

Chapter 5 Express.js POST Request

GET and POST both are two common HTTP requests used for building REST API's. POST requests are used to send large amount of data.

Express.js facilitates you to handle GET and POST requests using the instance of express.

Express.js POST Method

Post method facilitates you to send large amount of data because data is send in the body. Post method is secure because data is not visible in URL bar but it is not used as popularly as GET method. On the other hand GET method is more efficient and used more than POST.

Let's take an example to demonstrate POST method.

Example1:

Fetch data in JSON format

File: Index.html

```
<html>
<body>
<form action="http://127.0.0.1:8000/process_post" method="POST">
First Name: <input type="text" name="first_name"> <br>
Last Name: <input type="text" name="last_name">
<input type="submit" value="Submit">
</form>
</body>
</html>
```

File: post_example1.js

```
var express = require('express');
var app = express();
var bodyParser = require('body-parser');
// Create application/x-www-form-urlencoded parser
```

WEBISOFTTECH

Web-www.webisoftech.com

E-mail- enquiry@webisoftech.com

Phone- +91 9766962674

```
var urlencodedParser = bodyParser.urlencoded({ extended: false })
app.use(express.static('public'));
app.get('/index.html', function (req, res) {
  res.sendFile( __dirname + "/" + "index.html" );
})
app.post('/process_post', urlencodedParser, function (req, res) {
  // Prepare output in JSON format
  response = {
    first_name:req.body.first_name,
    last_name:req.body.last_name
  };
  console.log(response);
  res.end(JSON.stringify(response));
})
var server = app.listen(8000, function () {
  var host = server.address().address
  var port = server.address().port
  console.log("Example app listening at http://%s:%s", host, port)
})
```

Open the page index.html and fill the entries:

Now, you get the data in JSON format.