
Learning Material: Client-Side Programming

Introduction to Client-Side Programming

1. What is Client-Side Programming?

Client-side programming refers to code that runs **on the user's browser**, not on the server. It controls what the user **sees**, **clicks**, and **interacts** with on a webpage.

Why It's Important

- Makes websites **interactive and dynamic**
- Reduces load on the server
- Provides **immediate feedback** to users
- Allows animations, validations, pop-ups, and more

Languages Used

- ✓ **HTML** — structure
- ✓ **CSS** — design
- ✓ **JavaScript** — behavior (main client-side language)

2. JavaScript in Client-Side Programming

JavaScript adds interaction to web pages.

Basic Example

```
<button onclick="showMessage()">Click Me</button>

<script>
  function showMessage() {
    alert("Hello!");
  }
</script>
```

When the user clicks the button, the browser runs the JavaScript function.

Key Client-Side Features

1. DOM Manipulation

DOM = Document Object Model (the structure of a webpage).

JavaScript can modify page content instantly:

```
document.getElementById("title").innerHTML = "Welcome!";
```

2. Form Validation

Checks user inputs before sending data to the server.

```
if (username == "") {  
    alert("Username required!");  
}
```

Helps reduce errors and improves user experience.

3. Event Handling

Client-side code responds to user actions like:

- Click
- Hover
- Key press
- Scroll

Example

```
document.querySelector("#btn").onclick = function() {  
    console.log("Button clicked!");  
};
```

4. Client-Side Advantages

- ✓ Faster responses
 - ✓ Reduces server workload
 - ✓ Works even without constant internet
 - ✓ Enhances user experience
-