

크롤링: 웹 콘텐츠 수집

문과생을 위한 AI 활용 파이썬 코딩_4주차

HTML

- 웹 페이지의 구조와 내용 정의
- 태그(<tag>) 사용
- 웹 페이지의 뼈대 역할
- 예: <h1>, <p>, <div>,

CSS

- 디자인과 레이아웃 담당
- HTML 요소의 스타일 지정
- 색상, 크기, 배치, 여백 등 제어
- 예: color, font-size, margin

JavaScript

- 웹 페이지에 동적 기능 추가
 - 사용자 상호작용 처리
 - 데이터 조작 및 서버 통신
 - 예: 버튼 클릭, 폼 제출, 애니메이션
-

HTML(HyperText Markup Language)

- 웹 페이지의 뼈대와 구조를 만드는 마크업 언어
 - 텍스트, 이미지, 링크, 멀티미디어 같은 콘텐츠를 브라우저에 표시하기 위한 규칙 집합
 - 태그(tag)와 요소(element)로 구성되는 문서 구조
 - 브라우저가 HTML 코드를 해석해 화면에 출력하는 방식
 - 시각적 디자인은 CSS, 동적인 기능은 JavaScript와 함께 구현되는 웹 기술의 기초
-

HTML 파일의 구성 요소: 태그(Tag)와 요소(Element)

태그(Tag)

- HTML 문서를 구성하는 기본 단위
- <태그명> 형태로 시작 태그를 쓰고, </태그명> 형태로 종료 태그를 작성하는 규칙
- 웹 브라우저가 콘텐츠의 구조와 의미를 이해하도록 지시하는 역할

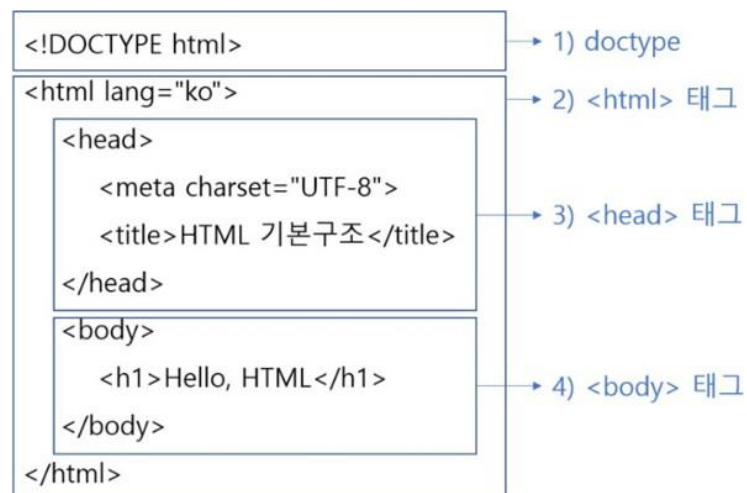
요소(Element)

- 시작 태그, 종료 태그, 그리고 그 사이의 콘텐츠를 모두 포함하는 전체 구조
 - 웹 문서 안에서 하나의 독립적인 구성 단위를 형성하는 역할
-



HTML 태그 구조

- 시작 태그: 태그 이름과 속성으로 구성되는 웹 요소의 시작 부분
- 태그 이름: 요소의 의미와 역할을 나타내는 이름
- 속성 이름: 추가 정보를 제공하기 위한 식별자
- 속성 값: 속성 이름에 할당되는 실제 값
- 내용: 브라우저에 표시되는 텍스트나 요소의 본문
- 종료 태그: 해당 요소의 끝을 알리는 표시



HTML 주요태그

- div와 span 태그는 비슷해 보이지만 div는 블록(block), span은 인라인 (inline) 형태라는 점이 중요.
- img는 시각 콘텐츠, a는 하이퍼링크를 담당.
- 태그는 속성을 통해 의미와 동작을 확장.

태그	설명	사용 예
h1~h6	제목을 입력할 때 사용 (h1이 가장 큰 제목, h6이 가장 작은 제목)	제일 큰 제목
p	하나의 문장을 입력할 때 사용	문장
div	박스형태의 구역 설정 (block 요소) (다른 태그들이 div 안에 모여있게 됨)	 제목 문장
span	줄 형태의 구역 설정 (inline 요소) (독립적으로 사용하지 않고 p태그 안에 span 태그가 들어감)	이렇게 span요소로 텍스트의 일부분 만 스타일을 적용할 수 있음
img	이미지와 관련된 태그 (속성명은 src, 속성값은 "이미지의 url 주소") 종료 태그(/img)가 없는 빈 태그(empty tag)	
a	하이퍼링크를 추가할 때 사용	HTML 링크
ul	unordered list : 기호로 된 리스트 li 태그가 하위 태그로 사용되어 내용을 채움	 - 기호 - 기호 - 기호
ol	ordered list : 순서가 있는 리스트 li 태그가 하위 태그로 사용되어 내용을 채움	 1번 2번 3번

CSS(Cascading Style Sheets)

- HTML로 작성된 웹페이지에 시각적 디자인을 적용하는 언어
 - HTML이 구조와 콘텐츠를 정의한다면, CSS는 색상, 글자 크기, 폰트, 배경색, 여백, 레이아웃 등 외형을 정의하는 역할
 - 특정 태그나 요소를 선택자(selector)로 지정하고, 그 안에 디자인 속성과 값을 담은 선언부(declaration block)를 작성
 - 웹페이지의 표현(스타일)과 구조(HTML)를 분리함으로써 코드의 재사용성과 유지보수성을 높이는 역할
-



CSS 구조

- 선택자(Selector): 스타일을 적용할 HTML 요소를 지정하는 부분
EX) a → 모든 하이퍼링크(a 태그)에 스타일 적용
- 선언부(Declaration Block): 중괄호 { } 안에 들어 있으며 스타일 속성과 값을 정의하는 부분
- 속성(속성명, Property): 어떤 스타일을 변경할지 결정하는 항목
EX) background-color, font-size, color
- 값(속성값, Value): 속성에 적용할 구체적인 값
EX) yellow, 16px, red

웹 크롤링할 때 CSS를 알아야 하는 이유

HTML은 구조, CSS는 이름표

- 웹페이지는 HTML로 뼈대를 만들고, CSS로 디자인을 입힘.
- CSS는 디자인할 부분을 구분하려고 클래스(class), 아이디(id) 같은 이름표를 붙임.

크롤러는 이 이름표를 따라감

- 크롤러가 HTML 코드를 그냥 읽으면 글자 덩어리밖에 안 보임.
- 그런데 CSS 덕분에 "이 부분은 뉴스 제목", "이건 날짜"처럼 위치와 역할이 정리돼 있음.
- 크롤러는 이 이름표를 이용해 필요한 정보만 골라서 수집함.

예를 들어

- CSS에 class="title"이 붙어 있으면 크롤러는 title 클래스를 찾아서 뉴스 제목만 모을 수 있음.
 - CSS에 class="date"가 있으면 날짜만 깔끔하게 모을 수 있음.
-

ID

- 특정 요소 하나를 고유하게 구분하기 위해 사용
- 한 문서 안에서 같은 ID 이름은 한 번만 사용 가능
- 주로 한 페이지에서 유일하게 존재하는 영역이나 특별히 개별 스타일이 필요한 요소에 적합
- CSS에서 ID 선택자는 #기호를 붙여서 선택
- 자바스크립트 같은 스크립트 언어에서도 고유 요소를 찾는 기준으로 많이 활용

CLASS

- 여러 요소를 묶어서 같은 스타일을 적용하기 위해 사용
 - 한 문서 안에서 같은 Class 이름을 여러 번 반복해서 사용 가능
 - 비슷한 성격이나 디자인을 공유하는 여러 요소에 동일한 스타일을 적용할 때 사용
 - CSS에서 Class 선택자는 .기호를 붙여서 선택
 - 한 요소에 여러 개의 Class를 동시에 부여해 다양한 스타일을 조합할 수도 있음
-

CSS 선택자와 웹 크롤링의 관계

CSS 선택자의 본래 역할

- CSS 선택자는 원래 웹페이지에서 특정 HTML 요소를 식별하고 스타일을 적용하기 위해 만들어짐.
- 예를 들어, #id명, .클래스명, 태그명 같은 규칙을 사용해 원하는 요소를 골라 스타일을 부여함.

웹 크롤링에서의 활용

- 웹 크롤링에서는 스타일을 입히는 대신, 데이터를 가져오기 위해 요소를 식별하는 데 같은 규칙을 사용함.
- 웹페이지는 수많은 태그와 중첩 구조를 갖는데, CSS 선택자를 이용하면 원하는 정보가 있는 요소만 정확히 선택할 수 있음.

Class 선택자

- 여러 요소를 묶어서 같은 스타일을 적용하기 위해 사용
 - 한 문서 안에서 같은 Class 이름을 여러 번 반복해서 사용 가능
 - 비슷한 성격이나 디자인을 공유하는 여러 요소에 동일한 스타일을 적용할 때 사용
 - CSS에서 Class 선택자는 .기호를 붙여서 선택
 - 한 요소에 여러 개의 Class를 동시에 부여해 다양한 스타일을 조합할 수도 있음
-

분류	설명	예
태그 선택	특정 태그를 선택	div --> <div> 태그를 선택
아이디 선택	id='속성값'인 태그를 선택	#query --> id의 속성값이 query인 태그 선택
클래스 선택	class='속성값'인 태그를 선택	.title --> class의 속성값이 title인 태그 선택
태그+아이디 선택	특정 태그 중 id가 '속성값'인 태그를 선택	input#query --> input 태그 중, id의 속성값이 query인 태그 선택
태그+클래스 선택	특정 태그 중 class가 '속성값'인 태그를 선택	p.title --> p 태그 중, class의 속성값이 title인 태그 선택

CSS selector 사용법

- CSS 선택자를 사용하여 HTML 요소를 식별하고, 해당 요소에 스타일 규칙을 적용
- 어떤 요소를 선택할지에 따라 태그, 아이디, 클래스 그리고 이들을 조합한 선택 방법이 존재함
- 속성이나 부모-자식 관계를 활용해 더 세밀하게 특정 요소만 지정할 수 있음
- 웹 크롤링 시 불필요한 데이터는 건너뛰고 원하는 정보만 빠르고 정확하게 추출하기 위해 이 선택 규칙이 중요함
- EX) soup.select('p') : HTML 문서 안의 모든 <p> 문단 요소를 선택/ soup.select('.title') : 클래스 이름이 title인 모든 요소를 선택/ soup.select('#main') : 아이디가 main인 단일 요소를 선택

Q&A