

# 렌더링

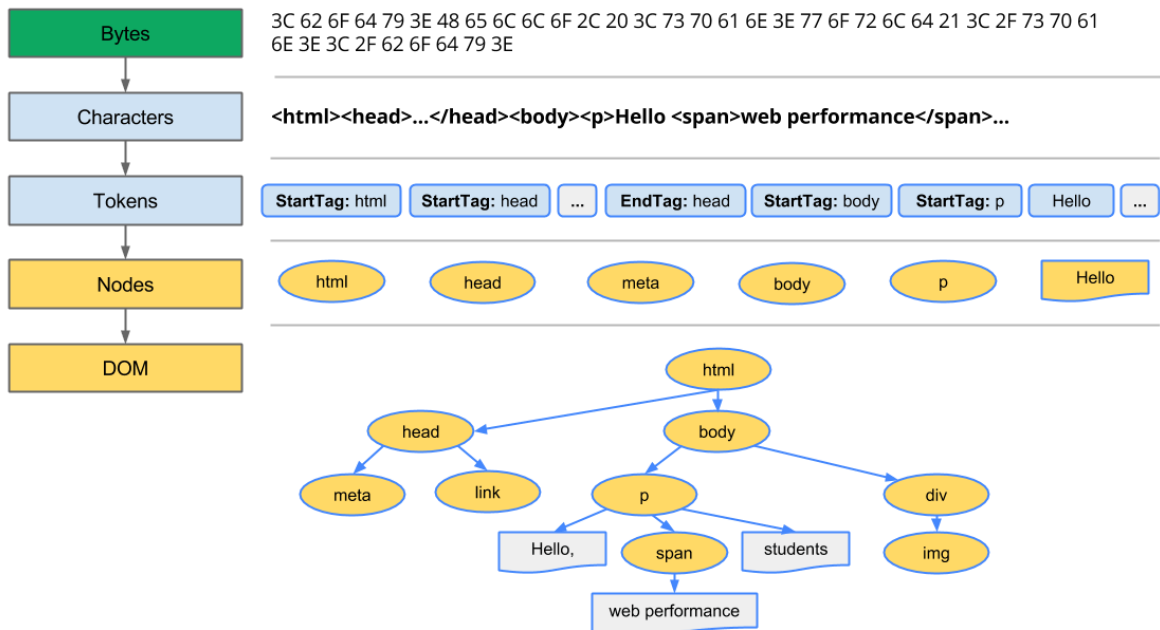
## 정의

- 브라우저로부터 받은 페이지에 대한 정보를 나타내는 현상

## 요약

- 바이트 → 문자 → 토큰 → 노드 → 객체 모델.
- HTML 마크업은 DOM(Document Object Model)으로 변환되고, CSS 마크업은 CSSOM(CSS Object Model)으로 변환됩니다.
- DOM 및 CSSOM은 서로 독립적인 데이터 구조입니다.
- Chrome DevTools Timeline을 사용하면 DOM 및 CSSOM의 생성 및 처리 비용을 수집하고 점검할 수 있습니다.

## 방법(DOM)



1. **변환:** 브라우저가 HTML의 원시 바이트를 디스크나 네트워크에서 읽어와서, 해당 파일에 대해 지정된 인코딩(예: UTF-8)에 따라 개별 문자로 변환합니다.
2. **토큰화:** 브라우저가 문자열을 [W3C HTML5 표준](#)에 지정된 고유 토큰으로 변환합니다(예: `<`, `>` 및 꺾쇠괄호로 묶인 기타 문자열). 각 토큰은 특별한 의미와 고유한 규칙을 가집니다.
3. **렉싱:** 방출된 토큰은 해당 속성 및 규칙을 정의하는 '객체'로 변환됩니다.
4. **DOM 생성:** 마지막으로, HTML 마크업이 여러 태그(일부 태그는 다른 태그 안에 포함되어 있음) 간의 관계를 정의하기 때문에 생성된 객체는 트리 데이터 구조 내에 연결됩니다. 이 트리 데이터 구조에는 원래 마크업에 정의된 상위-하위 관계도 포함됩니다. 즉, `HTML` 객체는 `body` 객체의 상위이고, `body`는 `paragraph` 객체의 상위인 식입니다.

# CSSDOM

- CSS만을 다루고 수정함으로서 효율적인 수정이 가능함

## 렌더링 트리

DOM과 CSSOM을 합치는 행위

1. HTML 마크업을 처리하고 DOM 트리를 빌드합니다.
2. CSS 마크업을 처리하고 CSSOM 트리를 빌드합니다.
3. DOM 및 CSSOM을 결합하여 렌더링 트리를 형성합니다.
4. 렌더링 트리에서 레이아웃을 실행하여 각 노드의 기하학적 형태를 계산합니다.
5. 개별 노드를 화면에 페인트합니다.

여기에 표시된 데모 페이지는 간단해 보일 수 있지만, 이 페이지에도 꽤 많은 작업이 필요합니다. DOM 또는 CSSOM이 수정된 경우, 화면에 다시 렌더링할 필요가 있는 픽셀을 파악하려면 이 프로세스를 다시 반복해야 합니다.

**\*주요 렌더링 경로를 최적화하는 작업\*** 은 위 단계에서 1단계~5단계를 수행할 때 걸린 총 시간을 최소화 하는 프로세스입니다.

