OpenYearRound

node.js

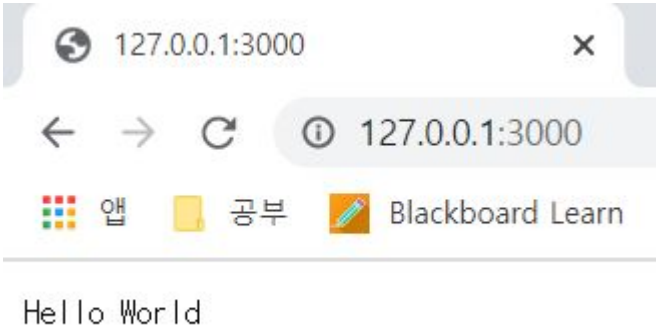
문지원 박정은 이현동

Node.js

- Chrome V8엔진

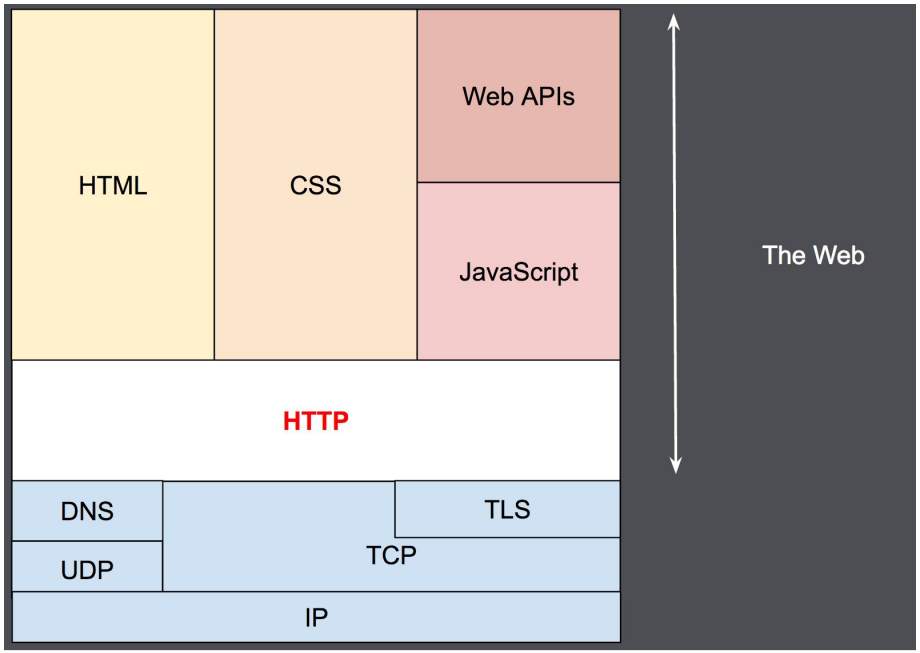
  ➔ 여러 OS환경에서 실행

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

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```



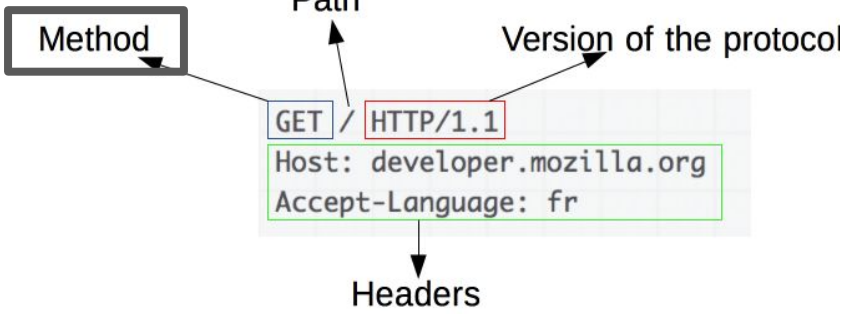
HTTP

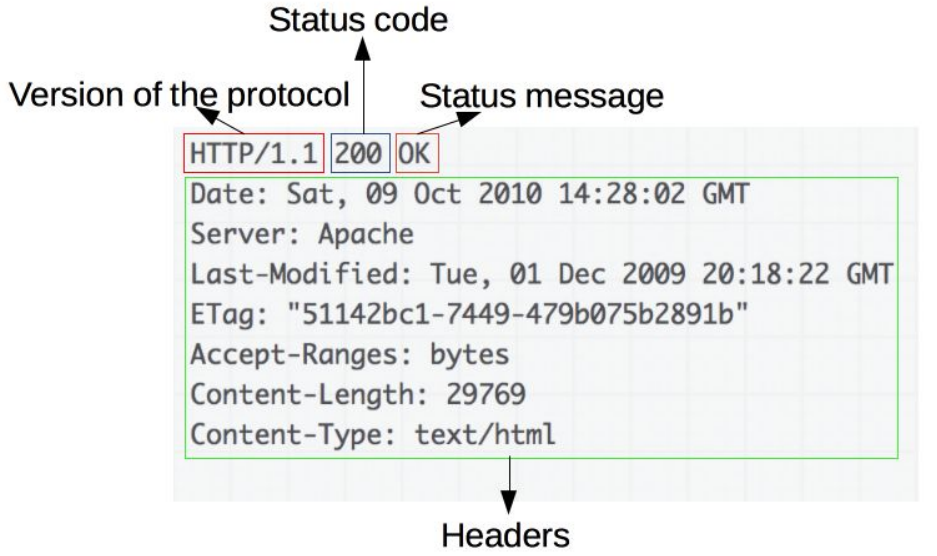
- HTTP는 HTML 문서와 같은 리소스들을 가져올 수 있도록 해주는 **프로토콜**
- **프로토콜**은 컴퓨터 사이에서 데이터의 교환 방식을 정의하는 규칙 체계



HTTP

- HTTP 메시지(Request, Response)





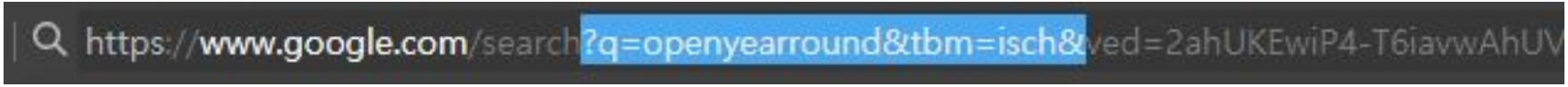
GET

get

겟

을 얻다

GET



- 어떠한 정보를 가져와서 조회하기 위해서 사용되는 방식
- URL에 key&value 쌍의 형태로 서버에 정보를 전송

특징

- URL에 변수(데이터)를 포함시켜 요청한다.
- 데이터를 Header(헤더)에 포함하여 전송한다.
- URL에 데이터가 노출되어 보안에 취약하다.
- 캐싱할 수 있다.

POST

- 데이터 변경
- GET보다 많은 양의 데이터 전송 가능
- 데이터를 body에 담아서 전달

  ➔ GET보다 보안성이 좋음
- body-parser 필요

  ➔ npm install body-parser

views/index.ejs

```
POST
<form action="/post" method="POST">
  <p>
    ID : <input type=”text” name="id">
  </p>
  <p>
    password : <input type=”password” name="pwd">
  </p>
  <p>
    <input type="submit">
  </p>
</form>
```

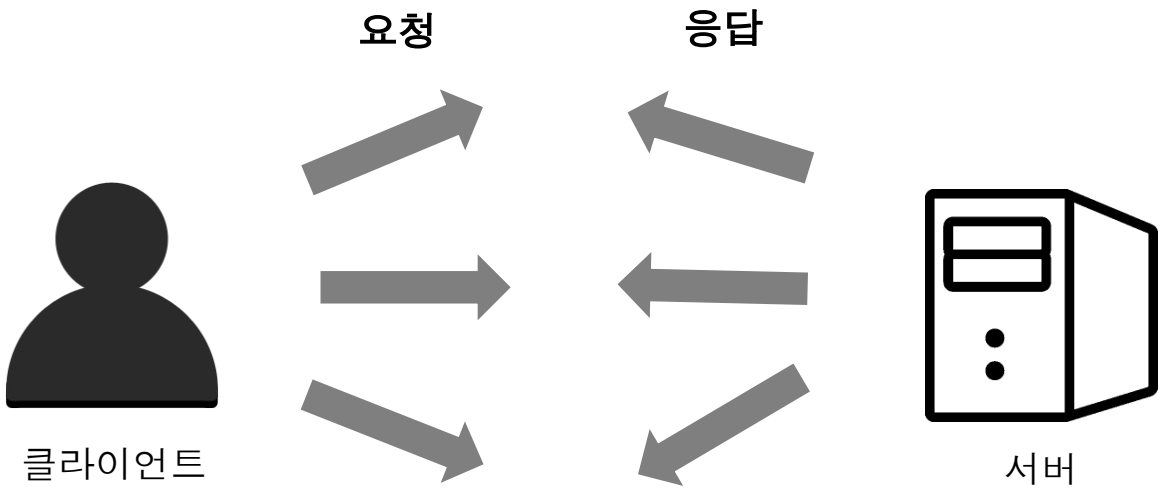
app.js

```
const express = require('express');
const app = express();
const bodyParser = require('body-parser');

app.use(bodyParser.urlencoded({extended: true}));
app.set('view engine', 'ejs');
app.set('views', './views');

app.get('/', (req, res)=>{
  res.render('index');
});
app.post('/post', (req, res)=>{
  res.send("POST");
});
```

GET / POST

처리 방식	GET 방식	POST 방식
URL에 데이터 노출 여부	O	X
URL 예시	http://localhost:8080/boardList?name=제목&contents=내용	http://localhost:8080/addBoard
데이터의 위치	Header(헤더)	Body(바디)
캐싱 가능 여부	O	X

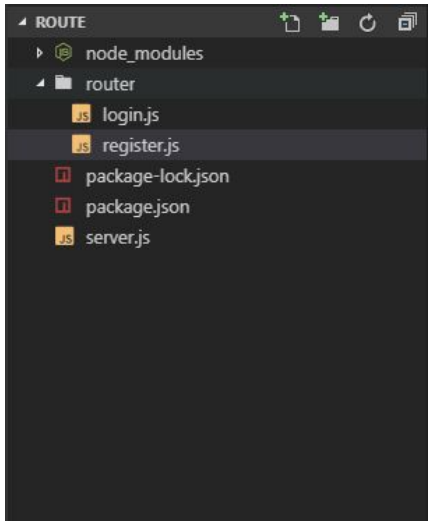
라우팅이란 ?



express

```
var express = require('express');
var app = express();

app.get('/', (req, res) => {
    res.send("익스프레스 테스트");
});

app.listen(5000, (req, res) => {
    console.log("서버 실행중..");
});
```

express를 활용한 라우팅



server.js

router
- login.js
- register.js

/router/login.js

```
const express = require('express');
const router = express.Router();

router.get('/', (req, res) => {
    res.send("로그인");
});

router.get('/logout', (req, res) => {
    res.send("로그아웃");
});

module.exports = router;
```

/router/register.js

```
const express = require('express');
const router = express.Router();

router.get('/', (req, res) => {
    res.send("회원가입");
});

router.get('/done', (req, res) => {
    res.send("회원가입 완료");
});

module.exports = router;
```

server.js

```
var express = require('express');
const app = express();

const login = require('./router/login');
const register = require('./router/register');

app.use('/api/login', login);
app.use('/api/register', register);

app.listen(5000, (req, res) => {
    console.log("서버 실행중..");
});
```

ex)
http://localhost:5000/api/login