

# HTTP

20200307 김예은

HTTP이란? HTML 문서와 같은 리소스들을 가져올 수 있도록 해주는 프로토콜이다.

HTTP를 사용하여 제어 가능한 기능 목록

- 캐시, origin 제약사항을 완화하기, 인증, 프록시와 터널링 (en-US), 세션

HTTP 단계

1. TCP 연결: 요청을 보내거나 응답을 받는데 사용. 클라이언트는 새 연결을 열거나, 기존 연결을 재사용하거나, 서버에 대한 여러 TCP연결을 열 수 있다.

2. HTTP 메시지 전송: 인간이 읽을 수 있다.

3. 서버에 전송된 응답 읽음

4. 연결을 닫거나 다른 요청들을 위해 재사용

HTTP 메시지의 타입

- 요청(requests): Method(ex. GET, POST, OPTIONS, HEAD 등), Path, Version of the protocol, Headers
- 응답(responses): Version of the protocol, **Status code, Status message**, Headers
- -> HTTP 상태 코드: 특정 HTTP 요청이 성공적으로 완료되었는지 알려준다.
- 정보를 제공하는 응답, 성공적인 응답, 리다이렉트, 클라이언트 에러, 서버 에러
- ex) 100 Continue: 지금까지의 상태가 괜찮으며 클라이언트가 계속해서 요청을 하거나 이미 요청을 완료한 경우에는 무시해도 됨
- 200 OK: 요청이 성공적으로 됨
- 302 Found: 이 응답 코드는 요청한 리소스의 URI가 일시적으로 변경되었음
- 400 Bad Request: 이 응답은 잘못된 문법으로 인하여 서버가 요청을 이해할 수 없음
- 404 Not Found: 서버는 요청 받은 리소스를 찾을 수 없음

결론

HTTP는 사용이 쉬운 확장 가능한 프로토콜. 헤더를 쉽게 추가하는 능력을 지닌 클라이언트-서버 구조는 HTTP가 웹의 확장된 수용력과 함께 발전할 수 있게 한다.