

Gesendet von sie:

I use a .js file with OAK REST commands. How can I display data posted to this site immediately ?

```
import { Application, Router } from "https://deno.land/x/oak/mod.ts";
```

```
const app = new Application();
```

```
const router = new Router();
```

```
let data = [];
```

```
router
```

```
.post("/data", async (context) => {
```

```
  const body = await context.request.body().value;
```

```
  data.push(body);
```

```
  context.response.body = { message: "Data received", data: body };
```

```
})
```

```
.get("/data", (context) => {
```

```
  context.response.body = data;
```

```
});
```

```
app.use(router.routes());
```

```
app.use(router.allowedMethods());
```

```
console.log("Server running on port 8000");
```

```
await app.listen({ port: 8000 });
```

🔗 **Handle POST Requests:** The /data endpoint handles POST requests to receive and store data. The data is stored in an array called data.

🔗 **Display Data Immediately:** To display the data immediately after posting, you can use a frontend framework like React or plain JavaScript to fetch and display the data. Here's a simple example using plain JavaScript:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

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

  <title>Display Data</title>

</head>

<body>

  <form id="dataForm">

    <input type="text" id="dataInput" placeholder="Enter data" required>

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

  </form>

  <div id="dataDisplay"></div>

  <script>

    const form = document.getElementById('dataForm');

    const dataDisplay = document.getElementById('dataDisplay');

    form.addEventListener('submit', async (e) => {

      e.preventDefault();

      const dataInput = document.getElementById('dataInput').value;

      // Post data to the server

      await fetch('http://localhost:8000/data', {

        method: 'POST',

        headers: {

          'Content-Type': 'application/json',

        },

        body: JSON.stringify({ data: dataInput }),

      });

      // Fetch and display the updated data
```

```
    const response = await fetch('http://localhost:8000/data');  
    const data = await response.json();  
    dataDisplay.innerHTML = JSON.stringify(data, null, 2);  
  });  
</script>  
</body>  
</html>
```