

Calma, Jimwell T.

WD-302

Get\_Sample

The image shows a web browser window on the left and a code editor on the right. The browser window displays a JSON response from a POST request to `http://127.0.0.1:3013/process_post`. The JSON object contains user information: `{ "first_name": "Jimwell", "last_name": "Pogi", "email": "asdsadsads@GMAIL.COM", "phone": "+639289606868", "gender": "male" }`.

The code editor on the right shows the `server.js` file. It defines an Express application with routes for `/`, `/about`, `/blog`, `/contact`, and `/process_post`. The `/process_post` route uses `urlencodedParser` to parse the request body and returns a JSON response with the user details. The server is listening on port 3013.

Below the code editor, a terminal window shows the command prompt where the server was started: `zsh: suspended node server.js` and `Server is running on http://localhost:3001`. The REST client window shows the details of the POST request to `http://localhost:3001/process_post`, including the headers and the JSON body.

```
127.0.0.1:3013/process_post
{"first_name":"Jimwell","last_name":"Pogi","email":"asdsadsads@GMAIL.COM","phone":"+639289606868","gender":"male"}

Activity_Method — node server.js — 76x24
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 midterms % cd activity_method
sjh-706@sjh-706-28 activity_method % node server.js
Server is running on http://localhost:3001
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 activity_method % node server.js
Server is running on http://localhost:3001
{
  first_name: 'Jimwell',
  last_name: 'Pogi',
  email: 'asdsadsads@GMAIL.COM',
  phone: '+639289606868',
  gender: 'male'
}
{
  first_name: 'Jimwell',
  last_name: 'Pogi',
  email: 'asdsadsads@GMAIL.COM',
  phone: '+639289606868',
  gender: 'male'
}

Midterms > Activity_Method > JS server.js > ...
1  var express = require('express');
2  var app = express();
3
4  app.use(express.static('Public'));
5  var bodyParser = require('body-parser');
6
7  var urlencodedParser = bodyParser.urlencoded({extended: false });
8  app.get('/', function (req, res) {
9    res.sendFile(__dirname + "/" + "index.html");
10 })
11
12 app.get('/about', function (req, res) {
13   res.sendFile(__dirname + "/" + "about.html");
14 })
15 app.get('/blog', function (req, res) {
16   res.sendFile(__dirname + "/" + "blog.html");
17 })
18 app.get('/contact', function (req, res) {
19   res.sendFile(__dirname + "/" + "contact.html");
20 })
21
22
23 app.post('/process_post', urlencodedParser, function (req, res) {
24   response = {
25     first_name: req.body.first_name,
26     last_name: req.body.last_name,
27     email: req.body.email,
28     phone: req.body.phone,
29     gender: req.body.gender
30   };
31   console.log(response);
32   res.end(JSON.stringify(response));
33 });
34
35 app.listen(3013, () => {
36   console.log('Server is running on http://localhost:3001');
37 });
```

## GET METHOD

The screenshot displays a development environment with three main components: a web browser, a code editor, and a terminal.

**Web Browser:** The address bar shows the URL `127.0.0.1:3025/process_get?first_name=Jimwell&last_name=Pogi`. The page content displays a JSON object: `{ "first_name": "Jimwell", "last_name": "Pogi" }`.

**Code Editor:** The editor shows two files: `index.html` and `server.js`.

**index.html:**

```
1 <form action="http://127.0.0.1:3025/process_get" method="GET">
2   First Name: <input type="text" name =
3     "first_name"><br>
4   Last Name: <input type="text"
5     name="last_name">
6   <input type="submit" value="Submit">
7 </form>
```

**server.js:**

```
1 const express = require('express');
2 const app = express();
3
4 app.get('/', function(req, res) {
5   res.sendFile(__dirname + '/' + 'index.html');
6 })
7
8 app.get('/process_get', function (req, res) {
9   response = {
10     first_name: req.query.first_name,
11     last_name: req.query.last_name
12   };
13
14   console.log(response);
15   res.end(JSON.stringify(response));
16 });
17
18 app.listen(3025, () => {
19   console.log('Server is running on
20     http://localhost:3001');
21 });
```

**Terminal:** The terminal shows the command `node server.js` being executed. The output indicates the server is running on `http://localhost:3001` and displays the received query parameters: `{ first_name: 'Jimwell', last_name: 'Pogi' }`.

## POST METHOD

The screenshot displays a development environment for a Node.js web server. On the left, a web browser at `127.0.0.1:3027/process_post` shows the JSON response: `{ "first_name": "Jimwell", "last_name": "asdasds" }`. The middle pane shows the `server.js` file with the following code:

```
6 var urlencodedParser = bodyParser.urlencoded({extended: false})
7 app.get('/', function (req, res) {
8   res.sendFile(__dirname + "/" + "index.html");
9 })
10
11 app.post('/process_post', urlencodedParser, function (req, res) {
12
13
14   response = {
15     first_name: req.body.first_name,
16     last_name: req.body.last_name
17
18   }
19   console.log(response);
20   res.end(JSON.stringify(response));
21 })
22 app.listen(3027, () => {
23   console.log('Server is running on http://localhost:3001');
24 })
```

The right pane shows the `index.html` file with a form:

```
1 <form action="http://127.0.0.1:3025/process_get" method="GET">
2   First Name: <input type="text" name = "first_name"><br>
3   Last Name: <input type="text" name="last_name">
4   <input type="submit" value="Submit">
5 </form>
```

The bottom terminal window shows the server startup and a successful POST request:

```
Node.js v18.13.0
sjh-706@sjh-706-28 Post_Method % node server.js
Server is running on http://localhost:3001
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 Post_Method % node server.js
Server is running on http://localhost:3001
{ first_name: 'Jimwell', last_name: 'asdasds' }
```

## Activity

The image shows a web development environment with three main components:

- Terminal (Left):** Displays the command-line interface for running a Node.js server. The commands and output are as follows:

```
sjh-706@sjh-706-28 Midterms % ls
pipe-reactive
sjh-706@sjh-706-28 Midterms % cd ..
sjh-706@sjh-706-28 ~ % cd 2160-JTC
sjh-706@sjh-706-28 2160-JTC % cd midterms
sjh-706@sjh-706-28 midterms % node server.js
Server is running on http://localhost:3001
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 midterms % cd activity_method
sjh-706@sjh-706-28 activity_method % node server.js
Server is running on http://localhost:3001
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 activity_method % node server.js
Server is running on http://localhost:3001
{
  first_name: 'Jimwell',
  last_name: 'Pogi',
  email: 'asdsadsads@GMAIL.COM',
  phone: '+639289606868',
  gender: 'male'
}
```
- Web Browser (Center):** Displays the "Calma's WCSERVER Page" at <http://localhost:3001>. The page title is "Calma's WCSERVER Page" and the subtitle is "This is the HOME page." The form contains the following fields:
  - First Name:
  - Last Name:
  - Enter your email:
  - Enter your phone number:
  - Gender: ☒ Male ☐ Female
  - Are you Happy? ☒
  - Submit button
- Code Editor (Right):** Shows the `index.html` file with the following HTML structure:

```
<html lang="en">
<head>
</head>
<body>
  <ul>
    <li><a class="active" href="#home">Home</a></li>
    <li><a href="#about">About</a></li>
    <li><a href="#blog">Blog</a></li>
    <li><a href="#contact">Contact</a></li>
  </ul>
  <h1>Calma's WCSERVER Page</h1>
  <h2>This is the HOME page.</h2>
  <form action="http://127.0.0.1:3013/process_post" method="post">
    <label for="first_name">First Name:</label>
    <input type="text" id="first_name" name="first_name"><br>
    <label for="last_name">Last Name:</label>
    <input type="text" id="last_name" name="last_name"><br>
    <label for="email">Enter your email:</label>
    <input type="email" id="email" name="email"><br>
    <label for="phone">Enter your phone number:</label>
    <input type="tel" id="phone" name="phone" pattern="\+63[
  </div class="gender-container">
    <label for="male">Male</label>
    <input type="radio" id="male" name="gender" value="male">
    <label for="female">Female</label>
    <input type="radio" id="female" name="gender" value="female">
  </div>
  <label for="happy">
    <input type="checkbox" id="happy" name="happy" value="happy">
    Are you Happy?
  </label><br>
```



127.0.0.1:3013/process\_post

```
{ "first_name": "Jimwell", "last_name": "Pogi", "email": "asdsadsads@GMAIL.COM", "phone": "+639289606868", "gender": "male" }
```

Activity\_Method — node server.js — 76x24

```
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 midterms % cd activity_method
sjh-706@sjh-706-28 activity_method % node server.js
Server is running on http://localhost:3001
^Z
zsh: suspended node server.js
sjh-706@sjh-706-28 activity_method % node server.js
Server is running on http://localhost:3001
{
  first_name: 'Jimwell',
  last_name: 'Pogi',
  email: 'asdsadsads@GMAIL.COM',
  phone: '+639289606868',
  gender: 'male'
}
{
  first_name: 'Jimwell',
  last_name: 'Pogi',
  email: 'asdsadsads@GMAIL.COM',
  phone: '+639289606868',
  gender: 'male'
}
}
```

2160-jtc

index.html # styles.css JS server.js

Midterms > Activity\_Method > JS server.js > ...  
1 var express = require('express');  
2 var app = express();  
3  
4 app.use(express.static('Public'));  
5 var bodyParser = require('body-parser');  
6  
7 var urlencodedParser = bodyParser.urlencoded({extended: false });  
8 app.get('/', function (req, res) {  
9 res.sendFile(\_\_dirname + "/" + "index.html");  
10 }  
11  
12 app.get('/about', function (req, res) {  
13 res.sendFile(\_\_dirname + "/" + "about.html");  
14 }  
15 app.get('/blog', function (req, res) {  
16 res.sendFile(\_\_dirname + "/" + "blog.html");  
17 }  
18 app.get('/contact', function (req, res) {  
19 res.sendFile(\_\_dirname + "/" + "contact.html");  
20 }  
21  
22  
23 app.post('/process\_post', urlencodedParser, function (req, res) {  
24 response = {  
25 first\_name: req.body.first\_name,  
26 last\_name: req.body.last\_name,  
27 email: req.body.email,  
28 phone: req.body.phone,  
29 gender: req.body.gender  
30 };  
31 console.log(response);  
32 res.end(JSON.stringify(response));  
33 }  
34  
35  
36 app.listen(3013, () => {  
37 console.log('Server is running on http://localhost:3001');  
});

Ln 21, Col 1 Sp