

3. HTTP 정보는 HTTP 메시지에 있다

➤ Books 그림으로 배우는 HTTP & Network Basic

- 1 HTTP 메시지
- 2 리퀘스트 메시지와 리스폰스 메시지의 구조
- 3 인코딩으로 전송 효율을 높이다
 - 메시지 바디와 엔티티 바디의 차이
 - 압축해서 보내는 콘텐츠 코딩
 - 분해해서 보내는 청크 전송 코딩
- 4 여러 데이터를 보내는 멀티 파트
- 5 일부분만 받는 레인지 리퀘스트
- 6 최적의 콘텐츠를 돌려주는 콘텐츠 네고시에이션

1 HTTP 메시지

리퀘스트 측 HTTP 메시지를 리퀘스트 메시지, 리스폰스 측 HTTP 메시지를 리스폰스 메시지라고 부른다.

HTTP 메시지는 메시지 헤더와 메시지 바디로 구성되어 있고 개행 문자로 구분한다. 메시지 바디는 항상 존재하는 것은 아니다.

2 리퀘스트 메시지와 리스폰스 메시지의 구조

메시지 헤더 내부를 보면 리퀘스트 메시지는 리퀘스트 라인, 리스폰스 메시지는 상태 라인을 가지고 있으며

공통적으로 리퀘스트(리스폰스) 헤더 필드, 일반 헤더 필드, 엔티티 헤더 필드 등을 포함한다.

- 리퀘스트 라인
리퀘스트에 사용하는 메소드, 리퀘스트 URI, 사용하는 HTTP 버전이 포함된다.
- 상태 라인
리스폰스 결과를 나타내는 상태 코드와 설명, 사용하는 HTTP 버전이 포함된다.
- 헤더 필드
여러 조건과 속성 등을 나타내는 각종 헤더 필드가 포함된다.

3 인코딩으로 전송 효율을 높인다

전송할 때 인코딩을 하면 다량의 액세스를 효율 좋게 처리할 수 있지만 CPU 등의 리소스는 보다 많이 소비하게 된다.

메시지 바디와 엔티티 바디의 차이

- 메시지(message)
HTTP 통신의 기본 단위
옥텟 시퀀스(Octet sequence)로 구성되고 통신을 통해 전송된다.
- 엔티티(entity)
페이로드(payload)로 전송되는 정보
엔티티 헤더 필드와 엔티티 바디로 구성된다.

HTTP 메시지 바디는 엔티티 바디를 운반하는 역할을 한다. 기본적으로 두 개의 바디는 같지만 전송 코딩이 적용된 경우 엔티티 바디의 내용이 변화하기 때문에 메시지 바디와 달라진다.

압축해서 보내는 콘텐츠 코딩

HTTP는 용량을 줄이기 위해 파일을 압축해서 보내는 기능인 콘텐츠 코딩이 구현되어 있다. 콘텐츠 코딩은 엔티티 정보를 유지한 채 압축하는 인코딩을 말한다.

분해해서 보내는 청크 전송 코딩

HTTP 통신에서는 리퀘스트 했었던 리소스 전부에서 엔티티 바디의 전송이 완료되지 않으면 브라우저에 표시되지 않는다.

사이즈가 큰 데이터를 전송하는 경우 엔티티 바디를 분할할 수 있는데 이를 청크 전송 코딩이라고 한다.

4 여러 데이터를 보내는 멀티 파트

HTTP는 멀티파트를 지원하고 있어서 하나의 메시지 바디 내부에 엔티티를 여러 개 포함시켜 보낼 수 있다.

주로 이미지나 텍스트 파일 등을 업로드할 때 사용되며 멀티파트에는 다음과 같은 것이 있다.

- **multipart/form-data**
Web 폼으로부터 파일 업로드에 사용된다.
- **multipart/byteranges**
상태 코드 206 리스폰스 메시지가 복수 범위의 내용을 포함할 때 사용된다.

5 일부분만 받는 레인지 리퀘스트

대용량의 이미지를 다운로드 하는 중 커넥션이 끊어지게 되면 처음부터 다시 다운로드를 해야 한다.

이러한 문제를 해결하기 위해 리쥔(resume)이라는 기능이 필요하게 되었다.

리쥔을 실현하기 위해서 엔티티의 범위를 지정해서 다운로드를 해야 하는데, 이렇게 범위를 지정하여 리퀘스트 하는 것을 레인지 리퀘스트(Range Request)라고 부른다.

6 최적의 콘텐츠를 돌려주는 콘텐츠 네고시에이션

서로 다른 언어를 사용하는 브라우저가 같은 URI에 액세스할 때 각각의 언어로 웹페이지를 표시한다.

이와 같은 구조를 콘텐츠 네고시에이션(Content Negotiation)이라고 부른다.

콘텐츠 네고시에이션은 제공하는 리소스를 언어와 문자 세트, 인코딩 방식 등을 기준으로 판단한다.

콘텐츠 네고시에이션에는 다음과 같은 종류들이 있다.

- **서버 구동형 네고시에이션(Server-driven Negotiation)**
서버 측에서 리퀘스트 헤더 필드의 정보를 참고해서 자동으로 처리한다.
브라우저에서 보내는 정보를 근거로 하기 때문에 유저에게 적절한 것이 선택되었다고 할 수 없다.
- **에이전트 구동형 네고시에이션(Agent-driven Negotiation)**
브라우저에 표시된 선택지 중 유저가 수동으로 선택한다.
- **트랜스페어런트 네고시에이션(Transparent Negotiation)**
서버 구동형과 에이전트 구동형을 혼합한 것이다.

서버와 클라이언트가 각각 콘텐츠 네고시에이션을 하는 방식이다.