

Handling Post

How to handle POST request

When building web applications, it's common to use HTTP POST requests to send data from the client (e.g., a form submission) to the server. In Express.js, handling POST requests involves using middleware to parse the incoming data and defining route handlers to process the data accordingly.

Middleware for Parsing JSON and Form Data:

Before handling POST requests, it's important to include middleware to parse the incoming data. Express provides built-in middleware for handling JSON and form data. Add the following middleware to your Express app:

```
// Middleware to parse JSON and form data
app.use(express.json());
app.use(express.urlencoded({ extended: true }));
```

Handling a POST Request:

```
// In-memory array to store text content
const textContent = [];

// Route to handle POST requests for adding text content
app.post('/add-content', (req, res) => {
  // Extract text content from the request body
  const newContent = req.body.content;

  // Validate the content (you might want to add more robust validation)
  if (!newContent) {
    // if there is an error it will send the code 400 and an error
    return res.status(400).json({ error: 'Content is required' });
  }

  // Add the content to the in-memory array
  textContent.push(newContent);

  // Respond with a success message
  res.status(201).json({ message: 'Content added successfully' });
});
```