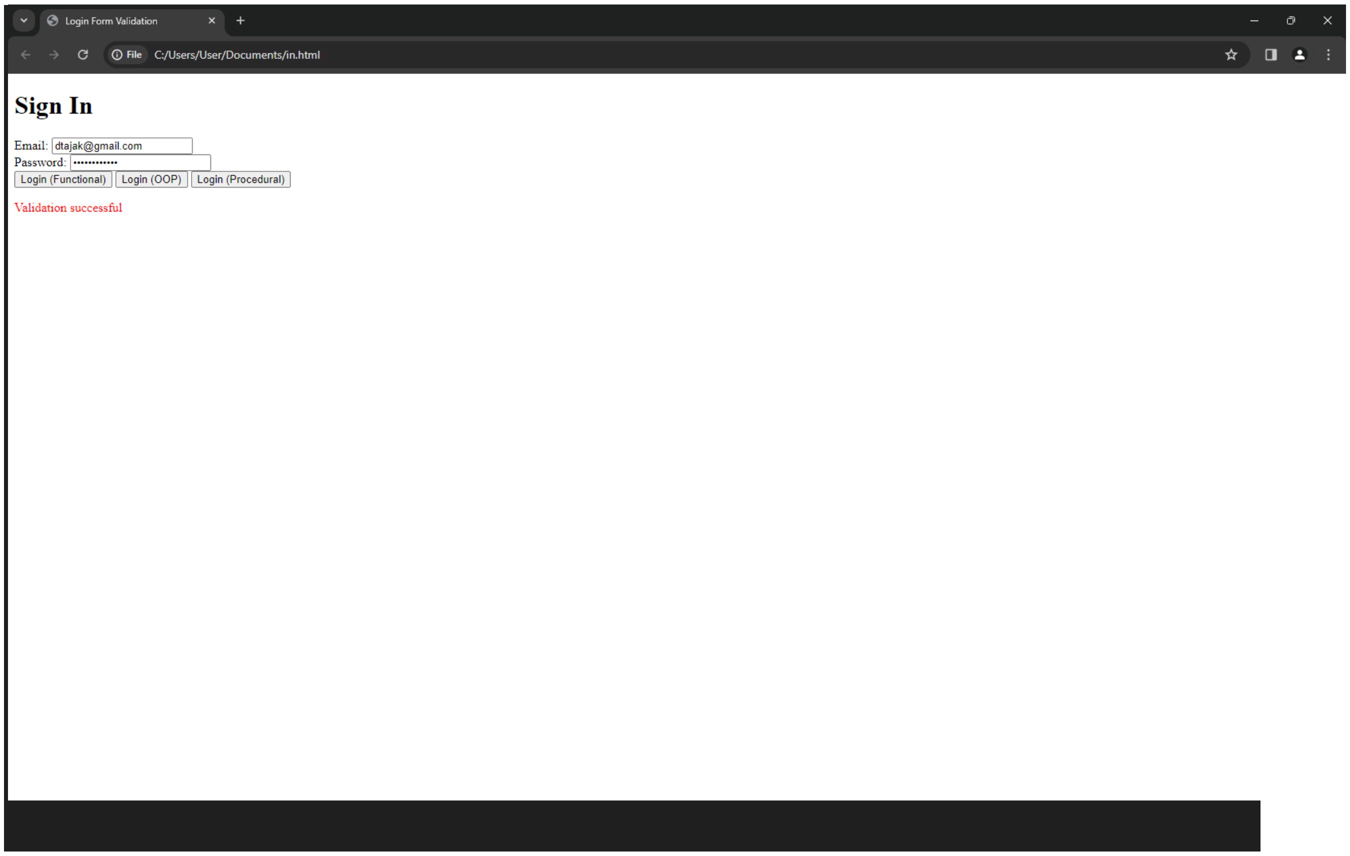
C:/Users/User/Documents/in.html

Sign In

Email: dtajak
Password: ••••••••••••

Login (Functional) Login (OOP) Login (Procedural)

Invalid email address



Sign In

Email: dtajak@gmail.com
Password: ••••••••••••
Login (Functional) Login (OOP) Login (Procedural)

Validation successful