

Web RTC

화상채팅



STEP
01

개발 환경



STEP
02

패키지 설치



STEP
03

WEB RTC



STEP
04

시연



socket.io

Web  RTC

Web  RTC



pug



socket.io



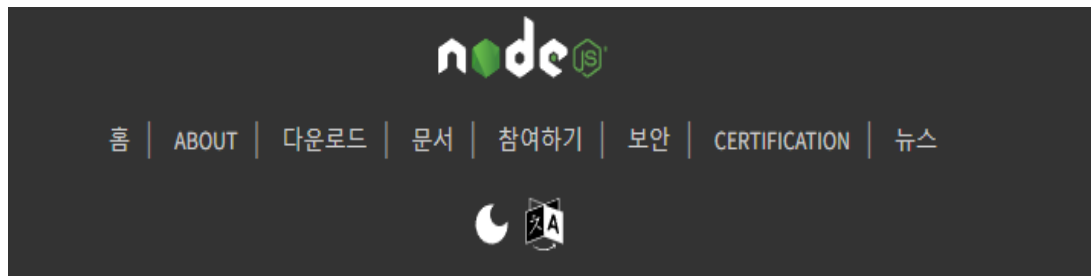
Visual Studio Code

CSS



JS





Node.js®는 Chrome V8 JavaScript 엔진으로 빌드된 JavaScript 런타임입니다.

다운로드 - Windows (x64)

18.16.0 LTS

안정적, 신뢰도 높음

[다른 운영 체제](#) | [변경사항](#) | [API 문서](#)

20.0.0 현재 버전

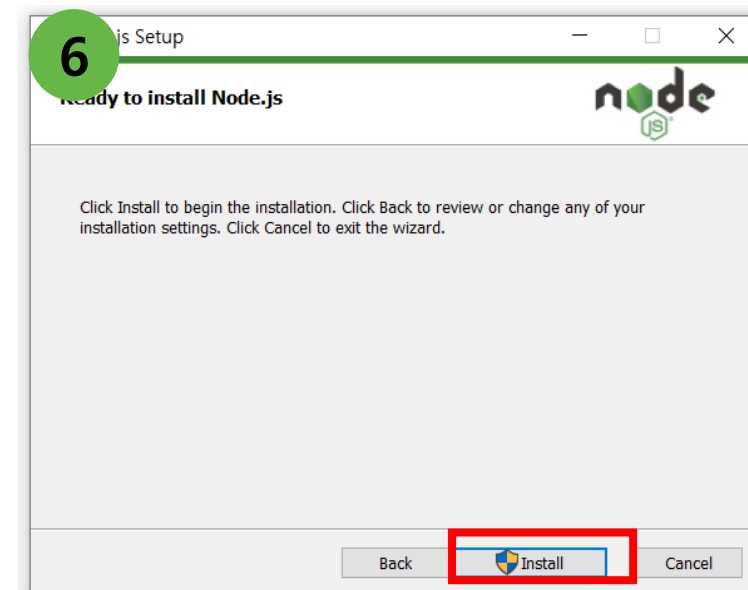
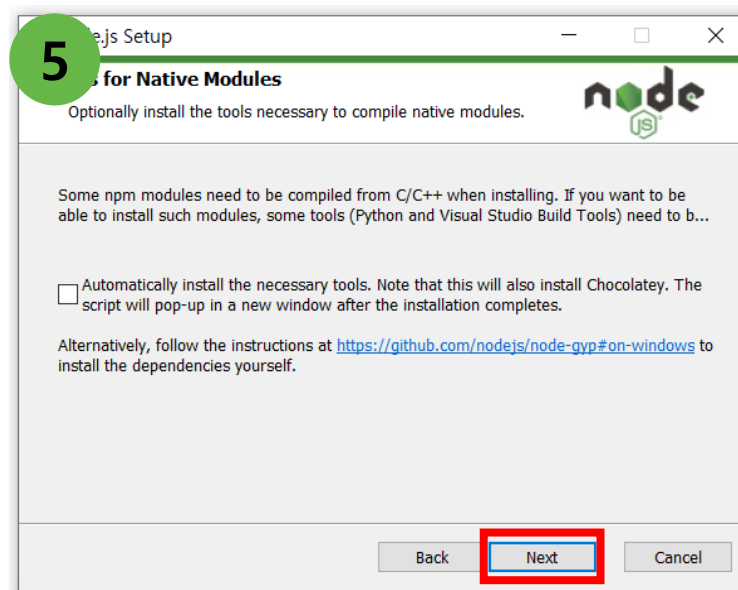
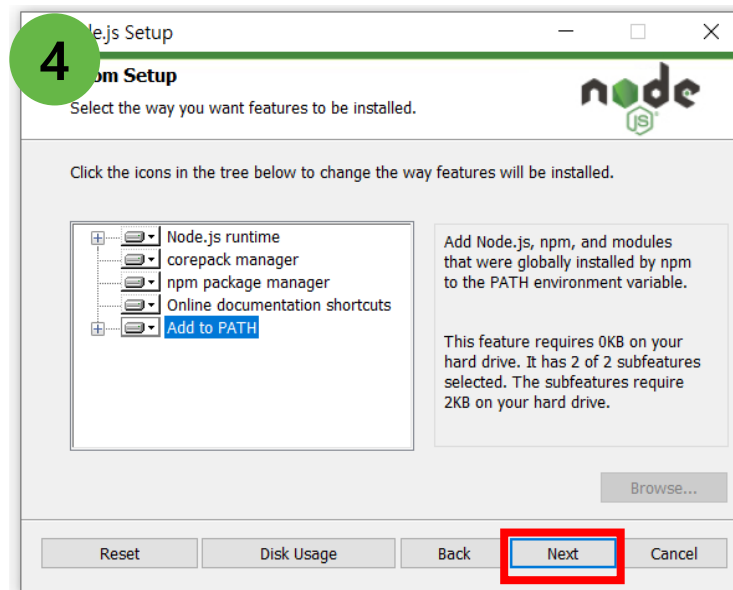
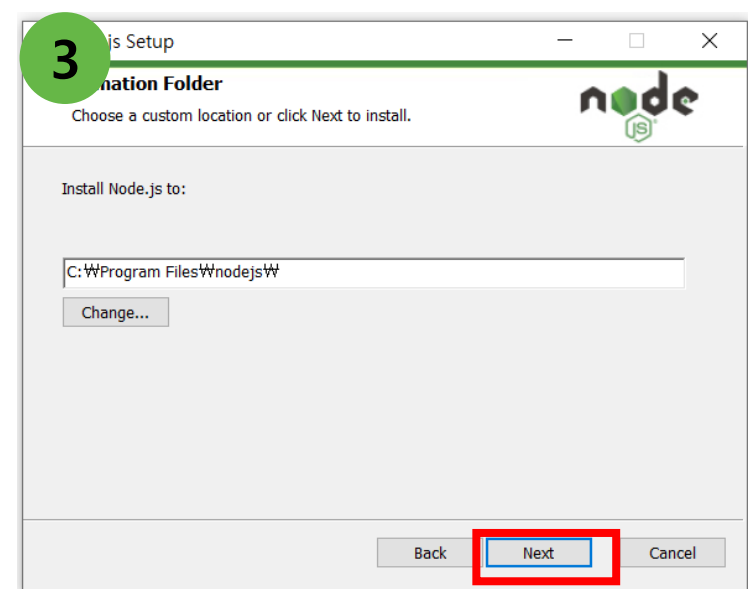
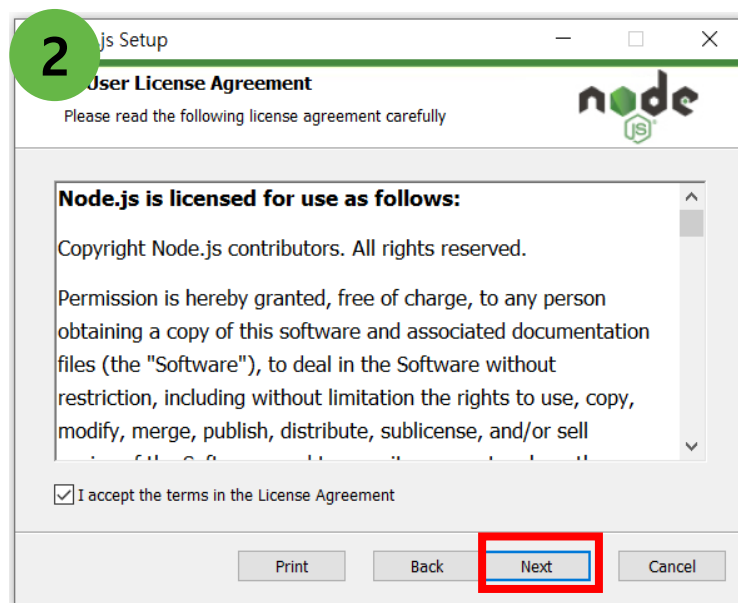
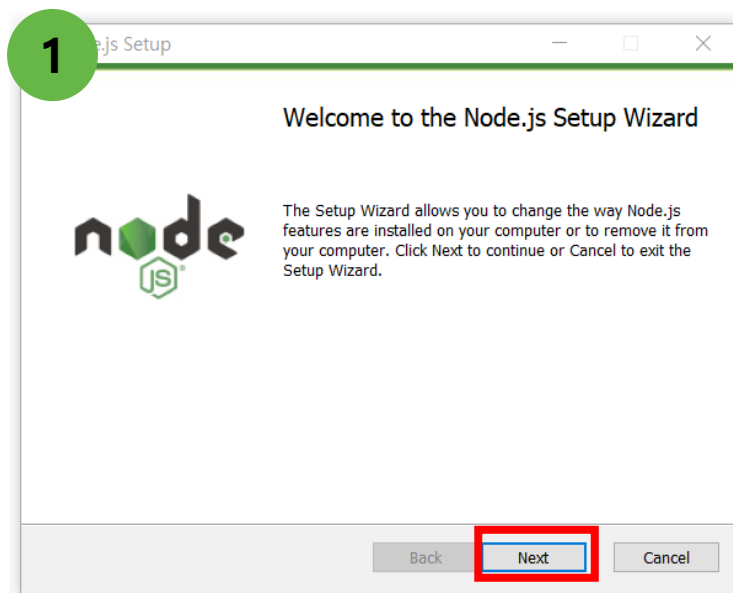
최신 기능

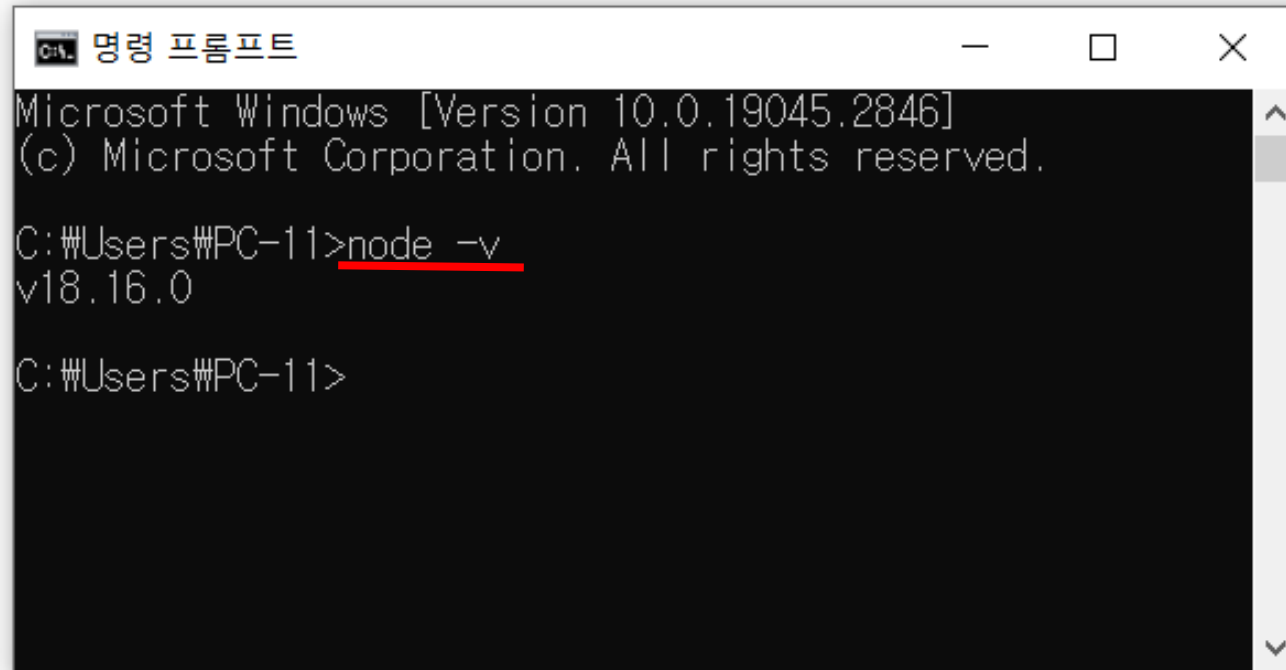
[다른 운영 체제](#) | [변경사항](#) | [API 문서](#)

LTS 일정은 [여기서](#) 확인하세요.

<https://nodejs.org/ko>

공식 홈페이지에서 다운로드 진행



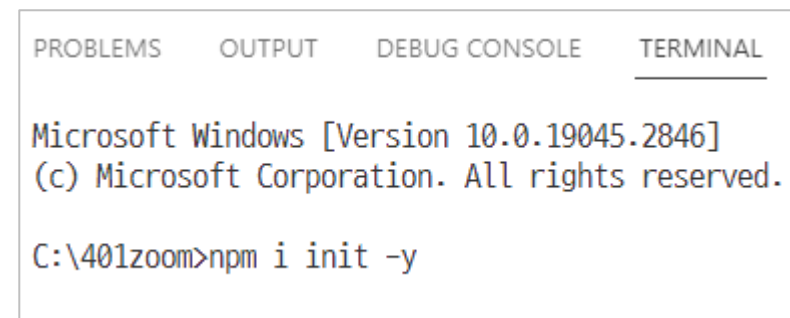
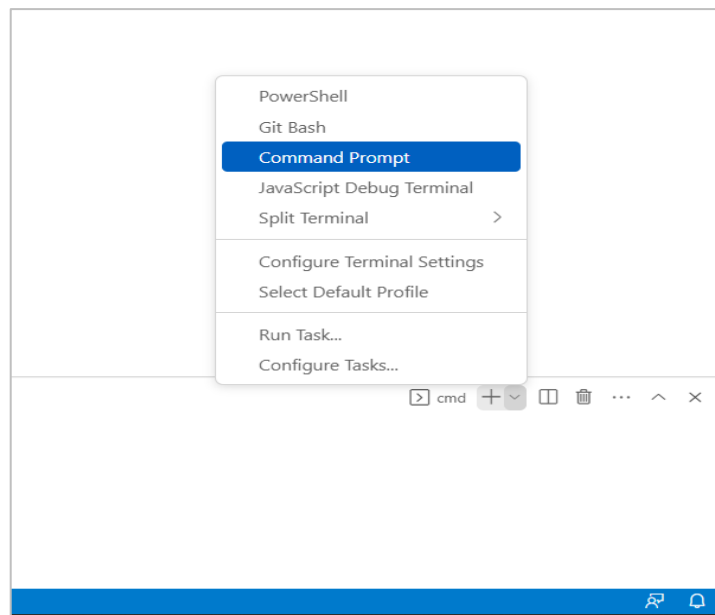
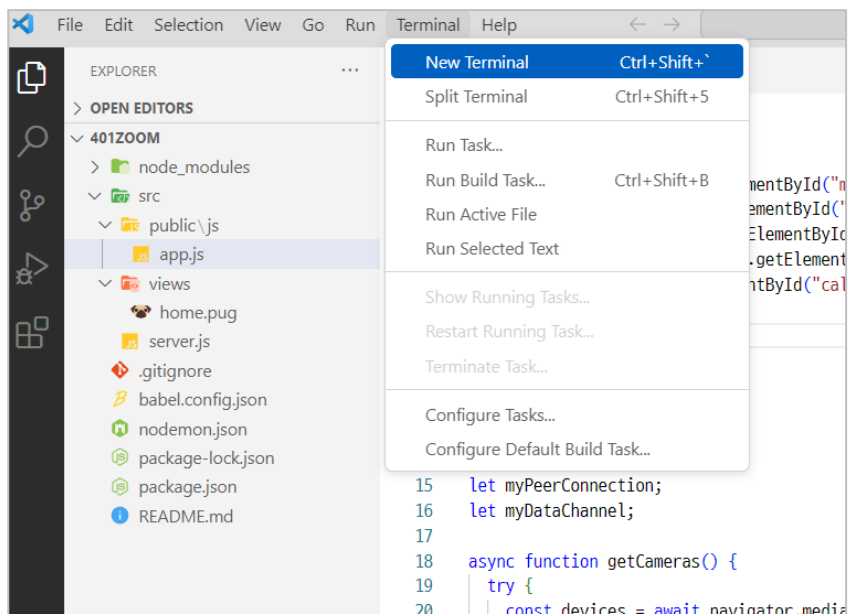


```
C:\> 명령 프롬프트
Microsoft Windows [Version 10.0.19045.2846]
(c) Microsoft Corporation. All rights reserved.

C:\> node -v
v18.16.0

C:\>
```

Node.js설치cmd 창에 node -v를 입력하면 버전 확인 가능



VS코드 상단에서 터미널 생성

하단에서command Prompt로 변경

command 입력

1 npm init -y

Node.js 프로젝트 생성 및 시작

2 npm i nodemon -D

코드 변경 시 서버 자동 재시작

3 npm i @babel/core @babel/cli @babel/node -D

버전의 차이 때문에 발생하는 스크립트 문법의

4 npm i @babel/preset-env -D

차이를 자동으로 맞춰서 실행되게 함

5 npm i express

웹 애플리케이션 프레임워크

6

npm i pug

Pug (HTML과 같으며 문법이 보다 깔끔함)

7

npm i ws

Websocket

8

npm i socket.io

Websocket 기반의 실시간 통신 라이브러리

9

npm i @socket.io/admin-ui

socket.io의 admin 대쉬보드 생성 시 사용(생략가능)

Node.js 설정 변경

package.json ×

package.json > ...

```
1  {
2    "name": "401zoom",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \\\"Error: no test specified\\\" && exit 1"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC",
12   "devDependencies": {
13     "@babel/cli": "^7.21.5",
14     "@babel/core": "^7.21.5",
15     "@babel/node": "^7.20.7",
16     "@babel/preset-env": "^7.21.5",
17     "nodemon": "^2.0.22"
18   },
19   "dependencies": {
20     "@socket.io/admin-ui": "^0.5.1"
```

package.json ●

package.json > ...

```
1  {
2    "name": "401zoom",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "dev": "nodemon"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC",
12   "devDependencies": {
13     "@babel/cli": "^7.21.5",
14     "@babel/core": "^7.21.5",
15     "@babel/node": "^7.20.7",
16     "@babel/preset-env": "^7.21.5",
17     "nodemon": "^2.0.22"
18   },
19   "dependencies": {
```

package.json ×

package.json > ...

```
1  {
2    "name": "401zoom",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "dev": "nodemon"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "MIT",
12   "devDependencies": {
13     "@babel/cli": "^7.21.5",
14     "@babel/core": "^7.21.5",
15     "@babel/node": "^7.20.7",
16     "@babel/preset-env": "^7.21.5",
17     "nodemon": "^2.0.22"
18   },
```

Node.js 패키지 설치가 끝난 후 package.json에서 내용 변경
"dev": "nodemon"

"main": "index.js"
내용 삭제

OpenSSL

Cryptography and SSL/TLS Toolkit

```
C:\Users\PC-13\Desktop\webRTC\webRTC_401>openssl req -nodes -new -x509 -keyout server.key -out server.cert
Generating a RSA private key
.+++++
.....+++++
writing new private key to 'server.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU] KR
State or Province Name (full name) [Some-State] Seoul
Locality Name (eg, city) [] Seoul
Organization Name (eg, company) [Internet Widgits Pty Ltd] local
Organizational Unit Name (eg, section) [] local
Common Name (e.g. server FQDN or YOUR name) [] local
Email Address [] test@test.com

C:\Users\PC-13\Desktop\webRTC\webRTC_401>
```

webRTC_401 폴더에서 cmd창 열고 명령어 실행 후 'server.cert' 'server.key' 생성

```
openssl req -nodes -new -x509 -keyout server.key -out server.cert
```

```
C:\Users\PC-13\Desktop\webRTC\webRTC_401>openssl req -new -key server.key -out server.csr
```

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:KR

State or Province Name (full name) [Some-State]:Seoul

Locality Name (eg, city) []:Seoul

Organization Name (eg, company) [Internet Widgits Pty Ltd]:local

Organizational Unit Name (eg, section) []:local

Common Name (e.g. server FQDN or YOUR name) []:local

Email Address []:test@test.com

Please enter the following 'extra' attributes to be sent with your certificate request

A challenge password []:java

An optional company name []:java

```
openssl req -new -key server.key -out server.csr
```

접근한 폴더에 'Server.csr' 생성

```
C:\Users\PC-13\Desktop\webRTC\webRTC_401>copy server.key server.key.origin
1개 파일이 복사되었습니다.

C:\Users\PC-13\Desktop\webRTC\webRTC_401>openssl rsa -in server.key.origin -out server.key
writing RSA key

C:\Users\PC-13\Desktop\webRTC\webRTC_401>openssl x509 -req -days 3650 -in server.csr -signkey server.key -out server.crt
Signature ok
subject=C = KR, ST = Seoul, L = Seoul, O = local, OU = local, CN = local, emailAddress = test@test.com
Getting Private key

C:\Users\PC-13\Desktop\webRTC\webRTC_401>_
```

copy server.key server.key.origin 입력 하여 기존의 개인키를 백업

openssl rsa -in server.key.origin -out server.key
입력하여 패스워드 제거 (테스트 편의성 제거 안하면 계속 입력하라고 함)

openssl x509 -req -days 3650 -in server.csr -signkey server.key -out server.crt
입력 시 server.crt 생성 끝

```
(base) C:\401zoom>npm run dev
```

```
> 401zoom@1.0.0 dev
```

```
> nodemon
```

```
[nodemon] 2.0.22
```

```
[nodemon] to restart at any time, enter `rs`
```

```
[nodemon] watching path(s): *.*
```

```
[nodemon] watching extensions: js,mjs,json
```

```
[nodemon] starting `babel-node src/server.js`
```

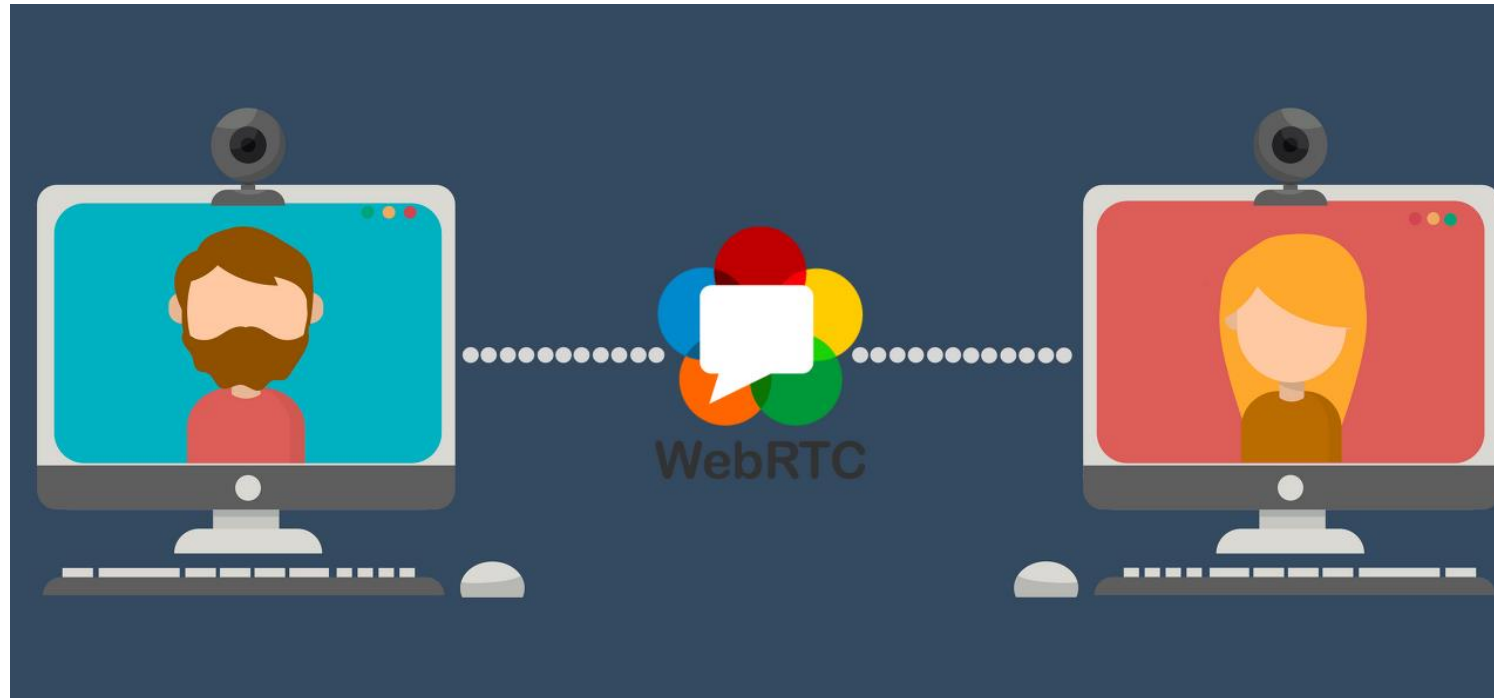
```
Listening on http://localhost:3000
```

```
█
```

VS코드터미널 창에 npm run dev에 입력 시 서버실행가능

WebRTC

Web Real-Time Communications



“별도의 소프트웨어 없이 음성, 영상 미디어, 텍스트, 파일 등을 브라우저끼리 주고 받을 수 있게 만든 기술”



Offer:

미디어 세션 정보와 관련된 초대장

Answer:

미디어 세션 정보와 관련된 응답 메시지(답장)

ICE(Interactive Connectivity Establishment):

중간 네트워크 장비들을 포함한 다양한 네트워크 환경에서도 두 웹 브라우저 간의 연결 설정을 해주는 기술(약도)

이 3가지의 기술들을 이용해 웹 브라우저 간의 연결을 **시그널링(Signaling)**이라고 함
시그널링이 성공하면 서버를 거치지 않고 웹 브라우저 끼리 통신

WEB-RTC 이해를 돕는 블로그

- <https://surprisecomputer.tistory.com/7?category=909008>
- <https://acstory.tistory.com/534>
- <https://jinn-blog.tistory.com/112>
- <https://millo-l.github.io/WebRTC-%EA%B5%AC%ED%98%84-%EB%B0%A9%EC%8B%9D-Mesh-SFU-MCU/>

시연