Created by Alan Vallavaraj, Lecturer, Ulster University / QA

Week 10 - Node JS, Express, and EJS: Step-by-Step Guide

1. Install Node.js

- Download Node.js from the Node.js website.

- Choose the LTS (Long Term Support) version.

- Install it by accepting all the default settings.

2. Create a Node.js Project

- Create a new project folder and open it in Visual Studio Code.

- Open a new terminal in VSCode: Terminal -> New Terminal.

- Initialize your project by running: npm init or npm init -y.

- A package.json file will be created, tracking the project dependencies.

3. Install Express

- Install Express by running the following command in the terminal: npm install express.

- This will update your package.json to include Express as a dependency.

4. Set Up a Basic Express Server

- Create a file named index.js.

- Add the following code to set up a basic Express server:

```
  const express = require('express');

  const app = express();

  app.listen(5000, () => { console.log('Server started'); });
```

5. Create Your First Route

- Add a simple route in index.js to handle a GET request to the home page:

```
  app.get('/', (req, res) => { res.send('<h1>Welcome to the Home Page</h1>'); });
```

- You can add more routes, such as an About page.

6. Handle Routes with Parameters

- Create dynamic routes using parameters, such as user profiles:

```
  app.get('/user/:userid', (req, res) => { res.send(`<h1>User ID: ${req.params.userid}</h1>`); });
```

7. Install and Set Up EJS for Templating

- Install EJS by running the following command: npm install ejs.

- Set EJS as the templating engine in index.js by adding: app.set('view engine', 'ejs');

8. Create EJS Templates

- Create a folder named views in your project directory.

- Inside the views folder, create your EJS templates (e.g., index.ejs).

- Modify your routes to render EJS templates instead of plain HTML.

9. Set Up Static Assets

- Create a folder named public in the root of your project.

- Place your static assets (CSS, JavaScript files, images) in this public folder.

- Tell Express where to find static files by adding: app.use(express.static('public'));

10. Pass Dynamic Content to EJS Templates

- You can pass dynamic data to your EJS templates from your routes:

```
  app.get('/', (req, res) => { res.render('index', { pagetitle: 'Welcome to the Homepage' }); });
```

- In your index.ejs template, you can access this data.

11. Conditional Logic in EJS Templates

- You can use conditional logic in your EJS templates.
- Example:

```
  app.get('/', (req, res) => { res.render('index', { name: 'Darren' }); });
  <% if(name == 'Darren') { %>
      <p>Welcome back, Darren!</p>
  <% } %>
```

Final Thoughts

This guide covers the key aspects of setting up and using Node.js, Express, and EJS for your project. However, these notes are meant to supplement the lecture. Be sure to refer to the recorded session for more in-depth explanations and examples.