

# **인터넷 프로그래밍**

**Final Project**

**20191164 최윤지**

# Crawling Site

<https://www.nypl.org/locations>

New York  
Public  
Library

LOG IN ▾ LOCATIONS GET A LIBRARY CARD GET EMAIL UPDATES ▾ DONATE SHOP

Books/Music/Movies Research Education Events Connect Give Get Help Search

Home / Locations

Welcome to The New York Public Library. Discover our locations across the Bronx, Manhattan, and Staten Island. <https://www.nypl.org/locations>

Find Your Library

Enter an address or landmark to search nearby or type in a Library name.

Open now

Filters

Borough + Accessibility + Amenities + Subject Specialties + Media Types +

[125th Street Library](#)

224 East 125th Street  
New York, NY 10035  
212-534-5050

Location is temporarily closed

[View on Map](#) | [Get Directions](#)

[Ford Street Library](#)

- 뉴욕 공공 도서관 정보 Crawling
- 제안서에서는 한국 공영 주차장 정보였으나 영어로 된 data를 읽어오기 위해 사이트를 변경
- 이름 /주소 / 이용시간 / 전화번호의 정보가 동일한 구조

# Crawling Site

<https://www.nypl.org/locations>

## 125th Street Library

224 East 125th Street  
New York, NY 10035  
212-534-5050

*Location is temporarily closed*

[View on Map](#) | [Get Directions](#)

## 53rd Street Library

18 West 53rd Street  
New York, NY 10019  
212-714-8400

 Fully Accessible

 Today's Hours: **11AM–6PM**

[View on Map](#) | [Get Directions](#)

## 58th Street Library



- 177개 정도의 도서관 데이터를 Crawling
  - 이름
  - 주소
  - 전화번호
  - 현재 이용 가능한지
  - 오픈 시간

# Crawling한 Data

```
[nodemon] starting `node app.js`
Example app listening on port 80!
데이터베이스에 연결되었습니다. : mongodb://localhost:27017/DB
125th Street LibraryGet Directions
53rd Street LibraryGet Directions
58th Street LibraryGet Directions
67th Street LibraryGet Directions
96th Street LibraryGet Directions
Aguilar LibraryGet Directions
Allerton LibraryGet Directions
Andrew Heiskell Braille and Talking Book LibraryGet Directions
Art & Architecture CollectionGet Directions
Art and Artifacts DivisionGet Directions
Battery Park City LibraryGet Directions
Baychester LibraryGet Directions
Belmont Library and Enrico Fermi Cultural CenterGet Directions
Billy Rose Theatre DivisionGet Directions
Bloomingdale LibraryGet Directions
Bronx Library CenterGet Directions
[nodemon] restarting due to changes...
Carl H. Pforzheimer Collection of Shelley and His CircleGet Directions
Castle Hill LibraryGet Directions
Center for Research in the HumanitiesGet Directions
Charleston LibraryGet Directions
Chatham Square LibraryGet Directions
Children's CenterGet Directions
City Island LibraryGet Directions
Clason's Point LibraryGet Directions
Columbus Library Get Directions
Countee Cullen LibraryGet Directions
DeWitt Wallace Periodical RoomGet Directions
Dongan Hills LibraryGet Directions
Dorot Jewish DivisionGet Directions
Eastchester LibraryGet Directions
Edenwald LibraryGet Directions
Epiphany LibraryGet Directions
Event Center & Rooftop TerraceGet Directions
Fort Washington LibraryGet Directions
```

```
[nodemon] starting `node app.js`
Example app listening on port 80!
데이터베이스에 연결되었습니다. : mongodb://localhost:27017/DB
224 East 125th Street
New York, NY 10035
18 West 53rd Street
New York, NY 10019
127 East 58th Street
New York, NY 10022
328 East 67th Street
New York, NY 10065
112 East 96th Street
New York, NY 10128
174 East 110th Street
New York, NY 10029
2740 Barnes Avenue
Bronx, NY 10467
40 West 20th Street
New York, NY 10011
Fifth Avenue and 42nd Street
New York, NY 10018
515 Malcolm X Boulevard (135th St and Malcolm X Blvd)
New York, NY 10037
175 North End Avenue
New York, NY 10282
2049 Asch Loop North
Bronx, NY 10475
610 East 186th Street
Bronx, NY 10458
40 Lincoln Center Plaza (entrance at 111 Amsterdam between 64th and 65th)
New York, NY 10023
150 West 100th Street
New York, NY 10025
310 East Kingsbridge Road
Bronx, NY 10458
Fifth Avenue and 42nd Street
```

```
[nodemon] starting `node app.js`
Example app listening on port 80!
데이터베이스에 연결되었습니다. : mongodb://localhost:27017/DB
212-534-5050
212-714-8400
212-759-7358
212-734-1717
212-289-0908
212-534-2930
718-881-4240
212-206-5400
212-930-0835
212-491-2241
212-790-3499
718-379-6700
718-933-6410
212-870-1639
212-222-8030
718-579-4244
212-930-0717
718-824-3838
```

```
데이터베이스에 연결되었습니다. : mongodb://localhost:27017/DB
Location is temporarily closed
```

```
Location is temporarily closed
```

```
Location is temporarily closed
```

```
[nodemon] starting `node app.js`
```

```
Example app listening on port 80!
데이터베이스에 연결되었습니다. : mongodb://localhost:27017/DB
```

```
11AM-6PM
10AM-6PM
11AM-7PM
10AM-7PM
Closed
10AM-6PM
12PM-7PM
10AM-7:45PM
Closed
Closed
```

```
10AM-6PM
10:30AM-6PM
10AM-7PM
9AM-9PM
11AM-5PM*
10AM-6PM
10AM-5:45PM
10AM-7PM
```

# Crawling한 Data를 JSON 형식으로 변형

```
var Crawler = ...require("crawler");
var jsonData

var c = new Crawler({
  maxConnections : 10,
  // This will be called for each crawled page
  callback : function (error, res, done) {
    if(error){
      console.log(error);
    }else{
      var $ = res.$;
      // $ is Cheerio by default
      // a lean implementation of core jQuery designed specifically for the server
      //console.log($(".title").text());

      const $bodyList = $("div#locations-list").find('li');

      var jsonArray = new Array();
      var i = 0
      $bodyList.each(function(i, elem) {
        let jsonObj = new Object();
        jsonObj.id = (i+1);
        jsonObj.title = $(this).find('div.css-1krdr49').find('a').text();
        //console.log(jsonObj.title);
        jsonObj.location = $(this).find('div.address.css-1se1pun').text();
        //console.log(jsonObj.location);
        jsonObj.number = $(this).find('div.phone.css-gbfm2m').text();
        //console.log(jsonObj.number);
        jsonObj.nonaval = $(this).find('div.css-ekwn3s').text();
        //console.log(jsonObj.nonaval);
        jsonObj.status = $(this).find('div.chakra-stack.css-ja1irr').find('div.css-1xsa88d').text();
        //console.log(jsonObj.status);

        jsonArray.push(jsonObj);
      });
    }
  }
});
```

npm Crawler package 를 이용

```
jsonData = JSON.stringify(jsonArray) ;
```

```
[nodemon] starting `node app.js`
Example app listening on port 80!
데이터베이스에 연결되었습니다. : mongodb://localhost:27017/DB
[{"id":1,"title":"125th Street LibraryGet Directions","location":"224 East 125th Street\nNew York, NY 10035","number":"212-534-5050","nonaval":"Location is temporarily closed","status":""}, {"id":2,"title":"53rd Street LibraryGet Directions","location":"18 West 53rd Street\nNew York, NY 10019","number":"212-714-8400","nonaval":"","status":"11AM-6PM"}, {"id":3,"title":"58th Street LibraryGet Directions","location":"127 East 58th Street\nNew York, NY 10022","number":"212-759-7358","nonaval":"","status":"10AM-6PM"}, {"id":4,"title":"67th Street LibraryGet Directions","location":"328 East 67th Street\nNew York, NY 10065","number":"212-734-1717","nonaval":"","status":"11AM-7PM"}, {"id":5,"title":"96th Street LibraryGet Directions","location":"112 East 96th Street\nNew York, NY 10128","number":"212-289-0908","nonaval":"","status":"10AM-7PM"}, {"id":6,"title":"Aguilar LibraryGet Directions","location":"174 East 110th Street\nNew York, NY 10029","number":"212-534-2930","nonaval":"","status":"Closed"}, {"id":7,"title":"Allerton LibraryGet Directions","location":"2740 Barnes Avenue\nBronx, NY 10467","number":"718-881-4240","nonaval":"","status":"10AM-6PM"}, {"id":8,"title":"Andrew Heiskell Braille and Talking Book LibraryGet Directions","location":"40 West 20th Street\nNew York, NY 10011","number":"212-206-5400","nonaval":"","status":"12PM-7PM"}, {"id":9,"title":"Art & Architecture CollectionGet Directions","location":"Fifth Avenue and 42nd Street\nNew York, NY 10018","number":"212-930-0835","nonaval":"","status":"10AM-7:45PM"}, {"id":10,"title":"Art and Artifacts DivisionGet Directions","location":"515 Malcolm X Boulevard (135th St and Malcolm X Blvd)\nNew York, NY 10037","number":"212-491-2241","nonaval":"","status":"Closed"}, {"id":11,"title":"Battery Park City LibraryGet Directions","location":"175 North End Avenue\nNew York, NY 10282","number":"212-790-3499","nonaval":"","status":"Closed"}, {"id":12,"title":"Baychester LibraryGet Directions","location":"2049 Asc h Loop North\nBronx, NY 10475","number":"718-379-6700","nonaval":"Location is temporarily closed","status":""}, {"id":13,"title":"Belmont Library and Enrico Fermi Cultural CenterGet Directions","location":"610 East 186th Street\nBronx, NY 10458","number":"718-933-6410","nonaval":"","status":"10AM-6PM"}, {"id":14,"title":"Billy Rose Theatre DivisionGet Directio ns","location":"40 Lincoln Center Plaza (entrance at 111 Amsterdam between 64th and 65th)\nNew York, NY 10023","number":"212-870-1639","nonaval":"","status":"10:30AM-6PM"}, {"id":15,"title":"Bloomingdale LibraryGet Directions","location":"150 West 100th Street\nNew York, NY 10025","number":"212-222-8030","nonaval":"","status":"10AM-7PM"}, {"id":16,"title":"Bro nx Library CenterGet Directions","location":"310 East Kingsbridge Road\nBronx, NY 10458","number":"718-579-4244","nonaval":"","status":"9AM-9PM"}, {"id":17,"title":"Carl H. Pforzheimer Collection of Shelley and His CircleGet Directions","location":"Fifth Avenue and 42nd
```

크롤링한 데이터를 JSON 형식으로 추출

# Crawling한 Data를 JSON 형식으로 변형

The screenshot shows the MongoDB Atlas interface. On the left, there's a sidebar with sections for DEPLOYMENT, DATABASES (Data Lake selected), SERVICES (Triggers, Data API, Data Federation), and SECURITY (Quickstart, Database Access, Network Access, Advanced). The main area shows a database named 'test' containing a collection named 'libraries'. The 'Collections' tab is selected. A red box highlights the 'TOTAL DOCUMENTS: 133' text. Below it, two documents are shown in a table, each with a red box highlighting its content:

Document ID	Content
_id: ObjectId('63ac3f2cf3f0e9784a5c7925')	<pre>_id: ObjectId('63ac3f2cf3f0e9784a5c7925') id: 1 title: "125th Street LibraryGet Directions" location: "224 East 125th Street New York, NY 10035" number: "212-534-5050" nonaval: "Location is temporarily closed" status: ""</pre>
_id: ObjectId('63ac3f2cf3f0e9784a5c7926')	<pre>_id: ObjectId('63ac3f2cf3f0e9784a5c7926') id: 2 title: "53rd Street LibraryGet Directions" location: "18 West 53rd Street New York, NY 10019" number: "212-714-8400" nonaval: "" status: "11AM-6PM"</pre>

A large red box on the right contains the Korean text: 사용자가 추가한 데이터 포함 총 133개 데이터 (crawling 한 데이터 127개)

Below the table, there are navigation buttons: PREVIOUS, NEXT, and a message icon.

# Backend

## Node js & Express

```
backend
  controllers
    JS librarys-controll...
  model
    JS Library.js
  node_modules
  routes
    JS library-routes.js
    JS app.js
    JS crawler.js
  {} package-lock.json
  {} package.json
```

기능 별 control 함수를 보기 편하게 따로 정의

MongoDB의 스키마를 정의하기 위한 model

Express를 이용한 Router 정의

```
{
  "name": "lastt",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  ▷ Debug
  "scripts": {
    "start": "nodemon app.js",
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "cors": "^2.8.5",
    "express": "^4.18.2",
    "mongoose": "^6.8.1",
    "nodemon": "^2.0.20",
    "start": "^5.1.0"
  }
}
```

Express, Cors, Mongoose package 사용

# Backend

## Node js & Express

```
const express = require("express");
const mongoose = require("mongoose");
const router = require("./routes/library-routes");
const cors = require("cors");
const app = express();

// Middlewares
// app.use('/', (req, res, next) => {
//   res.send("This is our starting app")
// })
app.use(express.json());
app.use(cors());
app.use("/librarys", router);

mongoose
  .connect(
    "mongodb+srv://admin:pwK0FUUrUExRufn8@cluster0.lm50x5v.mongodb.net/?retryWrites=true&w=majority"
  )
  .then(() => console.log("Connected To Database"))
  .then(() => {
    app.listen(8080);
  })
  .catch((err) => console.log(err));
```

app.js

```
const express = require("express");
const router = express.Router();
const Library = require("../model/Library");
const librarysController = require("../controllers/librarys-controller");

// router.get("/", async (req, res, next) => {
//   let lib;
//   try {
//     lib = await Library.find();
//   } catch (err) {
//     console.log(err);
//   }
//
//   if (!lib) {
//     return res.status(404).json({ message: "No library found" });
//   }
//   return res.status(202).json({ lib });
// });

router.get("/", librarysController.getAllLibrarys);
router.post("/", librarysController.addLibrary);
router.get("/:id", librarysController.getById);
router.put("/:id", librarysController.updateLibrary);
router.delete("/:id", librarysController.deleteLibrary);

module.exports = router;
```

routes/library-router.js

MongoDB 연결, 라우터 연결

라우터 정의, controller와 연결

# Backend

## Node js & Express

```
const Library = require("../model/Library");

const getAllLibrarys = async (req, res, next) => {
  let librarys;
  try {
    librarys = await Library.find();
  } catch (err) {
    console.log(err);
  }

  if (!librarys) {
    return res.status(404).json({ message: "No products found" });
  }
  return res.status(200).json({ librarys });
};

const getById = async (req, res, next) => {
  const id = req.params.id;
  let library;
  try {
    library = await Library.findById(id);
  } catch (err) {
    console.log(err);
  }
  if (!library) {
    return res.status(404).json({ message: "No Library found" });
  }
  return res.status(200).json({ library });
};
```

librarys-controller.js

기능 정의; 모든 데이터 보기, id로 검색, 데이터 추가 등의 기능

```
const addLibrary = async (req, res, next) => {
  const { id, title, location, number, nonaval, status } = req.body;
  let library;
  try {
    library = new Library({
      id,
      title,
      location,
      number,
      nonaval,
      status,
    });
    await library.save();
  } catch (err) {
    console.log(err);
  }

  if (!library) {
    return res.status(500).json({ message: "Unable To Add" });
  }
  return res.status(201).json({ library });
};

exports.getAllLibrarys = getAllLibrarys;
exports.addLibrary = addLibrary;
exports.getById = getById;
exports.updateLibrary = updateLibrary;
exports.deleteLibrary = deleteLibrary;
```

# Backend

## Postman

프론트엔드 구현 전 Postman 이용하여 테스트

The screenshot shows the Postman interface with a GET request to `http://localhost:8080/library`. The 'Body' tab is selected, showing an empty JSON object. The 'Headers' tab contains 9 items. The 'Tests' tab shows a test script: 

```
if(response.error){\n    console.log(response.error);\n}\nelse{\n    console.log('Success! Response: ' + response.data);\n}
```

. The 'Settings' tab shows a timeout of 10 seconds.

The screenshot shows a POST request to `http://localhost:8080/library`. The 'Body' tab is selected, containing the following JSON payload:

```
1 {\n2   "name": "postmantest",\n3   "location": "tete",\n4   "number": "010-1234-5678",\n5   "isaval": "1231"\n6 }
```

The 'Headers' tab shows 9 items. The 'Tests' tab contains a test script: 

```
if(response.error){\n    console.log(response.error);\n}\nelse{\n    console.log('Success! Response: ' + response.data);\n}
```

. The 'Settings' tab shows a timeout of 10 seconds.

The screenshot shows a GET request to `http://localhost:8080/librarys/no/no`. The 'Body' tab is selected, showing an empty JSON object. The 'Headers' tab contains 8 items. The 'Tests' tab shows a test script: 

```
if(response.error){\n    console.log(response.error);\n}\nelse{\n    console.log('Success! Response: ' + response.data);\n}
```

. The 'Settings' tab shows a timeout of 10 seconds.

# Backend

## Postman

프론트엔드 구현 전 Postman 이용하여 테스트

The screenshot shows the Postman interface with a GET request to `http://localhost:8080/library`. The 'Body' tab is selected, showing an empty JSON object. The 'Headers' tab contains 9 items. The 'Tests' tab shows a test script: 

```
if(response.error){\n    console.log(response.error);\n}\nelse{\n    console.log('Success! Response: ' + response.data);\n}
```

. The 'Settings' tab shows a timeout of 10 seconds.

The screenshot shows a POST request to `http://localhost:8080/library`. The 'Body' tab is selected, containing the following JSON payload:

```
1 {\n2   "name": "postmantest",\n3   "location": "tete",\n4   "number": "010-1234-5678",\n5   "isaval": "1231"\n6 }
```

The 'Headers' tab shows 9 items. The 'Tests' tab contains a test script: 

```
if(response.error){\n    console.log(response.error);\n}\nelse{\n    console.log('Success! Response: ' + response.data);\n}
```

. The 'Settings' tab shows a timeout of 10 seconds.

The screenshot shows a GET request to `http://localhost:8080/librarys/nothing`. The 'Body' tab is selected, showing an empty JSON object. The 'Headers' tab contains 8 items. The 'Tests' tab shows a test script: 

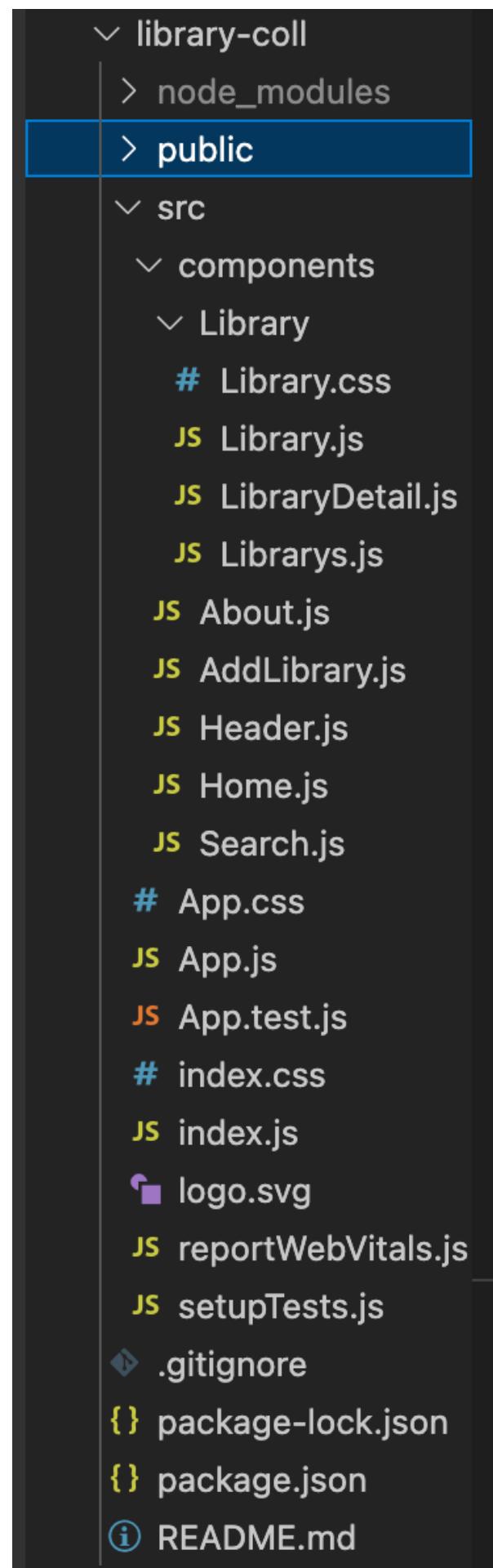
```
if(response.error){\n    console.log(response.error);\n}\nelse{\n    console.log('Success! Response: ' + response.data);\n}
```

. The 'Settings' tab shows a timeout of 10 seconds.

# FrontEnd

## React

npx create-react-app 로 프레임워크 구현



Components로 파트별 구현

```
"private": true,  
"dependencies": {  
  "@emotion/react": "^11.10.5",  
  "@emotion/styled": "^11.10.5",  
  "@mui/icons-material": "^5.11.0",  
  "@mui/material": "^5.11.2",  
  "@testing-library/jest-dom": "^5.16.5",  
  "@testing-library/react": "^13.4.0",  
  "@testing-library/user-event": "^13.5.0",  
  "axios": "^1.2.1",  
  "crawler": "^1.3.0",  
  "react": "^18.2.0",  
  "react-dom": "^18.2.0",  
  "react-router-dom": "^6.6.1",  
  "react-scripts": "5.0.1",  
  "web-vitals": "^2.1.4"  
},
```

## Move faster with intuitive React UI tools

MUI offers a comprehensive suite of UI tools to help you ship new features faster. Start with Material UI, our fully-loaded component library, or bring your own design system to our production-ready components.

Get started >

npm install @mui/material @emot...

Material UI 이용

npm react-router-dom, axios 이용

# FrontEnd

## React

```
import React from "react";
import Header from "./components/Header";
import { Route, Routes } from "react-router-dom";
import Home from "./components/Home";
import AddLibrary from "./components/AddLibrary";
import Librarys from "./components/Library/Librarys";
import About from "./components/About";
import LibraryDetail from "./components/Library/LibraryDetail";
function App() {
  return (
    <React.Fragment>
      <header>
        <Header />
      </header>
      <main>
        <Routes>
          <Route path="/" element={<Home />} exact />
          <Route path="/add" element={<AddLibrary />} exact />
          <Route path="/librarys" element={<Librarys />} exact />
          <Route path="/about" element={<About />} exact />
          <Route path="/librarys/:id" element={<LibraryDetail />} exact />
          <Route path="/search" element={<Search />} exact/>
        </Routes>
      </main>
    </React.Fragment>
  );
}

export default App;
```

app.js

```
import React, { useState } from "react";
import { AppBar, Tab, Tabs, Toolbar, Typography } from "@mui/material";
import LibraryBooksOutlinedIcon from "@mui/icons-material/LibraryBooksOutlined";
import { NavLink } from "react-router-dom";
const Header = () => {
  const [value, setValue] = useState();
  return (
    <div>
      <AppBar sx={{ backgroundColor: "#DC143C" }} position="sticky">
        <Toolbar>
          <NavLink to="/" style={{ color: "white" }}>
            <Typography>
              <LibraryBooksOutlinedIcon />
            </Typography>
          </NavLink>
          <Tabs
            sx={{ ml: "auto" }}
            textColor="inherit"
            indicatorColor="primary"
            value={value}
            onChange={(e, val) => setValue(val)}
          >
            <Tab LinkComponent={NavLink} to="/add" label="Add New" />
            <Tab LinkComponent={NavLink} to="/librarys" label="Librarys" />
            <Tab LinkComponent={NavLink} to="/about" label="About" />
            <Tab LinkComponent={NavLink} to="/search" label="Search" />
          </Tabs>
        </Toolbar>
      </AppBar>
    </div>
  );
}

export default Header;
```

header.js

ADD NEW LIBRARYS ABOUT SEARCH

path 별로 구현

Header 구현

# FrontEnd

## React

```
import { Button, Typography, Box } from "@mui/material";
import React from "react";
import { Link } from "react-router-dom";

const Home = () => {
  return (
    <div>
      <Box display="flex" flexDirection="column" alignItems="center">
        <Button
          LinkComponent={Link}
          to="/librarys"
          sx={{ marginTop: 15, background: "orangered" }}
          variant="contained"
        >
          <Typography variant="h3">View All Librarys</Typography>
        </Button>
      </Box>
    </div>
  );
};

export default Home;
```

Home.js

[VIEW ALL LIBRARYS](#)

[VIEW ALL LIBRARYS](#)

About.js

```
import { Box, Typography } from "@mui/material";
import React from "react";

const About = () => {
  return (
    <div>
      <Box display="flex" flexDirection="column" alignItems="center">
        <Typography sx={{ fontFamily: "monospace" }} variant="h2">
          This is a CRUD Application
        </Typography>
        <Typography sx={{ fontFamily: "monospace" }} variant="h3">
          By MERN STACK
        </Typography>
      </Box>
    </div>
  );
};

export default About;
```

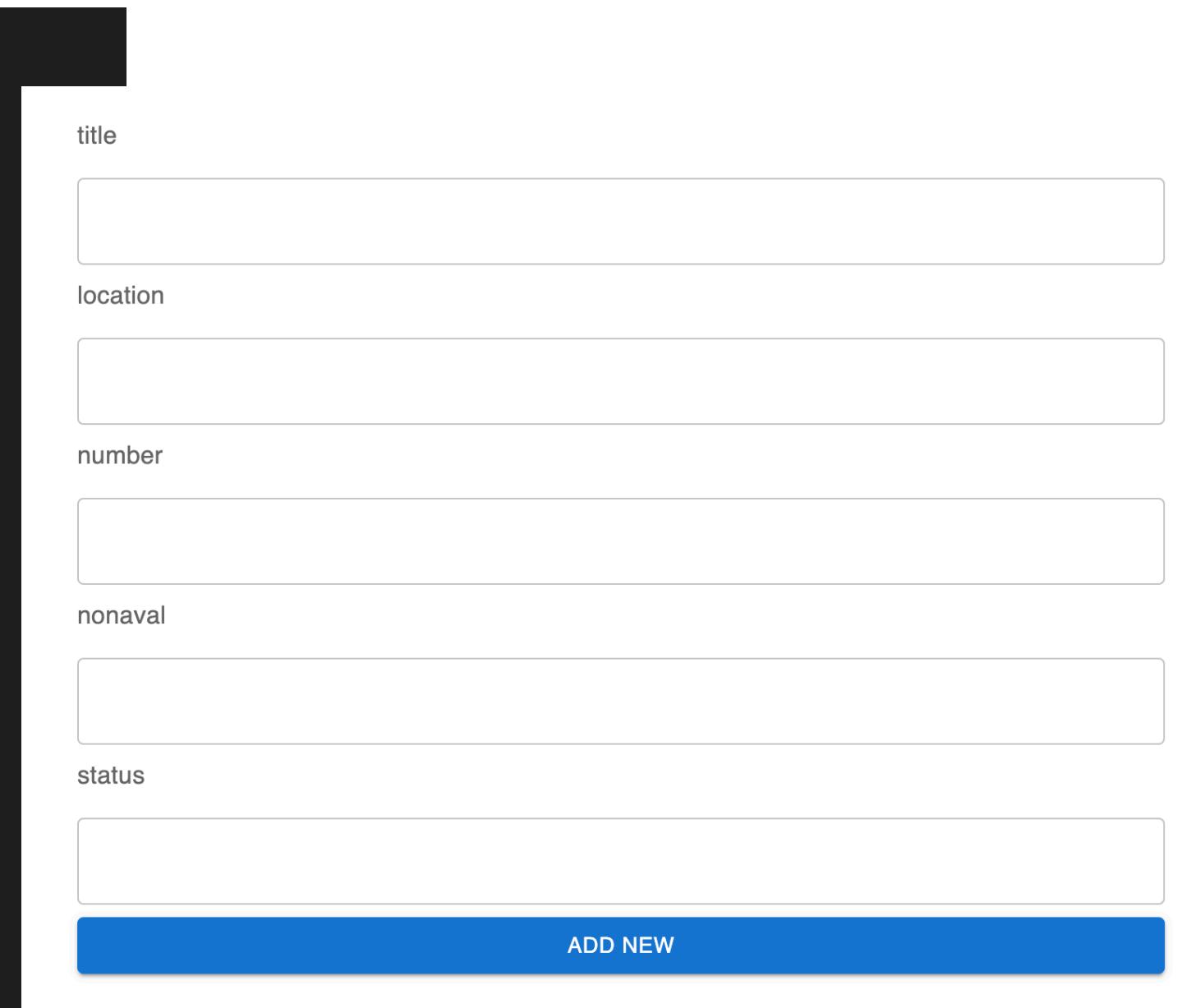
This is a CRUD Application  
By MERN STACK

# FrontEnd

## React

### Home.js

```
import {  
    Button,  
    Checkbox,  
    FormControlLabel,  
    FormLabel,  
    TextField,  
    Typography,  
} from "@mui/material";  
import { Box } from "@mui/system";  
import axios from "axios";  
import React, { useState } from "react";  
import { useNavigate } from "react-router-dom";  
  
//id, title, location, number, nonaval, status  
const AddLibrary = () => {  
    const history = useNavigate();  
    const [inputs, setInputs] = useState({  
        id:"",  
        title: "",  
        location: "",  
        number: "",  
        nonaval: "",  
        status: "",  
    });  
    const [checked, setChecked] = useState(false);  
    const handleChange = (e) => {  
        setInputs((prevState) => {  
            ...prevState,  
            [e.target.name]: e.target.value,  
        });  
        // console.log(e.target.name, "Value", e.target.value);  
    };  
  
    const sendRequest = async () => {  
        await axios  
            .post("http://localhost:8080/librarys", {  
                id:Number(inputs.id),  
                title: String(inputs.title),  
                location: String(inputs.location),  
                number: String(inputs.number),  
                nonaval: String(inputs.nonaval),  
                status: String(inputs.status),  
            })  
            .then((res) => res.data);  
    };  
  
    const handleSubmit = (e) => {  
        e.preventDefault();  
        console.log(inputs, checked);  
        sendRequest().then(() => history("/librarys"));  
    };  
    //id, title, location, number, nonaval, status  
    return (  
        <form onSubmit={handleSubmit}>  
            <Box  
                display="flex"  
                flexDirection="column"  
                justifyContent={"center"}  
                maxWidth={700}  
                alignContent={"center"}  
                alignSelf="center"  
                marginLeft={"auto"}  
                marginRight="auto"  
                marginTop={10}>  
                <FormLabel>title</FormLabel>  
                <TextField  
                    value={inputs.title}  
                    onChange={handleChange}  
                    margin="normal"  
                    fullWidth  
                    variant="outlined"  
                    name="title"  
                />  
                <FormLabel>location</FormLabel>  
                <TextField  
                    value={inputs.location}  
                    onChange={handleChange}  
                    margin="normal"  
                    fullWidth  
                    variant="outlined"  
                    name="location"  
                />  
                <FormLabel>nonaval</FormLabel>  
                <TextField  
                    value={inputs.nonaval}  
                    onChange={handleChange}  
                    type="number"  
                    margin="normal"  
                    fullWidth  
                    variant="outlined"  
                    name="nonaval"  
                />  
                <FormLabel>status</FormLabel>  
                <TextField  
                    value={inputs.status}  
                    onChange={handleChange}  
                    margin="normal"  
                    fullWidth  
                    variant="outlined"  
                    name="status"  
                />  
            </Box>  
            <Button variant="contained" type="submit">  
                Add New  
            </Button>  
        </form>  
    );  
};  
export default AddLibrary;
```



The screenshot shows a user interface for adding a new library entry. It features a horizontal form with five input fields: 'title', 'location', 'number', 'nonaval', and 'status'. Each field is accompanied by a label and a text input. Below the form is a blue 'ADD NEW' button.

title	<input type="text"/>
location	<input type="text"/>
number	<input type="text"/>
nonaval	<input type="text"/>
status	<input type="text"/>

ADD NEW

# FrontEnd

## React

```
ul {  
  display: flex;  
  flex-direction: row;  
  align-content: center;  
  justify-content: space-around;  
  flex-wrap: wrap;  
}  
  
li {  
  padding: 10px;  
  margin: 10px;  
  width: 250px;  
  max-width: 250px;  
  max-height: 400px;  
  list-style-type: none;  
  justify-content: flex-start;  
}  
  
.card {  
  margin: auto;  
  padding: 5px;  
  height: 100%;  
  width: 100%;  
  box-shadow: 5px 5px 10px #010101;  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  /* align-items: flex-start;  
  align-content: flex-start; */  
}  
  
.card > img {  
  width: 100%;  
  height: 50%;  
}  
  
.card article {  
  padding: 5px;  
}  
  
.card h3 {  
  padding: 4px;  
}
```

```
import { Button } from "@mui/material";  
import axios from "axios";  
import { Link, useNavigate } from "react-router-dom";  
import "./Library.css";  
const Library = (props) => {  
  const history = useNavigate();  
  const { _id, id, title, location, number, nonaval, status } = props.library;  
  const deleteHandler = async () => {  
    await axios  
      .delete(`http://localhost:8080/libraries/${_id}`)  
      .then((res) => res.data)  
      .then(() => history("/"))  
      .then(() => history("/libraries"));  
  };  
  
  return (  
    <div className="card">  
  
      <p>{id}</p>  
      <h3>{title}</h3>  
      <p>{location}</p>  
      <p>{number}</p>  
      <p>{nonaval}</p>  
      <p>{status}</p>  
      <Button LinkComponent={Link} to={`/libraries/${_id}`} sx={{ mt: "auto" }}>  
        Update  
      </Button>  
      <Button color="error" onClick={deleteHandler} sx={{ mt: "auto" }}>  
        Delete  
      </Button>  
    </div>  
  );  
};  
  
export default Library;
```

3

### 58th Street Library

#### Get Directions

127 East 58th Street New York, NY  
10022  
212-759-7358  
10AM–6PM

UPDATE

DELETE

# FrontEnd

## React

```
import React, { useEffect, useState } from "react";
import "./Library";
import axios from "axios";
import Library from "./Library";
const URL = "http://localhost:8080/librarys";
const fetchHandler = async () => {
  return await axios.get(URL).then((res) => res.data);
};

const Librarys = () => {
  const [librarys, setLibrarys] = useState();
  useEffect(() => {
    fetchHandler().then((data) => setLibrarys(data.librarys));
  }, []);
  console.log(librarys);
  return (
    <div>
      <ul>
        {librarys &&
          librarys.map((library, i) => (
            <li key={i}>
              <Library library={library} />
            </li>
          )));
      </ul>
    </div>
  );
};

export default Librarys;
```

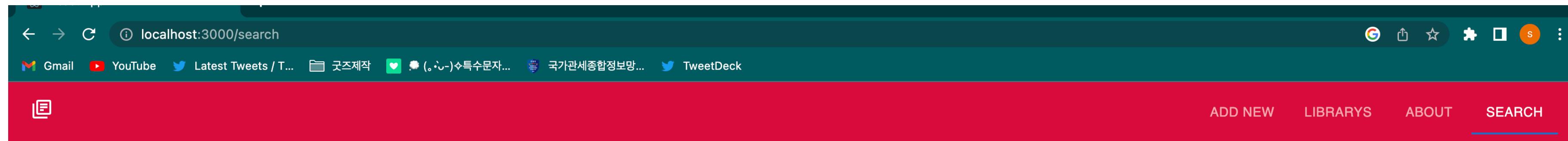
	NY 010-1234-5678	aaaa 123012310		1 125th Street Library Get Directions 224 East 125th Street New York, NY 10035 212-534-5050 Location is temporarily closed	2 53rd Street Library Get Directions 18 West 53rd Street New York, NY 10019 212-714-8400 11AM-6PM
	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE
3	<b>58th Street Library</b> Get Directions 127 East 58th Street New York, NY 10022 212-759-7358 10AM-6PM	<b>67th Street Library</b> Get Directions 328 East 67th Street New York, NY 10065 212-734-1717 11AM-7PM	<b>96th Street Library</b> Get Directions 112 East 96th Street New York, NY 10128 212-289-0908 10AM-7PM	<b>Aguilar Library</b> Get Directions 174 East 110th Street New York, NY 10029 212-534-2930 Closed	<b>Allerton Library</b> Get Directions 2740 Barnes Avenue Bronx, NY 10467 718-881-4240 10AM-6PM
4	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE
8	<b>Andrew Heiskell Braille and Talking Book Library</b> Collection Get Directions 40 West 20th Street New York, NY 10011 212-206-5400 10AM-5PM	<b>Art &amp; Architecture Collection</b> Get Directions Fifth Avenue and 42nd Street New York, NY 10018 212-930-0835 10AM-7:45PM	<b>Art and Artifacts Division</b> Get Directions 515 Malcolm X Boulevard (135th St and Malcolm X Blvd) New York, NY 10037 212-491-2241 11AM-5PM*	<b>Battery Park City Library</b> Get Directions 175 North End Avenue New York, NY 10282 212-790-3499 10AM-6PM	<b>Baychester Library</b> Get Directions 2049 Asch Loop North Bronx, NY 10475 718-379-6700 Location is temporarily closed
9	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE	UPDATE  DELETE
10					
12	localhost:3000/librarys	...	...	...	...
14	...	...	...	...	...
15	...	...	...	...	...
16	...	...	...	...	...
17	...	...	...	...	...

# 보완해야할 점

## Detail page, Search function, Delete



detail page 버튼 클릭 시, Url은 이동하나 내용 보이지 않음



Header 의 Search 버튼 클릭 시, 백엔드 구현, 프론트엔드 구현 추가 필요

Delete 프론트엔드만 구현, 백엔드 추가 필요

# **Github**

**<https://github.com/BandiRL/NYlibrary/>**

## **Demo video / Presentation**

**<https://www.youtube.com/watch?v=hqCgbDodc14>**

## **Presentation**

# **개발 환경**

**Mac 13.0.1 (22A400)**

**Visual Studio Code 1.74.2**

**Node js v18.12.1**

**MongoDB, MongoDB Atlas**

**Chrome**