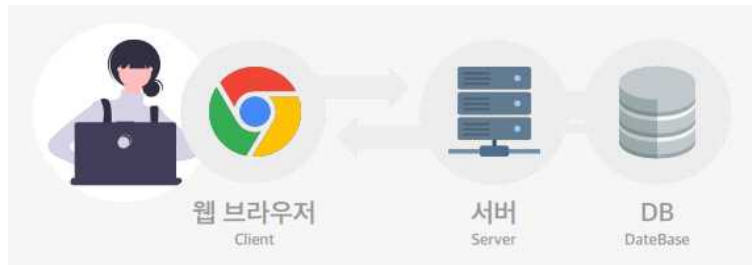


## Ch00. 기본기 다지기

### 1. 웹사이트의 기본원리와 개발영역

#### ① 웹사이트 기본 원리



#### ② 프론트엔드 개발



#### ③ 확장된 프론트엔드 개발



#### ④ 백엔드 개발



## 2. HTML, CSS 그리고 JS? 각 언어의 역할 이해

### ① HTML(Hyper Text Markup Language)



- 인터넷 브라우저 상에 보이는 웹페이지의 제목, 문단, 표, 이미지, 동영상 등 웹의 구조를 담당

### ② CSS(Cascading Style Sheets)

- 실제 화면에 표시되는 방법(색상, 크기, 폰트, 레이아웃 등)을 지정해 콘텐츠를 꾸며주는 시각적인 표현(정적)을 담당.



### ③ JS(JavaScript – ECMA Script)

- 콘텐츠를 바꾸고 움직이는 등 페이지를 동작시키는 동적 처리를 담당



## Ch01. vsCode 사용법

### 1. Project : vs code는 하나의 폴더를 하나의 Project로 인식합니다.

- 폴더의 이름은 영문, 소문자, 공백은 사용하지 않습니다.(공백대신 underscore 『 \_ 』 사용)

### 2. 파일 생성하기 : 두개의 파일을 생성해 봅니다.

- ① index.html, main.css
- ② css 라는 이름의 폴더를 생성한 후 main.css → css 폴더로 이동합니다.

### 3. vsCode 한글적용하기

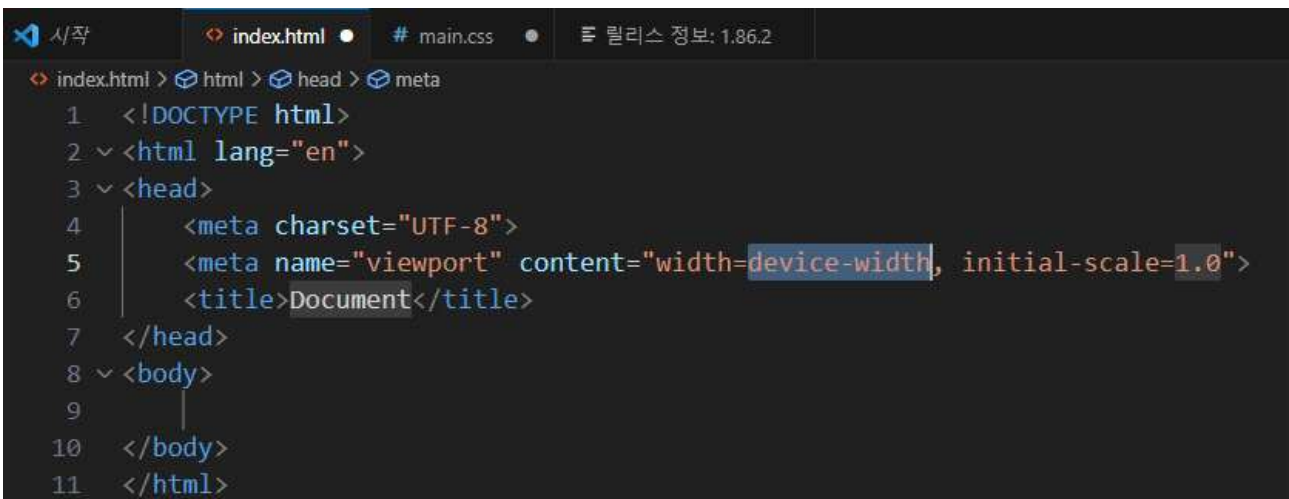
- ① 좌측 Activity Bar → 확장(Extension) → 검색 : korean
- ② 우측 korean language pack for ... 선택 후 설치



- ③ vsCode Editor를 종료 후 재시작합니다.

### 4. 새 파일 만들고 HTML Layout 적용하기

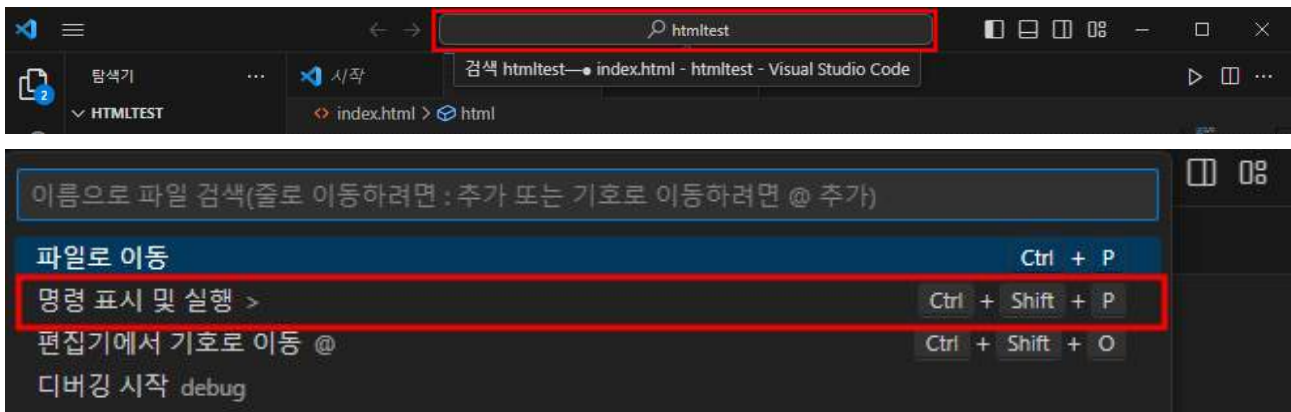
- ① 새 문서 → index.html
- ② 우측 Code Window → ! + tab : HTML 기본 구조가 만들어 짐.



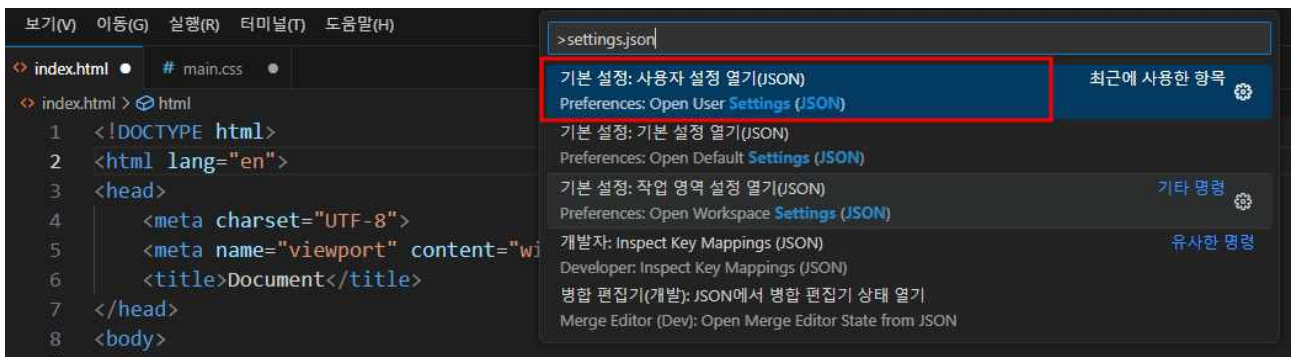
- 문서 기본 언어를 한글로 설정 : <html lang="en"> → "ko" 로 변경

## 5. 한번에 ko 로 바꾸기

① 검색에서 명령 표시 및 실행 > ctrl + shift + P 선택



## ② settings.json 입력



### ⑤ 설정 값 변경하기



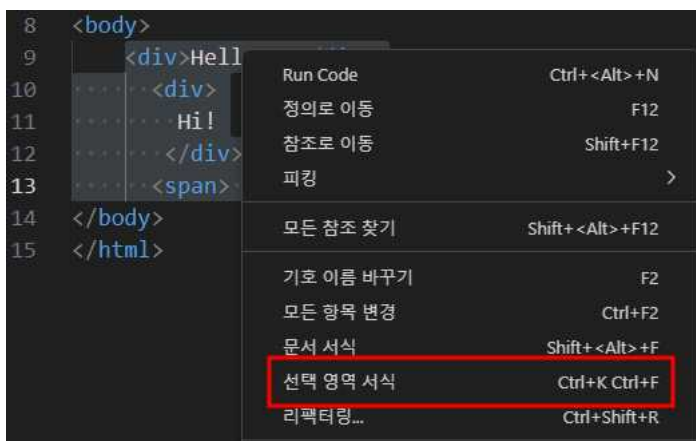
## ⑥ vsCode 재시작 하기

## 6. 코드 정렬하기

① 정렬이 제대로 되지 않은 코드

```
8 <body>
9     <div>Hello~~ </div>
10 <div>
11     Hi!
12 </div>
13 <span> </span>
14 </body>
15 </html>
```

② 블록 설정 후 → 마우스 오른쪽 클릭

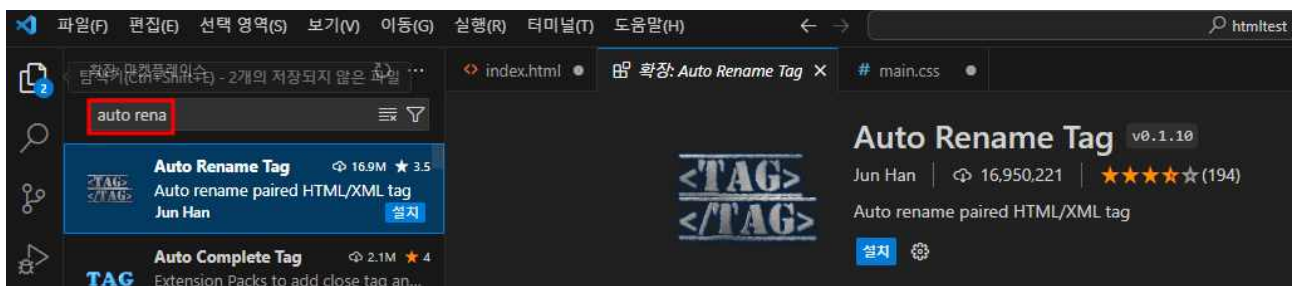


→ 단축키 : Ctrl + K + F

## 7. 태그 이름 한꺼번에 바꾸기

① <div> 태그를 한꺼번에 <span> 태그로 바꾸자 → 현재는 쌍으로 이루어진 태그의 수정이 필요하면 앞 뒤 태그를 모두 수정해 주어야 한다.

② 확장 → 검색 → auto rename

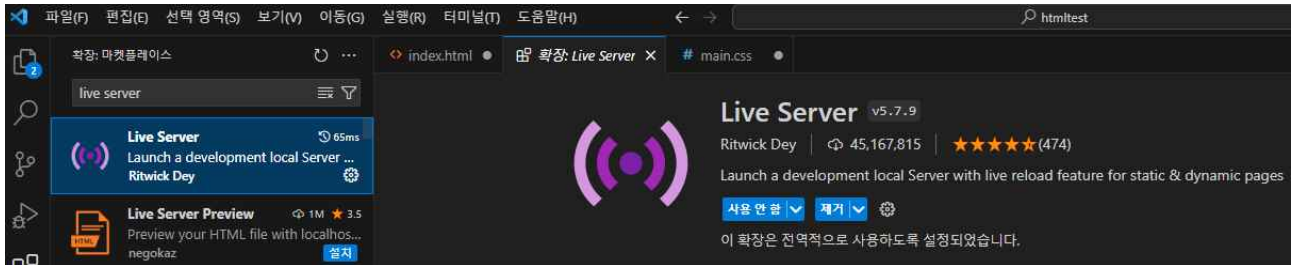


③ 쌍이 되는 태그가 같이 변경되고 편집.

## 8. 작성한 내용 브라우저로 확인하기

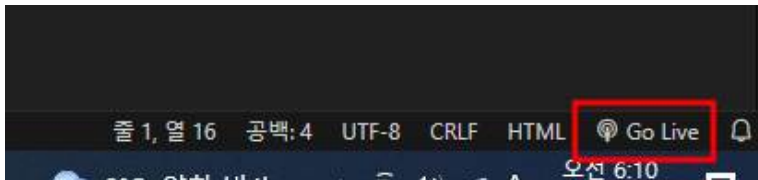
① 프로그램 개발 시 마다 변경한 파일의 내용이 즉시 반영되어 결과를 브라우저에서 확인 가능하게 함.

② 확장 → 검색 → live server → 설치

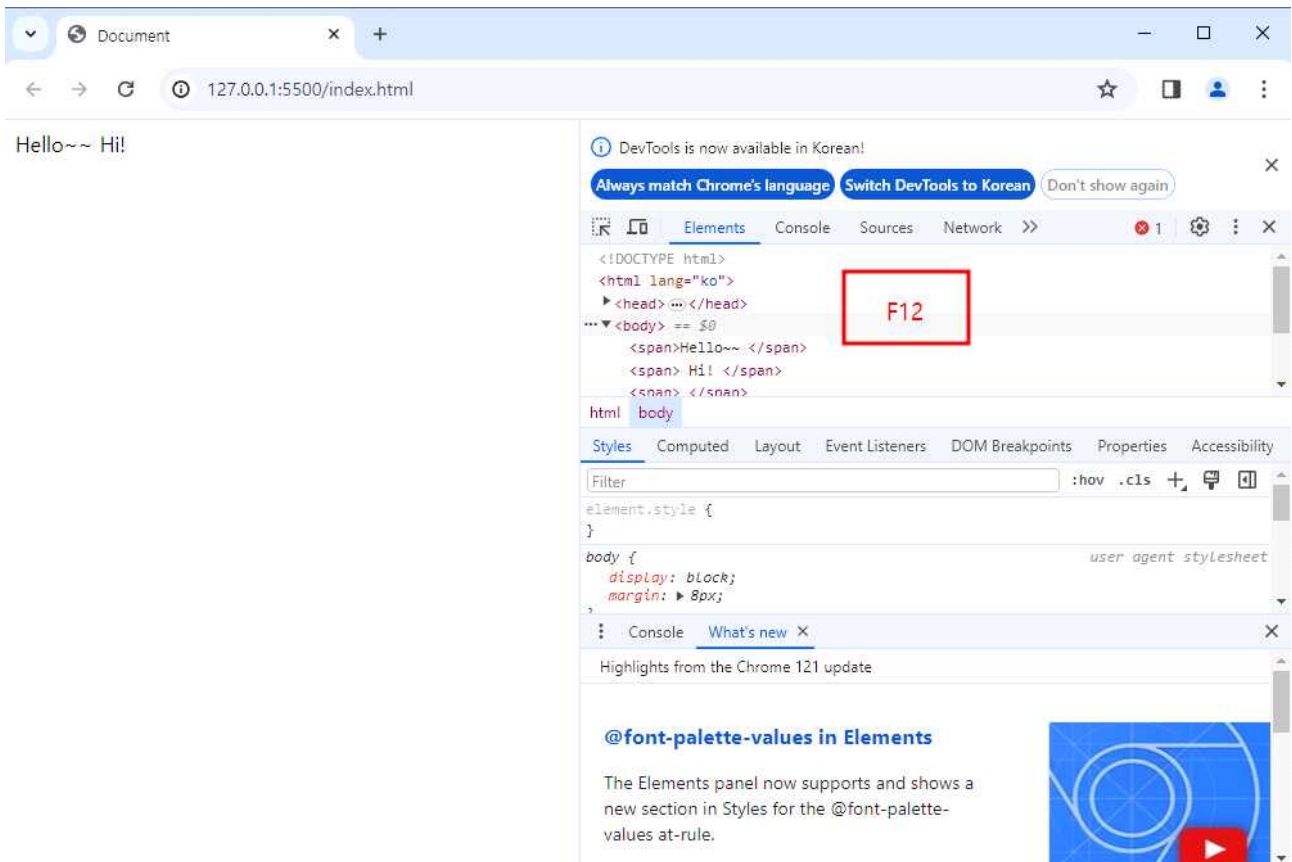


③ 실행하기

- 마우스 우클릭 → Open with Live Server (Alt + L + O)
- 화면 우측 하단 아이콘 사용



④ 브라우저에서 개발자 모드 들어가기 → F12



⑤ Live Server 종료하기

- 마우스 우클릭 → Open with Live Server (Alt + L + O)
- 화면 우측 하단 아이콘 사용



## 9. 에디터 Tab Size 조정하기

- ① 현재 기본 값은 4로 설정되어 있음(space 4칸)
- ② 코드 양이 많아 질 경우 가독성이 떨어 질 수 있음.
- ③ Settings.json → "editor.tabSize" : 2



## Ch02. HTML 구조와 파일의 연결

### 1. HTML 구조

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Document</title>
7 </head>
8 <body>
9     <h1>Hello world</h1>
10 </body>
11 </html>
```

#### ① 문서 형식 선언, **<!DOCTYPE html>**

- 문서 형식 선언, 줄여서 DOCTYPE은 문서의 내용이 시작되기 전에 해당 문서가 어떤 마크업 언어 형식으로 작성되었는지를 명시하는 역할을 합니다.
- DOCTYPE의 뒤에 html이라고 쓰여 있는 것은 '이 문서는 HTML5로 작성 되었습니다'라는 뜻입니다. 과거 HTML의 이전 버전들의 문서 형식은 비교적 표기법이 복잡한데 반해, HTML5는 웹 표준 기술로써 이전 버전들을 수용하고 계속적으로 발전하고 있으므로 표기법이 무척 단순합니다.
- 문서 형식 선언은 웹 브라우저가 해당 문서를 해석할 때나 문서의 유효성 검사 등을 할 때 참고 사항 역할을 합니다.

#### ② 문서의 시작과 끝, **<html lang="ko">**

- 문서 형식 선언 이후 실제 문서의 내용을 표시하는 태그가 바로 <html> 태그입니다.
- 여는 태그가 문서의 시작을, 닫는 태그가 문서의 끝을 의미합니다.
- 이 태그에는 선택적으로 lang이라는 속성을 추가할 수 있는데, 이는 문서의 주요 언어를 표기하기 위해 추가하는 속성입니다. 한국어를 사용한 경우에는 위 예시에서 표기한 것처럼 "ko"를 입력하고, 영어를 뜻하는 "en"이나 일본어를 의미하는 "ja" 등 다양한 언어 코드를 속성값으로 입력할 수 있습니다.
- <html> 태그의 안에는 실제 문서의 내용들이 포함되는데, 내용으로는 문서가 표시할 콘텐츠 뿐만 아니라 문서에 대한 정보(설정값)도 함께 포함될 수 있습니다.

#### ③ 문서의 정보, **<head>**

- <html> 태그의 안에는 <head>와 <body>라는 두 개의 하위 태그가 자리하고 있습니다.
- <head>의 역할은 브라우저에게 문서의 정보를 전달하는 것입니다. <head>의 안에 작성되는 태그들은 웹 브라우저 화면에 표시될 콘텐츠를 나타내는 것은 아니지만, 웹페이지의 품질에 영향을 주는 중요한 정보들입니다.

#### ④ 인코딩 방식 설정, **<meta charset="utf-8">**

<meta> 태그는 문서의 키워드 또는 설정 등 문서와 관련된 여러 가지 항목들을 지정할 때 사용하는 태그인데, 그 중 대표적인 항목이 바로 문서의 인코딩 방식을 설정해주는 문자셋(character set) 설정입니다.



다. 세상에는 한국어, 영어, 일본어 등 다양한 언어 및 문자가 존재합니다. 그리고 컴퓨터가 이를 해석하기 위해서는 각 문자에 맞는 인코딩 방식을 사용해야 합니다. <meta> 태그의 문자셋 설정을 이용해 인코딩 방식을 utf-8 방식으로 설정하면, 한글을 비롯한 세상의 모든 언어를 표시할 수 있습니다.

⑤ 문서의 제목, **<title>**

- <title> 태그는 문서의 제목을 입력하는 태그입니다.
- 여기에 입력한 제목은 웹 브라우저의 탭 메뉴에 표시됩니다.
- 웹페이지를 방문하는 웹 이용자는 탭 메뉴에 표시되는 텍스트를 보고 웹페이지가 제공하는 콘텐츠에 대해 짐작할 수 있으므로 이는 무척 중요한 정보입니다.
- 또한 문서의 제목은 검색 엔진이 웹페이지를 탐색할 때에도 영향을 끼칩니다.

⑥ 화면에 표시될 콘텐츠, **<body>**

- <body> 태그는 웹 브라우저 화면에 표시될 콘텐츠를 입력하는 태그입니다. 이 안에는 텍스트나 이미지, 미디어 요소 등 다양한 콘텐츠들을 포함하여 웹페이지를 풍성하게 꾸밀 수 있습니다.

## 2. CSS, js 파일을 HTML에 연결하기

① index.html과 연결할 main.css 파일 생성하기

② main.css에 입력하기

```
h1{  
  color : red;  
}
```

③ main.css를 index.html에 연결하기

```
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Document</title>  
  <link rel="stylesheet" href="/css/main.css">  
  <script defer src="/js/main.js"></script>  
</head>
```

④ js 폴더 만들고 main.js 파일 준비하기

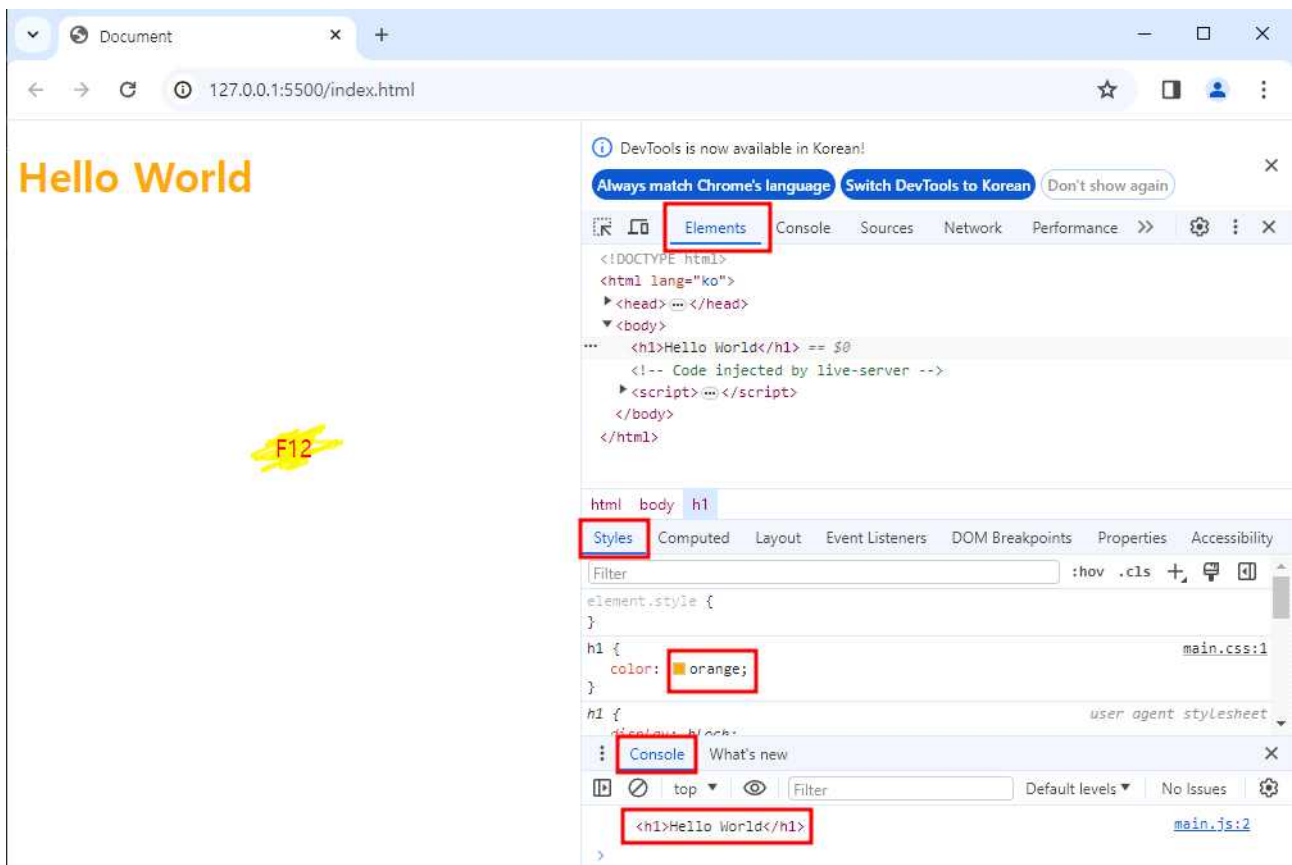
```
const el = document.querySelector('h1')  
console.log(el)
```

⑤ main.js를 index.html에 연결하기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/css/main.css">
  <script defer src="/js/main.js"></script>
</head>
```

- defer : html body 부분 처리를 지원하는 코드로 페이지가 모두 로드된 후 해당 외부 스크립트가 실행되어야 함을 의미함.

⑥ 테스트 하기



### 3. 웹페이지를 나누고 연결하기

#### ① index.html과 연결할 about.html 파일 생성하기

```
<body>
  <h1>About</h1>
</body>
```

#### ② index.html에서 about.html 파일 연결하기

```
<body>
  <h1>About</h1>
  <a href="/about.html">About...</a>
</body>
```

#### ③ about.html 에서 index.html 파일로 연결하기

```
<body>
  <h1>About</h1>
  <a href="/index.html">Main...</a>
</body>
```

### 4. 코드 주석처리하기(Comment) - 단축 키 : 『Ctrl + /』

#### ① HTML 파일 내 주석처리

```
<body>
  <!--
    <h1>About</h1>
    <a href="/index.html">Main...</a>
  -->
</body>
```

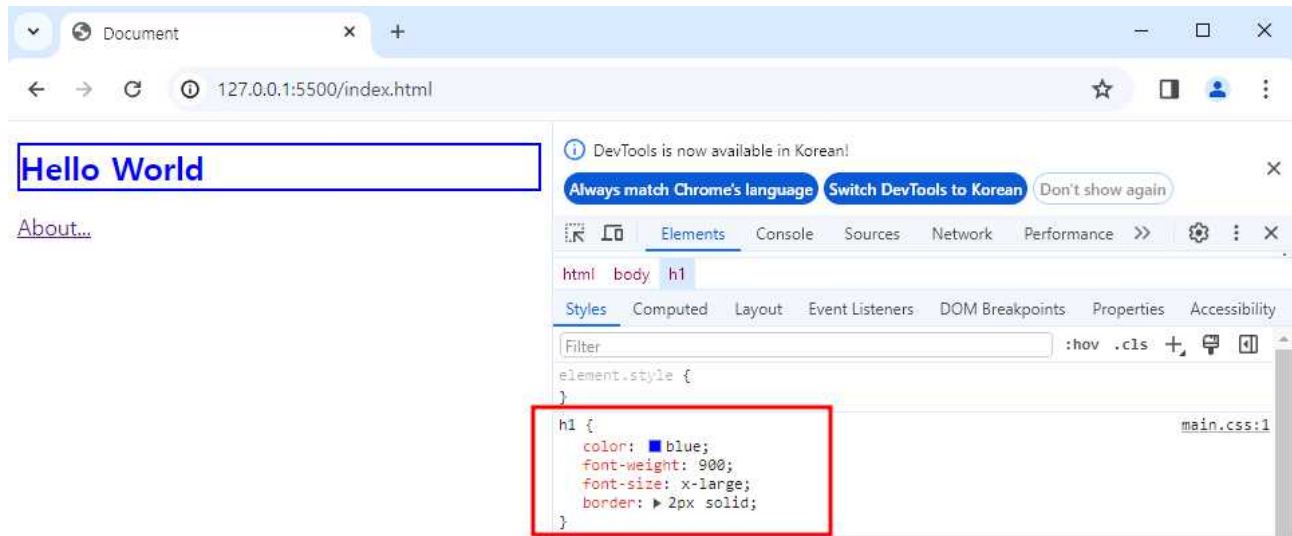
#### ② CSS 파일에서의 주석 처리

```
h1{
  /* color: blue; */
}
```

### ③ JS 파일에서의 주석 처리

```
const el = document.querySelector('h1')  
// console.log(el)
```

## 5. 브라우저 도구를 사용해서 편집하고 수정 후 Source에 적용하기



## Ch03. HTML 기본 요소 익히기

### 1. HTML 기본 문법

#### ① HTML 기본 구조

```
<h1>Hello world!</h1>
```

```
<태그>내용</태그>
```

```
<열린태그>내용</닫힌태그>
```

```
<시작태그>내용</종료태그>
```

#### ② 만들어 보기

```
<h1 class="title">Hello world!</h1>
```

```
<태그 속성="값">내용</태그>
```

- class : 해당 태그의 별칭 정도로 이해하면 됩니다.

#### ③ 종료 태그가 없는 요소도 있다.

```
<br>
```

```

```

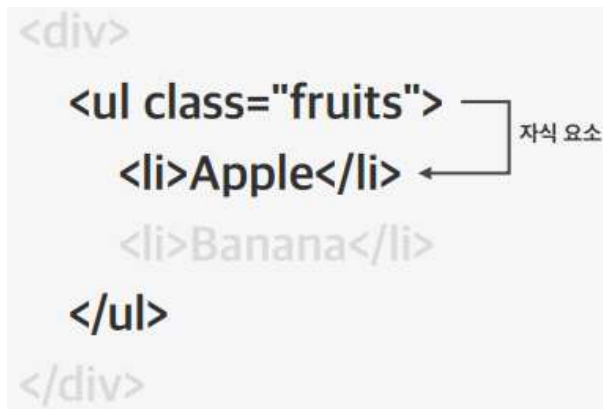
### 2. HTML 요소의 구조

#### ① 요소는 모두 가족이다.

```
<div>
  <ul class="fruits">
    <li>Apple</li>
    <li>Banana</li>
  </ul>
</div>
```

```
<div>
  <ul class="fruits">
    <li>Apple</li>
    <li>Banana</li>
  </ul>
</div>
```

부모 요소



## 2. 블록요소와 인라인 요소(상자와 글자)

### ① 상자 요소(Block Elements)

- 상자(레이아웃)을 만들기 위한 요소
- div, h1, h2, h3, p, ul, header 등...

### ② 글자요소(Inline Element)

- 글자를 만들기 위한 요소
- span, a ...

### ③ block-inline-test.html

```
<body>
  <div>Hello~</div>
  <div>Hello~</div>
  <div>Hello~</div>
  <span>World!</span>
  <span>World!</span>
  <span>World!</span>
</body>
```

### ④ 브라우저 F12 → Element에서 확인

<div> : 한줄 전체 차지  
<span> : 글자 영역만 차지  
두 번째 <div>Hello~</div> 선택 및 삭제 → 226x0 이렇게 보임!  
세 번째 <div>Hello~</div> 태그 우측 마우스 버튼 → HTML 로 수정  
<div>Hello~<br>Hello</div> → 두줄로 표시 됨.

⑤ <div> 태그와 <span> 태그의 차이

- 정렬 : align



- 가로너비 : width



- 세로너비





- 상자크기 : block – 자유롭게 지정가능, inline – 상자크기 지정 불가능



⑥ 요소에 css 적용해 보기 → F12 → style

- <div>Hello~</div>

```
element.style{
  border : 3px solid; /* 반영 됨.
  height : 50px;    /* 가로는 최대한 늘어나고 높이는 50px에 맞춘다.
  width : 150px;    /* 가로 너비도 반영된다.
}
```

- <span>World!</span>

```
element.style{
  border : 3px solid; /* 반영 됨.
  height : 50px;    /* 반영 ×
  width : 150px;    /* 반영 ×
}
```

## Ch04. CSS 기본 요소 익히기

### 1. CSS 기본 문법

```
h1 { color: red; }  
선택자 { 속성: 값; }
```

```
h1 {  
  color : red;  
  font-size : 30px;  
}
```

### 2. HTML 선택자

① 태그선택자(Type Selector) : HTML 태그를 이름으로 선택

```
<div>  
  <ul>  
    <li>사과</li>  
    <li>딸기</li>  
    <li>오렌지</li>  
  </ul>  
  <div>당근</div>  
  <p>토마토</p>  
  <span>오렌지</span>  
</div>
```

#### 태그 선택자

Type Selector  
HTML 태그 이름으로 선택해요.

```
li {  
  color: ■ red;  
}
```

② 클래스 선택자 : HTML class 속성의 값으로 선택

```
<div>  
  <ul>  
    <li>사과</li>  
    <li>딸기</li>  
    <li class="orange">오렌지</li>  
  </ul>  
  <div>당근</div>  
  <p>토마토</p>  
  <span class="orange">오렌지</span>  
</div>
```

#### 클래스 선택자

Class Selector  
HTML class 속성의 값으로 선택해요.

```
.orange {  
  color: ■ red;  
}
```

- ③ 하위(후손) 선택자 : HTML의 모든 하위 요소 선택

```
<div>
  <ul>
    <li>사과</li>
    <li>딸기</li>
    <li class="orange">오렌지</li>
  </ul>
  <div>당근</div>
  <p>토마토</p>
  <span class="orange">오렌지</span>
</div>
<span class="orange">오렌지</span>
```

## 하위(후손) 선택자

Descendant Combinator  
HTML 요소의 하위 요소를 선택해요.

```
div .orange {
  color: ■ red;
}
```

- ④ 가상클래스 : hover → HTML 요소에 마우스 커서가 올라가 있는 동안 선택



## 가상 클래스 :hover

Pseudo-Classes :hover  
HTML 요소에 마우스 커서가 올라가 있는 동안 선택해요.

```
a:hover {
  color: ■ red;
}
```

```
<a href="https://www.naver.com">NAVER</a>
```

- ⑤ 가상클래스 : active → HTML 요소에 마우스를 클릭하고 있는 동안 실행



## 가상 클래스 :active

Pseudo-Classes :active  
HTML 요소에 마우스를 클릭하고 있는 동안 선택해요.

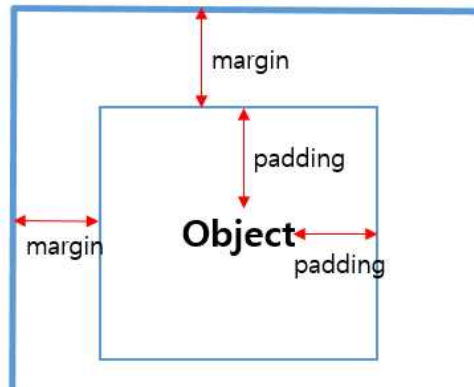
```
a:active {
  color: ■ red;
}
```

```
<a href="https://www.naver.com">NAVER</a>
```

### 3. 모양을 만드는 CSS 속성

① Box 모델 : HTML 요소 형태를 만든다.

▶ padding과 margin의 차이



- 요소의 가로 너비  
width: 100px;  
max-width / min-width
- 요소의 세로 높이  
height: 100px  
max-height / min-height
- 요소의 외부(바깥) 여백  
margin-top: 20px  
margin-top / margin-bottom / margin-left / margin-right
- 요소의 내부(안쪽) 여백  
padding-top: 20px  
padding-top / padding-bottom / padding-left / padding-right
- 요소의 테두리 선  
border: 4px solid red;
- 요소의 모서리 둥글게  
border-radius: 10px
- 요소의 그림자

```
box-shadow: 4px 4px 10px rgba(0,0,0,0.5);
```

X축 Y축 흐림 색상

: rgba(red, green, blue, alpha) - alpha(투명도)

▶ box-shape.html

```
<body>
  <div class="container">  /!-- .container(클래스 선택자) 입력 후 Tab or Enter -->
    <div class="item"> 1 </div>
    <div class="item"> 2 </div>
  </div>
</body>
```

▶ box-shape.css

```
.container{
  backgroud-color : orange;
  height : 200px;
  padding-top : 30px;
  padding-left : 30px;
  width : 300px;
}

.item{
  background-color : royalblue;
  height : 70px;
  max-width : 150px; /* 0~150까지 크기 조정됨.
  margin-bottom : 30px;
  padding-top : 20px;
  border : 4px solid red;
  border-radius : 30px;
  box-shodow : 30px 10px 30px rgba(0, 0, 0, 0.5);
}
```

② 글꼴, 문자 : 문자 서체의 종류, 문자의 정렬 등...

Hello world  
Good morning~

- 글꼴의 크기

width: 100px;

max-width / min-width

- 요소의 세로 높이

height: 100px

max-height / min-height

- 요소의 외부(바깥) 여백

margin-top: 20px

margin-top / margin-bottom / margin-left / margin-right

- 요소의 내부(안쪽) 여백

padding-top: 20px

padding-top / padding-bottom / padding-left / padding-right

- 요소의 테두리 선

border: 4px solid red;

- 요소의 모서리 둥글게

border-radius: 10px

- 요소의 그림자

**box-shadow: 4px 4px 10px rgba(0,0,0,0.5);**  
X축 Y축 흐림 색상

: rgba(red, green, blue, alpha) - alpha(투명도)

▶ font-test.html

```
<body>
  <div class="container">  /!-- .container(클래스 선택자) 입력 후 Tab or Enter -->
    동해물과 백두산이 마르고 닳도록
    하느님이 보우하사 우리나라 만세
    무궁화 삼천리 화려강산
    대한사람 대한으로 길이 보전하세
  </div>
</body>
```

▶ box-shape.css

```
.container{
  font-size : 22px; /* 기본 font-size : 16px
  font-weight : 700; /* 기본 400, bold : 700
  font-family : serif; /* 바탕체 : serif, 고딕 : sans-serif
  line-height : 2; /* 줄높이 : 1.3 ~ 1.4가 적당
  text-align : center
  color : blue;
  text-decoration : underline;
  word-break : keep-all;
  /* 글자꾸밈 : 한글은 한 글자 단위로 줄바꿈 처리되고, 영어는 단어 단위로 줄바꿈 됨.
  /* 한글 단어 띄어쓰기 : word-break : keep-all; --- 필수
}
```

③ 배경(backgrounds) : 배경색상이나 이미지 추가 등, Box 로 생성한 요소의 배경 지정

▶ background-img-test.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/css/background-img-test.css">
  <title>Document</title>
</head>
<body>
  <div class="container"></div>
```

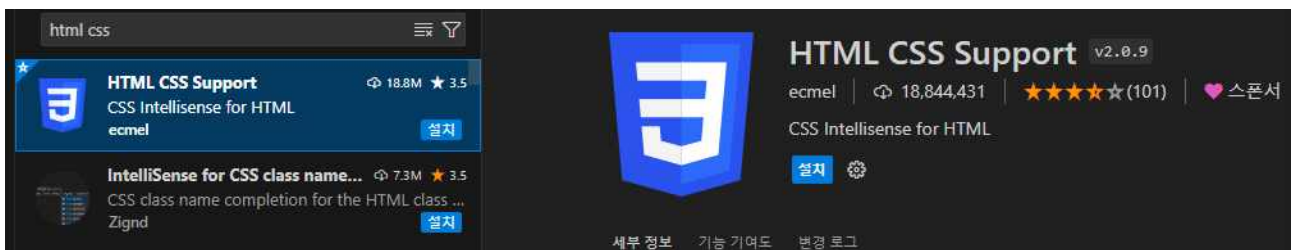


```
</body>
</html>
```

▶ background-img-test.css

```
.container{
  width: 100px;
  height: 150px;
  background-color: royalblue;
  background-image: url("/img/kitty.png");
  background-size: 50px;
  background-repeat: no-repeat;
  background-position: center;
}
```

※ HTML에서 CSS 편집을 쉽게 해 주는 확장 팩(HTML CSS Support)



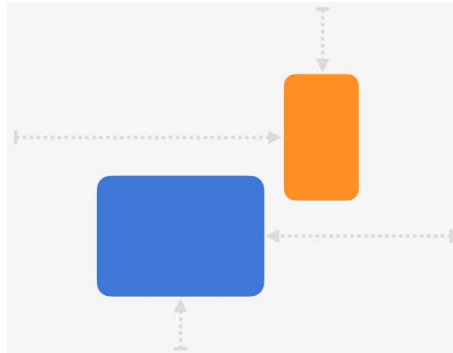
※ CSS에서 HTML에서 사용한 속성이름을 자동완성해 주는 확장 팩(HTML to CSS autocompletion)



※ setting.json에 추가하기

```
"css.enabledLanguages": [
  "html",
  "javascript"
]
```

④ 배치(Position) : HTML 요소의 위치 지정



⑤ 정렬 : Flexible, boxes, flex

- HTML 요소들을 수평 혹은 수직으로 정렬



① 1단계

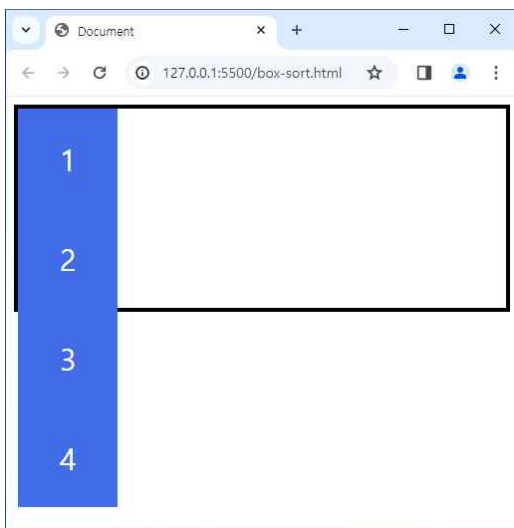
▶ box-sort.html

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/css/box-sort.css">
  <title>Document</title>
</head>
<body>
  <div class="container">
    <div class="item">1</div>
    <div class="item">2</div>
    <div class="item">3</div>
    <div class="item">4</div>
  </div>
</body>
```

▶ box-sort.css

```
.container{
  height: 200px;
  border: 4px solid black;
}
```

```
.container .item{
  width: 100px;
  height: 100px;
  background-color: royalblue;
  display: flex;
  justify-content: center;
  align-items: center;
  color: white;
  font-size: 30px;
}
```



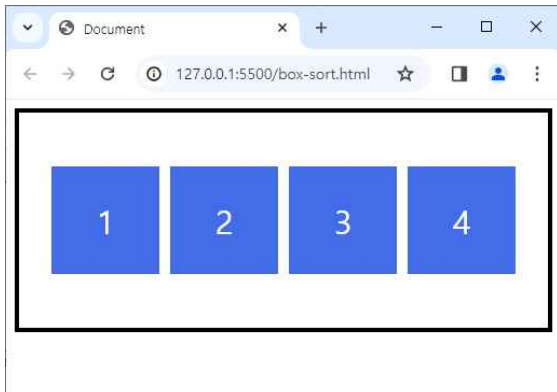
② 2단계

▶ box-sort.css

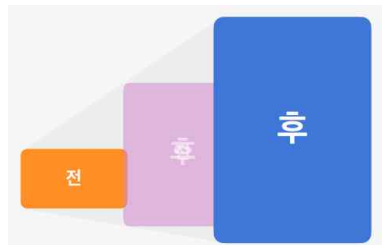
```
.container{
  height: 200px;
  border: 4px solid black;
  display: flex;
  justify-content: center;
  align-items: center;
  gap: 10px;
}

.container .item{
  width: 100px;
  height: 100px;
  background-color: royalblue;
  display: flex;
  justify-content: center;
```

```
align-items: center;
color: white;
font-size: 30px;
}
```



- ⑥ 전환(Transitions) - 일종의 애니메이션 효과
- HTML 요소의 이전과 이후 상태를 자연스럽게 처리



애니메이션 추가하기

① 1단계

▶ animation-test.html

```
<div class="container">
  <div class="item">
    animation
  </div>
</div>
```

▶ animation-test.css

```
.container{
  height: 100px;
  border: 4px solid black;
}

.container .item{
```

```
width: 100px;
height: 100px;
background-color: royalblue;
display: flex;
color: wheat;
align-items: center;
justify-content: center;
}
```



## ② 2단계

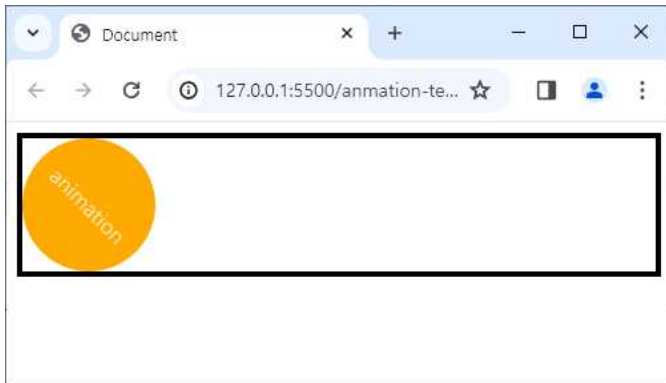
### ▶ animation-test.css

```
.container{
  height: 100px;
  border: 4px solid black;
}

.container .item{
  width: 100px;
  height: 100px;
  background-color: royalblue;
  display: flex;
  color: wheat;
  align-items: center;
  justify-content: center;
  transition: 1s;
}

.container .item:hover{
```

```
transform: rotate(45deg);
background-color: orange;
border-radius: 50%;
}
```



화면 크기에 맞게 CSS 다루기 : 반응형 web site 제작 시 사용(미디어쿼리)

① 1단계

► animation-test.html

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/css/media-query.css">
</head>
<body>
  <div class="container">
    <div class="item">
      animation
    </div>
  </div>
</body>
```

► media-query.css

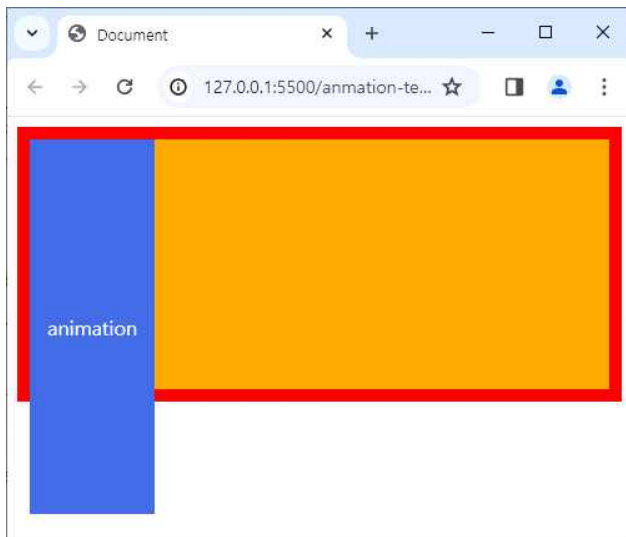
```
.container{
  height: 100px;
  border: 10px solid black;
}

.container .item{
  width: 200px;
  height: 100px;
}
```

```
background-color: royalblue;
display: flex;
align-items: center;
justify-content: center;
color: white;
}

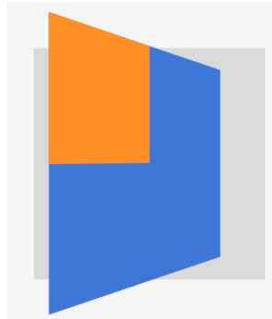
@media(max-width : 500px){
  .container{
    height: 200px;
    border: 10px solid red;
    background-color: orange;
  }

  .container .item{
    width: 100px;
    height: 300px;
  }
}
```





⑦ 변환(Transforms) : HTML 요소에 회전, 비율, 기울이기, 이동, 원근감 효과 추가



① 1단계

► box-transform.html

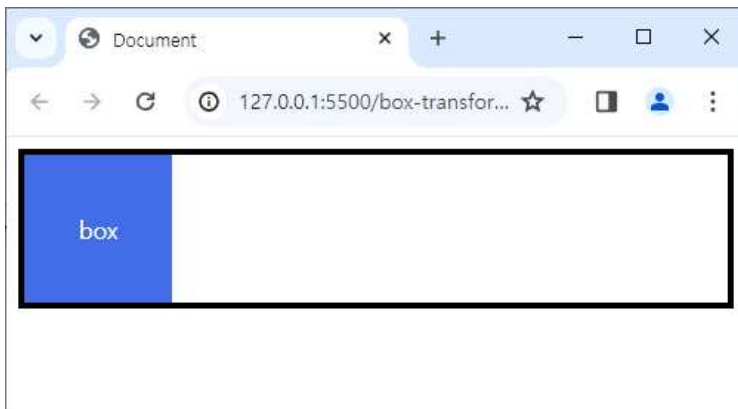
```
<link rel="stylesheet" href="/css/box-transform.css">
<title>Document</title>
</head>
<body>
  <div class="container">
    <div class="item">
      box
    </div>
  </div>
</body>
```

► box-transform.css

```
.container{
  height: 100px;
  border: 4px solid black;
}

.container .item{
  width: 100px;
  height: 100px;
  background-color: royalblue;

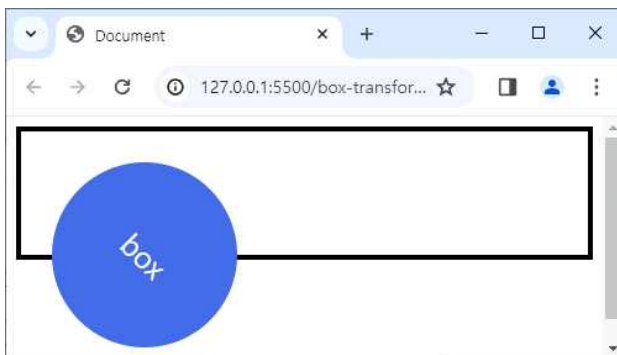
  display: flex;
  color: white;
  align-items: center;
  justify-content: center;
}
```



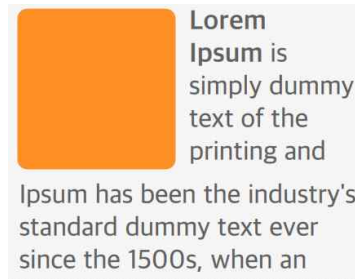
## ② 2단계

### ▶ box-transform.css

```
.container{  
  height: 100px;  
  border: 4px solid black;  
}  
  
.container .item{  
  width: 100px;  
  height: 100px;  
  background-color: royalblue;  
  
  display: flex;  
  color: white;  
  align-items: center;  
  justify-content: center;  
  
  transform: translateX(50px) translateY(50px) rotate(45deg) scale(1.5);  
  border-radius: 50%;  
}
```

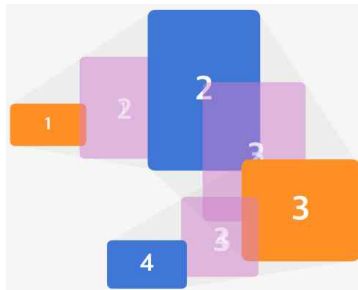


⑧ 띄움(Flotations) : HTML 요소를 글자위에 띄움.



-현재는 flexbox나 CSS Grid와 같은 레이아웃 기술을 이용하여 더욱 효율적이고 다양한 레이아웃을 구성하고 있어 참고 정도로만 보자

⑨ 애니메이션(Animation) - 전환(Transitions)보다 복잡한 효과 구현 가능



⑩ 그리드(Grid) : HTML 요소들의 레이아웃 지정(복잡한 구성 사용)



⑪ 다단(Columns) : HTML 요소의 문장을 마치 신문처럼 여러 단으로 구성

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown	the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived
--	---

⑫ 필터(Filter) : HTML 요소에 여러 가지 그래픽 효과 추가

- 흐리게 : blur
- 무채색 : gray scale
- 색상반전