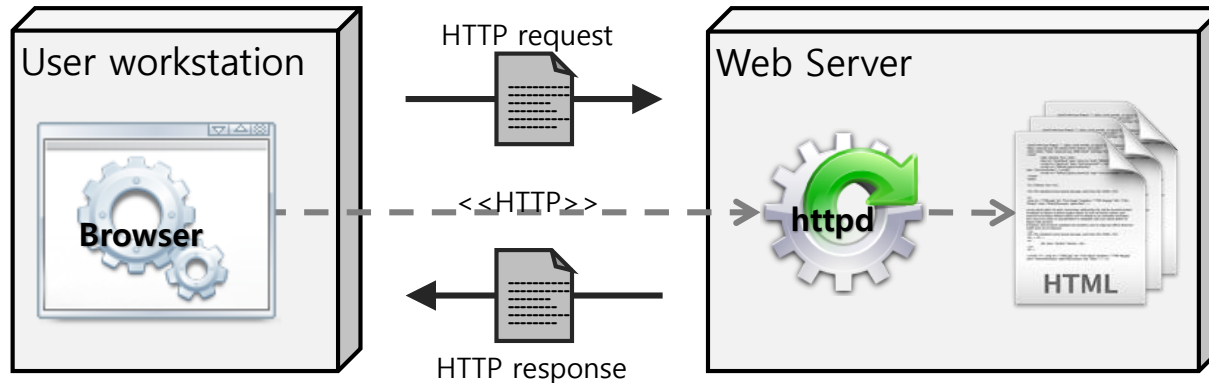


HTTP Client-Server Architecture

- ✓ 브라우저가 서버의 페이지를 요청하면 서버는 해당 파일을 찾는 다음 HTTP 응답을 통해 클라이언트에 전송
- ✓ 브라우저는 응답된 페이지를 해석하여 화면에 보여준다.
- ✓ request 정보는 사용자가 원하는 파일 또는 리소스의 위치와 브라우저에 관한 정보를 포함한다.
- ✓ response 정보는 요청한 자원에 관한 정보를 가지고 있으며, 일반적으로 텍스트 형태이며, 그래픽 등을 바이너리 정보를 포함할 수 도 있다.



HTTP Message

- ✓ 웹 어플리케이션은 요청과 응답으로 서버의 데이터를 클라이언트에 보여준다.
- ✓ 모든 것은 요청과 응답으로 이루어진다.

HTTP Request 메시지의 구성

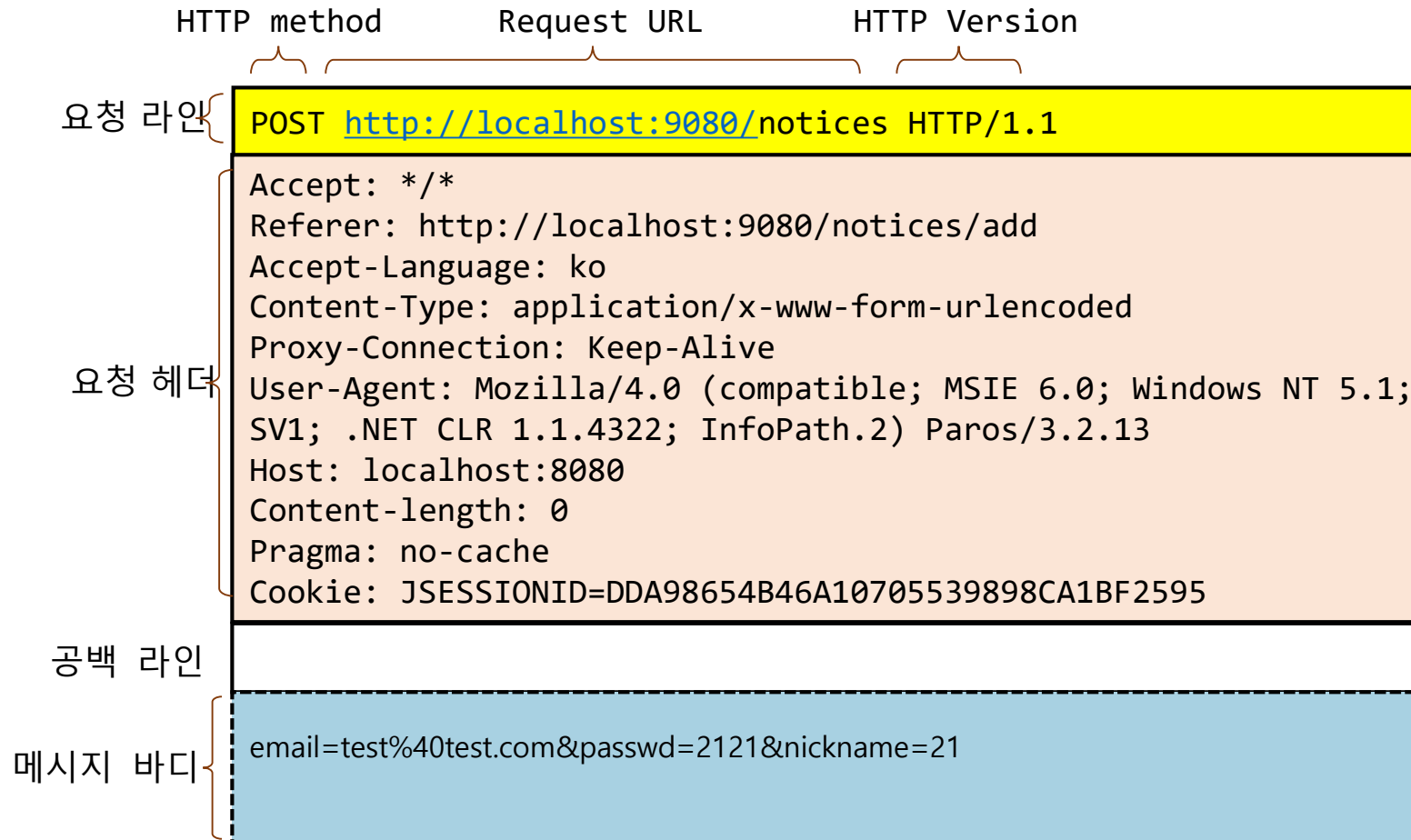


HTTP Response 메시지의 구성



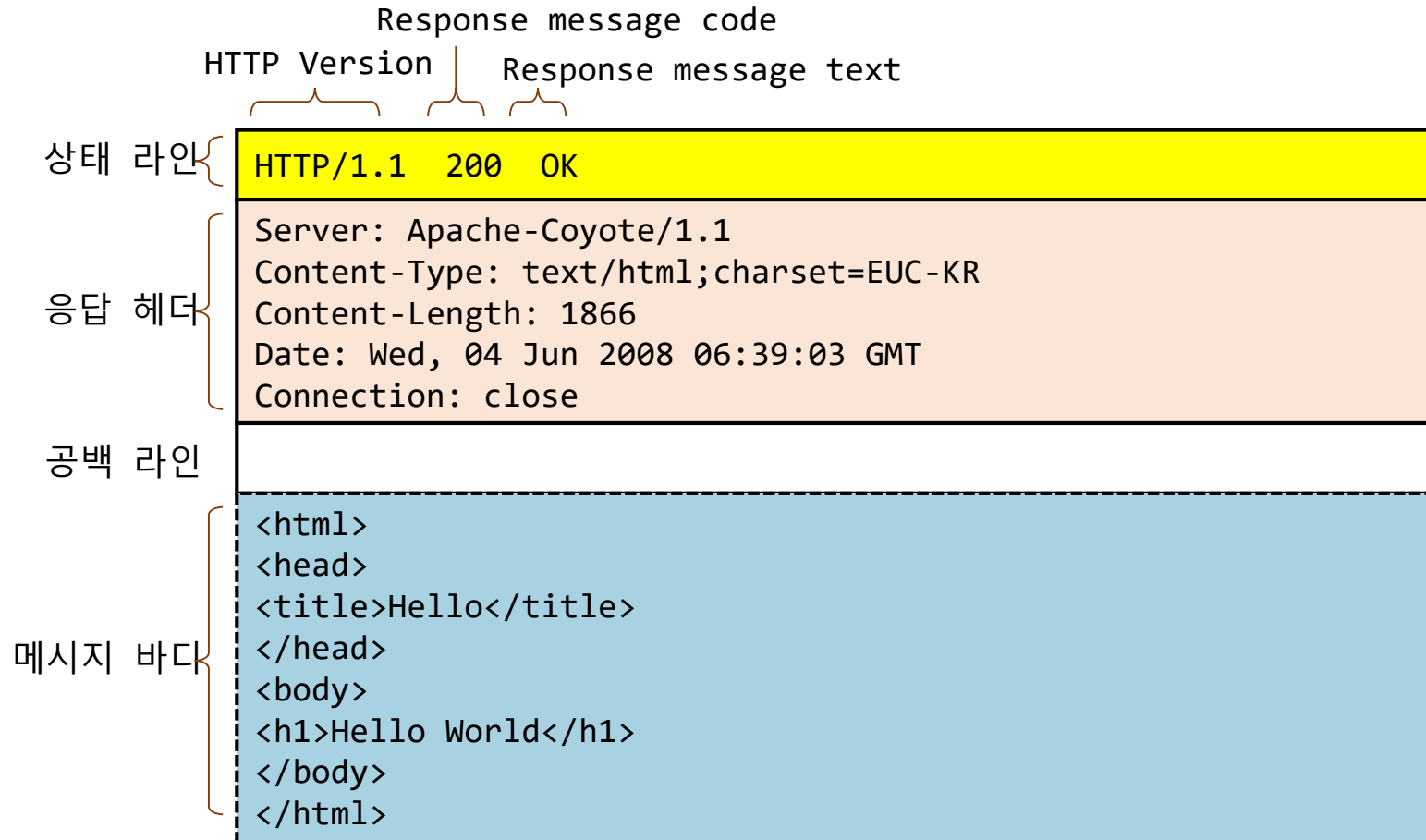
HTTP Request

- ✓ HTTP 요청은 서버의 주소와 요청 방식(GET or POST)을 포함한다.
- ✓ 그 외에 헤더정보에는 클라이언트 환경을 알 수 있는 많은 정보들이 있다.



HTTP Response

- ✓ HTTP 응답에는 응답상태와 헤더, 그리고 메시지 바디를 포함한다.
- ✓ 브라우저는 메시지 바디의 내용을 파싱하여 브라우저 화면에 보여준다.



HTTP response message code

응답코드	응답메시지(설명)
100	Continue (클라이언트로부터 일부 요청을 받았으며 나머지 정보를 계속 요청함)
200	OK (요청이 성공적으로 수행되었음)
201	Created (PUT 메서드에 의해 원격지 서버에 파일이 정상적으로 생성됨)
202	Accepted (웹서버가 명령 수신함)
203	Non-authoritative information (서버 클라이언트 요구 중 일부만 전송)
204	No content (사용자 요구를 처리하였으나 전송할 데이터가 없음)
301	Moved permanently (요청 URL 정보를 타 URL에 요청함)
302	Moved temporarily (요청 URL 정보를 타 URL에 요청함)
304	Not modified (컴퓨터 로컬의 캐시 정보를 이용함, 대개 그림 파일 등은 웹 서버에게 요청하지 않음)
400	Bad request (사용자의 잘못된 요청을 처리할 수 없음)
401	Unauthorized (인증이 필요한 페이지를 요청한 경우)
403	Forbidden (접근금지, 디렉터리 리스팅 요청 및 관리자 페이지 접근 등을 차단하는 경우)
404	Not found (요청한 페이지 없음)
405	Method not allowed (허용되지 않는 http method 사용함)
407	Proxy authentication (프락시 인증이 요구되는 경우)
408	Request timeout (요청 시간 초과)
410	Gone (영구적으로 사용 금지)
412	Precondition failed (전체 조건 실패)
414	Request-URL too long (요청된 URL 길이가 긴 경우)
500	Internal server error (내부 서버 오류)
501	Not implemented (웹 서버가 처리할 수 없음)
503	Service unavailable (서비스 제공 불가)
504	Gateway timeout (게이트웨이 시간초과)
505	HTTP version not supported (해당 http 버전 지원되지 않음)

HTML 요청

➤ HTML 요청 (HTML태그는 GET,POST 2가지만 지원함)

- ✓ 회원 목록 : GET <http://localhost:9080/members> members
- ✓ 회원 등록 양식 : GET <http://localhost:9080/members/add> addForm
- ✓ 회원 등록 : POST <http://localhost:9080/members/add> add
- ✓ 회원 조회 : GET <http://localhost:9080/members/{id}/> member
GET <http://localhost:9080/members?id=test@kh.com>
- ✓ 회원 수정 양식 : GET <http://localhost:9080/members/{id}/edit> editForm
GET <http://localhost:9080/members/edit?id=test@kh.com>
- ✓ 회원 수정 : POST <http://localhost:9080/members/{id}/edit> edit
- ✓ 회원 삭제 : GET <http://localhost:9080/members/{id}/del> del

HTTP 메서드 종류 주요 메서드

- GET: 리소스 조회
- POST: 요청 데이터 처리, 주로 등록에 사용
- PUT: 리소스를 대체, 해당 리소스가 없으면 생성
- PATCH: 리소스 부분 변경
- DELETE: 리소스 삭제

HTML 요청 & HTTP API요청

➤ HTTP API요청 (다양한 Method를 지원함)

- ✓ 회원 등록 : POST <http://localhost:9080/members>
- ✓ 회원 목록 : GET <http://localhost:9080/members>
- ✓ 회원 조회 : GET <http://localhost:9080/members/{id}>;
- ✓ 회원 수정 : PATCH <http://localhost:9080/members/{id}>
- ✓ 회원 삭제 : DELETE <http://localhost:9080/members/{id}>

HTTP 요청 데이터

➤ GET - 쿼리 파라미터

- ✓ /url?name=홍길동&age=20
- ✓ 메시지 바디 없이, URL의 쿼리 파라미터에 데이터를 포함해서 전달
- ✓ 예) 검색, 필터, 페이징등에서 많이 사용하는 방식

➤ POST - HTML Form

- ✓ content-type: application/x-www-form-urlencoded
- ✓ 메시지 바디에 쿼리 파라미터 형식으로 전달 username=hello&age=20
- ✓ 예) 회원 가입, 상품 주문, HTML Form 사용

➤ HTTP message body에 데이터를 직접 담아서 요청

- ✓ HTTP API에서 주로 사용, JSON, XML, TEXT
- ✓ 데이터 형식은 주로 JSON 사용
- ✓ POST, PUT, PATCH

HTTP 응답 데이터

➤ 단순 텍스트

- ✓ Ex) ok , nok .

➤ HTML응답

- ✓ 정적 리소스(HTML, CSS, JAVASCRIPT, IMG등)
- ✓ 동적HTML
 - Templat엔진(thymeleaf, jsp)에서 렌더링된 HTML페이지

➤ JSON 응답

- ✓ HTTP API - MessageBody

서버사이드 렌더링, 클라이언트 사이드 렌더링

➤ SSR - 서버 사이드 렌더링

- ✓ HTML 최종 결과를 서버에서 만들어서 웹 브라우저에 전달
- ✓ 주로 정적인 화면에 사용
- ✓ 관련기술: JSP, 타임리프 -> 백엔드 개발자

➤ CSR - 클라이언트 사이드 렌더링

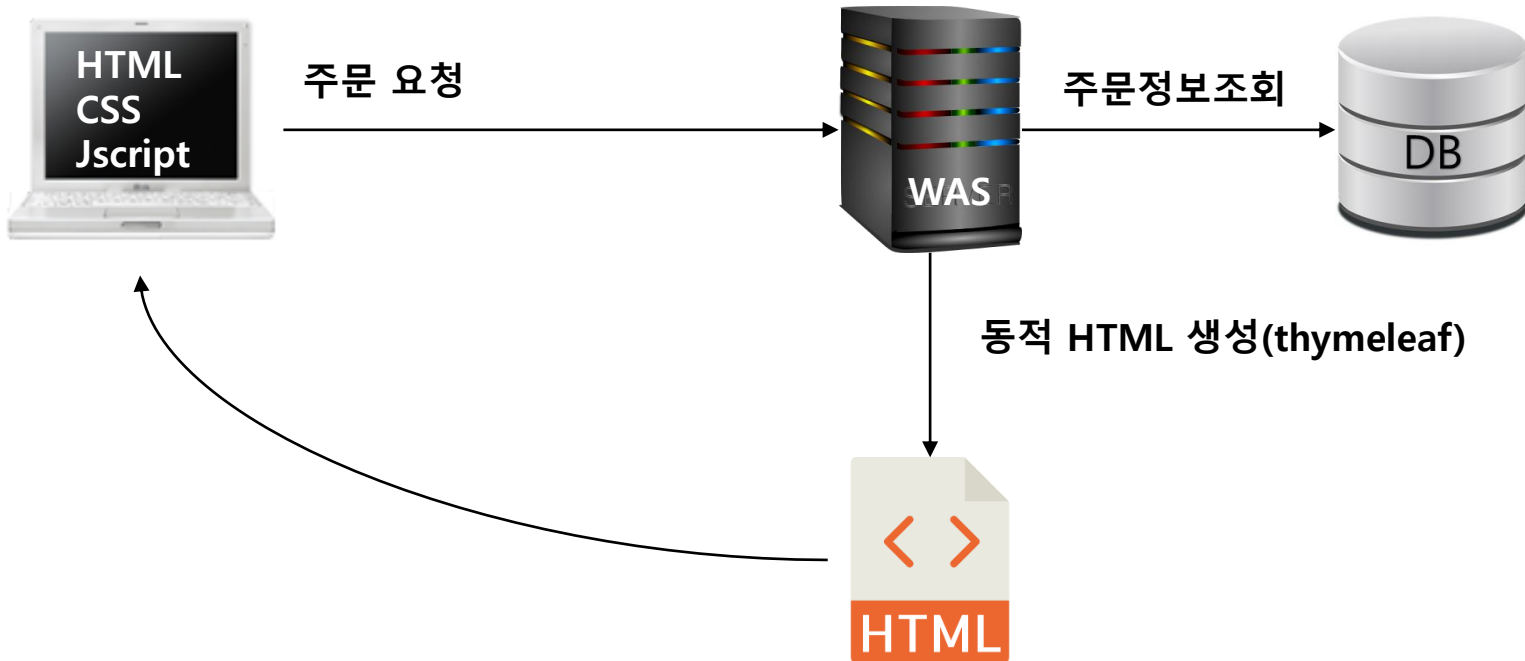
- ✓ HTML 결과를 자바스크립트를 사용해 웹 브라우저에서 동적으로 생성해서 적용
- ✓ 주로 동적인 화면에 사용, 웹 환경을 마치 앱 처럼 필요한 부분부분 변경할 수 있음
- ✓ 예) 구글 지도, Gmail, 구글 캘린더
- ✓ 관련기술: React, Vue.js -> 웹 프론트엔드 개발자

➤ 참고

- ✓ React, Vue.js를 CSR + SSR 동시에 지원하는 웹 프레임워크도 있음
- ✓ SSR을 사용하더라도, 자바스크립트를 사용해서 화면 일부를 동적으로 변경 가능

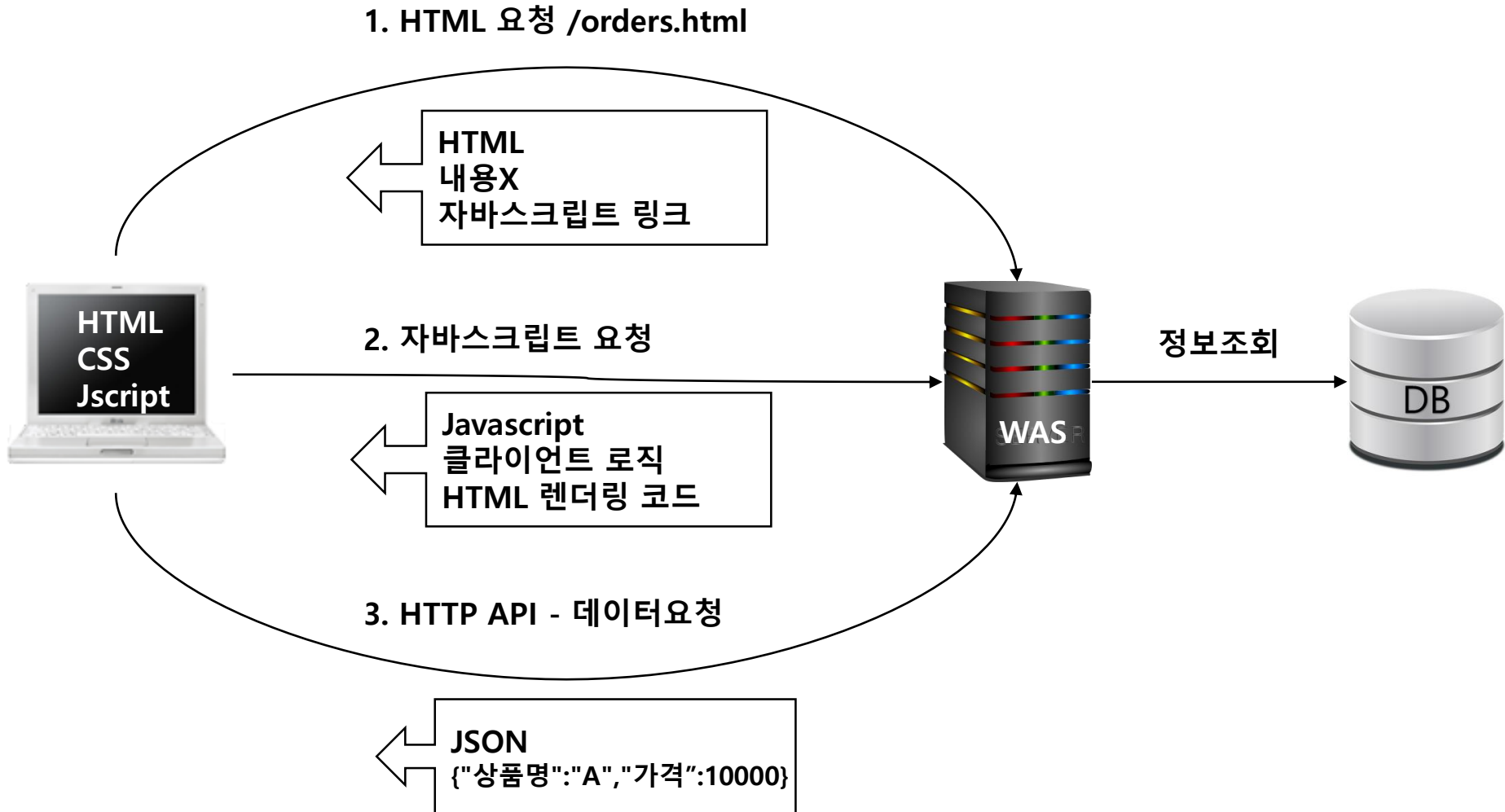
SSR - 서버 사이드 렌더링

- 뷰를 서버에서 생성한다.
- 서버에서 최종 HTML을 생성해서 클라이언트에 전달



CSR - 클라이언트 사이드 렌더링

- 뷰는 클라이언트에서 전담 한다.
- 서버는 HTTP API만 구축하고 클라이언트와 데이터만 송수신한다



Library & Framework 의 차이점

➤ 누가 코드를 제어하는가?

✓ library

- 내가 코드를 제어한다면 Library ex) JQuery, React
- 프로젝트관점에서 라이브러리는 쉽게 대체가 가능
- 조립컴퓨터, 맞춤옷

✓ framework

- 누군가 정해진 규칙을 따른다면 Framework
 - ex) spring framework, bootstrap, Django, flask, Vue.js
- 브랜드컴퓨터(완제품),기성복