**Ex:** 1

**Date:** 25.6.25

**Development of Toy Store using NodeJS Server without Express**

**AIM:**

To develop an toy store shop website and host it using a NodeJS static server created with core modules such as http, fs and path without using any external frameworks like Express, to serve static HTML ,CSS ,and JavaScript files to the client brower.

**PROCEDURE:**

**STEP 1:Project Initialization:** Create a new project folder (e.g., static-server).

**STEP 2;Navigate to Folder:** Open a terminal or command prompt and navigate into the newly created project folder.

**STEP3:Create package.json:** Run the command npm init -y to create a default package.json file.

**STEP 4:Create HTML File:** Create a new file named index.html in the project root.

**STEP 5:Create CSS File:** Create a second file named style.css in the project root.

**STEP 6:Create Server File:** Create the main server logic file named server.js.

**STEP 7:Write HTML Content:** Add the necessary HTML boilerplate to index.html, including a heading and a <link> tag pointing to /style.css.

**STEP 8:Write CSS Rules:** Add CSS rules to style.css to style the elements from the HTML file.

**STEP 9:Import Modules:** In server.js, import the required Node.js modules: http, fs, and path.

**STEP 10:Implement Server Logic:** Write the code to create an HTTP server that listens for requests.

**STEP 11:Implement Routing:** Inside the server, check the request URL (req.url). If it's '/', serve index.html. If it's '/style.css', serve style.css.

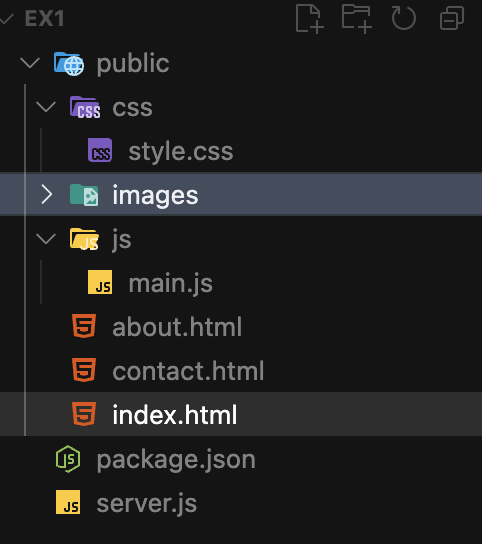
**STEP 12:Set Content-Type:** For each file type, set the correct MIME type in the response header (text/html for HTML, text/css for CSS).

**STEP 13:Handle Errors:** Implement a basic 404 "Not Found" response for any other requested URL.

**STEP 14:Start the Server:** Add the code to make the server listen on a specific port (e.g., 3000).

**STEP 15:Run and Test:** Execute the server using the command node server.js and open http://localhost:3000 in a web browser to verify the result.

**DESIGN:**



**IMPLEMENTATION:**

**1.index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Toy Store</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header>

<h1>Welcome to the Toy Store!</h1>

<p>Your one-stop shop for fun and learning.</p>

</header>

<section class="toy-gallery">

<div class="toy-card">

<img src="https://via.placeholder.com/150?text=Teddy+Bear" alt="Teddy Bear">

<h2>Teddy Bear</h2>

<p>$15.99</p>

</div>

<div class="toy-card">

<img src="https://via.placeholder.com/150?text=Toy+Car" alt="Toy Car">

<h2>Toy Car</h2>

<p>$9.99</p>

</div>

<div class="toy-card">

<img src="https://via.placeholder.com/150?text=Building+Blocks" alt="Building Blocks">

<h2>Building Blocks</h2>

<p>$12.49</p>

</div>

<div class="toy-card">

<img src="https://via.placeholder.com/150?text=Doll" alt="Doll">

<h2>Doll</h2>

<p>$14.99</p>

</div>

</section>

<button class="shop-btn" onclick="showAlert()">Shop Now</button>

<script src="script.js"></script>

</body>

</html>

**2.style.css**

body {

font-family: 'Comic Sans MS', 'Comic Sans', cursive, sans-serif;

background: linear-gradient(135deg, #f9d423 0%, #ff4e50 100%);

margin: 0;

padding: 0;

min-height: 100vh;

}

header {

text-align: center;

padding: 2rem 1rem 1rem 1rem;

color: #fff;

}

h1 {

font-size: 2.5rem;

margin-bottom: 0.5rem;

letter-spacing: 2px;

}

.toy-gallery {

display: flex;

flex-wrap: wrap;

justify-content: center;

gap: 2rem;

padding: 2rem 1rem;

}

.toy-card {

background: #fffbe7;

border-radius: 16px;

box-shadow: 0 4px 12px rgba(0,0,0,0.08);

padding: 1rem;

width: 180px;

text-align: center;

transition: transform 0.2s;

}

.toy-card:hover {

transform: scale(1.05) rotate(-2deg);

box-shadow: 0 8px 24px rgba(0,0,0,0.15);

}

.toy-card img {

width: 100%;

border-radius: 12px;

margin-bottom: 0.5rem;

}

.toy-card h2 {

font-size: 1.2rem;

margin: 0.5rem 0 0.25rem 0;

color: #ff4e50;

}

.toy-card p {

font-size: 1rem;

color: #333;

margin: 0;

}

.shop-btn {

display: block;

margin: 2rem auto;

padding: 1rem 2.5rem;

font-size: 1.2rem;

background: #ff4e50;

color: #fff;

border: none;

border-radius: 32px;

cursor: pointer;

box-shadow: 0 2px 8px rgba(0,0,0,0.10);

transition: background 0.2s, transform 0.2s;

}

.shop-btn:hover {

background: #f9d423;

color: #ff4e50;

transform: scale(1.08);

}

**3.server.js**

const http = require('http');

const fs = require('fs');

const path = require('path');

const PORT = 3000;

const server = http.createServer((*req*, *res*) => {

if (*req*.url === '/' || *req*.url === '/index.html') {

fs.readFile(path.join(\_\_dirname, 'index.html'), (*err*, *data*) => {

if (*err*) {

*res*.writeHead(500, { 'Content-Type': 'text/plain' });

*res*.end('Internal Server Error');

return;

}

*res*.writeHead(200, { 'Content-Type': 'text/html' });

*res*.end(*data*);

});

} else if (*req*.url === '/style.css') {

fs.readFile(path.join(\_\_dirname, 'style.css'), (*err*, *data*) => {

if (*err*) {

*res*.writeHead(404, { 'Content-Type': 'text/plain' });

*res*.end('Not Found');

return;

}

*res*.writeHead(200, { 'Content-Type': 'text/css' });

*res*.end(*data*);

});

} else if (*req*.url === '/script.js') {

fs.readFile(path.join(\_\_dirname, 'script.js'), (*err*, *data*) => {

if (*err*) {

*res*.writeHead(404, { 'Content-Type': 'text/plain' });

*res*.end('Not Found');

return;

}

*res*.writeHead(200, { 'Content-Type': 'application/javascript' });

*res*.end(*data*);

});

} else {

*res*.writeHead(404, { 'Content-Type': 'text/plain' });

*res*.end('Not Found');

}

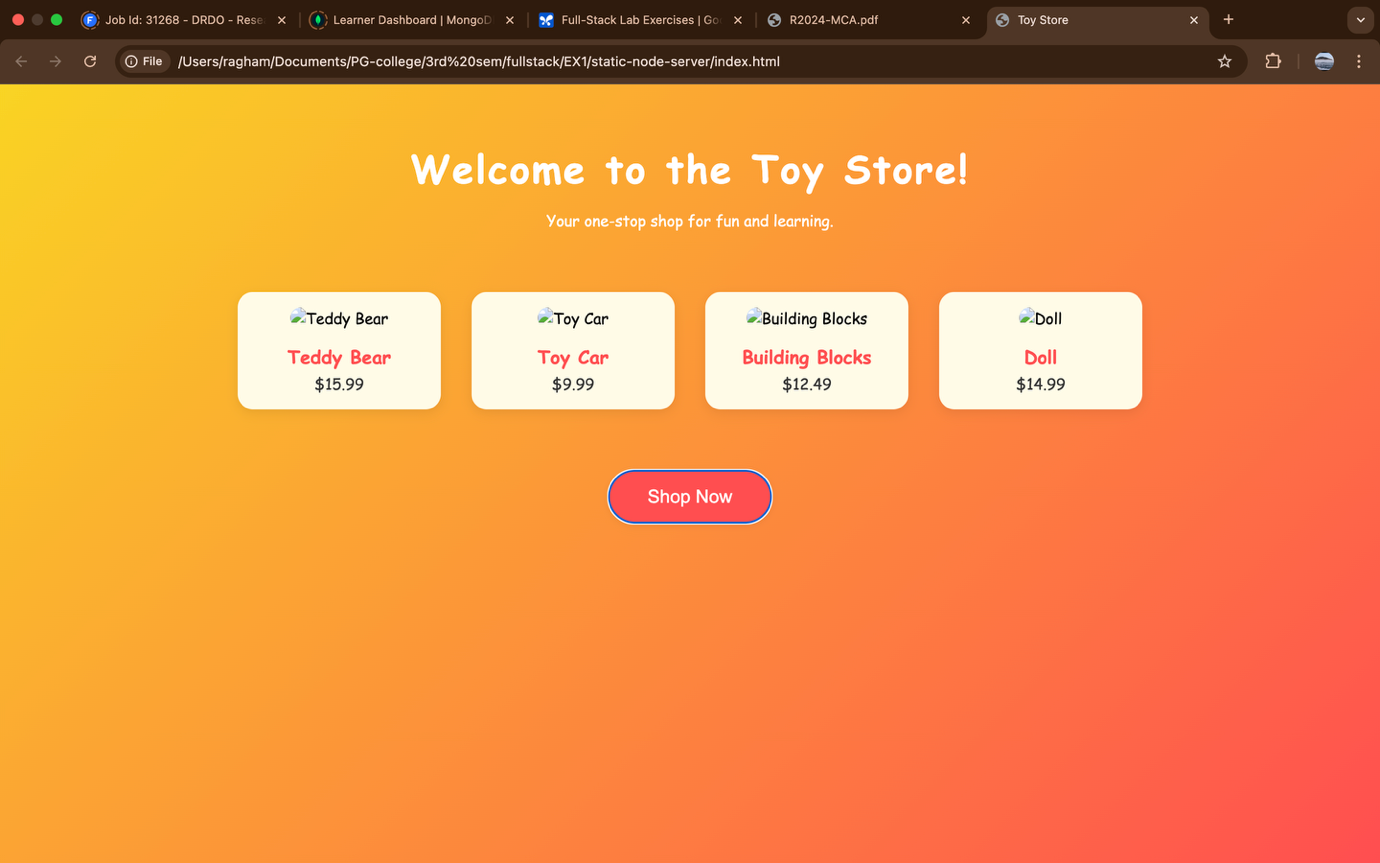
});

server.listen(PORT, () => {

console.log(`Server running at http://localhost:${PORT}/`);

});

**OUTPUT SCREEN SHOT:**

****

|  |  |
| --- | --- |
| **COE(30)** |  |
| **RECORD(20)** |  |
| **VIVA(10)** |  |
| **TOTAL(60)** |  |

**RESULT:**

The online Toy store website was successfully hosted using a NodeJS static server without Express. The static file were served correctly and the application worked as expected at <http://localhost:3000/>.