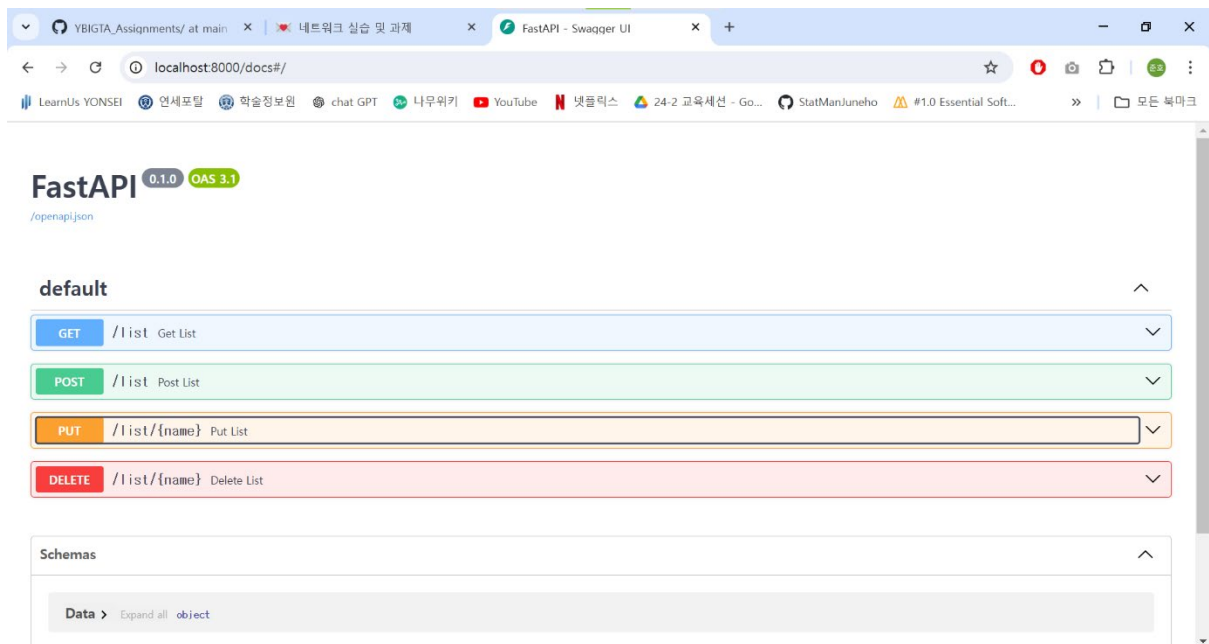


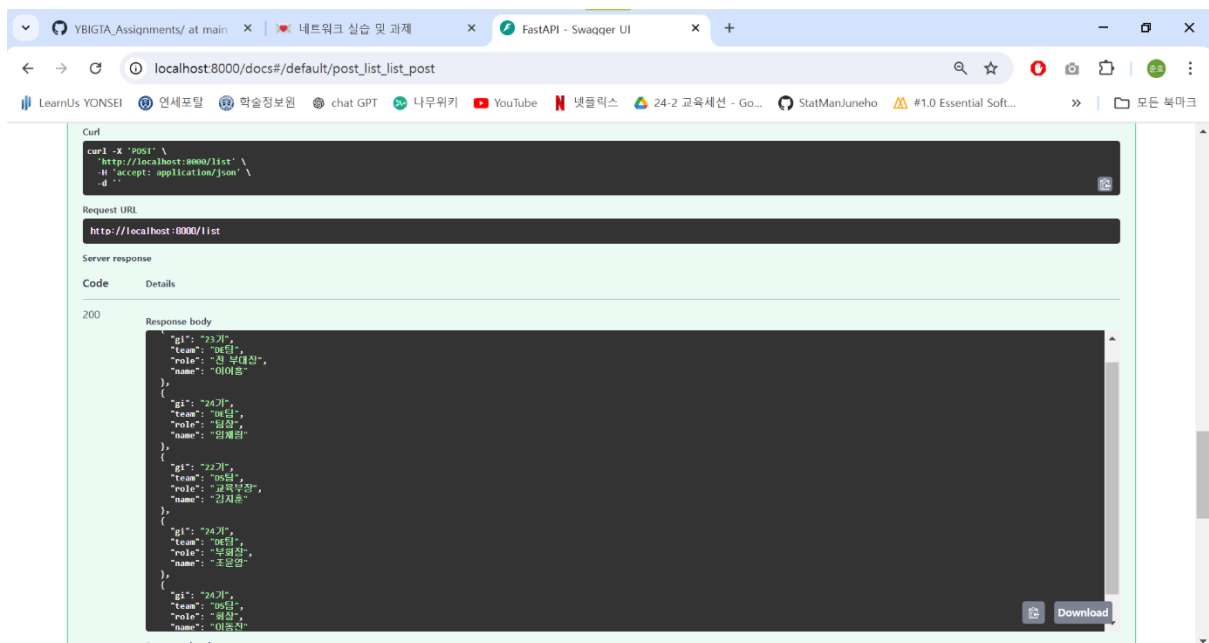
Report

김준호

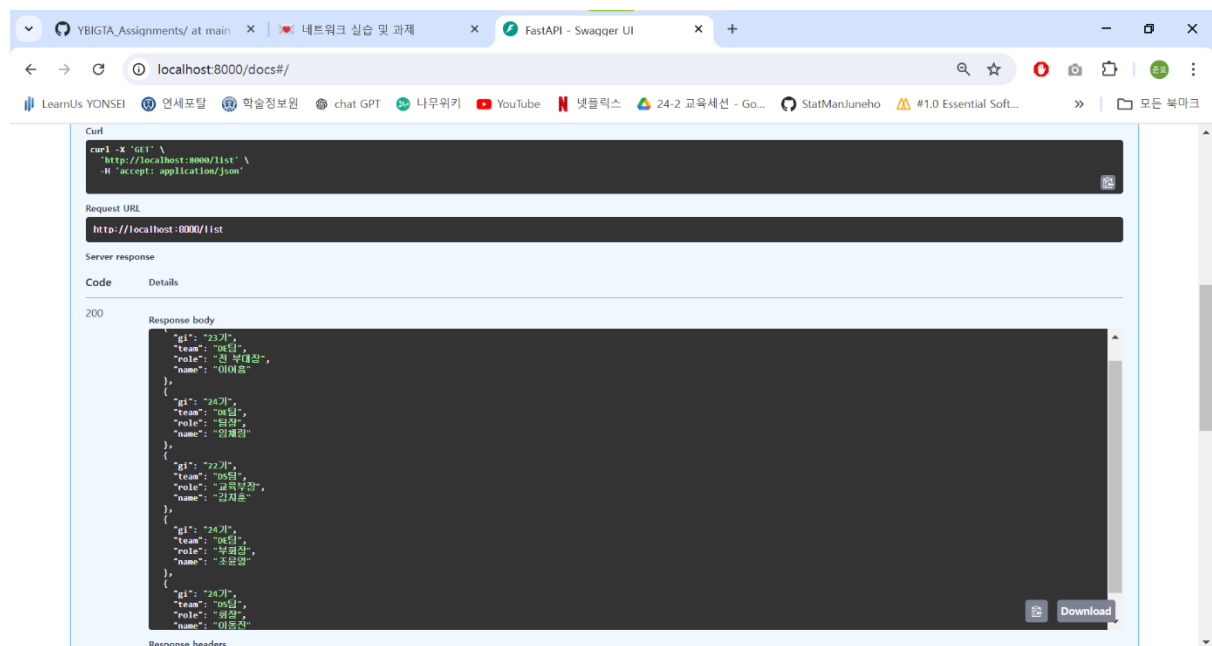
network_assn.py를 실행한 후 localhost:8000/docs에 접속하면 다음과 같이 페이지를 확인할 수 있다.



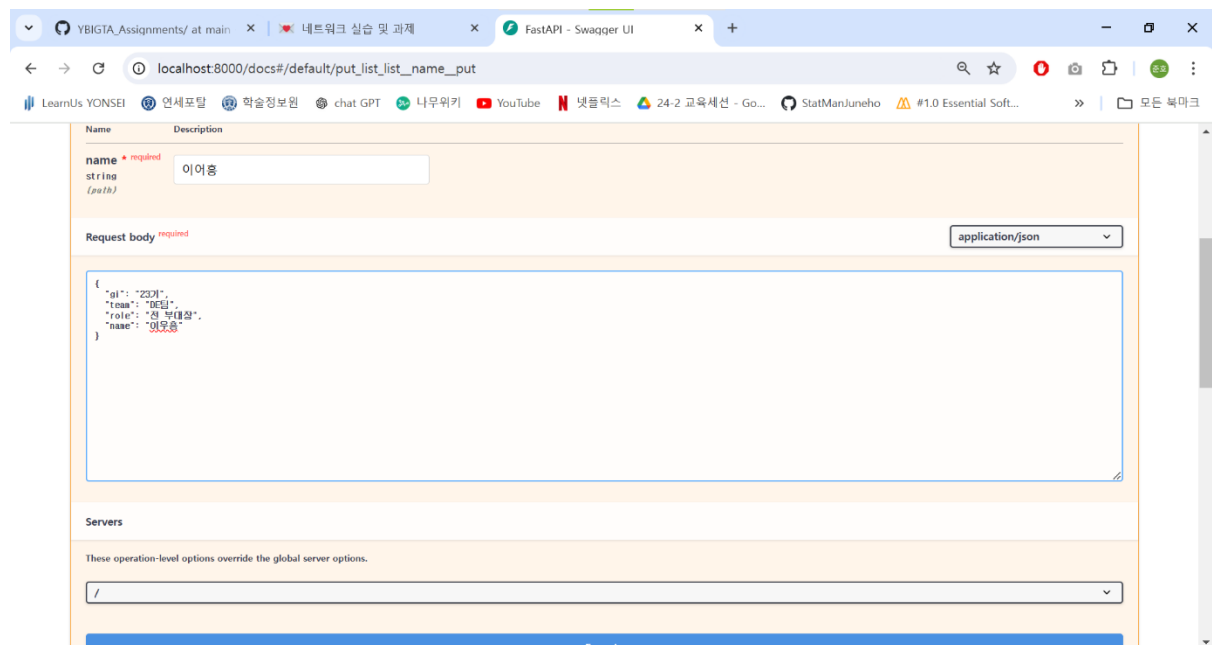
먼저 Post에서, 다음과 같이 총 5명 인원의 출석부를 불러올 수 있다.



그 다음, Get을 통해 출석부를 다시 확인해보자



그런데 DE팀 전 부대장님 성함이 이어흥으로 잘못 나타났다. Put에서 이우흥으로 고쳐보자.



실행 결과는 다음과 같이 나타난다.

The screenshot shows the Swagger UI interface for a FastAPI application. The URL is `localhost:8000/docs#/default/put_list_list_name_put`. The 'Responses' section is expanded, showing a 200 status code. The response body is a JSON object:

```
{
  "g1": "23기",
  "team": "06팀",
  "role": "전 부대장",
  "name": "이우승"
}
```

The response headers are also visible:

```
content-length: 71
content-type: application/json
date: Wed, 24 Jul 2024 17:15:22 GMT
server: uvicorn
```

Get에서 다시 한번 확인해보자. 잘 고쳐진 걸 확인할 수 있다.

The screenshot shows the Swagger UI interface for a FastAPI application. The URL is `localhost:8000/docs#/default/get_list_list_get`. The 'Responses' section is expanded, showing a 200 status code. The response body is a JSON array of team members:

```
[
  {
    "g1": "23기",
    "team": "06팀",
    "role": "전 부대장",
    "name": "이우승"
  },
  {
    "g1": "24기",
    "team": "06팀",
    "role": "임재관",
    "name": "임재관"
  },
  {
    "g1": "22기",
    "team": "05팀",
    "role": "김재환",
    "name": "김재환"
  },
  {
    "g1": "24기",
    "team": "06팀",
    "role": "부대장",
    "name": "조문영"
  },
  {
    "g1": "24기",
    "team": "05팀",
    "role": "조문영",
    "name": "이동진"
  }
]
```

The response headers are also visible:

```
content-length: 408
content-type: application/json
date: Wed, 24 Jul 2024 17:15:22 GMT
server: uvicorn
```

마지막으로 출퇴한 부대장을 출석부에서 Delete 해보자.

The screenshot shows the Swagger UI for the DELETE endpoint `/list/{name}`. The 'Parameters' section has a table with one parameter:

Name	Description
name * required	

The parameter type is 'string (path)' and its value is '이우훈'. Below the parameters is a 'Servers' section with a dropdown menu set to '/'. An 'Execute' button is visible. The 'Responses' section shows a table with one response:

Code	Description	Links
200	Successful Response	No links

The media type is set to 'application/json'.

Execute 결과 다음과 같은 결과창이 나타난다.

The screenshot shows the 'Responses' section of the Swagger UI after executing the DELETE request. It displays the following information:

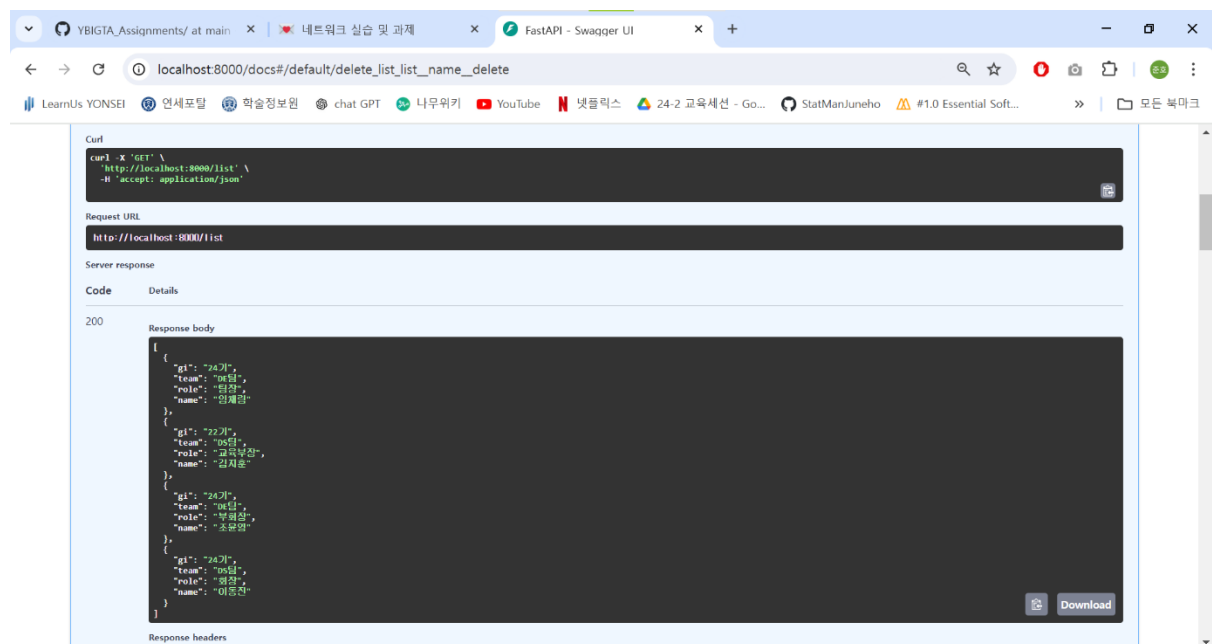
- Curl:** `curl -X 'DELETE' \`
`'http://localhost:8000/list/{name}' \`
`-H 'accept: application/json'`
- Request URL:** `http://localhost:8000/list/{name}`
- Server response:**
 - Code:** 200
 - Details:**
 - Response body:** `{`
 `"id": "23기",`
 `"team": "팀장",`
 `"role": "진 부대장",`
 `"name": "이우훈"`
`}`
 - Response headers:** `content-length: 71`
`content-type: application/json`
`date: Wed, 24 Jul 2024 17:17:54 GMT`
`server: uvicorn`

At the bottom, the 'Responses' table is shown with one response:

Code	Description	Links
200	Successful Response	No links

The media type is set to 'application/json'.

Get에서 다시 확인해보자. Delete 작업이 잘 수행된 걸 확인할 수 있다.



코드에 대한 자세한 설명:

```
class Data(BaseModel):
```

Pydantic의 BaseModel을 상속받아 데이터 모델을 정의하고, 각 속성(gi, team, role, name)의 타입을 정의한다.

```
def post_list():
```

/list 경로로 POST 요청이 들어오면 student_list를 생성하여 data_store에 추가한다.

```
def get_list():
```

/list 경로로 GET 요청이 들어오면 data_store의 모든 데이터를 반환한다.

```
def put_list(name: str, data: Data):
```

/list/{name} 경로로 PUT 요청이 들어오면 name에 해당하는 데이터를 data로 업데이트하고, 해당 이름의 데이터가 없으면 404 상태 코드를 반환한다.

```
def delete_list(name: str):
```

/list/{name} 경로로 DELETE 요청이 들어오면 name에 해당하는 데이터를 삭제하고, 해당 이름의 데이터가 없으면 404 상태 코드를 반환한다.

```
if __name__ == '__main__':
```

애플리케이션을 실행하기 위해 uvicorn 서버를 사용하고, localhost의 8000번 포트에서 애플리케이션을 실행한다.

FastAPI가 자동으로 Swagger UI를 통해 API 문서를 생성하여, localhost:8000/docs에서 API 문서를 확인할 수 있다.