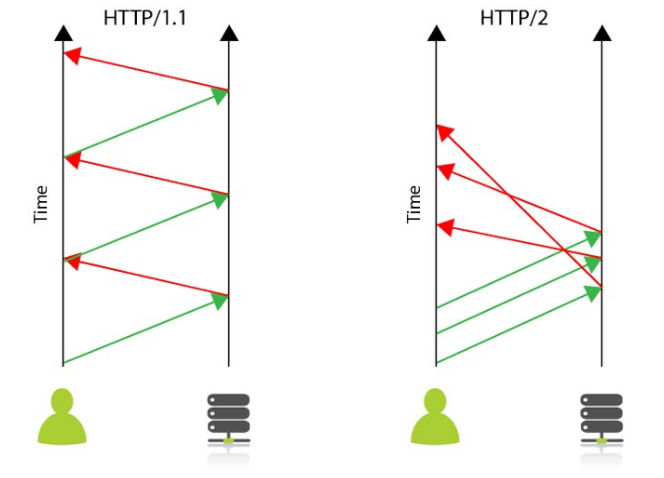
**HTTP 1.1과 2.0 차이점**



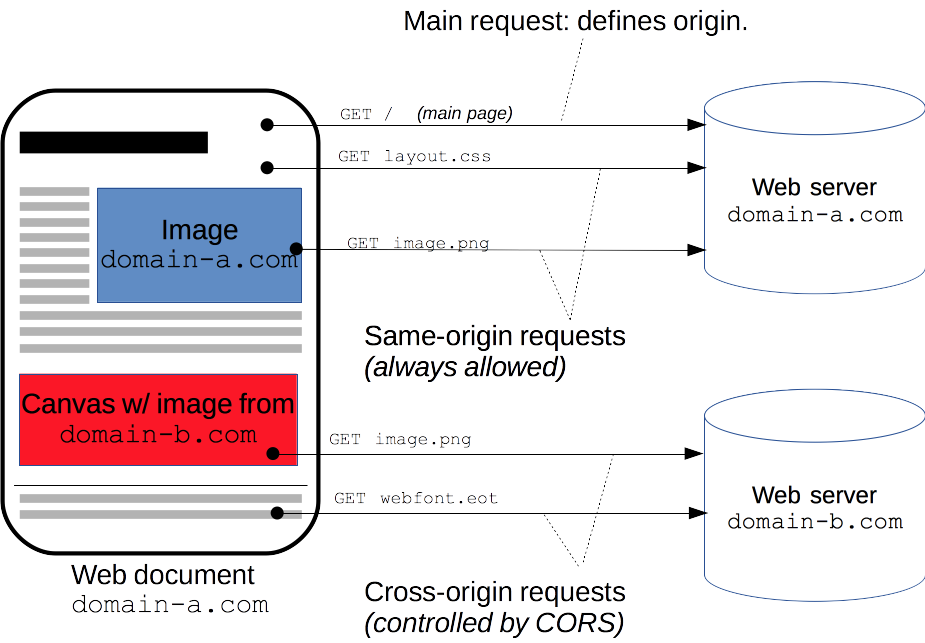
HTTP 1.1 특징 및 문제점

* Connection 한 개당 하나의 요청을 처리하도록 설계되었다.
* 요청과 응답이 순차적으로 이루어진다.
* HOL(Head Of Line) Blocking이 발생한다.
* 앞선 요청에 의해 다음 요청이 지연된다.
* 무거운 Header 구조를 가진다.
* 매 요청마다 중복된 Header 값을 전송한다.
* 서버 도메인에 관련된 Cookie 정보도 Header에 함께 포함되어 전송한다.

HTTP 2.0 특징

* Multiplexed Streams
* Stream으로 응답을 요청 순서에 상관없이 받는다.
* Stream Prioritization
* 응답에 대한 우선순위를 정해 우선순위가 높을수록 응답을 빨리 한다.
* Server Push
* 서버가 클라이언트의 요청 없이 응답을 보낸다.
* Header Compression
* HPACK 압축 방식으로 Header의 크기를 경량화 합니다.

**CORS**



Cross-Origin Resource Sharing

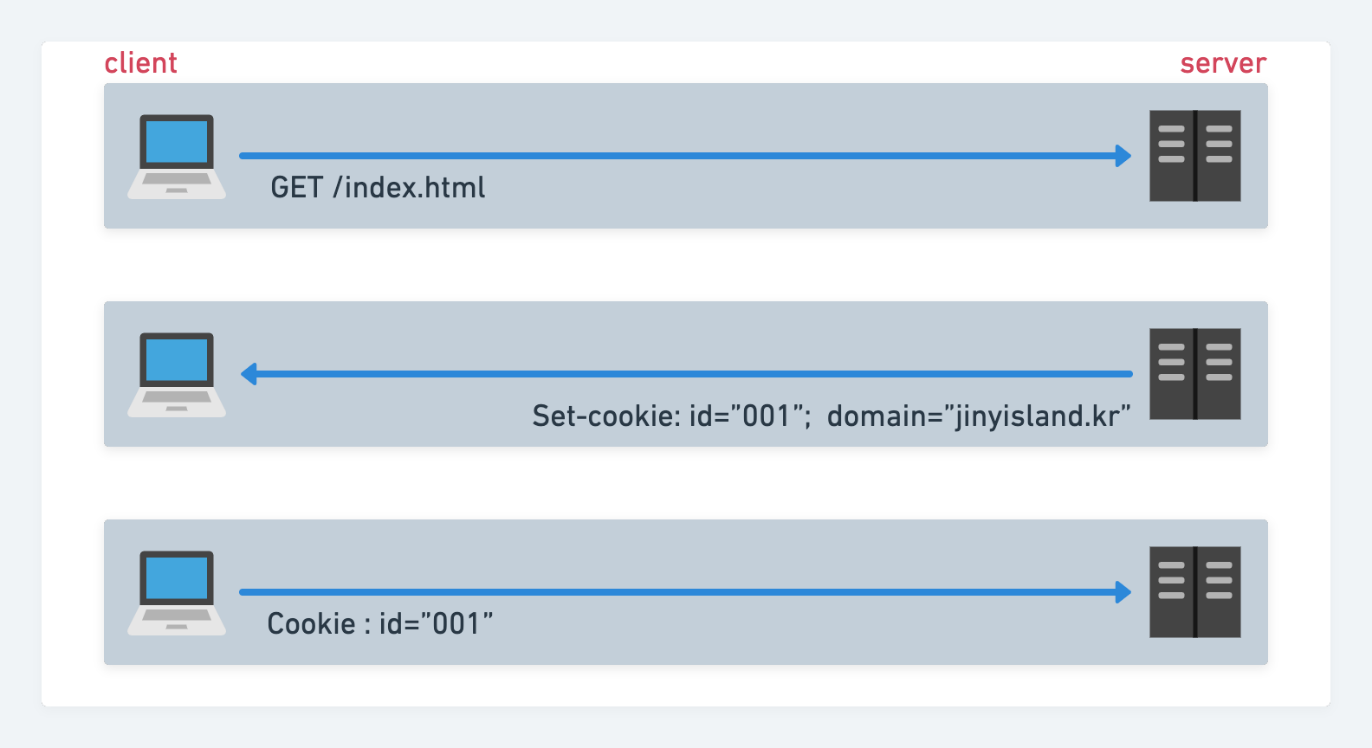
교차 출처 리소스 공유

* 자신의 출처(도메인, 프로토콜, 포트)가 다른 서버의 리소스에 대한 접근 권한을 브라우저에게 알려준다.
* 보안상의 이유로, 브라우저는 Cross-Origin HTTP 요청을 제한한다.
* CORS 체제는 브라우저와 서버 간의 안전한 교차 출처 요청 및 데이터 전송을 지원한다.

CORS는 왜 필요한가?

* CORS가 없으면 모든 곳에서 데이터를 요청할 수 있게 된다.
* 다른 사이트에서 원래 사이트를 흉내낼 수 있다.
* 세션을 탈취하여 정보를 추출할 수 있다.
* 정보를 통해 공격할 수 있다.
* 공격을 할 수 없도록 브라우저에서 보호하고, 필요한 경우만 서버와 협의하여 요청할 수 있도록 하기 위해 필요하다.

**Cookie 사용 예시**



클라이언트가 서버에서 받은 Cookie를 저장

HTTP 요청시 서버로 전달

* 사용자 로그인
* 아이디 또는 비밀번호 저장 체크
* 사용자 인증을 통해 유효 시간동안 인증 유지가 된다.
* 웹 브라우저에 저장되기 때문에 보안에 민감한 데이터는 저장하지 않는다.
* 세션 관리
* 세션은 서버의 자원을 사용한다.
* 광고 정보
* 검색 정보, 검색 날짜, 검색 시간 등의 정보를 통해 트래킹한다.
* 쇼핑몰 장바구니 기능
* 저장된 고객 정보를 통해 장바구니를 보도록 한다.
* 팝업창
* ‘오늘 더 이상 이 창을 보지 않음’ 체크
* SNS 트래킹
* 사용자 정보를 통한 친구 추천이 가능하다.