

NODEJS SERVER USING EXPRESS THAT STORES DATA FROM A FORM AS A JSON FILE

EXP NO: 2

AIM: -

Create a NodeJS server using Express that stores data from a form as a JSON file and displays it in another page. The redirect page should be prepared using Handlebars.

ALGORITHM: -

1. Initialize a Node.js project **using** `npm init -y`.
2. Install required packages: Express for server handling, body-parser for parsing form data, hbs for rendering Handlebars templates, fs (built-in) for writing/reading JSON files.
3. Import required modules (express, body-parser, fs, path).
4. Initialize the Express app, Set up body-parser middleware to handle urlencoded form data, Set up the view engine as hbs (Handlebars), Define the path to your views folder using `app.set()`.
5. Create a GET route / that renders a Handlebars template form.hbs.
6. Create a POST route /submit that:
 - a. Retrieves form data from `req.body`.
 - b. Writes the data to a `data.json` file using `fs.writeFile`.
 - c. Redirects the user to /display.
7. Create a GET route /display that:
8. Reads the `data.json` file using `fs.readFile`.
9. Parses the data. Renders a Handlebars template `display.hbs` with the data passed as context.
10. Create `views/form.hbs` containing a form that POSTs to /submit.
11. Create `views/display.hbs` to show the submitted data using Handlebars variables (like `{{name}}`, `{{email}}`, etc.).
12. Start the Express server on a specific port (e.g., 3000).
13. Open the browser and go to `http://localhost:3000` to test.

CODE: -

Server.js

```
const express = require('express');
const bodyParser = require('body-parser');
const fs = require('fs');
const path = require('path');

const app = express();
const PORT = 3000;

// Set up view engine
app.set('view engine', 'hbs');
app.set('views', path.join(__dirname, 'views'));

// Middleware
app.use(bodyParser.urlencoded({ extended: false }));
app.use(express.static('public'));

// Routes

// Show the form
app.get('/', (req, res) => {
  res.render('form');
});

// Handle form submission
app.post('/submit', (req, res) => {
  const formData = req.body;

  // Save to JSON file
  fs.writeFile('data.json', JSON.stringify(formData, null, 2), (err) => {
    if (err) return res.status(500).send('Error saving data.');
```

```

});
});

// Display submitted data
app.get('/display', (req, res) => {
  fs.readFile('data.json', 'utf-8', (err, data) => {
    if (err) return res.status(500).send('Error reading data.');
```

const parsedData = JSON.parse(data);

```

    res.render('display', parsedData); // data passed as context
  });
});

app.listen(PORT, () => {
  console.log(`Server running at http://localhost:${PORT}`);
});
```

Form HDS:-

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Submit Form</title>
</head>
<body>
  <h1>Submit Your Info</h1>
  <form action="/submit" method="POST">
    <label>Name: <input type="text" name="name" required></label><br><br>
    <label>Email: <input type="email" name="email" required></label><br><br>
    <label>Message: <textarea name="message"
required></textarea></label><br><br>
    <button type="submit">Submit</button>
  </form>
</body>
```

</html>

Display HDS

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Display Info</title>

</head>

<body>

<h1>Submitted Information</h1>

<p>Name: {{ name }}</p>

<p>Email: {{ email }}</p>

<p>Message: {{ message }}</p>

Back to form

</body>

</html>

RESULTS: -

Thus, the Create a NodeJS server using Express that stores data from a form as a JSON file and displays it in another page. The redirect page should be prepared using Handlebars.