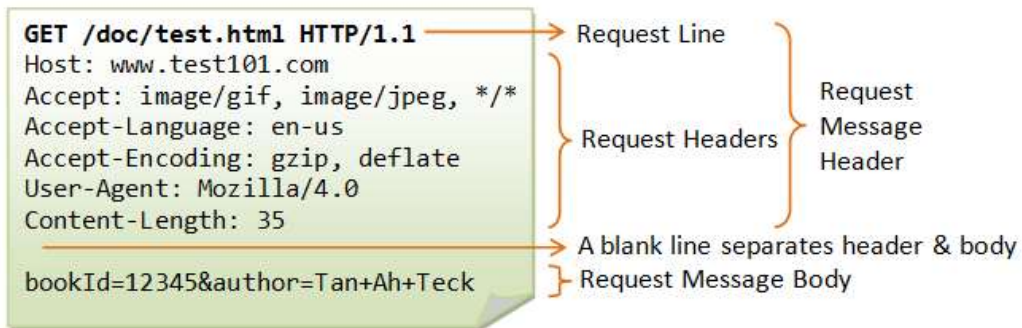


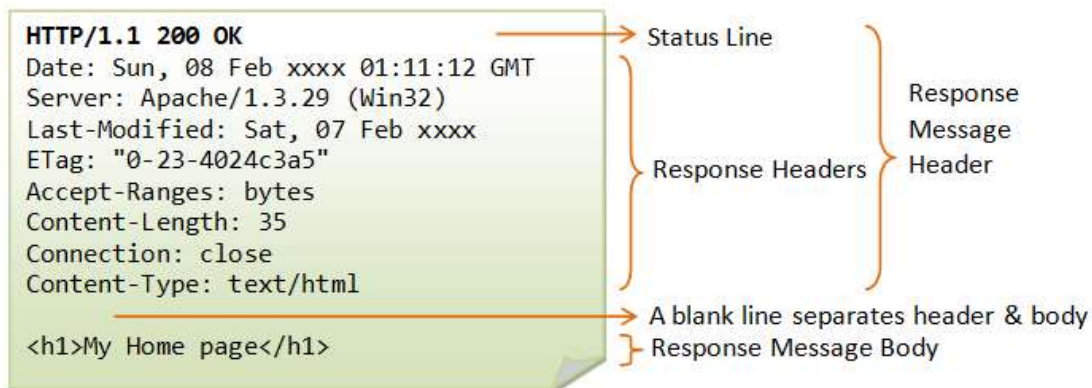
HTTP Message

HTTP Message

request



response



HTTP Message 구조

시작줄 : 실행되어야 할 요청 또는 요청 수행에 대한 성공 또는 실패가 기록되어있음

요청 시작줄 : HTTP 메서드, URL, HTTP버전

ex) GET https://dapi.kakao.com/v3/search/book?query=a HTTP/1.1

응답 시작줄 : 응답의 시작줄은 상태 줄이라고 불린다.

프로토콜 버전, 상태코드, 상태 텍스트

ex) HTTP/1.1 404 Not Found.

header : 요청에 대한 설명 또는 메시지 본문(body)에 대한 설명

body(본문): 추가 또는 업데이트를 하기 위한 데이터 또는 사용자가 요청한 데이터가 담겨있다.

모든 요청이나 응답에 body가 존재하는 것은 아니다.

HTTP Message

HTTP 상태코드(<https://developer.mozilla.org/ko/docs/Web/HTTP/Status>)

상태코드 종류

1xx : 요청을 받았으며 프로세스를 계속 진행

2xx : 통신 성공

3xx(redirection) : 요청 완료를 위한 추가 요청이 필요함

4xx : 클라이언트 오류(요청 문법이 틀렸거나, 존재하지 않는 resource를 요청했거나...)

5xx : 서버 오류(요청 문법은 맞았지만 서버가 해당 요청에 응답할 수 없음)

자주 볼 상태코드

200 : 성공!

400 : 잘못된 문법의 요청, bad request

401 : 비인증, 서버에서 요구하는 방식으로 스스로를 인증해야함

403 : 콘텐츠에 접근할 권한이 없음

404 : 요청한 리소스를 찾을 수 없음

405 : 허용되지 않은 http 메서드

500 : 서버가 처리방법을 모르는 상황 발생

503 : 요청한 서비스가 현재 불가능, 서버과부하로 발생

JSON

JSON

JSON(JavaScript Object Notation) : 자바스크립트 객체 문법.

전송할 데이터를 자바스크립트의 객체 문법에 따라 구조화하는 문자기반 데이터 형식
구조가 자바스크립트 객체일 뿐 자바스크립트 뿐 아니라 모든 언어에서 JSON을 읽고 쓰기 위한 기능을 제공한다.
현재 통신에서 가장 많이 사용되는 데이터 형식이다.

```
ex) {
  "employees":
  [
    {
      "name": "Raj",
      "email": "raj@gmail.com",
      "age": 32
    },
    {
      "name": "Mohan",
      "email": "Mohan@yahoo.com",
      "age": 21
    }
  ]
}
```

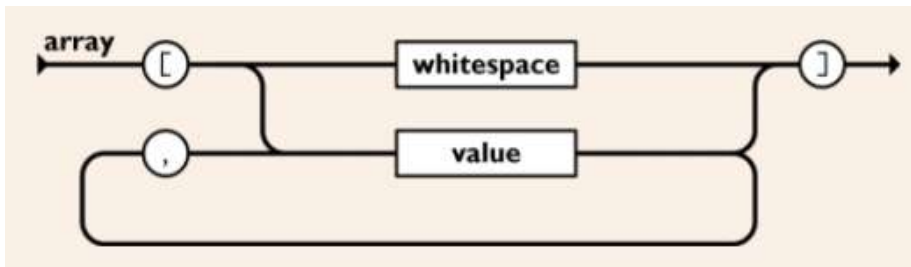
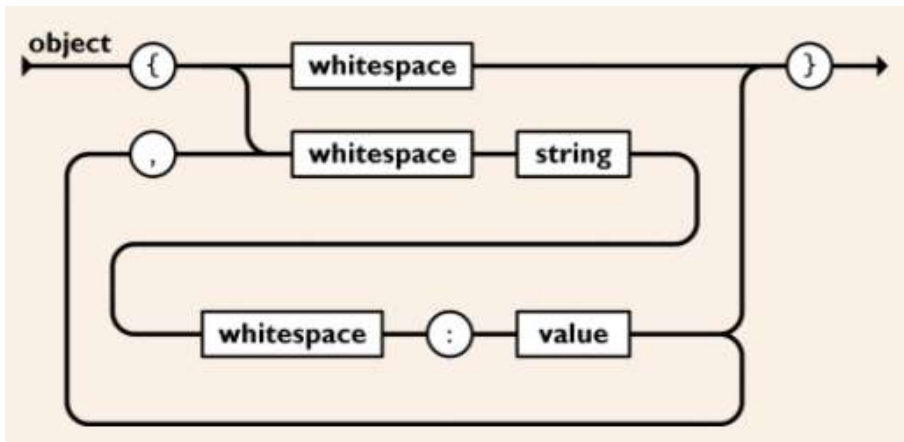
JSON

JSON에는 object, array, string, number, boolean, null을 담을 수 있다.

객체는 속성과 값으로 이루어져 있다.

,를 사용해 여러 속성을 표현할 수 있다.

자바스크립트와 마찬가지로 JSON으로 표현 가능한 모든 타입을 속성 값으로 사용할 수 있다.

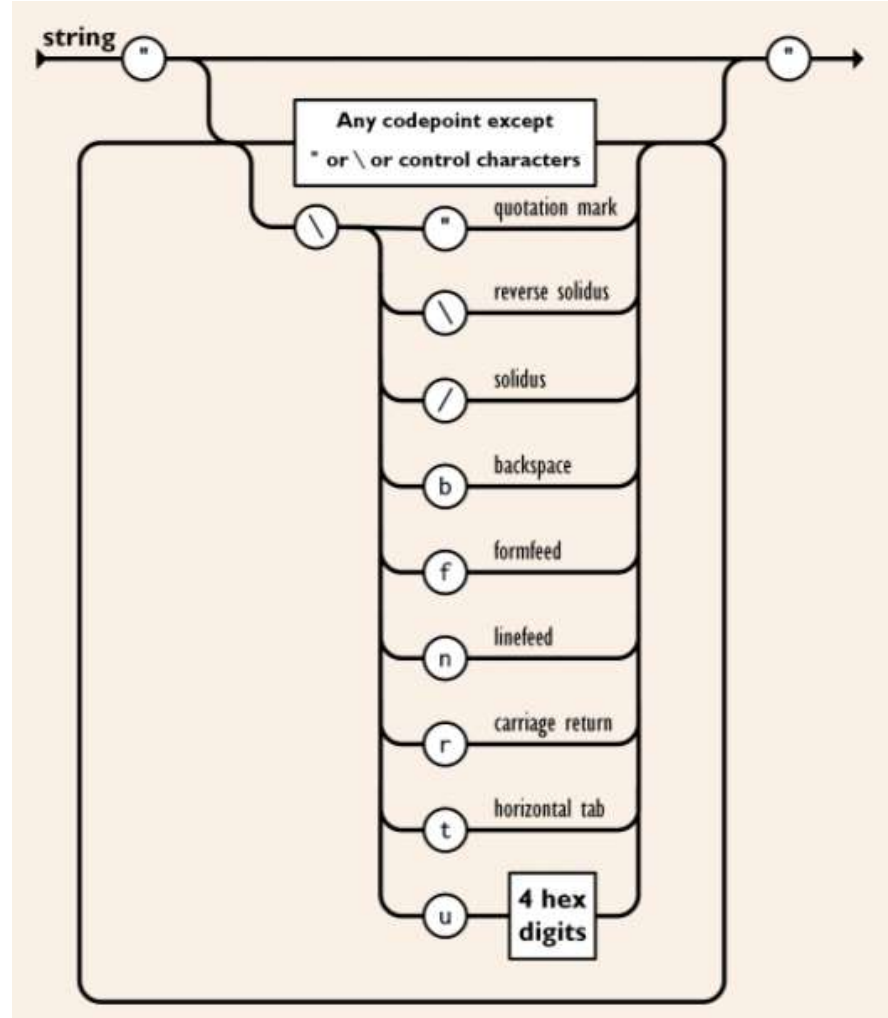
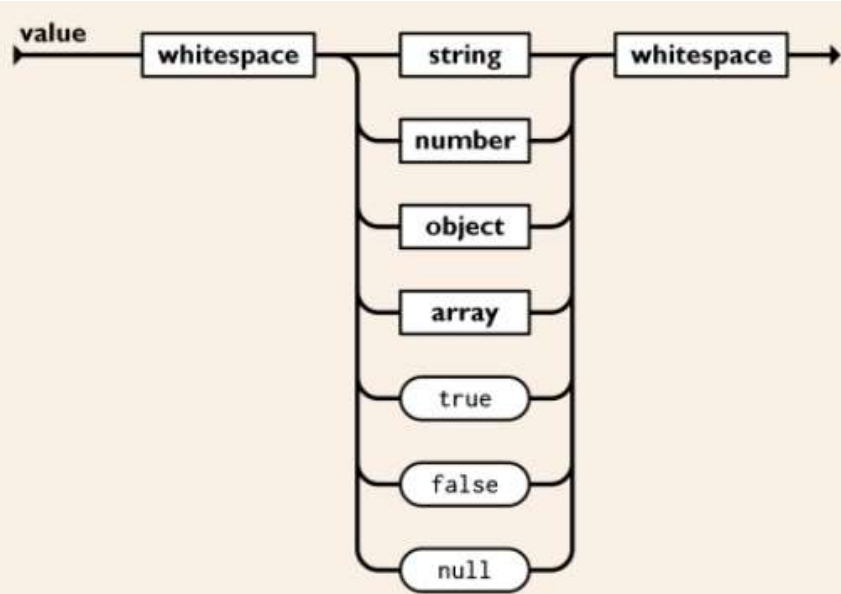


JSON

*** JSON에서 문자열 표준은 "" (Double Quote) 이다.

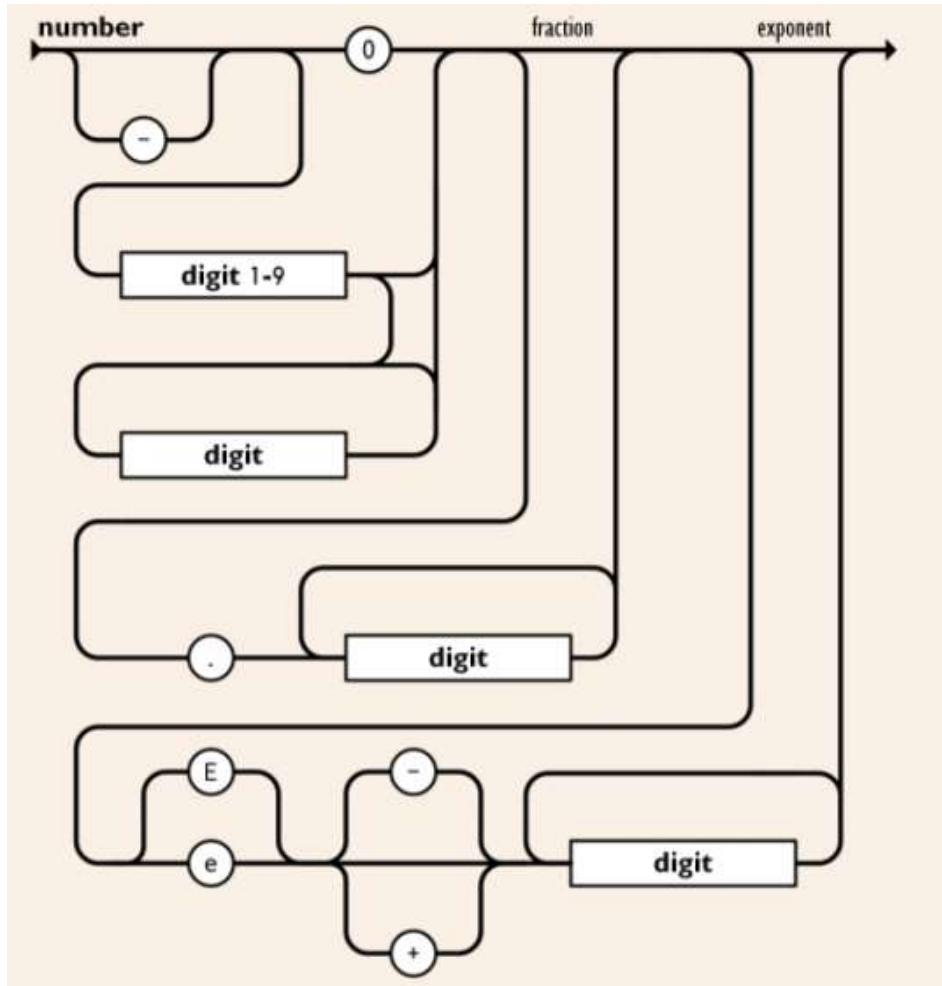
formfeed : 프린트 할 경우 다음 페이지로 이동

4 hex digits : 16진수 4자리로 표현된 유니코드



JSON

숫자는 정수, 실수, 지수를 표현할 수 있다.
8진수나 16진수 등을 표현할 수는 없다.



JSON

JSON Parsing

자바스크립트에서는 json을 다루기 위한 JSON 객체를 제공한다.

JSON 객체의 parse 메서드를 사용해 JSON을 자바스크립트 객체로
stringify 메서드를 사용해 자바스크립트 객체를 JSON으로 파싱할 수 있다.

```
1 const json = '{"result":true, "count":42}';
2 const obj = JSON.parse(json);
3
4 console.log(obj.count);
5 // expected output: 42
6
7 console.log(obj.result);
8 // expected output: true
9
```

```
1 console.log(JSON.stringify({ x: 5, y: 6 }));
2 // expected output: '{"x":5,"y":6}'
3
4 console.log(JSON.stringify([new Number(3), new String('false'), new Boolean(false)]));
5 // expected output: "[3,\"false\",false]"
6
7 console.log(JSON.stringify({ x: [10, undefined, function() {}, Symbol('')] }));
8 // expected output: '{"x":[10,null,null,null]}'
9
10 console.log(JSON.stringify(new Date(2006, 0, 2, 15, 4, 5)));
11 // expected output: "\"2006-01-02T15:04:05.000Z\""
12
```

JSON

JSON 공식사이트 : <https://www.json.org/json-ko.html>

JSON 유효성 검사 : <https://jsonlint.com/>