

# [2강] HTTP, JSON, API 개념

🕒 Created	@Aug 13, 2020 6:04 PM
🏷 Tags	
📝 메모	

HTTP의 개념과 HTTP 메소드의 종류, JSON의 개념과 기본적인 사용법에 대해서 알아봅니다.

그리고 API와 REST API의 개념을 예시를 통해 알아보도록 하겠습니다.

## HTTP

Transfer Protocol

HTTP의 구성

HTTP의 요청 메소드

## JSON

특징

MDN json (link)([mdn.github.io](https://mdn.github.io))

## API

API의 종류

## REST API

아키텍처란?

REST의 구성요소

# HTTP

H yper

T ext

T ransfer

P rotocol

참조를 통해 한 문서에서 관련된 다른 문서들로 넘나들며 원하는 정보를 얻을 수 있게 해주는 텍스트 (예: 유튜브 - 영상 하나만 보는게 아니라 다른 영상들도 추천받고 그쪽으로 넘어가는데 넘어갈 때 사용하는 텍스트)

# Transfer Protocol

인터넷을 통해서 정보를 주고받을때 지켜야하는 규칙

## HTTP의 구성

주고받는 과정이기 때문에 Request 요청과 Response 응답으로 이루어짐

## HTTP의 요청 메소드

**Get**: URL(Uniform Resource Locator)에 표시된 리소스를 가져오기 (**읽기**)

⇒ 주소를 통해 리소스에 다가가는 방법.

**POST**: body에 정보를 담아 서버에 입력 (**쓰기**)

예) 페이스북에 글을 올릴 때 작성한 텍스트를 body에 담아 서버에 입력요청.

**PUT**: URL에 표시된 리소스와 바꿔치기 (**수정**)

🔍 post처럼 body에 정보 담아 보내면 PUT으로 title 보내고 content는 빈 채로 보내면 title 바뀌고 내용은 없는 상태로 post 됨?

**PATCH**: PUT과 다르게 '일부만' 수정 (**수정**)

🔍 title만 body에 실어서 서버에 입력했을 때 title만 변경이 되고 content는 내버려둠.

**DELETE**: URL에 표시된 특정 리소스를 삭제 (**삭제**)

## JSON

**J** ava

**S** cript

**O** bject

**N** otation

Key : Value 형식. Python의 dict type과 유사.

데이터 교환할 때 자주 사용함 ⇒ 프로토콜이 필요하고 정보가 뭐가 어쨌다고?

예전엔 XML 형식을 사용함

## 특징

- 기존 XML보다 가볍다 ⇒ 경제적이다
- 많은 프로그래밍 언어가 지원한다
- 전송시: 직렬화 과정

네트워크상에서 정보 보낼 때는 string으로 보내야하기 때문에 json을 ⇒ string으로 변환

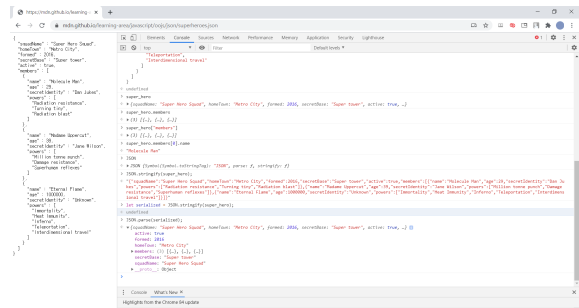
- 수신시: 역직렬화 과정  
string을 ⇒ object 형식으로 만들어서 수신

## 직렬, 역직렬 실습(JSON)

`stringify` = 직렬

`parse` = 역직렬

```
> JSON.stringify(super_hero);
'{"name":"Super Hero Squad","homeTown":"Metro City","format":2016,"secretBase":"Super base","active":true,"members":[{"name":"Molecule Man","age":25,"secretIdentity":"Max 36","powers":["radiation resistance","flaming hair","radiation blast"]},{name":"Hulk","secretIdentity":"Bruce Banner","age":30,"secretIdentity":"Bruce Banner","powers":["superhuman strength"]},{name":"Hulk","secretIdentity":"Bruce Banner","age":30,"secretIdentity":"Bruce Banner","powers":["superhuman strength"]]}]}'
> JSON.parse(JSON.stringify(super_hero));
{"name":"Super Hero Squad","homeTown":"Metro City","format":2016,"secretBase":"Super base","active":true,"members":[{"name":"Molecule Man","age":25,"secretIdentity":"Max 36","powers":["radiation resistance","flaming hair","radiation blast"]},{name":"Hulk","secretIdentity":"Bruce Banner","age":30,"secretIdentity":"Bruce Banner","powers":["superhuman strength"]},{name":"Hulk","secretIdentity":"Bruce Banner","age":30,"secretIdentity":"Bruce Banner","powers":["superhuman strength"]}]}
> JSON.stringify(super_hero);
'{"name":"Super Hero Squad","homeTown":"Metro City","format":2016,"secretBase":"Super base","active":true,"members":[{"name":"Molecule Man","age":25,"secretIdentity":"Max 36","powers":["radiation resistance","flaming hair","radiation blast"]},{name":"Hulk","secretIdentity":"Bruce Banner","age":30,"secretIdentity":"Bruce Banner","powers":["superhuman strength"]},{name":"Hulk","secretIdentity":"Bruce Banner","age":30,"secretIdentity":"Bruce Banner","powers":["superhuman strength"]}]}'
```



## MDN json ([link](https://developer.mozilla.org/en-US/docs/Glossary/JSON))([mdn.github.io](https://mdn.github.io))

moz://a 재단에서 운영하는 개발자 네트워크

웹 기술 관련된 내용이 많음

## API

Application ⇒ 휴대폰 앱이 아니라 웹상에서 돌아가는 응용프로그램들 (카카오지도, 네이버지도, 기상청 날씨예보 등 우리가 사용하는 다양한 서비스들을 포함함)

Programming

Interface

서비스들이 사용하는 데이터에 접근하고 조작하는 도구 (TV의 리모컨같은 기능)

회사에서 API를 제공해주면 프로그래밍언어를 사용해 해당 서비스? 제공 자료?를 발전할 수 있음

## API의 종류

**SOAP** Simple Object Access Protocol

XML을 쉽게 주고받는 프로토콜의 일종 ⇒ 좀 어려움 (REST 먼저 하는게 좋음)

**REST** Representation State Transfer

일종의 아키텍처 ??

**GraphQL** Graph Query Language

API 서버 만들 때 REST를 사용해서 만들긴 하는데 그 한계를 페이스북에서 극복하고 만들었음.

## REST API

일종의 소프트웨어 아키텍처

어디에 뭘 붙이고 사용자들이 어떻게 사용하고 사용하는 방법은 어떻게되고... .. 이런걸 REST가 정해놓은것임.

(장점은 따로 알아서 공부할 것)

**R**

**E** presentational

**S** tate

**T** ransfer

## 아키텍처란?

소프트웨어를 설계하는 지침과 원칙 ⇒ 반드시!!! 지킬 필요는 없지만 지키면 좋음

## REST의 구성요소

## REST의 구성요소

**자원** GET /likelion/member/8<sup>th</sup>/list  
PATCH /likelion/member/8<sup>th</sup>/허태정

**행위** GET /likelion/member/8<sup>th</sup>/list  
PATCH /likelion/member/8<sup>th</sup>/허태정

**표현** {  
    "members" : ["김멋사", ... , "허멋사"]  
}

---

😞 뭘소린지 모르겠어서 일단 넘김