What are Parameters in Node.js?

In Node.js, parameters are essential for handling data in HTTP requests. They allow you to pass information between the client and server, enabling dynamic interactions. Parameters can come from different parts of an HTTP request

Types of parameters

- **Query Parameters**: Added to the URL after ? (e.g., ?name=John&age=30).
- **Route Parameters**: Embedded directly in the URL path (e.g., /users/123).
- **Body Parameters**: Sent in the request body, often used in POST or PUT requests.
- **Header Parameters**: Included in the HTTP headers for metadata or authentication.

Query Parameters in Node.js

```
const url = require('url');
const parsedUrl = url.parse(req.url, true);
const queryParams = parsedUrl.query;
```

Query parameters are a common way to pass data in URLs. In Node.js, you can easily access them using the url module.

Query parameters are ideal for filtering, sorting, or pagination in APIs.

Route Parameters in Node.js

```
const pathSegments = req.url.split('/').filter(Boolean);
const userId = pathSegments[2]; // Assuming the URL is /api/users/123
```

Route parameters are dynamic segments of a URL path. They’re perfect for identifying specific resources, like a user ID or product slug.

Body Parameters in Node.js

```
let body = '';
req.on('data', chunk => body += chunk.toString());
req.on('end', () => {
    const parsedBody = JSON.parse(body);
    console.log(parsedBody); // { name: 'John', age: 30 }
});
```

- Body parameters are sent in the request body, typically in POST, PUT, or PATCH requests. They’re used for submitting data like form inputs or JSON payloads.
- Body parameters are essential for creating and updating resources.

Header Parameters in Node.js

```
const headers = req.headers;
console.log(headers.authorization); // Access the Authorization header
```

- Header parameters are part of the HTTP headers and are often used for metadata, authentication, or content negotiation
- Headers are crucial for secure and efficient communication between clients and servers.