

NODE-EJS



Templating

EJS (Embedded JavaScript Templates)

EJS is a simple templating language that lets you generate HTML markup with plain JavaScript.

@devwithpayal

NODE-EJS

Using EJS

```
app.set("view engine", "ejs");  
  
app.get("/", (req, res) => {  
  res.render("home.ejs");  
});
```

NODE-EJS

Views Directory

```
const path = require("path");  
  
app.set("views", path.join(__dirname, "/views"));
```

Make sure to use `__dirname` with two underscores to get the directory name of the current module.

This will set the "views" directory to be in the same directory as your script (assuming that's where your "views" directory is located).

NODE-EJS

Interpolation Syntax

Interpolation refers to
embedding expressions into
marked up text.

NODE-EJS

Passing data to EJS

```
app.get("/rolldice", (req, res) => {  
  let num = Math.floor(Math.random() * 6) + 1;  
  res.render("rollDice.ejs", { diceVal: num });  
});
```

```
<h1>Your dice gave value : <%= diceVal %></h1>
```


NODE-EJS

Conditional Statements in EJS

Adding conditions inside EJS

```
<% if(diceVal == 6) { %>  
<h2>Nice! Roll dice again.</h2>  
<% } %>
```

NODE-EJS

Loops in EJS

```
<% for(user of followers) { %>
<li><%= user %></li>
<% } %>
```

This EJS code uses the `<% %>` tags for control flow to execute the JavaScript code inside.

The loop iterates over each user in the followers array and generates an `` element containing the user information.

When rendering this EJS template with actual data, it will create a list of `` elements, each representing a user from the followers array.

NODE-EJS

Serving Static Files

`app.use(express.static(folder_name))`

```
app.use(express.static(path.join(__dirname, "public")));
```


NODE-EJS

Includes

```
<%- include("includes/head.ejs"); %>
```

When you render index.ejs, it will include the content of head.ejs at the specified location.

Make sure your file paths are correct based on your project structure.