**Name: Chauhan Parth Dineshkumar**

**Enrollment: 22012012041**

**Class: 5CEIT**

**Batch: 5B-5**

**AIM : Create application which perform basic functionalities like insert, update, delete and display records with database- Inventory Management**

**Code:**

const url = require('url');

const express = require('express');

const bodyParser = require('body-parser');

const mysql = require('mysql');

const app = express();

const port = 4590;

app.use(bodyParser.urlencoded({ extended: false }));

var con = mysql.createConnection({

host: 'localhost',

user: 'root',

password: '',

database: 'inventory'

});

con.connect(function (err) {

if (err) {

console.log(err.message);

}

console.log("Connected!");

});

app.get('/', (req, res) => {

var fetchedData = 'select \* from products';

con.query(fetchedData, (err, result) => {

if (err) {

console.error('Error fetching data: ' + err.message);

res.status(500).send('Error fetching data from the database');

}

else {

const tableRows = result.map((product) =>

`

<tr>

<td>${product.name}</td>

<td>${product.price}</td>

<td>${product.dealer}</td>

</tr>

`);

const html = `

<!DOCTYPE html>

<html>

<head>

<title>Inventory Management</title>

<style>

/\* Reset some default styles \*/

body,

h1 {

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

text-align: center;

}

nav {

font-size: larger;

text-align: left;

background-color: #333;

color: #fff;

padding: 10px 10px;

}

nav ul {

list-style-type: none;

padding: 0;

}

nav ul li {

display: inline;

margin-right: 20px;

}

nav a {

text-decoration: none;

color: #fff;

font-weight: bold;

}

table {

width: 80%;

margin: 20px auto;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 0 5px rgba(0, 0, 0, 0.3);

}

th, td {

padding: 10px;

text-align: center;

}

th {

background-color: #333;

color: #fff;

}

tr:nth-child(even) {

background-color: #f2f2f2;

}

/\* Add more styles as needed \*/

</style>

</head>

<body>

<nav>

<ul>

<li><a href="/">Home</a></li>

<li><a href="/add">Add</a></li>

<li><a href="/modify">Modify</a></li>

<li><a href="/delete">Delete</a></li>

</ul>

</nav>

<h2>Item List</h2>

<table>

<tr>

<th>Name</th>

<th>Price</th>

<th>Dealer</th>

</tr>

${tableRows.join('')}

</table>

</body>

</html>

`

res.write(html)

}

});

});

app.get('/add', (req, res) => {

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Product Form</title>

<style>

/\* Reset some default styles \*/

body,

h1 {

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

text-align: center;

}

nav {

font-size: larger;

text-align: left;

background-color: #333;

color: #fff;

padding: 10px 10px;

}

nav ul {

list-style-type: none;

padding: 0;

}

nav ul li {

display: inline;

margin-right: 20px;

}

nav a {

text-decoration: none;

color: #fff;

font-weight: bold;

}

.info{

margin-top:10px;

}

form {

max-width: 500px;

margin: 20px auto;

background-color: #fff;

padding: 20px;

border-radius: 5px;

box-shadow: 0 0 5px rgba(0, 0, 0, 0.2);

}

.form-field {

margin-bottom: 10px;

}

h1 {

background-color: #333;

color: #fff;

padding: 20px 0;

margin: 0;

}

label {

display: block;

margin-top:10px;

margin-bottom: 5px;

}

input[type="text"],

input[type="number"] {

width: 95%;

padding: 10px;

border: 1px solid #ccc;

border-radius: 3px;

font-size: 16px;

}

button[type="submit"] {

background-color: #333;

color: #fff;

border: none;

padding: 10px 20px;

cursor: pointer;

font-size: 16px;

border-radius: 3px;

}

button[type="submit"]:hover {

background-color: #555;

}

/\* Add more styles as needed \*/

</style>

</head>

<body>

<nav>

<ul>

<li><a href="/">Home</a></li>

<li><a href="/add">Add</a></li>

<li><a href="/modify">Modify</a></li>

<li><a href="/delete">Delete</a></li>

</ul>

</nav>

<form id="productForm" action="/submit" method="POST">

<h1 class="info">Product Information Form</h1>

<div class="form-field">

<label for="name">Product Name:</label>

<input type="text" id="name" name="name" required minlength="5">

</div>

<div class="form-field">

<label for="price">Price (in rupees):</label>

<input type="number" id="price" name="price" required min="1">

</div>

<div class="form-field">

<label for="dealer">Dealer Name:</label>

<input type="text" id="dealer" name="dealer" required minlength="5">

</div>

<button type="submit">Submit</button>

</form>

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

</body>

</html>

`);

});

app.post('/submit', (req, res) => {

const productName = (req.body.name).toLowerCase();

const price = req.body.price;

const dealer = (req.body.dealer).toLowerCase();

const addQuery = 'INSERT INTO products (name, price, dealer) VALUES (?, ?, ?)';

con.query(addQuery, [productName, price, dealer], (err, result) => {

if (err) {

console.error('Error inserting data: ' + err.message);

res.status(500).send('Error inserting data into the database');

}

else {

console.log('Data inserted into the database');

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Product insertion page</title>

</head>

<body>

<script>

alert('Data inserted successfully!');

window.location.href = '/add'; // Redirect back to the form

</script>

</body>

</html>

`);

}

})

})

app.get('/delete', (req, res) => {

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Delete Item</title>

<style>

/\* Reset some default styles \*/

body,

h1 {

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

text-align: center;

}

h1 {

background-color: #333;

color: #fff;

padding: 20px 0;

margin: 0;

}

nav {

font-size: larger;

text-align: left;

background-color: #333;

color: #fff;

padding: 10px 10px;

}

nav ul {

list-style-type: none;

padding: 0;

}

nav ul li {

display: inline;

margin-right: 20px;

}

nav a {

text-decoration: none;

color: #fff;

font-weight: bold;

}

.container {

max-width: 500px;

margin: 20px auto;

background-color: #fff;

padding: 20px;

border-radius: 5px;

box-shadow: 0 0 5px rgba(0, 0, 0, 0.2);

}

label {

display: block;

margin-top:10px;

margin-bottom: 5px;

}

input[type="text"] {

width: 100%;

padding: 10px;

margin-bottom: 10px;

border: 1px solid #ccc;

border-radius: 3px;

font-size: 16px;

}

button[type="submit"] {

background-color: #ff4d4d;

color: #fff;

border: none;

padding: 10px 20px;

cursor: pointer;

font-size: 16px;

border-radius: 3px;

}

button[type="submit"]:hover {

background-color: #ff1a1a;

}

</style>

</head>

<body>

<nav>

<ul>

<li><a href="/">Home</a></li>

<li><a href="/add">Add</a></li>

<li><a href="/modify">Modify</a></li>

<li><a href="/delete">Delete</a></li>

</ul>

</nav>

<div class="container">

<h1>Delete Item</h1>

<form id="deleteForm" action="/delete" method="POST">

<div class="form-group">

<label for="item">Enter Item Name to Delete:</label>

<input type="text" id="name" name="name" required minlength="4">

<small>Minimum 4 characters required.</small>

</div>

<button type="submit">Delete</button>

</form>

</div>

</body>

</html>

`)

});

app.post('/delete', (req, res) => {

const productName = req.body.name.toLowerCase();

// Check if the product exists in the database

const checkQuery = 'SELECT \* FROM products WHERE name = ?';

con.query(checkQuery, [productName], (err, result) => {

if (err) {

console.error('Error checking data: ' + err.message);

res.status(500).send('Error checking data in the database');

} else {

if (result.length === 0) {

// Product not found, so provide a message

res.send('Data does not exist in the database');

} else {

// Product found, so delete it

const deleteQuery = 'DELETE FROM products WHERE name = ?';

con.query(deleteQuery, [productName], (err, deleteResult) => {

if (err) {

console.error('Error deleting data: ' + err.message);

res.status(500).send('Error deleting data from the database');

} else {

console.log('Data deleted from the database');

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Product removal page</title>

</head>

<body>

<script>

alert('Data deleted successfully!');

window.location.href = '/delete'; // Redirect back to the form

</script>

</body>

</html>

`);

}

});

}

}

});

});

app.get('/modify', (req, res) => {

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Modify Product</title>

<style>

/\* Reset some default styles \*/

body, h1 {

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

text-align: center;

}

h1 {

background-color: #333;

color: #fff;

padding: 20px 0;

margin: 0;

}

nav {

font-size: larger;

text-align: left;

background-color: #333;

color: #fff;

padding: 10px 10px;

}

nav ul {

list-style-type: none;

padding: 0;

}

nav ul li {

display: inline;

margin-right: 20px;

}

nav a {

text-decoration: none;

color: #fff;

font-weight: bold;

}

.container {

max-width: 500px;

margin: 20px auto;

background-color: #fff;

padding: 20px;

border-radius: 5px;

box-shadow: 0 0 5px rgba(0, 0, 0, 0.2);

}

label {

display: block;

margin-top:10px;

margin-bottom: 5px;

}

input[type="text"],

input[type="number"] {

width: 95%;

padding: 10px;

margin-bottom: 10px;

border: 1px solid #ccc;

border-radius: 3px;

font-size: 16px;

}

button[type="submit"] {

background-color: #333;

color: #fff;

border: none;

padding: 10px 20px;

cursor: pointer;

font-size: 16px;

border-radius: 3px;

}

button[type="submit"]:hover {

background-color: #555;

}

</style>

</head>

<body>

<nav>

<ul>

<li><a href="/">Home</a></li>

<li><a href="/add">Add</a></li>

<li><a href="/modify">Modify</a></li>

<li><a href="/delete">Delete</a></li>

</ul>

</nav>

<div class="container">

<h1>Modify Product</h1>

<form id="modifyForm" action="/modify" method="POST">

<div class="form-group">

<label for="name">Product Name:</label>

<input type="text" class="form-control" id="name" name="name" required minlength="5">

</div>

<div class="form-group">

<label for="price">Price (in rupees):</label>

<input type="number" class="form-control" id="price" name="price" required min="1">

</div>

<div class="form-group">

<label for="dealer">Dealer Name:</label>

<input type="text" class="form-control" id="dealer" name="dealer" required minlength="5">

</div>

<button type="submit" class="btn btn-primary">Modify</button>

</form>

</div>

</body>

</html>

`);

});

app.post('/modify', (req, res) => {

const productName = req.body.name.toLowerCase();

const price = req.body.price;

const dealer = req.body.dealer.toLowerCase();

// Check if the product already exists in the database

const checkQuery = 'SELECT \* FROM products WHERE name = ?';

con.query(checkQuery, [productName], (err, result) => {

if (err) {

console.error('Error checking data: ' + err.message);

res.status(500).send('Error checking data in the database');

} else {

if (result.length === 0) {

// Product not found, so insert it

const addQuery = 'INSERT INTO products (name, price, dealer) VALUES (?, ?, ?)';

con.query(addQuery, [productName, price, dealer], (err, insertResult) => {

if (err) {

console.error('Error inserting data: ' + err.message);

res.status(500).send('Error inserting data into the database');

} else {

console.log('Data inserted into the database');

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Modification page</title>

</head>

<body>

<script>

alert('Data inserted successfully!');

window.location.href = '/modify'; // Redirect back to the form

</script>

</body>

</html>

`);

}

});

} else {

// Product found, so update it

const updateQuery = 'UPDATE products SET price = ?, dealer = ? WHERE name = ?';

con.query(updateQuery, [price, dealer, productName], (err, updateResult) => {

if (err) {

console.error('Error updating data: ' + err.message);

res.status(500).send('Error updating data in the database');

} else {

console.log('Data updated in the database');

res.send('Data updated successfully');

res.write(`

<!DOCTYPE html>

<html>

<head>

<title>Modification page</title>

</head>

<body>

<script>

alert('Data modified successfully!');

window.location.href = '/modify'; // Redirect back to the form

</script>

</body>

</html>

`);

}

});

}

}

});

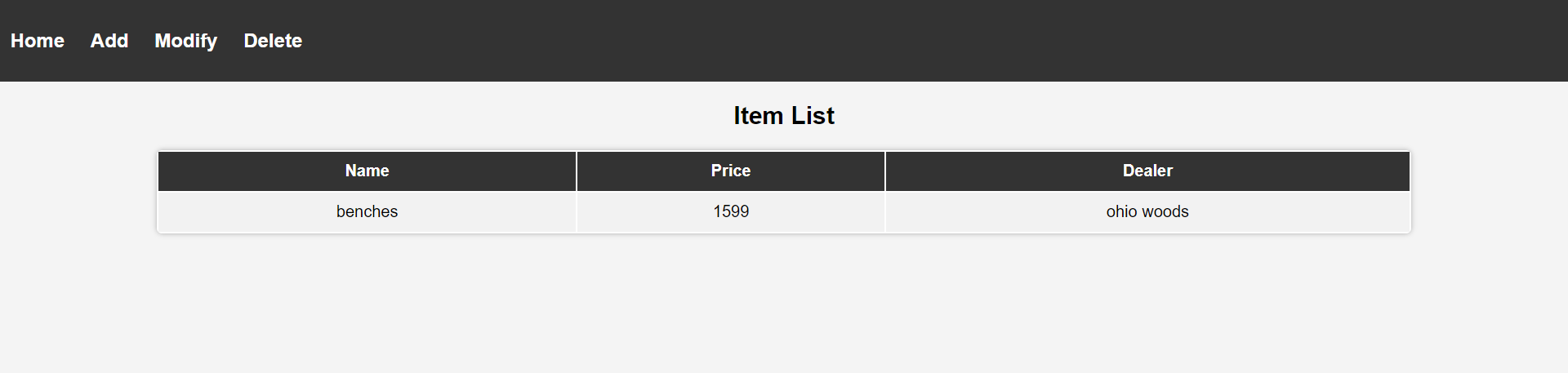
})

app.listen(port, () => {

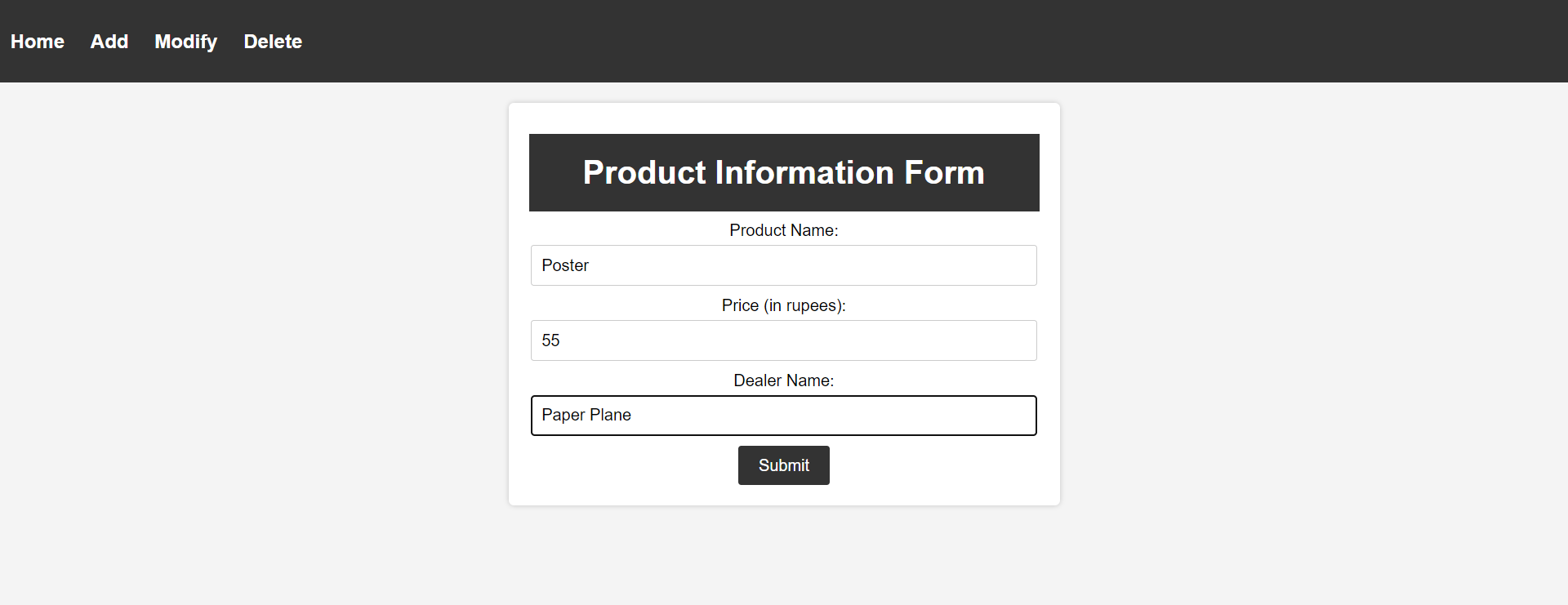
console.log(`Server is running on http://localhost:${port}`);

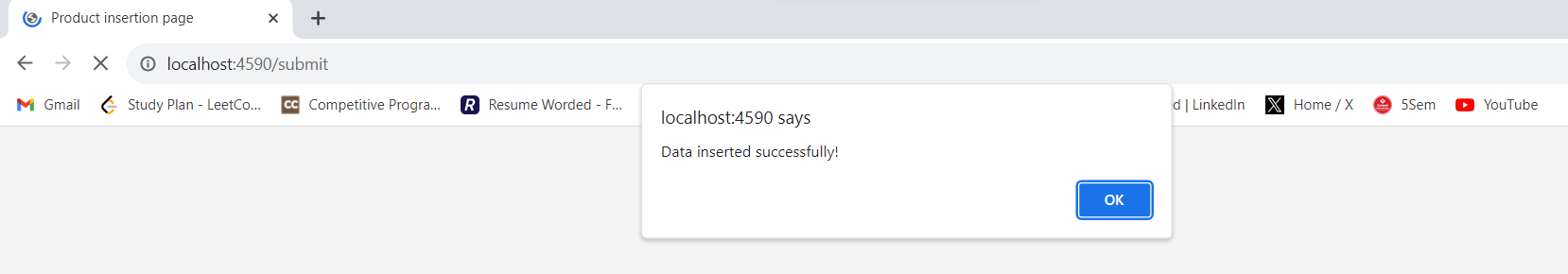
});

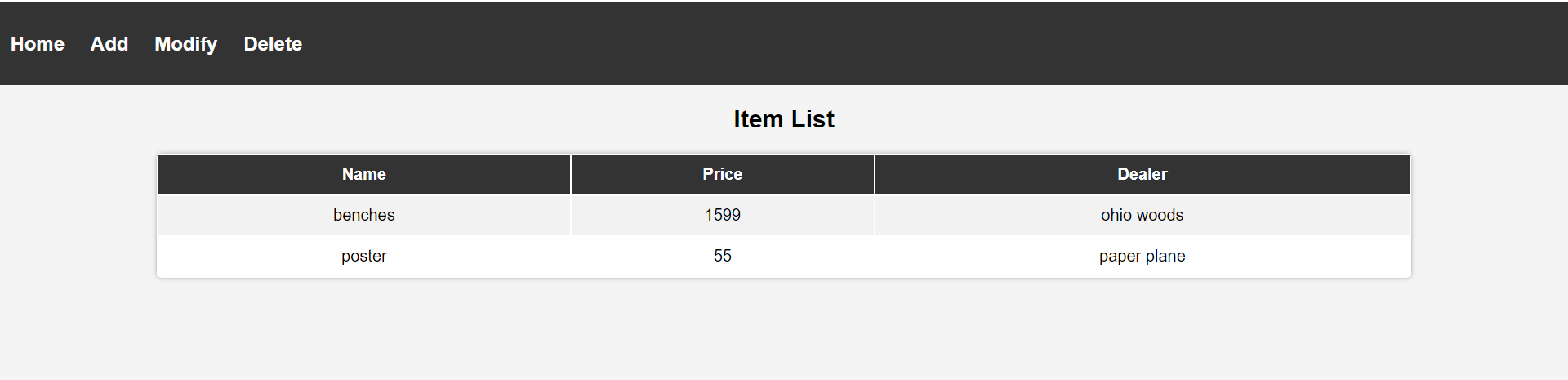
**Output:**

****

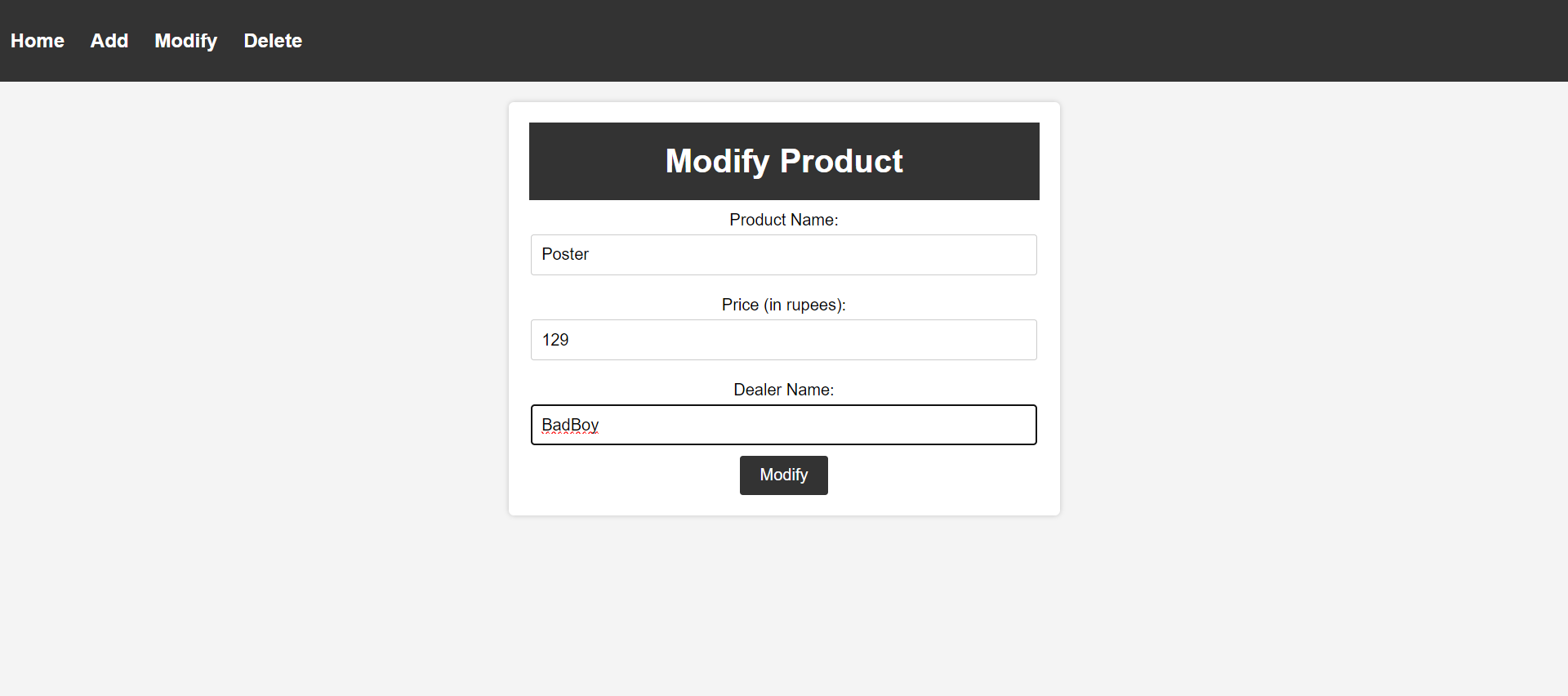
**Inserting data –**

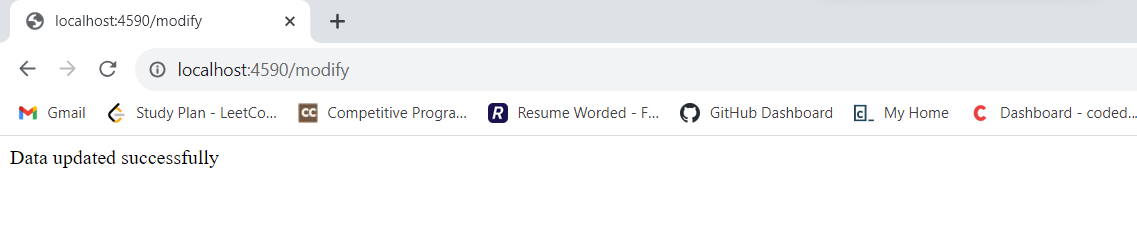
****

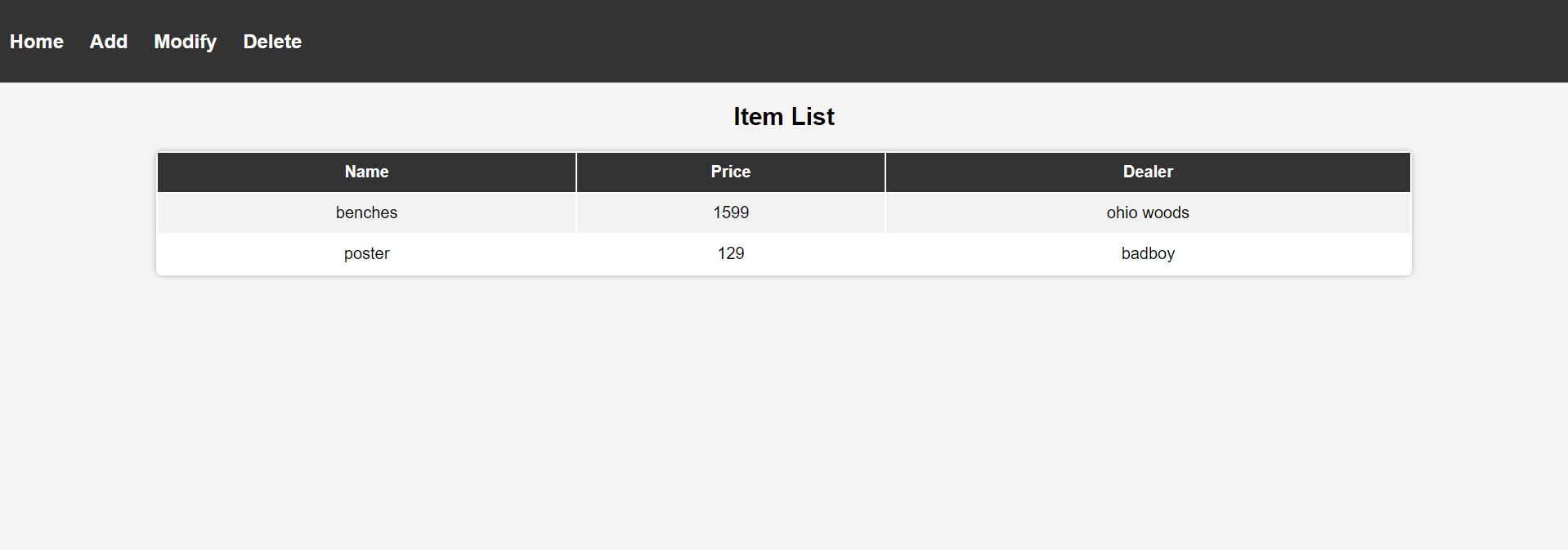
****

****

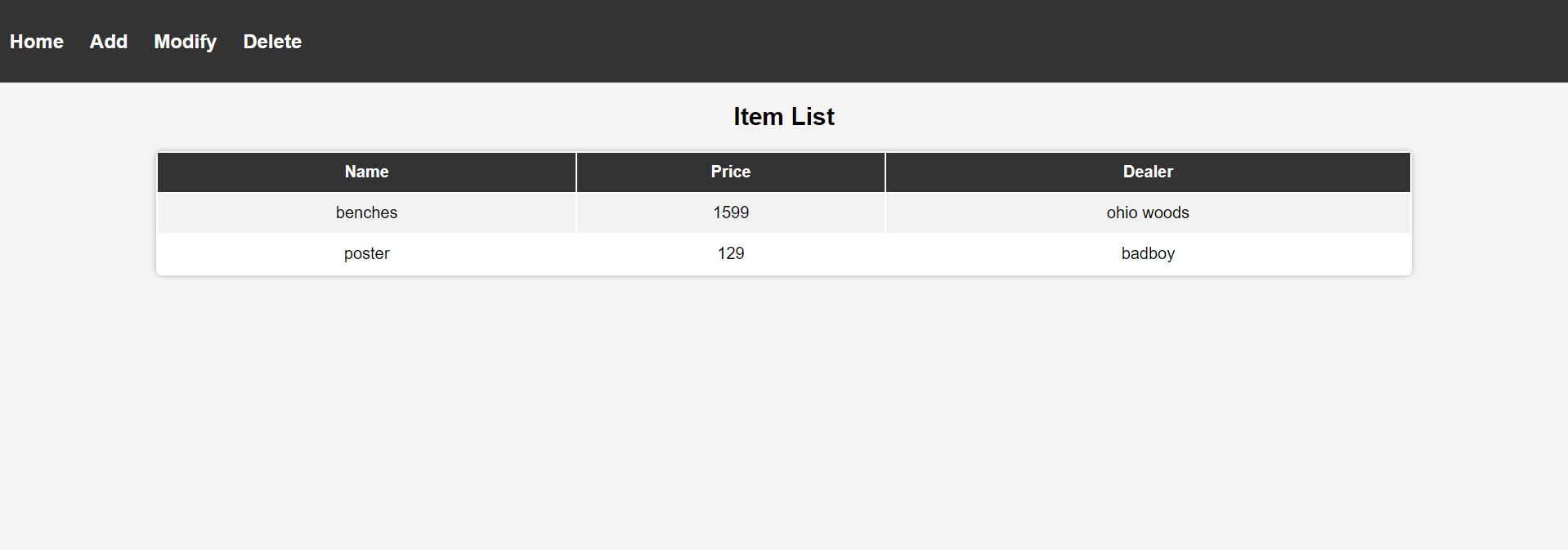
**Modifying Data –**

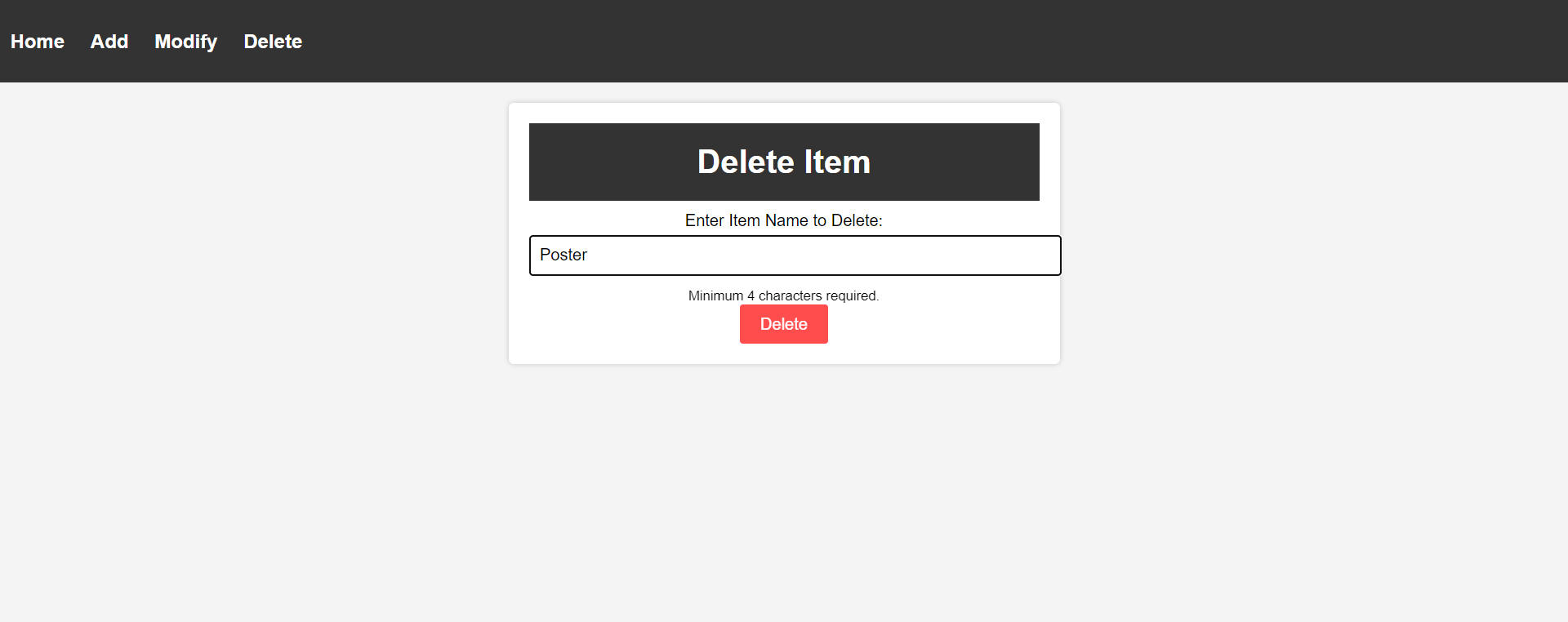
****

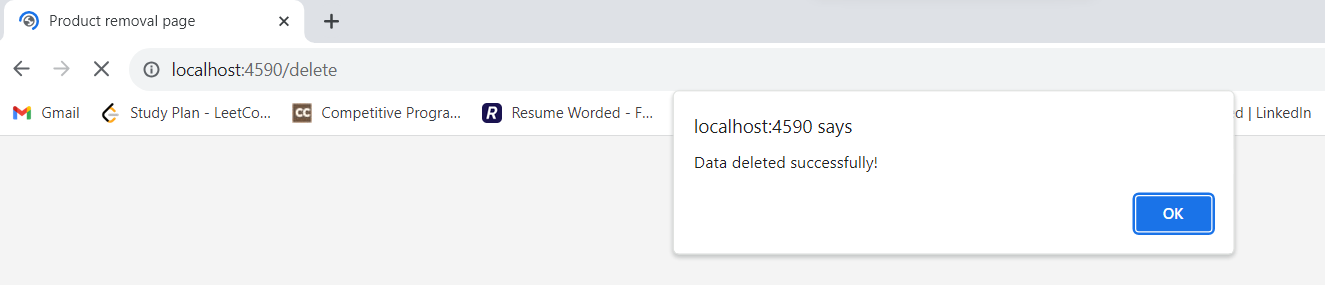
****

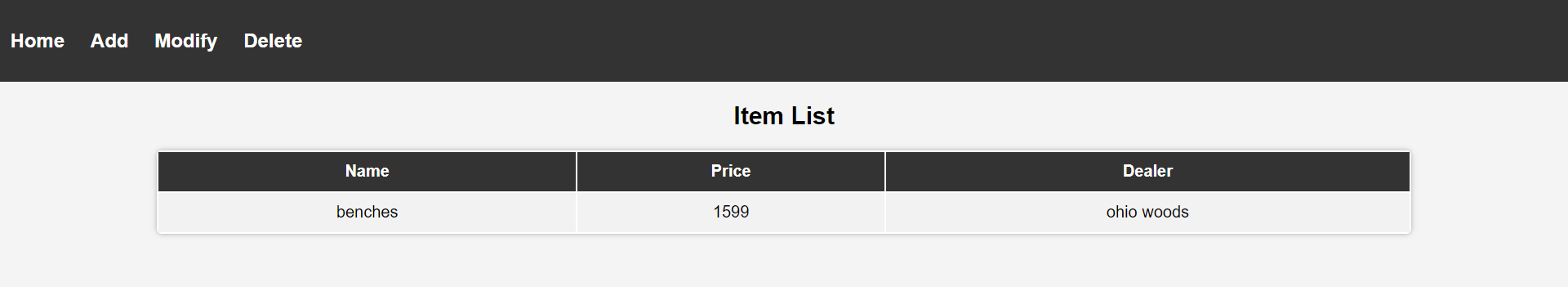
****

**Deleting Data –**

****

****

****

****