

8월 27일 (금)

1. Codecademy - Async JavaScript and HTTP Requests - APIs and HTTP Requests

Article 1. HTTP Requests

- Background
- What is HTTP ?
- HTTP & TCP : How it Works
- An Analogy
- What is HTTPS ?

- Background

우리의 웹 브라우저가 인터넷과 어떻게 소통하는지에 대한 기초지식을 이해해보자.

인터넷은 다른 서버에서 호스팅된 resource 들의 묶음으로 구성되어있다. resource 는 웹에서 HTML 파일, stylesheets, images, videos, script 를 포함하는 어떤 개체를 의미한다. 인터넷에서 한 콘텐츠에 접근하기 위해, 우리의 브라우저는 서버에게 원하는 리소스에 대한 요청을 보내야하고, 받아온 리소스를 우리에게 보여준다. 이러한 요청과 응답의 프로토콜을 통해 우리가 브라우저 상에서 원하는 페이지를 볼 수 있는 것이다.

- What is HTTP ?

HTTP는 "HyperText Transfer Protocol" 을 의미한다. 인터넷 상에서 요청과 응답을 구성하기 위해 사용되어진다. HTTP는 네트워크를 통해 한 지점에서 다른 지점으로 데이터가 이동

할 수 있도록 요구한다.

리소스들의 이동은 TCP(Transmission Control Protocol) 사용으로 발생하게 된다. TCP 는 너의 브라우저와 서버 사이의 채널(연결통로)을 관리한다. TCP 는 어떤 것을 다른 곳으로 보내길 원하는 하나의 컴퓨터 또는 장치들의 많은 인터넷 연결 타입을 관리하기위해 사용된다.

HTTP는 장치의 양쪽 연결이 소통하기 위해 따라야하는 커맨드 언어이다.

• HTTP & TCP : How it Works

브라우저에 www.codecademy.com과 같은 주소를 입력하면 해당 URL(Uniform Resource Locator)에 응답하는 서버에 TCP 채널을 열도록 명령한다. URL은 너에게 도달하는 방법을 설명하므로 집 주소 또는 전화번호와 같다.

이 경우 요청을 하는 컴퓨터를 클라이언트라고 한다. 내가 요청하고 있는 URL 은 서버에 속한 주소이다.

TCP연결이 되면, 클라이언트는 서버에 HTTP GET 요청을 보내 표시할 웹페이지를 검색한다. 서버가 응답을 보낸 후 TCP 연결을 닫는다. 브라우저에서 웹 사이트를 다시 열거나 브라우저가 서버에서 자동으로 무언가를 요청하면 위에서 설명한 동일한 프로세스를 따르는 새로운 TCP 연결이 생성된다. GET 요청은 클라이언트가 호출할 수 있는 HTTP메서드의 한 종류이다. 이 문서에서 일반적인 다른 항목(POST, PUT, DELETE)에 대해 알아볼 수 있을 것이다.

가장 일반적인 요청 유형인 GET 요청을 사용하여 컴퓨터(클라이언트)가 웹에서 리소스에 접근하는 방법에 대한 예를 살펴보자.

예시로 다음 상황을 가정해보자. 너는 코드카데미 홈페이지(<http://codecademy.com>)에 새로 나온 강의를 확인하길 원한다. 브라우저에 코드카데미의 URL 을 입력하면, 브라우저는 http 부분을 추출하고, 사용할 네트워크 프로토콜의 이름임을 인식한다. 그런 다음, URL에서 도메인 이름을 가져와 인터넷 도메인 이름 서버에 IP(인터넷 프로토콜) 주소를 반환하도록 요청한다.

이제 클라이언트는 목적지(코드카데미홈페이지)의 IP주소를 알게 된다. 그런 다음 지정된 http 프로토콜을 사용하여 해당 주소의 서버에 대한 연결을 연다. 호스트의 IP주소와 임의의

데이터 페이로드를 포함하는 클라이언트는 서버에 GET 요청을 보내는 것을 시작할 것이다. GET 요청에는 다음 텍스트가 포함되어 있다.

```
GET / HTTP/1.1
Host: www.codecademy.com
```

위 코드는 요청의 유형, www.codecademy.com 의 경로, HTTP/1.1 로 나누어서 식별할 수 있다.

HTTP/1.1 은 첫번째 HTTP를 수정한 것이다. **HTTP/1.1**에서 모든 리소스 요청에 서버에 대한 별도의 연결이 필요하다. **HTTP/1.1** 는 두 번 이상 연결을 사용하므로, 페이지가 검색된 후에도 추가 내용이 검색된다. 따라서

HTTP/1.1 요청은 HTTP/1.0 보다 사용하는 요청 지연시간이 짧다.

요청의 두 번째 행은 "www.codecademy.com"인 서버의 주소를 포함한다. 브라우저에게 어떤 데이터를 전송하도록 선택하는지에 따라 추가 줄이 있을 수도 있다.

서버가 요청된 경로를 찾을 수 있는 경우, 서버는 다음과 같이 헤더로 응답할 수 있다.

```
HTTP/1.1 200 OK
Content-Type: text/html
```

이 헤더 다음에는 요청된 콘텐츠가 나온다. 이 경우 www.codecademy.com 를 렌더링하는데 필요한 정보가 된다. 헤더의 첫 번째 줄인 HTTP/1.1 200 OK는 클라이언트가 통신하려는 프로토콜(HTTP/1.1)과 서버에서 리소스가 발견되었음을 나타내는 HTTP 상태 코드를 서버가 이해하는지 확인하는 것입니다. 두 번째 행인 Content-Type: text/html은 클라이언트로 보낼 내용의 유형을 표시합니다.

클라이언트가 요청한 경로를 찾을 수 없는 경우 서버는 다음과 같이 헤더로 응답합니다.

```
HTTP/1.1 404 NOT FOUND
```

- An Analogy (비유)

브라우저의 실제 작동 상태를 확인하기 어렵기 때문에, HTTP 작동 방식을 이해하기 어려울 수 있다. 비유를 통해 지금까지 배운 내용을 복습해보자.

인터넷이 마을이라고 상상해 보세요. 사용자는 고객이고 주소에 따라 연락할 수 있는 위치가 결정됩니다. Codecademy.com과 같은 시내 기업은 전송된 요청을 처리합니다. 다른 집들은 당신 같은 다른 고객들로 가득 차 있고 시내에 있는 이 업체들로부터 답변을 기다리고 있습니다. 이 마을은 빛의 속도로 움직이는 열차를 타고 이동할 수 있는 우편 배달 직원들로 구성된 정신나간 빠른 우편 서비스도 갖추고 있습니다.

조간 신문을 읽고 싶다고 가정해 보십시오. 검색하려면 HTTP라는 언어로 필요한 내용을 기록하고 로컬 메일 배달 직원에게 특정 비즈니스에서 검색하도록 요청해야 합니다. 우편 배달원은 동의하고 집과 회사 사이에 거의 즉각적으로 철도 선로(연결)를 건설하며, "TCP"라고 표시된 열차 차량을 타고 사용자가 제공한 회사 주소로 이동합니다.

회사에 도착하자마자, 그녀는 그 요청을 이행할 준비가 된 몇몇 무료 직원 중 첫 번째에게 물어봅니다. 직원이 요청한 신문 페이지를 검색하여 찾을 수 없는 경우 메일 배달원에게 다시 전달합니다.

우편 배달원은 광속 열차로 돌아와 돌아오는 길에 선로를 찢고 "404를 찾을 수 없습니다"라는 문제가 있었다고 말합니다. 당신이 쓴 글의 철자를 확인하고 나면 당신은 신문 제목 철자를 잘못 썼다는 것을 알게 됩니다. 사용자가 수정하고 수정된 제목을 메일 배달 담당자에게 제공합니다.

이번에는 우편 배달원이 업체에서 그것을 회수할 수 있습니다. 이제 다음 페이지를 읽기로 결정할 때까지 안심하고 신문을 읽을 수 있습니다. 그러면 다시 요청하여 메일 배달원에게 전달합니다.

- **What is HTTPS ?**

HTTP 요청은 특정 네트워크 연결 지점의 모든 사용자가 읽을 수 있으므로 이 프로토콜을 사용하여 신용 카드 또는 암호와 같은 정보를 전달하는 것은 좋지 않을 수 있습니다. 다행히 많은 서버가 HTTP Secure의 줄임말로 HTTPS를 지원하므로 사용자가 보내고 받는 데이터를 암호화할 수 있습니다. HTTPS에 대한 자세한 내용은 Wikipedia에서 확인할 수 있습니다.

HTTPS는 중요한 정보 또는 개인 정보를 웹 사이트에 전달하거나 웹 사이트에서 수신할 때 사용해야 합니다. 그러나 서버를 설정하는 것은 기업에 달려 있습니다. HTTPS를 지원하려면 기업이 인증 기관의 인증서를 신청해야 합니다.

Article 2. An Overview of HTTP (Advanced)

- HTTP 기반 시스템의 구성요소
- HTTP의 기초적인 측면
- HTTP로 제어할 수 있는 것
- HTTP 흐름
- HTTP 메시지
- HTTP 기반 API
- 결론

출처: <https://developer.mozilla.org/ko/docs/Web/HTTP/Overview>

오늘의 단어

- Analogy : 유사성, 비유, 유추, 유사한
- juncture : 시점, 단계