

HUFS Missing Semester

개발자로 살아남기 위한 API

#4 Python으로 진행하는 간단한 REST API 실습



박용훈

발표자 / 실습 진행자

Global business & Technology 학부 19학번
멋쟁이사자처럼

**본 강의는 한국외국어대학교 AI교육원의
지원을 받아 제작되었음을 알려드립니다.**

CONTENTS

1. Json-server setting

2. REST API 실습 (Python, Postman)

a. GET : 조회(index)

b. POST : 생성

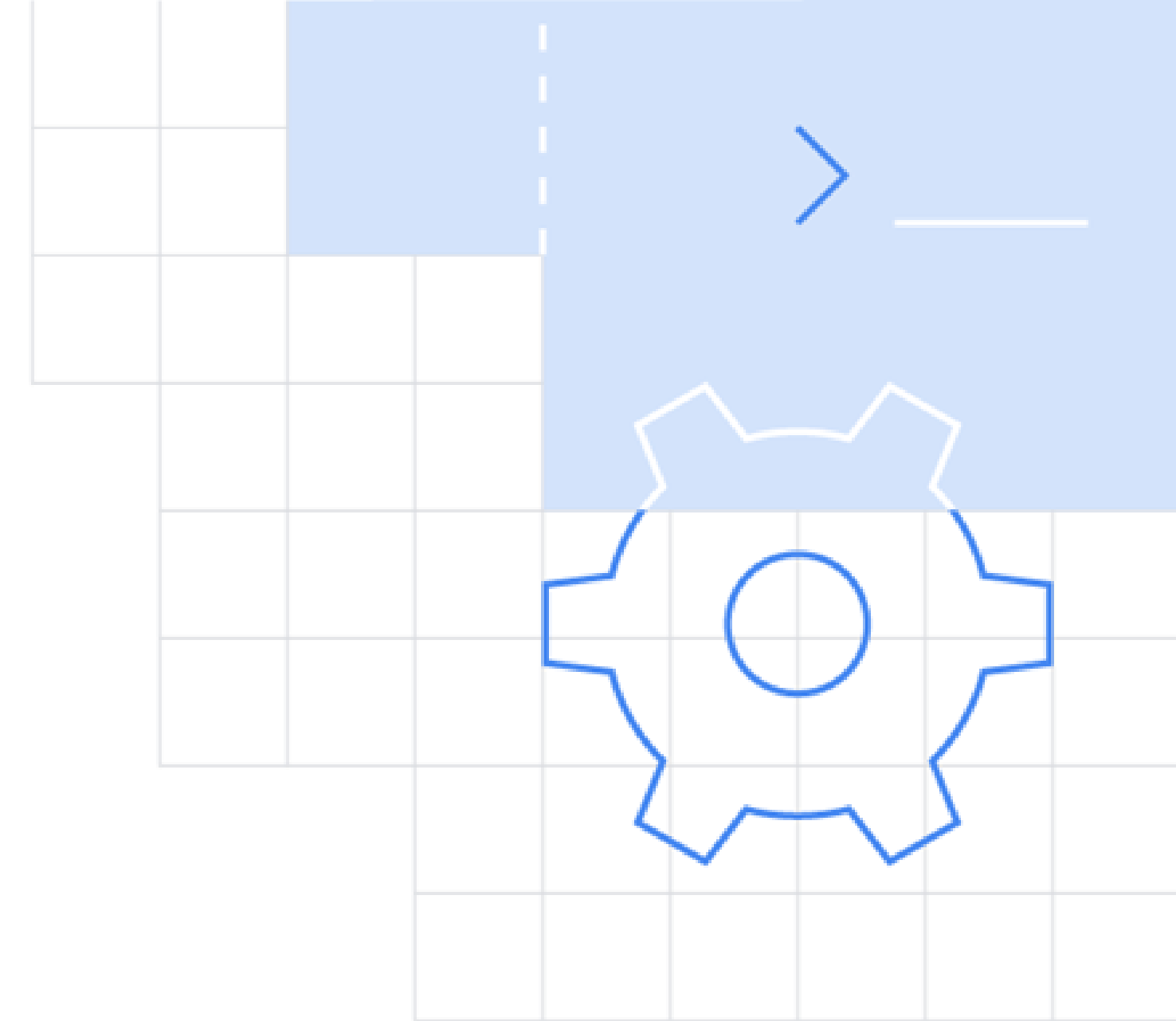
c. PUT : 특정 전체 갱신

d. PATCH : 특정 일부 갱신

e. DELETE : 삭제



#1 Json-server 설치



Json Server

- json 파일을 사용하여 간단한 REST API Mock server를 구축할 수 있는 툴
- 간단한 공부용에 사용하기 적합



Json-server setting

- 폴더 생성 & json-server 설치

- ✓ restAPI-exam라는 이름의 폴더 생성 후 경로 진입

```
$ mkdir restAPI  
$ cd restAPI
```

- ✓ Package.json을 기본 값으로 생성

```
$ npm init -y
```

- ✓ Npm을 사용하여 json-server 라이브러리 설치

```
$ npm install json-server
```



Json-server setting

```
{
  "todos": [
    {
      "id": 1,
      "content": "HTML",
      "completed": false
    },
    {
      "id": 2,
      "content": "CSS",
      "completed": true
    },
    {
      "id": 3,
      "content": "Javascript",
      "completed": false
    },
    {
      "id": 4,
      "content": "Python",
      "completed": true
    }
  ]
}
```



restAPI.json 파일 생성

Json-server setting

```
{
  "name": "yonpark",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node_modules/.bin/json-server --watch restAPI.json --port 8001"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "json-server": "^0.16.3"
  }
}
```



Package.json 수정

Json-server setting

- Json-server 실행

- ✓ Package.json에 정의된 start 명령어 실행

```
$ npm start
```



Json-server setting

터미널에 다음과
같이 뜬다면 성공!

```
> yonpark@1.0.0 start C:\Users\pyw08\Desktop\examjs\yonpark  
> json-server --watch restAPI.json --port 8001
```

```
\{^_^}/ hi!
```

```
Loading restAPI.json  
Done
```

Resources

<http://localhost:8001/todos>

Home

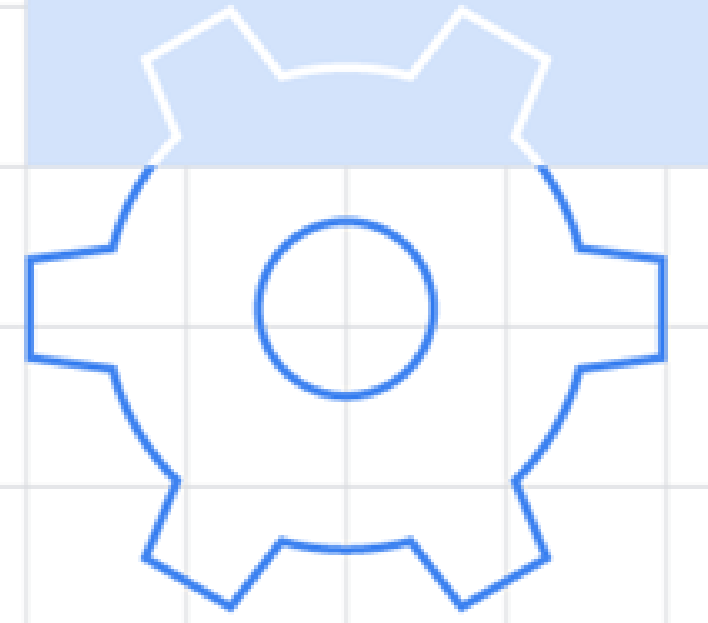
<http://localhost:8001>

```
Type s + enter at any time to create a snapshot of the database  
Watching...
```



#2 REST API 실습

With Python, Postman



REST API 실습

- 가상환경 설정

- ✓ 파이썬 가상환경 추가

```
$ python -m venv venv
```

- ✓ 파이썬 가상환경 진입

```
$ .\venv\Scripts\activate.bat
```

```
터미널: Local × Local (2) × +  
  
C:\Users\pyw08\Desktop\examjs\yonpark>python -m venv venv  
  
C:\Users\pyw08\Desktop\examjs\yonpark>.\venv\Scripts\activate.bat  
  
(venv) C:\Users\pyw08\Desktop\examjs\yonpark>
```



REST API 실습

- Requests 라이브러리 설치

- ✓ requests 라이브러리 설치

```
$ pip install requests
```

- ✓ requests 라이브러리 설치 확인

```
$ pip list
```

```
터미널: Local × Local (2) × +
(venv) C:\Users\pyw08\Desktop\examjs\yonpark>pip list
Package            Version
-----
certifi             2021.5.30
charset-normalizer  2.0.3
idna                3.2
pip                20.2.3
requests            2.26.0
setuptools          49.2.1
urllib3             1.26.6
```

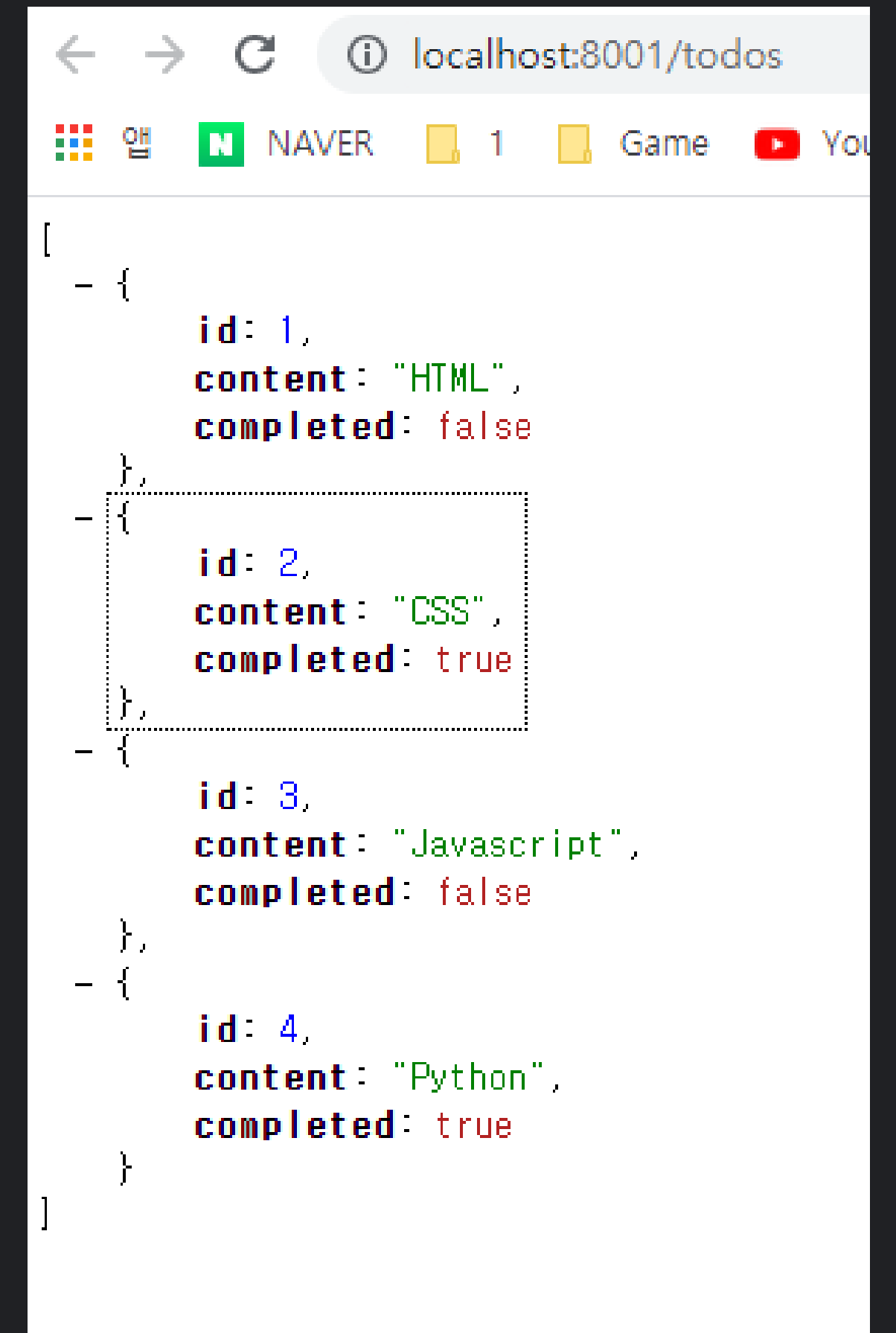


REST API 실습

```
import requests
import json

url_item = "http://localhost:8001/todos"
response = requests.get(url_item)

print(response.text)
```



```
[
  - {
      id: 1,
      content: "HTML",
      completed: false
    },
  - {
      id: 2,
      content: "CSS",
      completed: true
    },
  - {
      id: 3,
      content: "Javascript",
      completed: false
    },
  - {
      id: 4,
      content: "Python",
      completed: true
    }
]
```

GET : 모든 todo를 조회



REST API 실습

```
import requests
import json

url_items = "http://localhost:8001/todos"

new_item = {
    "id": 5,
    "content": "R",
    "completed": False
}
response = requests.post(url_items, data=new_item)

print(response.text)
```



POST : 새로운 todo 생성

REST API 실습

```
import requests
import json

url_items = "http://localhost:8001/todos/4"

new_item = {
    "id": 4,
    "content": "CPP",
    "completed": False
}
response = requests.put(url_items, data=new_item)

print(response.json())
```



PUT : 특정 리소스의 **전체 갱신**

REST API 실습

```
import requests
import json

url_items = "http://localhost:8001/todos/4"

new_item = {"content": "Dart"}

response = requests.patch(url_items, data=new_item)

print(response.text)
```



PATCH : 특정 리소스의 일부 갱신

REST API 실습

```
import requests
import json

url_items = "http://localhost:8001/todos/5"

response = requests.delete(url_items)

print(response.text)
```



DELETE : id를 사용하여 todo를
특정하고 삭제

감사합니다