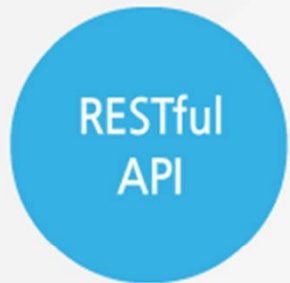


웹의 이해

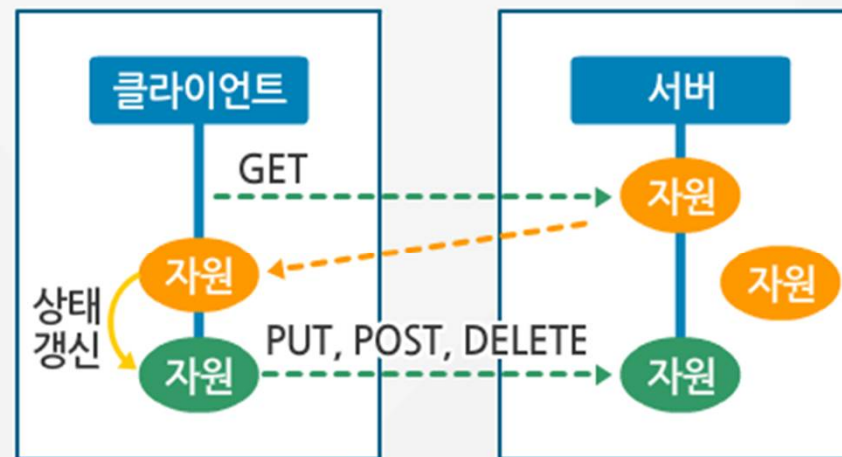
- client vs server
- Request vs Response
- Request 방법 :
 - GET
 - POST
- Response 방법
 - html, 이미지, css, json, xml, ...
- 웹서버 vs WAS 서버
- url vs port
- 쿠키 vs 세션
- string vs stream



2. 클라이언트 - 서버 모델



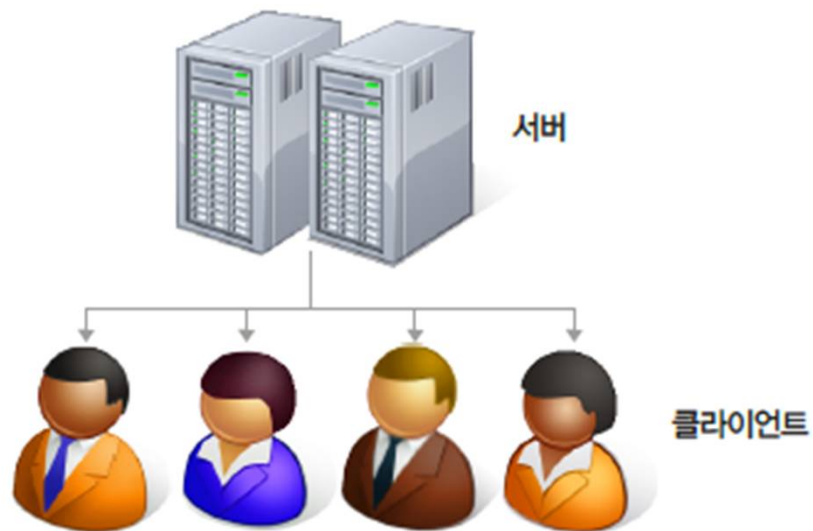
- 웹 이외의 애플리케이션에도 사용
- 사물 간의 통신을 위한 API
- 객체 간 동배 간으로 자원을 확장하고, 재사용하는 개념



[RESTful API의 클라이언트-서버 모델]

server vs client

- 클라이언트(사용자) : 요청하는 쪽
- 서버(제공자) : 응답하는 쪽
- 웹 : 클라이언트가 서버에 **요청**하면 요청에 **응답**하여 웹 페이지를 제공하는 장소

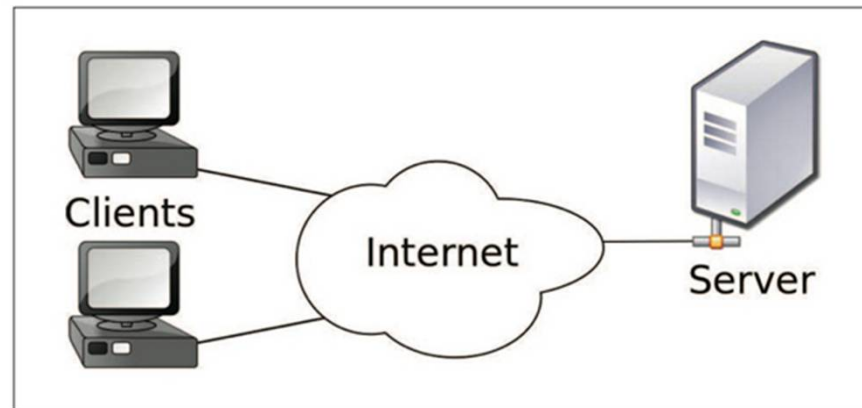


서버와 클라이언트

server vs client

- 클라이언트 프로그램

- ◉ 클라이언트에서 실행되는 프로그램
- ◉ ex) 웹 브라우저
- ◉ HTML, CSS, 자바스크립트로 구성



- 서버 프로그램

- ◉ 클라이언트의 요청에 따라 적절한 데이터를 제공.
- ◉ 서버에서 실행되는 프로그램
- ◉ 자바, C#, PHP, 자바스크립트와 같은 언어로 개발

request(요청) vs response(응답)

- 치킨 주문과 웹 페이지 주문
 - ⊙ 웹 서버 : 치킨집
 - ⊙ 웹 페이지 : 치킨

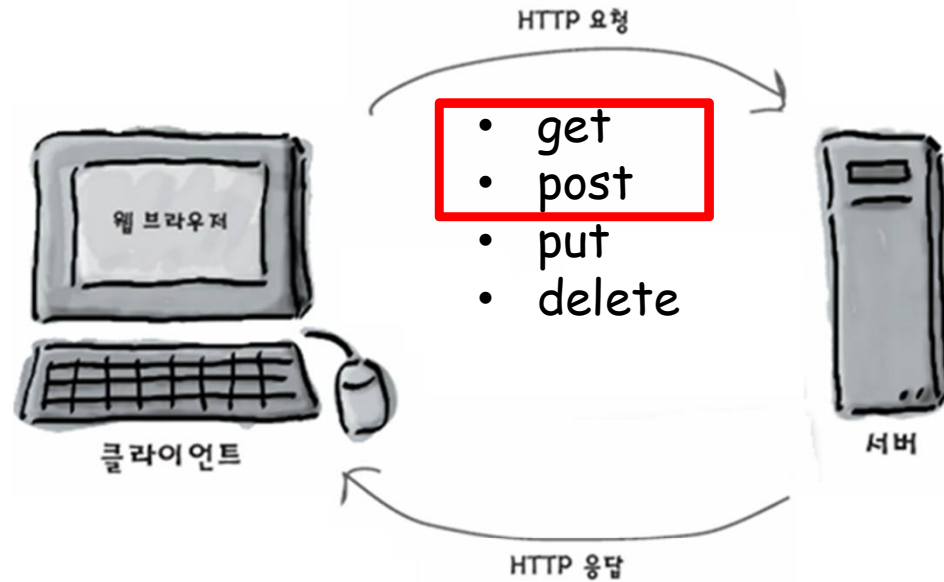


요청과 응답 과정



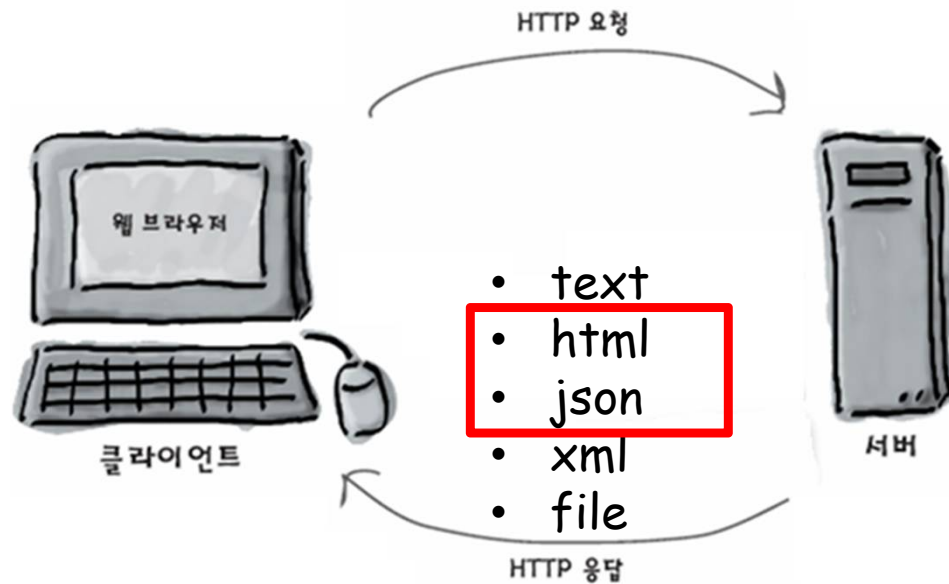
서버의 요청과 응답

Request(요청): GET vs POST



	GET	POST
정보 노출	URL을 통해 노출	HTTP 헤더 속으로 감춰짐
전달 가능한 정보의 양	URL 주소의 한계 길이인 4096 바이트를 넘을 수 없음(이론적으로는 URL 주소의 길이 제한 없음)	이론적으로는 무한대지나 현실적으로는 웹서버의 응답 지연 시간만큼 전송 가능
정보 전달 방식	URL 뒤에 ?를 붙이거나 HTML 폼 형식에서 이용 가능	반드시 HTML 폼을 통해서만 사용할 수 있음
사용 범위	포털 사이트의 검색어 전달, 게시판 페이지 번호 등 정보 위험도와 관계 없는 부분에서 많이 사용됨	회원 아이디, 비밀번호, 개인 정보 등 개인 정보 전송에 많이 사용됨

Response(응답):json vs xml



```
{  
  hey: "guy",  
  anumber: 243,  
  anobject: {  
    whoa: "nuts",  
    anarray: [  
      1,  
      2,  
      "thr<h1>ee"  
    ],  
    more: "stuff"  
  },  
  awesome: true,  
  bogus: false  
}
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<xypair>  
  <xaxis axistype="independent">  
    <property>Wavelength</property>  
    <value>254.0</value>  
    <unit>nm</unit>  
  </xaxis>  
  <yaxis axistype="dependent">  
    <property>Absorbance</property>  
    <value>0.1234</value>  
    <unit>none</unit>  
  </yaxis>  
</xypair>
```

request(요청) vs response(응답)

