



Send JSON-format student data from node.js server via NGRok

19673 Jisen Fang



Table of contents

1. Concept
2. student_info.js
3. node.js server
4. Localhost result
5. Set up NGRok
6. Start NGRok
7. Result



Concept

JSON: JSON stands for JavaScript Object Notation. It is a lightweight format for storing and transporting data that often used in data sent from a server to a web page.

Node.js: Node.js is a back-end JavaScript runtime environment that uses JavaScript on the server.

NGRok: Ngrok could map your local private address to a public address in this project.



student_info.js

```
const http = require('http');

const students = [
  { id: 11111, name: 'Bruce Lee', score: 84 },
  { id: 22222, name: 'Jackie Chen', score: 93 },
  { id: 33333, name: 'Jet Li', score: 88 },
];
```

```
const server = http.createServer((req, res) => {
  if (req.url.startsWith('/api/score?student_id=')) {
    const studentId = parseInt(req.url.split('=')[1]);
    const student = students.find(student => student.id ===
studentId);

    if (student) {
      res.writeHead(200, { 'Content-Type': 'application/json'
});
      res.end(JSON.stringify(student));
    } else {
      res.writeHead(404, { 'Content-Type': 'text/plain' });
      res.end('Student not found');
    }
  } else {
    res.writeHead(404, { 'Content-Type': 'text/plain' });
    res.end('Invalid URL');
  }
});

const PORT = 8000;
server.listen(PORT, () => {
  console.log(`Server is running on port ${PORT}`);
});
```



node.js server

1. Create a student-server directory

```
mkdir student-server
```

```
cd student-server
```

2. Initialize a Node.js Project

```
npm init -y
```

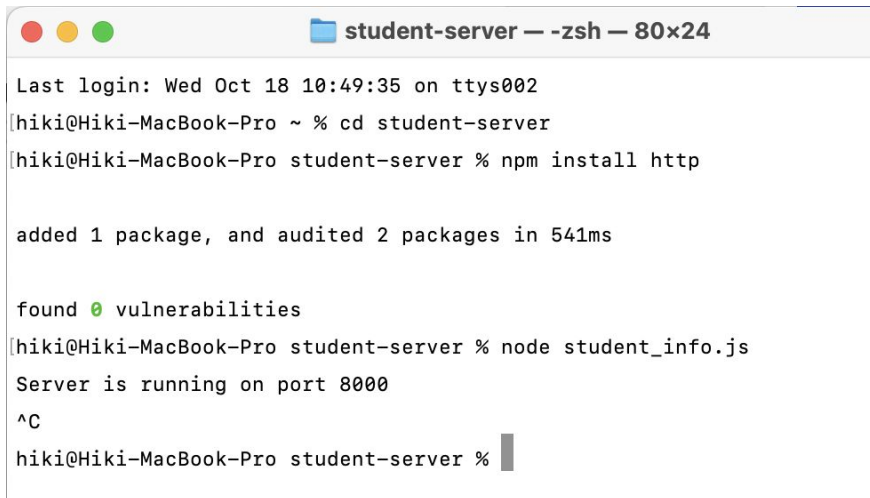
3. Copy the student_info.js to the directory

4. Install required dependencies

```
npm install http
```

5. Run the server

```
node student_info.js
```



```
student-server — zsh — 80x24

Last login: Wed Oct 18 10:49:35 on ttys002
hiki@Hiki-MacBook-Pro ~ % cd student-server
hiki@Hiki-MacBook-Pro student-server % npm install http

added 1 package, and audited 2 packages in 541ms

found 0 vulnerabilities
hiki@Hiki-MacBook-Pro student-server % node student_info.js
Server is running on port 8000
^C
hiki@Hiki-MacBook-Pro student-server %
```



localhost

localhost:8000/api/score?student_id=11111

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
{
  "id": 11111,
  "name": "Bruce Lee",
  "score": 84
}
```

localhost:8000/api/score?student_id=33333

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
{
  "id": 33333,
  "name": "Jet Li",
  "score": 88
}
```

localhost:8000/api/score?student_id=44444

Student not found

2. Add authToken, you need to sign up a account for authToken

```
ngrok config add-authtoken <token>
```

```
remote: Compressing objects: 100% (37/37), done.
remote: Total 128 (delta 24), reused 49 (delta 17), pack-reused 60
Receiving objects: 100% (128/128), 19.38 KiB | 4.84 MiB/s, done.
Resolving deltas: 100% (35/35), done.
Tapped 1 cask (13 files, 28.8KB).

==> Downloading https://bin.equinox.io/a/mCL51Hi52gn/ngrok-v3-3.4-darwin-arm64
##### 100.0%

==> Installing Cask ngrok

==> Linking Binary 'ngrok' to '/opt/homebrew/bin/ngrok'

🍺 ngrok was successfully installed!

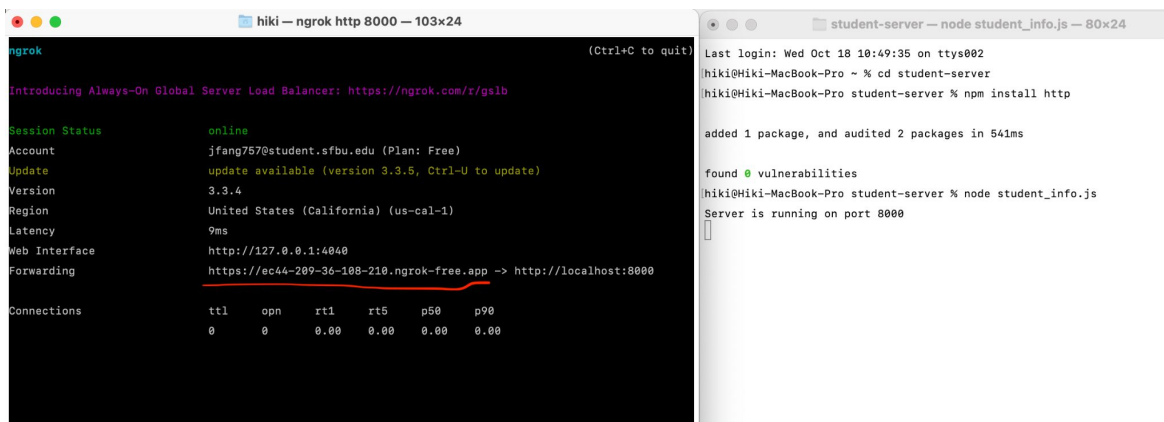
[hiki@Hiki-MacBook-Pro ~ %] ngrok config add-authtoken 2Zmm... RaT9yqcPQPC6rNq
Authtoken saved to configuration file: /Users/hiki/Library/Application Support/ngrok/ngrok.yml

[hiki@Hiki-MacBook-Pro ~ %] ngrok http 80
hiki@Hiki-MacBook-Pro ~ % ngrok http 8000
```

Start NGrok

1. Start a tunnel

`ngrok http 8000`



The image shows two terminal windows. The left window, titled 'hiki — ngrok http 8000 — 103x24', displays the ngrok interface. It shows session status as 'online', account as 'jfang757@student.sfbu.edu (Plan: Free)', and version as '3.3.4'. The forwarding URL is 'https://ec44-209-36-108-210.ngrok-free.app', which is underlined with a red line. The right window, titled 'student-server — node student_info.js — 80x24', shows the execution of 'npm install http' and 'node student_info.js', resulting in the message 'Server is running on port 8000'.

```
ngrok
(CTRL+C to quit)

Introducing Always-On Global Server Load Balancer: https://ngrok.com/t/gslb

Session Status      online
Account             jfang757@student.sfbu.edu (Plan: Free)
Update              update available (version 3.3.5, Ctrl-U to update)
Version             3.3.4
Region              United States (California) (us-cal-1)
Latency             9ms
Web Interface        http://127.0.0.1:4040
Forwarding           https://ec44-209-36-108-210.ngrok-free.app -> http://localhost:8000

Connections
  ttl  opn  rt1  rt5  p50  p90
    0    0    0.00 0.00 0.00 0.00

Last login: Wed Oct 18 10:49:35 on ttys002
hiki@Hiki-MacBook-Pro ~ % cd student-server
hiki@Hiki-MacBook-Pro student-server % npm install http

added 1 package, and audited 2 packages in 541ms

found 0 vulnerabilities
hiki@Hiki-MacBook-Pro student-server % node student_info.js
Server is running on port 8000
```

Use the address above the red line to access server

Ex: `http://ec44-209-36-108-210.ngrok-free.app/api/score?student_id=11111`

Result

←	→	↻	🔒	https://ec44-209-36-108-210.ngrok-free.app/api/score?student_id=11111
JSON	Raw Data	Headers		
Save	Copy	Collapse All	Expand All	🔍 Filter JSON
id:	11111			
name:	"Bruce Lee"			
score:	84			

←	→	↻	🔒	https://ec44-209-36-108-210.ngrok-free.app/api/score?student_id=22222
JSON	Raw Data	Headers		
Save	Copy	Collapse All	Expand All	🔍 Filter JSON
id:	22222			
name:	"Jackie Chen"			
score:	93			

←	→	↻	🔒	https://ec44-209-36-108-210.ngrok-free.app/api/score?student_id=33333
JSON	Raw Data	Headers		
Save	Copy	Collapse All	Expand All	🔍 Filter JSON
id:	33333			
name:	"Jet Li"			
score:	88			

←	→	↻	🔒	https://ec44-209-36-108-210.ngrok-free.app/api/score?student_id=44444
---	---	---	---	---

Student not found