



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

19673 Jisen Fang



# Table of contents

1. Concept
2. Student Info
3. student\_info.js
4. node.js server
5. Localhost result
6. Set up NGROK
7. Start NGROK
8. 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

Student ID	Student Name	Score
11111	Bruce Lee	84
22222	Jackie Chen	93
33333	Jet Li	88



# 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(function(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);
    res.end();
  }
});

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
```

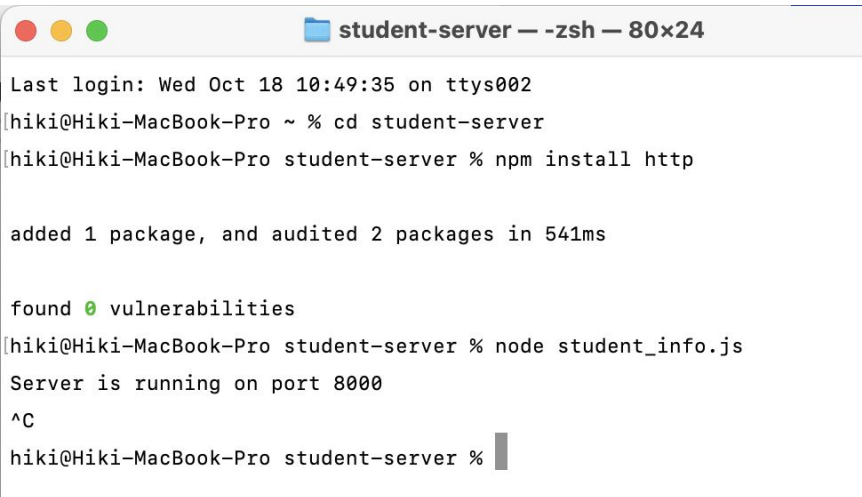
3. Copy the student\_info.js to the directory

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 2Zm... RaT9yqcPQC6rNq
Authtoken saved to configuration file: /Users/hiki/Library/Application Support/ngrok/ngrok.yml

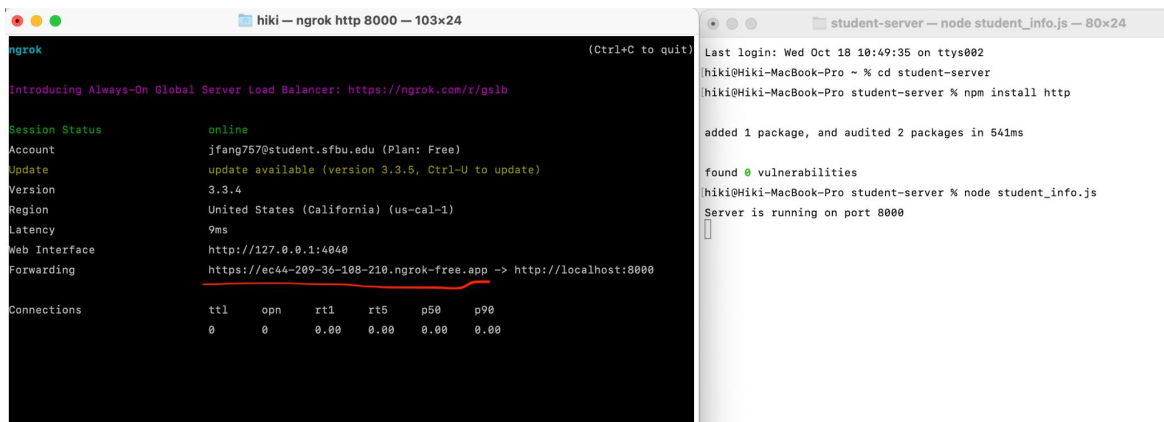
hiki@Hiki-MacBook-Pro ~ % ngrok http 80
hiki@Hiki-MacBook-Pro ~ % ngrok http 8000
```



# 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'. It also shows a forwarding URL: `https://ec44-209-36-108-210.ngrok-free.app`, which is underlined in red. The right window, titled 'student-server — node student\_info.js — 80x24', shows the output of running a Node.js script. It displays the last login time, the current directory, the command to install http, the number of packages added and audited, the number of vulnerabilities found, and 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