

```

// Import the http module
const {createServer} = require("http");
// Create a null object
const methods = Object.create(null);

// Create a server
createServer((request, response) => {
  // Get the handler for the current request method or use the notAllowed function
  let handler = methods[request.method] || notAllowed;

  // Call the handler function with the request
  handler(request)
  // If there is an error catch it
  .catch(error => {
    // Return the status if there exists if not return a 500 error.
    if (error.status !== null) return error;
    return {body: String(error), status: 500};
  })

  // If there is no error then
  .then(({body, status = 200, type = "text/plain"}) => {
    // Send the response with the status, type, and body
    response.writeHead(status, {"Content-Type": type});
    // If body is a readable stream, pipe it to the writeable stream
    if (body && body.pipe) body.pipe(response);
    // Otherwise end the response
    else response.end(body);
  });
// Listen for requests on port 8000.
}).listen(8000);

// async function to handle not allowed methods
async function notAllowed(request) {
  // return 405 status and a not allowed message
  return {
    status: 405,
    body: `Method ${request.method} not allowed.`
  };
}

```

The first block of code creates a server that handles requests  
The second block handles any requests received that the server does not support.