

1. Make the 4 route - req object using body,params,query,

body

The screenshot shows the Thunder Client interface. In the top navigation bar, there are tabs for 'JS server.js', 'JS requestRoute.js', 'New Request', 'requestController.js', and 'Response'. The 'Body' tab is selected. Below it, the 'JSON' tab is also selected. The JSON content pane contains the following code:

```
1 {
2   "name": "React",
3   "age": 22
4 }
```

The response pane shows the following details:

- Status: 200 OK
- Size: 48 Bytes
- Time: 34 ms

The response body is:

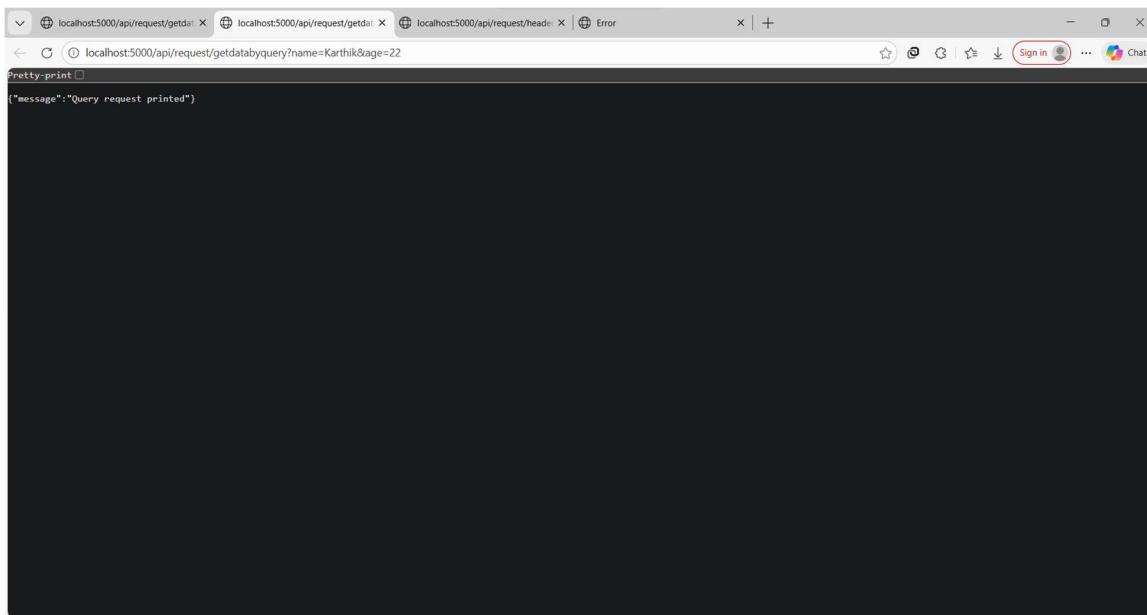
```
1 {
2   "message": "Request object printed in terminal"
3 }
```

Params

The screenshot shows a browser window with several tabs open. The active tab is a POST request to 'http://localhost:5000/api/request/getdata'. The response body is:

```
{"message": "Param request printed"}
```

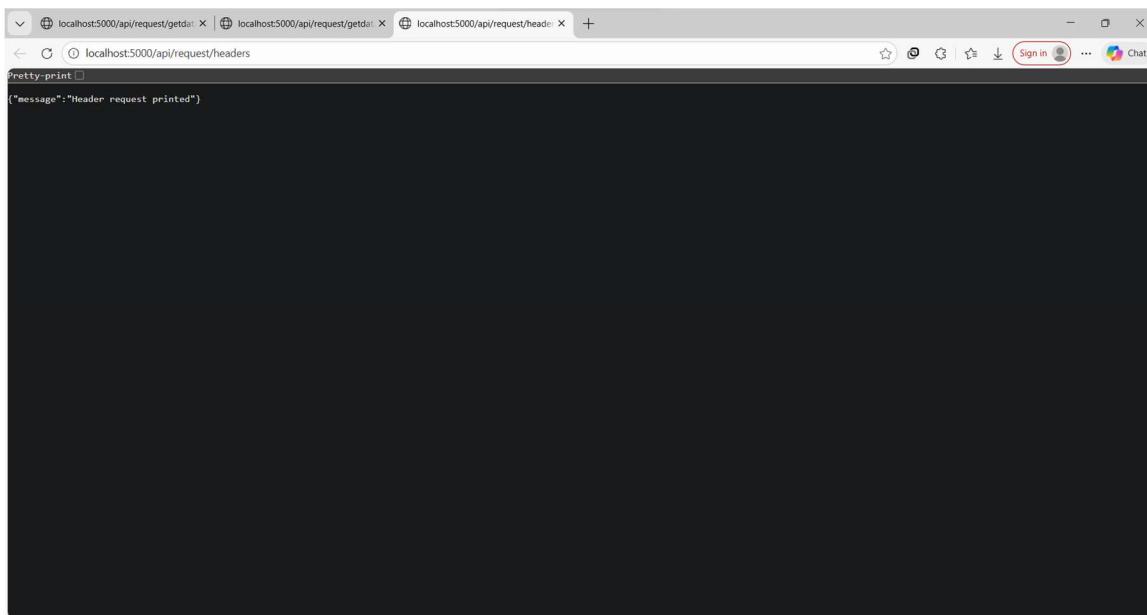
Query



A screenshot of a browser window with three tabs open. The active tab shows the URL `localhost:5000/api/request/getdata`. The page content is a JSON object:

```
{"message": "Query request printed"}
```

Header



A screenshot of a browser window with three tabs open. The active tab shows the URL `localhost:5000/api/request/headers`. The page content is a JSON object:

```
{"message": "Header request printed"}
```

2. What is Request Object?

The Request Object (`req`) is created by Express.js and contains all information about the incoming HTTP request from the client.

It includes:

- URL
- HTTP Method (GET, POST, PUT, DELETE)
- Headers
- Body Data
- Query Parameters
- Route Parameters

We use the request object whenever we need to read or access data sent by the client.

3. What is the use of Body Data? When do we use it?

req.body contains data sent by the client in the request body.

Used in:

- POST
- PUT
- PATCH

We use body data when:

- Submitting forms
- Sending JSON data
- Creating or updating database records

Example:

```
app.post("/register", (req, res) => {
  console.log(req.body.name);
});
```

Note: We must use middleware like express.json() to access body data.

4. What is the use of Headers Data? When do we use it?

req.headers contains metadata about the request.

Used for:

- Authentication tokens
- Content-Type
- Authorization
- Language preferences

Example:

```
console.log(req.headers.authorization);
```

Headers are used to send extra information that is not part of the main data.

5. What is the use of Params Data? When do we use it?

req.params contains route parameters.

Used when:

- Fetching a specific resource by ID
- Creating dynamic routes

Example:

```
app.get("/user/:id", (req, res) => {
  console.log(req.params.id);
});
```

If URL is /user/101, then output will be 101.

6. What is the use of Query Data? When do we use it?

req.query contains query string parameters.

Used for:

- Filtering
- Searching
- Sorting
- Pagination

Example:

```
app.get("/products", (req, res) => {
  console.log(req.query.category);
});
```

If URL is /products?category=mobile, then output will be mobile.

7. What is Request Object? When do we use it? (Summary)

The request object (req) in Express.js is used to access all data sent by the client to the server.

We use it:

- To read form data -> req.body
- To get route values -> req.params
- To read search filters -> req.query

- To check authentication -> req.headers
- To know request method -> req.method