

Request object in Node.js

The req object represents the HTTP Request and has properties for the request query string, parameters, body, HTTP Header ...

The req object represents the HTTP Request and has properties for the request query string, parameters, body, HTTP Header, .

Before entering the chapter, we will learn what the middleware concept is:

According to vi.wikipedia: Middleware is computer software that connects software components or applications together. This type of software includes a set of services that allow interaction between processes running on one or more machines.

Properties of the Request object in Node.js

The table below lists some commonly used properties of the Request object in Node.js:

Stt Properties & Description

1 **req.app**

This property holds a reference to the Express application using Middleware

2 **req.baseUrl**

The URL link on which a router is mounted

3 **req.body**

Contains key-value pairs of data submitted in the body section of the Request. By default, it is undefined, and is

generated when you use a Middleware to parse the body of the request (**body-parser** example)

4 **req.cookies**

When using cookie-parser middleware, this attribute is an object that contains Cookies sent by Request

5 **req.fresh**

Determine whether or not Request is "fresh." This attribute is in contrast to req.stale.

6 **req.hostname**

Contains the hostname section from the "Host" section of the HTTP header.

7 **req.ip**

Remote IP address of the request.

8 **req.path**

Contains the path part of the URL in the Request.

9 **req.protocol**

The protocol string of Request, which is "http" or "https" when requested with TLS

10 **req.query**

An object contains an attribute for each query string parameter in Router

11 **req.route**

A string representing the current route is connected to

12 **req.secure**

Is true if a TLS connection is successfully established

13 **req.signedCookies**

When using cookie-parser Middleware, this property contains Cookies sent by Request

14 **req.stale**

Determining whether or not Request is "stale", this attribute is in contrast to the req.fresh attribute.

15 **req.subdomains**

An array consists of subdomains in the domain name section of the Request

Method of Request object in Node.js

Method req.accepts (types)

```
req . accepts ( types )
```

This method checks whether the Content-type is acceptable, based on the Request HTTP Header field of the Request. For example:

```
// Accept: text/html req . accepts ( 'html' ); // => "html" // Accept: text/*,
```

Method req.get (field)

```
req . get ( field )
```

This method returns the specific Header field of the Request. For example:

```
req . get ( 'Content-Type' ); // => "text/plain" req . get ( 'content-type' );
```

Method req.is (type)

```
req . is ( type )
```

This method returns true if the Content-Type field of the Request is connected to the Type MIME type specified by the type parameter. For example:

```
// With Content-Type: text/html; charset=utf-8 req . is ( 'html' ); req . is
```

Req.param method (name [, defaultValue])

```
req . param ( name [, defaultValue ])
```

This method returns the value of the name parameter. For example:

```
// ?name=tobi req . param ( 'name' ) // => "tobi" // POST name=tobi req . param
```

According to Tutorialspoint

Previous article: RESTful API in Node.js

Next lesson: Response object in Node.js

You finished reading the article "**Request object in Node.js**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.