



SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NAAC-UGC with 'A' Grade

Approved by AICTE, Recognized by UGC & Affiliated to Anna
University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY

Course Code and Name : 19TS601 FULL STACK DEVELOPMENT

Unit 3 : NODEJS AND EXPRESS

Topic : Template Engine



Template Engine

- EJS stands for Embedded JavaScript.
- It is a template engine designed to be used with server-side JavaScript environments like NodeJS and
- It is used to generate dynamic content in a web page.
- It simplifies the generation of HTML by allowing you to embed JavaScript code directly in HTML.



How to use EJS?

- EJS is a template engine that is used to write JavaScript within HTML to generate dynamic content.
- EJS file is denoted by .ejs extension.
- EJS allows you to embed JavaScript code directly within HTML templates using special tags such as `<% %>` and `<%= %>`.
- To use EJS, we have to install it by using the `npm install EJS` command in our project folder.



Output Data (Variable Interpolation)

- Used to print values in the template.
- **<p>Welcome, <%= name %>!</p>**
- **<%= %>: Escapes HTML and prints the value.**



Unescaped Output

- Used when you **don't want** to escape HTML characters.
- `<p><%- htmlContent %></p>`
- `<%- %>`: Prints without escaping HTML (use carefully to avoid XSS attacks).



JavaScript Logic (Control Flow)

- If-Else Condition
- `<% if (user.isLoggedIn) { %>`
- `<p>Welcome, <%= user.name %>!</p>`
- `<% } else { %>`
- `<p>Please log in.</p>`
- `<% } %>`
- `<% %>`: Runs JavaScript code without output.



Loop (forEach Example)

- ``
- `<% users.forEach(user => { %>`
- `<%= user.name %> - <%= user.age %> years old`
- `<% }) %>`
- ``
- Loops through users and prints user details.



Include Partial Templates

- Used to **reuse templates** (e.g., header, footer).
 - Eg:ejs
 - `<%- include("header") %>`
 - `<h1>Home Page</h1>`
 - `<%- include("footer") %>`
- `<%- include("filename") %>`: Inserts another EJS file.



Passing Data from Express to EJS

- Express Route Example:
- Eg:js
- `app.get("/", (req, res) => {`
- `res.render("index", { name: "Alice", users: [{ name: "Bob", age: 30 }]`
- `});`
- `});`
- Data { name: "Alice", users: [...] } is passed to index.ejs.



Commenting in EJS

- Eg:ejs
- `<%-- This is an EJS comment, not visible in the output --%>`
- Use `<%-- --%>` for comments that won't appear in the rendered HTML.



Index.js

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>EJS Example</title>
```

```
</head>
```

```
<body>
```

```
  <%- include("header") %>
```

```
  <h1>Welcome, <%= name %>!</h1>
```

```
  <% if (users.length > 0) { %>
```




```
<% users.forEach(user => { %>
```

```
  <li><%= user.name %> - <%= user.age %> years old</li>
```

```
<% }) %>
```

```
</ul>
```

```
<% } else { %>
```

```
  <p>No users available.</p>
```

```
<% } %>
```

```
<%- include("footer") %>
```

```
</body>
```

```
</html>
```



Syntax	Purpose
<code><%= value %></code>	Output variable (escaped)
<code><%- value %></code>	Output variable (unescaped)
<code><% code %></code>	JavaScript logic (no output)
<code><% if (...) { %> ... <% } %></code>	Conditional statements
<code><% users.forEach(...) { %></code>	Looping (iteration)
<code><%- include("file") %></code>	Include another EJS template
<code><%-- Comment --%></code>	EJS comment



- **Features of EJS Templating Engine:**

- It supports variable substitution by using `<%= %>` tag.
- It supports conditionals and loops directly in the template by using `<% %>` tag.
- It can be easily used with NodeJS frameworks like ExpressJS.
- It supports reusable components by using include.



- Step 1: Install Dependencies
- Run the following command to install **Express.js** and **EJS**:
- For eg:sh
- `npm install express ejs`
- Step 2: Create an Express Server (server.js) Create a file named server.js and set up an Express app that renders an EJS template.
- For Eg:server.js



```
const express = require("express");
const app = express();
// Set EJS as the template engine
app.set("view engine", "ejs");
// Define a route to render an EJS template
app.get("/", (req, res) => {
  const users = [
    { name: "Alice", age: 25 },
    { name: "Bob", age: 30 },
    { name: "Charlie", age: 35 }
  ];
```




```
res.render("index", { title: "User List", users });  
});  
  
// Start the server  
app.listen(3000, () => {  
  console.log("Server running at http://localhost:3000");  
});
```



- **Step 3: Create an EJS Template (views/index.ejs)**
- **Inside the views/ folder, create a file named index.ejs.**



- `<!DOCTYPE html>`
- `<html lang="en">`
- `<head>`
- `<meta charset="UTF-8">`
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`
- `<title><%= title %></title>`
- `</head>`
- `<body>`
- `<h1><%= title %></h1>`



```
<ul>
```

```
<% users.forEach(user => { %>
```

```
<li><%= user.name %> - <%= user.age %> years old</li>
```

```
<% }); %>
```

```
</ul>
```

```
</body>
```

```
</html>
```

Step 4: Run the Application

Start the server:sh



- **Step 4: Run the Application**

- Start the server:

For Eg:Sh

```
node server.js
```



JSON Objects

- Now, open **<http://localhost:3000>** in your browser. You should see:

For Eg: sql

User List

- Alice - 25 years old
- Bob - 30 years old
- Charlie - 35 years old



Thank
You!

dreamstime

