DAY 11

INDEX.HTML

```
<body>
 <div class="container"> ... </div>
</body>
```

- The main content is inside a container div that centers everything.

Login Box
```
<div class="login-box"> ... </div>
```

This is split into **two panels**:

Left Panel (Social Login)
```
<div class="left-panel">
 <div class="logo"><h1>14 PRINTS</h1></div>
 <h2>Login using social media...</h2>
 <button class="social-btn fb">...</button>
</div>
```

- Shows your brand/logo name ("14 PRINTS").
- Prompts the user to log in via **Facebook**, **Twitter**, or **Google**.

Right Panel (Email Login Form)
```
<div class="right-panel">
 <h2>Login to your account</h2>
 <form> ... </form>
</div>
```

- Contains a form with:
    - **Email input** (with floating label)
    - **Password input** (with floating label and eye icon □ to toggle visibility)
    - **Remember me** checkbox and **Forgot password** link
    - **Login button**

STYLE.CSS

Main Layout

body { ... }

- Sets a **dark blue background** and uses a **nice font**
- Makes sure the login box is **centered on the screen**
- The whole screen height is used (100vh)

.container { ... }

- Keeps the login box in a fixed width (not too wide)

.login-box { ... }

- Main white box that contains both panels
- Adds **rounded corners**, **shadow**, and **horizontal layout**

Left Panel (Social Media Login)

.left-panel { ... }

- Covers 55% of the login box
- Shows a **background image** with a color overlay
- Text is **white** with shadows for readability
- Uses vertical alignment and center text

.social-btn { ... }

- Styled buttons for Facebook, Twitter, and Google
- Each button has a unique background color
- Bold text, rounded edges, and hover effect

Right Panel (Email/Password Login Form)

.right-panel { ... }

- Covers 60% (overlapping width—OK for spacing)
- White background with padding
- Headings and text are **dark blue**

.input-group input + label { ... }

- Stylish input fields with **floating labels**
- When typing or focusing, label moves up for a neat look

.toggle-icon { ... }

- Adds a clickable **eye icon** inside the password field to show/hide the password

Extra Features

.options { ... }

- Displays "Remember me" and "Forgot password" side by side

.email-btn { ... }

- A bright blue **login button** with hover effect

Responsive Design

@media (max-width: 768px) { ... }

- On small screens (like phones), the layout changes:
    - Left and right panels **stack vertically**
    - Panels take full width and get some padding

SCRIPT.JS

1. Stop the Form from Actually Submitting

```
document.querySelector('form').addEventListener('submit', function(e) {
  e.preventDefault();
  alert("Login functionality not implemented.");
});
```

This part **stops the form from doing a real login** when you press the "Login with email" button.

Instead, it shows a message box saying:

"Login functionality not implemented."

You can replace this later with real login code.

2. Show or Hide Password

```
document.getElementById('togglePassword').addEventListener('click', function () {
  const passwordInput = document.getElementById('password');
  const type = passwordInput.type === 'password' ? 'text' : 'password';
  passwordInput.type = type;
  this.textContent = type === 'password' ? '□' : '▢';
});
```

When you click on the eye icon, it **shows your password** so you can see what you're typing. When you click again, it **hides it back**.
The icon also changes between and based on whether the password is visible.

3. Check if the Email Looks Valid

```
document.getElementById('email').addEventListener('input', function () {
  const email = this.value;
  if (!email.includes('@') || !email.includes('.')) {
    this.style.borderColor = 'red';
  } else {
    this.style.borderColor = 'green';
  }
```

});

This watches what the user types in the email box.

If the email **doesn't contain** @ and . → it shows a **red border** to warn you it's not valid.

If it's a proper email format → border turns **green**, showing it's okay.