

```

C:\Users\natha>curl -v localhost:8081
* Host localhost:8081 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:8081...
* Connected to localhost (::1) port 8081
* GET / HTTP/1.1
* Host: localhost:8081
* User-Agent: curl/8.9.1
* Accept: */*

* Request completely sent off
* HTTP/1.1 500 Internal Server Error
* X-Powered-By: Express
* Access-Control-Allow-Origin: *
* Content-Security-Policy: default-src 'none'
* X-Content-Type-Options: nosniff
* Content-Type: text/html; charset=utf-8
* Content-Length: 1217
* Date: Thu, 07 Nov 2024 19:06:00 GMT
* Connection: keep-alive
* Keep-Alive: timeout=5

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Error</title>
</head>
<body>
<pre>TypeError: res.send is not a function<br> &nbsp; &nbsp; &nbsp;at C:\319NodeJs\activity17\app.js:13:9<br> &nbsp; &nbsp; &nbsp;at Layer.handle [as handle_request] (C:\319NodeJs\activity17\node_modules\express\lib\router\layer.js:95:5)<br> &nbsp; &nbsp; &nbsp;at next (C:\319NodeJs\activity17\node_modules\express\lib\router\route.js:149:13)<br> &nbsp; &nbsp; &nbsp;at Route.dispatch (C:\319NodeJs\activity17\node_modules\express\lib\router\route.js:119:3)<br> &nbsp; &nbsp; &nbsp;at Layer.handle [as handle_request] (C:\319NodeJs\activity17\node_modules\express\lib\router\layer.js:95:5)<br> &nbsp; &nbsp; &nbsp;at C:\319NodeJs\activity17\node_modules\express\lib\router\index.js:284:15<br> &nbsp; &nbsp; &nbsp;at Function.process_params (C:\319NodeJs\activity17\node_modules\express\lib\router\index.js:346:12)<br> &nbsp; &nbsp; &nbsp;at next (C:\319NodeJs\activity17\node_modules\express\lib\router\index.js:280:10)<br> &nbsp; &nbsp; &nbsp;at jsonParser (C:\319NodeJs\activity17\node_modules\body-parser\lib\types\json.js:113:7)<br> &nbsp; &nbsp; &nbsp;at Layer.handle [as handle_request] (C:\319NodeJs\activity17\node_modules\express\lib\router\layer.js:95:5)</pre>
</body>
</html>

```

Here is the first Image. It was given a 500 Internal Server Error instead of a 404 not found, but I believe the reason for this is that the Method that we made for “/” did not have a req or res argument and was left blank.

```

C:\Users\natha>curl -v localhost:8081/ListRobots
* Host localhost:8081 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:8081...
* Connected to localhost (::1) port 8081
* GET /ListRobots HTTP/1.1
* Host: localhost:8081
* User-Agent: curl/8.9.1
* Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< X-Powered-By: Express
< Access-Control-Allow-Origin: *
< Content-Type: text/html; charset=utf-8
< Content-Length: 563
< ETag: W/"233-K2ZLhNgjAHiBRs0ZF7RndwVj6Is"
< Date: Thu, 07 Nov 2024 18:46:32 GMT
< Connection: keep-alive
< Keep-Alive: timeout=5
<
[
  {
    "id": 1,
    "name": "Robot 1",
    "price": 24.50,
    "description": "This is a description of Robot 1",
    "imageUrl": "https://robohash.org/robot1"
  },
  {
    "id": 2,
    "name": "Robot 2",
    "price": 32.90,
    "description": "This is a description of Robot 2",
    "imageUrl": "https://robohash.org/robot2"
  },
  {
    "id": 3,
    "name": "Robot 3",
    "price": 20.50,

```

Status code is 200 OK which means it was successful and the Content type is text/html and the body is the Robot info text that is returned by the server on the bottom.

```
C:\Users\natha>curl -v localhost:8081/ListRobots
* Host localhost:8081 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:8081...
* Connected to localhost (::1) port 8081
> GET /ListRobots HTTP/1.1
> Host: localhost:8081
> User-Agent: curl/8.9.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 404 Not Found
< X-Powered-By: Express
< Access-Control-Allow-Origin: *
< Content-Type: text/html; charset=utf-8
< Content-Length: 563
< ETag: W/"233-K2ZlhNgjAHiBRs0ZF7RndwVj6Is"
< Date: Thu, 07 Nov 2024 18:39:13 GMT
< Connection: keep-alive
< Keep-Alive: timeout=5
<
[
  {
    "id": 1,
    "name": "Robot 1",
    "price": 24.50,
    "description": "This is a description of Robot 1",
    "imageUrl": "https://robohash.org/robot1"
  },
  {
    "id": 2,
    "name": "Robot 2",
    "price": 32.90,
    "description": "This is a description of Robot 2",
    "imageUrl": "https://robohash.org/robot2"
  },
  {
    "id": 3,
    "name": "Robot 3",
    "price": 20.50,
    "description": "This is a description of Robot 3",
    "imageUrl": "https://robohash.org/robot3"
  }
]
```

The Status code is now 404 Not Found but it still sends the data through since the 404 was just a hardcoded value and the request is still valid.

The screenshot shows a web browser window with the address bar displaying `localhost:8081/listRobots`. The page content shows a JSON array of robot data, including Robot 1, Robot 2, and Robot 3, with their respective IDs, names, prices, descriptions, and image URLs.

The Chrome DevTools Network tab is open, showing a single request named `listRobots`. The request details are as follows:

- Request URL:** `http://localhost:8081/listRobots`
- Request Method:** `GET`
- Status Code:** `404 Not Found`
- Remote Address:** `[::1]:8081`
- Referrer Policy:** `strict-origin-when-cross-origin`

The **Response Headers** section shows the following details:

- Access-Control-Allow-Origin:** `*`
- Connection:** `keep-alive`
- Content-Length:** `563`
- Content-Type:** `text/html; charset=utf-8`
- Date:** `Thu, 07 Nov 2024 18:50:54 GMT`
- ETag:** `W/"233-K2ZlhNgjAHiBRs0ZF7RndwVj6Is"`
- Keep-Alive:** `timeout=5`

Here it shows the Status code 404 Not Found that we hardcoded in

```
C:\Users\natha>curl -v localhost:8081/
* Host localhost:8081 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:8081...
* Connected to localhost (::1) port 8081
> GET / HTTP/1.1
> Host: localhost:8081
> User-Agent: curl/8.9.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< X-Powered-By: Express
< Access-Control-Allow-Origin: *
< Content-Type: text/html; charset=utf-8
< Content-Length: 89
< ETag: W/"59-BtUmF6XeSSLFK8IpxkvPoZ0vBeY"
< Date: Thu, 07 Nov 2024 18:54:17 GMT
< Connection: keep-alive
< Keep-Alive: timeout=5
<
<h1 style='color:Green;background-color: black;border: 0px; '>Hello World From Node </h1>* Connection #0 to host localhost left intact

C:\Users\natha>
```

localhost:8081

Issues · cs309 / 2024 · spin the wheel - Go · 11.2.21_robots.json · Node.js — Run Java · Activity17 In-class · Activity17 In-class · localhost:8081

My Apps Dashboar... · Gmail · Outlook · CyHire · GitHub · Canvas · Projects - GitLab · 9anime - Watch Ani... · YouTube · Team - Iowa State ... · mmoamenn/Lucky...

Hello World From Node

Elements · Console · Sources · Network

Filter · Invert · More filters

Fetch/XHR · Doc · CSS · JS · Font · Img · Media · Manifest · WS · Wasm · Other

20 ms · 40 ms · 60 ms · 80 ms · 100 ms

Name	Status	Type	Initiator	Size	Time
localhost	200	docume...	Other	349 B	3 ms

```
C:\Users\natha>curl -v localhost:8081/person
* Host localhost:8081 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:8081...
* Connected to localhost (::1) port 8081
> GET /person HTTP/1.1
> Host: localhost:8081
> User-Agent: curl/8.9.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< X-Powered-By: Express
< Access-Control-Allow-Origin: *
< Content-Type: application/json; charset=utf-8
< Content-Length: 60
< ETag: W/"3c-BeqiMqtijU9zNSyXx+uf+ZbBKAc"
< Date: Thu, 07 Nov 2024 18:58:10 GMT
< Connection: keep-alive
< Keep-Alive: timeout=5
<
{"name":"alex","email":"alex@mail.com","job":"software dev"}* Connection #0 to host localhost left intact

C:\Users\natha>
```

The screenshot shows a web browser window with a REST client interface. The address bar shows the URL `localhost:8081/person`. The interface includes a "Pretty-print" button and a text area containing the JSON response: `{"name":"alex","email":"alex@mail.com","job":"software dev"}`. On the right side, there is a "Network" tab showing a single request. The request details are as follows:

Name	Status	Type	Initiator	Size	Time
person	200	document	Other	327 B	3 ms