



2020 혁신성장 청년인재 집중양성 사업

빅데이터 수업 1주차

# 리눅스 web서버와 React & HTML5

#Hyper-V #WSL2 #CentOS #Ubuntu

시작하기에 앞서



# A table of Contents

1

리눅스 web 서버와 Windows 개발환경

2

React 사용해 보기 – Front End

3

HTML5

# Part 1

리눅스 web 서버와 Windows 개발환경



## 리눅스 web 서버와 Windows 개발환경

리눅스 console 버전으로 변경 하는 방법

- 현재 리눅스 버전 확인

```
# systemctl get-default
```

- CentOS 8 변경 옵션

multi-user.target: analogous to runlevel 3

graphical.target: analogous to runlevel 5

- console 버전으로 변경

```
# systemctl set-default multi-user.target.target
```



# 리눅스 web 서버와 Windows 개발환경

## User에게 임시 root 권한 주기

- err message  
sudoers 설정 파일에 없습니다. 이 시도를 보고합니다

- sudo는 임시로 root 권한을 주는 명령어
- sudoers에 등록되어야지만 임시 root 권한을 쓸 수 있음.

```
$ sudo visudo -f /etc/sudoers
```

- 변경옵션

```
remaper ALL=(ALL:ALL) ALL
```



# 리눅스 web 서버와 Windows 개발환경

## FILE 에게 권한 주기

```
$ ls -al
```

```
drwxr-xr-x 2 root root 4096 Apr 22 16:59 conory
```

파일Type   퍼미션정보   링크수   소유자   소유그룹   용량   생성날짜   파일이름

- 파일 Type : "d" -> 디렉토리 , "l" -> 링크파일 , "-" -> 일반파일 등등..
- **퍼미션정보** : 해당 파일에 어떠한 퍼미션이 부여되어있는 지 표시
- **링크수** : 해당 파일이 링크된 수! 링크는 윈도우의 "바로가기"
- "in [대상파일] [링크파일]"
- **소유자** : 해당 파일의 소유자 이름
- **소유그룹** : 해당 파일을 소유한 그룹 이름
- **용량** : 파일의 용량
- **생성날짜** : 파일이 생성된 날짜
- **파일이름** : 파일이름



# 리눅스 web 서버와 Windows 개발환경

## FILE 에게 권한 주기

```
$ ls -al
```

```
d rwxr-xr-x  2 root root 4096 Apr 22 16:59 conory
```

파일Type   퍼미션정보   링크수   소유자   소유그룹   용량   생성날짜   파일이름

- 읽기 ( r ) : 파일의 읽기권한(4)
- 쓰기 ( w ) : 파일의 쓰기권한(2)
- 실행 ( x ) : 파일의 실행권한(1)

chmod [변경될 퍼미션값] [변경할 파일]

chown [변경할 소유자] [변경할 파일]



# >>>> 리눅스 web 서버와 Windows 개발환경

## FILE 에게 권한 주기

파일 속성 바꾸기 X

"공개" 디렉터리의 새로운 속성을 선택하십시오.

소유자 권한

☒ 읽기(R) ☐ 쓰기(W) ☒ 실행(E)

그룹 권한

☐ 읽기(A) ☐ 쓰기(R) ☐ 실행(X)

공개 권한

☐ 읽기(D) ☐ 쓰기(I) ☐ 실행(C)

숫자값(N) 500

원래 파일의 권한을 유지하려면 해당 자리에 x를 지정하면 됩니다.

☐ 하위 디렉터리로 이동(U)

☒ 모든 파일과 디렉터리에 적용(A)  
☐ 파일에만 적용(F)  
☐ 디렉터리에만 적용(L)

확인 취소



## 리눅스 web 서버와 Windows 개발환경

nodejs 설치하기

```
$ sudo dnf install nodejs
```

```
$ node -v
```

```
$ npx create-react-app my-app
```

```
$ cd my-app
```

```
$ npm start
```



## 리눅스 web 서버와 Windows 개발환경

**3000 port open**

```
# firewall-cmd --state
```

```
# firewall-cmd --zone=public --list-all
```

```
# firewall-cmd --permanent --zone=public --add-port=3000/tcp
```

```
# firewall-cmd --reload
```

```
# firewall-cmd --zone=public --list-all
```



## 리눅스 web 서버와 Windows 개발환경

### **Network On/Off**

```
$ sudo systemctl restart NetworkManager.service
```

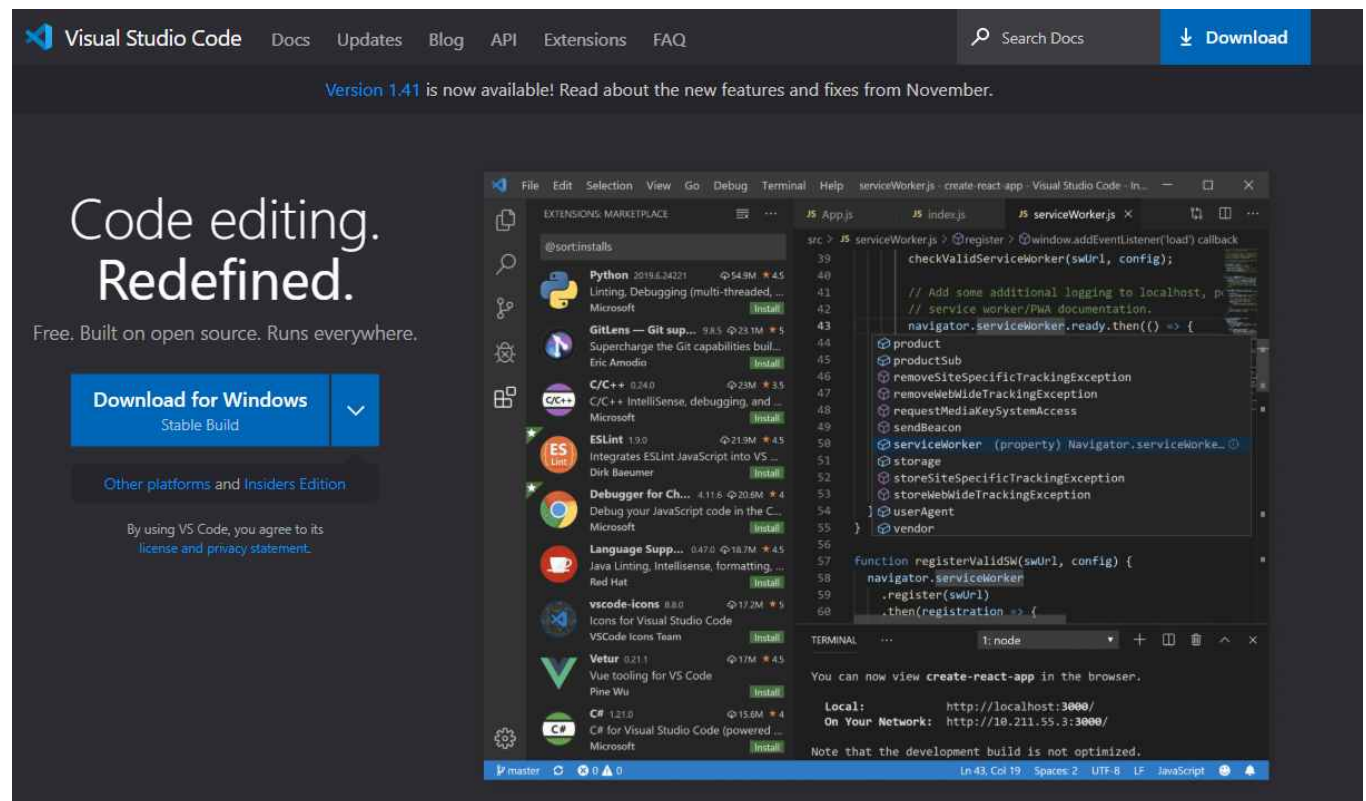
```
$ sudo nmcli net off
```

```
$ sudo nmcli net on
```

# >>>> 리눅스 web 서버와 Windows 개발환경

Visual Studio Code 설치

<https://code.visualstudio.com/>

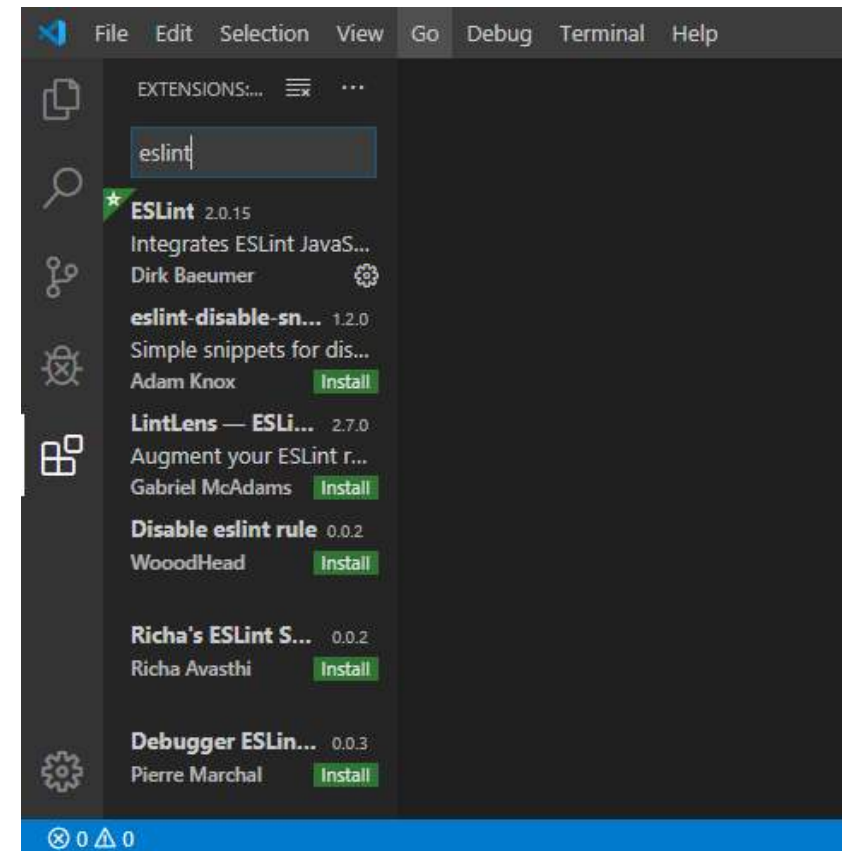


# >>>> 리눅스 web 서버와 Windows 개발환경

## Visual Studio Code 설치

### 확장프로그램 설치

- ESLint
- Reactjs Code Snippets
- Prettier-Code formatter
- code-eol
- ftp-simple





# 리눅스 web 서버와 Windows 개발환경

step 1.

- VSCode 마켓플레이스에서 ftp-simple을 Install 한다.

step 2.

- ftp-simple install 후에 VSCode 재시작(Reload).

step 3.

- 키보드 F1버튼 -> ftp-simple : Config - FTP connection setting



## 리눅스 web 서버와 Windows 개발환경

step 4.

```
[  
  {  
    "name": "homepage",  
    "host": "192.168.0.123",  
    "port": 22,  
    "type": "sftp",  
    "username": "gaesalgu",  
    "password": "gaesalgu",  
    "path": "/home/www/html/",  
    "autosave": true,  
    "confirm": true  
  }  
]
```





## 리눅스 web 서버와 Windows 개발환경

step 5.

- 지금 편집하고 있는 해당 파일 ftp-simple-temp.json 저장.

step 6.

- 키보드 F1버튼 -> ftp-simple : Remote 선택.

step 7.

- 아래 터미널창에 출력 탭 선택 -> 옆에 스피너 선택. 기본값은 아마 작업일것. 클릭하고 ftp-simple로 변경. Remote시 로그가 출력되니 로그 확인할것



## 리눅스 web 서버와 Windows 개발환경

step 8.

- ftp-simple이 접속이 잘되는지 확인. 아무 응답없다면 안된것. 주소나 유저네임 패스워드 등 정보들이 틀렸는지 확인. 다시 Remote하여도 응답하지 않는다면 VSCode 재시작. 그 후 다시 원격연결.

step 9.

- 접속이 되면 재시작 되면서 VSCode에 탐색기 항목에 경로를 지정했던 원격 폴더 생성이 됩니다.

# 접속이 됐을경우 다운로드가 제대로 안됐다면 [DIR]폴더명 으로 표시가 될것입니다. 해당 폴더를 더블클릭하여 열려고 할경우 다운로드가 시작되어 정상적으로 표시가 됩니다.



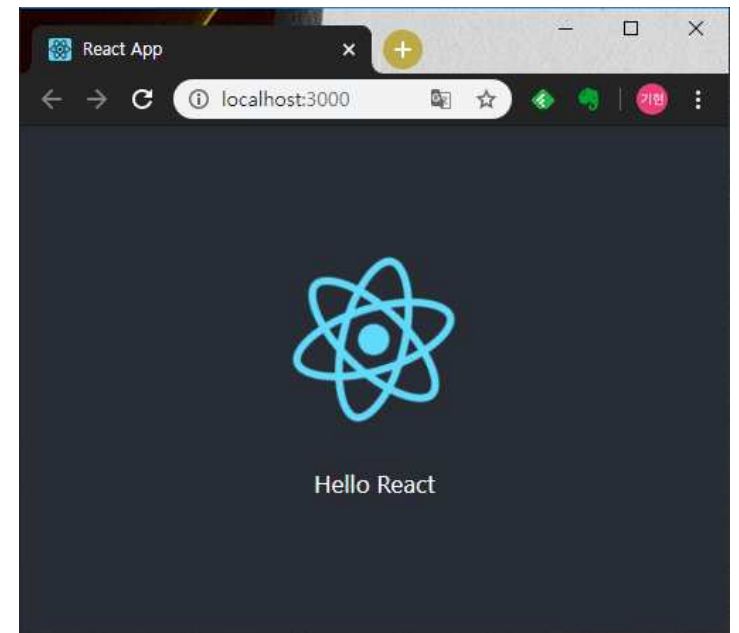
# Part 2

React 사용해 보기 – Front End



## React 사용해 보기 – Front End

```
function App() {  
  return (  
    <div className="App">  
      <header className="App-header">  
        <img src={logo} className="App-logo" alt="logo" />  
        <p>Hello React</p>  
      </header>  
    </div>  
  );  
}
```



# Introduction JSX

01

import

```
import React from 'react';  
import logo from './logo.svg';  
import './App.css';
```

# Introduction JSX

02

function App()

```
function App() {  
  ...  
  return (  
    <div className="App">  
      ...  
    </div>  
  );  
}
```

# Introduction JSX

02

function App()

```
function App() {  
  const name = '리액트';  
  return (  
    <>  
      {/* 주석은 이렇게 작성합니다. */}  
      <div  
        className="react" // 시작 태그를 여러 줄로 작성하게 된다면 여기에 주석을 작성 할 수 있습니다.  
      >  
        {name}  
      </div>  
      // 하지만 이런 주석이나 /* 이런 주석은 페이지에 그대로 나타나게 됩니다. */  
      <input />  
    </>  
  );  
}
```

# Introduction JSX

02

function App()

App.css 추가

```
.react {  
  background: aqua;  
  color: black;  
  font-size: 48px;  
  font-weight: bold;  
  padding: 16px;  
}
```



# Introduction JSX

03

export

```
export default App;
```

# Introduction JSX

04

간단한 연산자와 조건문

```
function App() {  
  const name = '리액트';  
  return (  
    <div>{name} === '리액트' && <h1>리액트입니다</h1></div>  
  );  
}
```

# Introduction JSX

04

간단한 연산자

```
function App() {  
  const name = '리액트';  
  return (  
    <div>{  
      name === '리액트' ?  
        (<h1>리액트입니다</h1>) :  
        (<h1>리액트가 아닙니다</h1>)  
    }</div>);  
}
```

# Introduction JSX

04

간단한 연산자

```
function App() {  
  const name = undefined;  
  return (  
    <div>{name} || '리액트'</div>);  
}
```

# Introduction Component

01

함수형 컴포넌트

```
function App() {  
  ...  
  return (  
    <div className="App">  
      ...  
    </div>  
  );  
}
```

# Introduction Component

02

클래스 컴포넌트

```
import React, {Component} from 'react';
import './App.css';

class App extends Component {
  render(){
    const name = 'react'
    return <div className="react">{name}</div>
  }
}
export default App;
```

# Introduction Component

03

클래스 컴포넌트 (MyComponent.js)

MyComponent.js

```
import React from 'react'
```

```
const MyComponent = () => {  
  return <div>"처음 만든 내 컴포넌트"</div>  
};
```

```
export default MyComponent
```

# Introduction Component

03

클래스 컴포넌트 (MyComponent.js)

App.js

```
import React, {Component} from 'react';  
import Mycomponent from './MyComponent'
```

```
const App=()=> {  
  return <Mycomponent/>  
}  
export default App;
```



# Introduction Component

04

클래스 컴포넌트 (MyComponent.js) – props 사용하기

MyComponent.js

```
import React from 'react'
```

```
const MyComponent = props => {  
  return <div>안녕하세요. 제 이름은  
    {props.name} 입니다. </div>  
};
```

```
export default MyComponent
```

# Introduction Component

04

클래스 컴포넌트 (MyComponent.js) – props 사용하기

App.js

```
import React, {Component} from 'react';  
import Mycomponent from './MyComponent'
```

```
const App=()=> {  
  return <Mycomponent name="remaper"/>  
}  
export default App;
```

# Introduction Component

04

클래스 컴포넌트 (MyComponent.js) – props 사용하기

MyComponent.js

```
import React from 'react'

const MyComponent = props => {
  return (
    <div>
      안녕하세요. 제 이름은 {props.name} 입니다. <br/>
      children 값은 {props.children} 입니다.
    </div>
  )
};

MyComponent.defaultProps = {
  name : '기본 이름'
};

export default MyComponent
```

# Introduction Component

04

클래스 컴포넌트 (MyComponent.js) – props 사용하기

App.js

```
import React, {Component} from 'react';
import Mycomponent from './MyComponent'

const App=()=> {
  return <Mycomponent /*name="remaper"*/>김기현
</Mycomponent>
}
export default App;
```

# Introduction Component

05

클래스 컴포넌트 (MyComponent.js) – 비구조화 할당

```
const MyComponent = props => {  
  const {name , children} = props;  
  return (  
    <div>  
      안녕하세요. 제 이름은 {name} 입니다. <br/>  
      children 값은 {children} 입니다.  
    </div>  
  )  
};
```

# Introduction Component

05

클래스 컴포넌트 (MyComponent.js) – 비구조화 할당

```
const MyComponent = ({name , children}) => {  
  return (  
    <div>  
      안녕하세요. 제 이름은 {name} 입니다. <br/>  
      children 값은 {children} 입니다.  
    </div>  
  )  
};
```

# Introduction Component

05

클래스 컴포넌트 (MyComponent.js) – 비구조화 할당

```
const MyComponent = ({name , children}) => {  
  return (  
    <div>  
      안녕하세요. 제 이름은 {name} 입니다. <br/>  
      children 값은 {children} 입니다.  
    </div>  
  )  
};
```

# Introduction Component

06

클래스 컴포넌트 (Counter.js) – state

```
import React, { Component } from 'react';

class Counter extends Component {
  state = {
    number: 0,
    fixedNumber: 0
  };
  render() {
    const { number, fixedNumber } = this.state;

  }
}

export default Counter;
```



# Introduction Component

06

## 클래스 컴포넌트 (Counter.js) – state

```
render() {  
  const { number, fixedNumber } = this.state;  
  return (  
    <div>  
      <h1>{number}</h1>  
      <h2>바뀌지 않는 값: {fixedNumber}</h2>  
      <button  
        onClick={() => {  
          this.setState(  
            {  
              number: number + 1  
            }  
          ) => {  
            console.log('방금 setState 가 호출되었습니다. ');  
            console.log(this.state);  
          }  
        }  
      >  
      +1  
    </button>  
    </div>  
  );  
};
```

# Introduction Component

06

클래스 컴포넌트 (Counter.js) – state

App.js

```
import React from 'react';  
import Counter from './counter'
```

```
const App=()=> {  
  return <Counter />  
}  
export default App;
```

# Introduction Component

06

클래스 컴포넌트 (Counter.js) – state

App.js

```
import React from 'react';  
import Counter from './counter'
```

```
const App=()=> {  
  return <Counter />  
}  
export default App;
```

# Introduction CSS

```
.react {  
  background: aqua;  
  color: black;  
  font-size: 48px;  
  font-weight: bold;  
  padding: 16px;  
}
```

```
function App() {  
  const name = '리액트';  
  return (  
    <>  
      { /* 주석은 이렇게 작성합니다. */}  
      <div  
        className="react"  
        // 시작 태그를 여러 줄로 작성하게 된다면 여기에 주석을 작  
        성 할 수 있습니다.  
      >  
        {name}  
      </div>  
      // 하지만 이런 주석이나 /* 이런 주석은 페이지에 그대로 나타나  
      게 됩니다. */  
      <input />  
    </>  
  );  
}
```

---

**예제를 보면서 같이 코딩 합니다**

---



# Part 3

HTML5



## HTML5

```
<!DOCTYPE html>  
<html> </html>  
<head> </head>  
<meta>  
<title> </title>  
<body> </body>
```



# HTML5

예제 1-1 CSS를 이용하여 글자 크기와 색상 바꾸기

01\hello-css.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8'>
<title>HTML 기본 구조</title>
<style>
body {
  font-size: 20px;
  color: red;
}
</style>
</head>
<body>
  안녕하세요, 반갑습니다~~~
</body>
</html>
```

파일의 위치이며 C:\source\챕터\  
파일 이름입니다. C:\source는 생략하  
고 늘 챕터\파일 이름만 적혀있습니다.

CSS예요!



# HTML5

예제 2-1 글 제목 만들기 02\heading.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8'>
</head>
<body>
```

책에서는 학습에 필요한 코드의 일부만 보여줍니다.  
음영으로 표기된 코드는 앞으로 나올 예제에서는 삭제  
될 예정이며, 전체 코드는 예제 파일에 있습니다.

- 1 <h1>글 제목 1</h1>
- 2 <h2>글 제목 2</h2>
- 3 <h3>글 제목 3</h3>
- 4 <h4>글 제목 4</h4>
- 5 <h5>글 제목 5</h5>
- 6 <h6>글 제목 6</h6>

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

코드의 빨간 원문자는 설명을  
위한 내용이며 실제 코드에 포  
함되지 않습니다.

실행결과

heading.html

file:///C:/source/02/heading.html

글 제목 1

글 제목 2

글 제목 3

글 제목 4

글 제목 5

글 제목 6

그림 2-1 heading.html의 실행 화면



# HTML5

## 예제 2-2 단락 나누기

02\p.html

```
<h3>요리란?</h3>
```

```
<p>요리는 먹기 좋게 가공한 음식 혹은 그 가공 행위 자체를 말합니다. 요리는 크게 나누어 지역  
별로, 시대별로 그 종류를 구분할 수 있습니다.</p>
```

1

```
<p>전통적인 한국 정식은 밥, 국, 김치와 함께 나오는 많은 반찬들로 이루어집니다. 한국 요리는  
주로 쌀을 주식으로 하며 반찬은 참기름, 된장, 간장, 소금, 마늘, 생강 그리고 고추장 등의 양  
념으로 맛을 냅니다.</p>
```

2



# HTML5

## 예제 2-3 글자 두껍게 하기

02\b.html

❶ <h3>라면 맛있게 끓이는 법</h3>

<p>냄비에 물 2컵 반을 붓고 ❷<b>건더기 수프</b>와 ❸<b>분말수프</b>를 ❹<b>미리</b> 넣고  
끓입니다. 면을 넣기 전에 수프를 ❺<b>먼저 넣어 끓이는 것</b>이 비결입니다.</p>



# HTML5

예제 2-5 <br> 태그와 nbsp;의 사용

02\br-nbsp.html

```
<p>안녕 ①&nbsp;&nbsp;&nbsp;하세요. ②<br><br>  
반갑 &nbsp;&nbsp;&nbsp;습니다!!!</p>
```



# HTML5

## 예제 2-6 <ul>, <li> 태그로 목록 만들기

02\food-fair.html

```
<h3>음식 박람회 입장 안내</h3>
```

```
<ul>
```

```
  <li>초대권 소지자 : 등록데스크에서 본인 확인 후 교환권 지급</li>
```

```
  <li>사전등록자 : 등록데스크에서 본인 확인 후 교환권 지급</li>
```

```
  <li>일반 관람객 : 일반 관람객은 매표소에서 입장권을 구입하여 입장</li>
```

```
</ul>
```



# HTML5

예제 2-7 <ol>, <li> 태그로 목록 만들기

02\fish-cake.html

```
<h3>얼큰 어묵탕 만들기</h3>
```

```
<ol>
```

```
  <li>냄비에 물을 붓고 국물용 티백을 넣고 야채를 썰어 넣어주세요.</li>
```

```
  <li>물이 끓으면 티백을 건져내고 어묵을 넣어주세요.</li>
```

```
  <li>간장과 고춧가루를 넣고 간을 맞춰주세요.</li>
```

```
  <li>준비한 그릇에 담아주면 완성입니다!</li>
```

```
</ol>
```



# HTML5

## 예제 2-8 설명 글 달기

02\comment.html

❶ <!-- <h1>이 아닌 중간 크기의 <h3> 태그를 사용했습니다. ❷-->

<h3>얼큰 어묵탕 만들기</h3>

<ol>❶ <!-- <ol> 태그를 사용하면 순서가 각 항목에 붙습니다. ❷-->

<li>냄비에 물을 붓고 국물용 티백을 넣고 야채를 썰어 넣어주세요.</li>

<li>물이 끓으면 티백을 건져내고 어묵을 넣어주세요.</li>

<li>간장과 고춧가루를 넣고 간을 맞춰주세요.</li>

<li>준비한 그릇에 담아주면 완성입니다!</li>

</ol>



# HTML5

## 예제 3-1 이미지 삽입하기

03\image.html

```
<h3>반려동물이란?</h3>
```

```
<p>사람과 더불어 살아가는 동물이란 의미로 1983년 오스트리아 빈에서 열린 애완동물 국제 심포지엄에  
서 처음으로 제안되었다 합니다.</p>
```

```
❶ <img src='dog.jpg'>
```

이미지 파일 포맷	파일 확장자	주로 사용되는 곳
JPG	.jpg	사진 이미지
GIF	.gif	아이콘과 같은 컴퓨터 그래픽 이미지 간단한 애니메이션 이미지
PNG	.png	사진과 그래픽 이미지에 모두 사용 가능
SVG	.svg	컴퓨터 그래픽 이미지와 로고 이미지



# >>>> HTML5

예제 3-2 이미지 크기 지정하기

03\image-size.html

```
<h3>게코 도마뱀</h3>
```

- ❶ <img src='./img/lizard.jpg'>
- ❷ <img src='./img/lizard.jpg' width='200' height='200'>
- ❸ <img src='./img/lizard.jpg' width='150'>

속성	의미
src	삽입되는 이미지 파일의 이름과 경로
width	이미지의 너비 지정
height	이미지의 높이 지정
title	이미지 위에 마우스를 올렸을 때 나타나는 이미지 설명 글

예제 3-3 이미지 설명 글 지정하기

```
<h3>크라운 피시</h3>
```

```
<img src='./img/fish.jpg' ❶title='영화 니모의 주인공'>
```



# HTML5

예제 3-4 오디오 재생하기

03\audio.html

```
<audio ❶src='bass.mp3' ❷controls ❸autoplay ❹loop></audio>
```

- controls  
오디오 플레이어를 화면에 표시
- autoplay  
자동 재생
- loop  
자동 반복



# HTML5

예제 3-5 비디오 재생하기

03\video.html

```
<video ❶width='320' ❷height='240' ❸autoplay ❹controls ❺loop>
  <source src='sample-video.mp4' type='video/mp4'>
  <source src='sample-video.webm' type='video/webm'>
  <source src='sample-video.ogg' type='video/ogg'>
</video>
```

- width 및 height  
비디오 플레이어의 너비와 높이 지정
- controls  
비디오 화면에 마우스 올렸을 때 플레이어가 화면에 표시
- 특정 비디오 파일 재생하려면 해당 포맷의 코덱 프로그램 있어야 함



# HTML5

예제 3-6 링크 걸기

03\main.html

```
<h2>글자에 링크 걸기</h2>
<h3>❶<a href='page1.html'>게코 도마뱀</a></h3>
<h3>❷<a href='page2.html'>크라운 피시</a></h3>

<h2>이미지에 링크 걸기</h2>
❸<a href='page1.html'><img src='img/lizard.jpg' width='100'></a>
❹<a href='page2.html'><img src='img/fish.jpg' width='100'></a>
```

- `<a>` `</a>`  
링크 거는 역할
- `href`  
클릭 시 이동할 주소



# HTML5

## 예제 3-8 새로운 탭으로 링크 걸기

03\new-tab.html

```
<h2>새로운 탭으로 링크 걸기</h2>
```

- ❶ <h3><a href='http://www.facebook.com' target='\_blank'>페이스북</a></h3>
- ❷ <h3><a href='http://www.twitter.com' target='\_blank'>트위터</a></h3>
- ❸ <h3><a href='http://www.pinterest.com' target='\_blank'>핀터레스트</a></h3>



# HTML5

예제 4-1 열차 시간표 만들기

04\train.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8'>
```

```
<style>
table, tr, th, td {
  border: solid 1px black;
  border-collapse: collapse;
  padding: 8px;
}
</style>
```

편의를 위해 추가한 CSS 코드입니다.  
자세한 내용은 2부에서 배울 예정입니다.

```
</head>
<body>
<h3>KTX 열차 시간표</h3>
```

```
❶<table>
❷<tr>
❸<th>출발</th> <th>도착</th> <th>객실</th> <th>요금</th>
</tr>
```



# HTML5

```
<tr>
  <td>서울(17:25)</td> <td>부산(20:07)</td> <td>일반실</td> <td>50,800원</td>
</tr>
<tr>
  <td>서울(18:00)</td> <td>부산(20:35)</td> <td>특실</td> <td>83,700원</td>
</tr>
<tr>
  <td>서울(18:15)</td> <td>부산(20:50)</td> <td>일반실</td> <td>50,800원</td>
</tr>
</table>
</body>
</html>
```



## HTML5

- `<table>`  
표에 넣고자 하는 내용
- `<tr>`  
하나의 행
- `<th>`  
테이블 각 열의 제목  
"table header"
- `<td>`  
열 제목 나타내는 첫 번째 행의 셀 제외한 각각의 셀 표현



# HTML5

예제 4-2 일기예보 만들기

04\weather.html

```
<h3>일기예보</h3>
<table>
  <tr>
    <th @rowspan='2'>지역</th><th @colspan='2'>27일(목)</th><th @colspan='2'>28일(금)</th>
    <th @colspan='2'>29일(토)</th>
  </tr>
  <tr>
    <td>오전</td><td>오후</td><td>오전</td><td>오후</td><td>오전</td><td>오후</td>
  </tr>
  <tr>
    <td>서울<br>경기도</td>
    <td><img src='img/shine.png'></td><td><img src='img/shine.png'></td>
    <td><img src='img/shine.png'></td>
    <td><img src='img/rain.png'></td><td><img src='img/rain.png'></td>
    <td><img src='img/rain.png'></td>
  </tr>
  <tr>
    <td>제주도</td>
    <td><img src='img/shine.png'></td><td><img src='img/shine.png'></td>
    <td><img src='img/shine.png'></td>
    <td><img src='img/rain.png'></td><td><img src='img/rain.png'></td>
    <td><img src='img/rain.png'></td>
  </tr>
</table>
```

- rowspan
  - 행 합치는 데 사용
  - 합치고자 하는 셀의 개수를 속성값으로 지정
- colspan
  - 열을 합치는 데 사용



# HTML5

## 04\text-pass.html

```
<form>
아이디 : ❶<input type='text'><br>
비밀번호 : ❷<input type='password'>
</form>
```

## 04\radio-checkbox.html

[illegible]



# HTML5

예제 4-5 파일 선택 창

04\file.html

```
<form>
파일 첨부 : ❶<input type='file'>
</form>
```

예제 4-6 선택 박스

04\select.html

```
<form>
이메일 : <input type='text'> @
❶<select>
  ❸<option>선택</option>
  ❹<option>naver.com</option>
  ❺<option>hanmail.net</option>
  ❻<option>gmail.com</option>
  ❼<option>직접입력</option>
❷</select>
</form>
```



# HTML5

예제 4-5 파일 선택 창

04\file.html

```
<form>
파일 첨부 : ❶<input type='file'>
</form>
```

예제 4-6 선택 박스

04\select.html

```
<form>
이메일 : <input type='text'> @
❶<select>
  ❸<option>선택</option>
  ❹<option>naver.com</option>
  ❺<option>hanmail.net</option>
  ❻<option>gmail.com</option>
  ❼<option>직접입력</option>
❷</select>
</form>
```

# HTML5

예제 5-1 CSS를 이용한 글자 색상 바꾸기

05\css.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8'>
```

```
<style>
①h3 {
  ②color: red;
}
③p {
  ④color: blue;
}
</style>
```

<head> 안에 위치하며 <style> </style>로 둘러싸여 본문을 꾸며줍니다.

CSS 설정은 <body> </body>로 둘러싸인 이 부분에 적용됩니다.

```
</head>
<body>
  ⑤<h3>여행(travel)의 어원</h3>

  ⑥<p>여행을 뜻하는 영어 단어 'travel'의 어원은 'travail(고통, 고난)'이라고 합니다. 교통이 발달하지 않은 과거에는 여행이 고통이나 고난이었던 거지요. 여행이 오락이나 쾌락으로 여겨지게 된 것은 교통수단이 조금 더 발달된 19세기에 이르러서입니다.</p>
</body>
</html>
```

- <head> 태그 내 <style> 태그
  - HTML 문서에 CSS 삽입
- <style> 태그 내 h3
  - CSS 선택자
  - HTML 문서에서 꾸미고자 하는 영역 선택
- color; red;
  - CSS 명령
  - 선택자 h3가 선택한 글자의 색상 적색으로 변경

# HTML5

예제 5-2 글자 스타일 지정하기

05\text-style.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8'>
<style>
```

```
h2 {
  color: blue;
  text-shadow: 2px 2px 10px gray;
}
```

<h2> </h2>로 둘러싸인 콘텐츠를 꾸며주는 부분입니다.

```
p {
  color: #444444;
  font-size: 18px;
  font-family: '바탕';
  line-height: 150%;
}
```

<p> </p>로 둘러싸인 콘텐츠를 꾸며주는 부분입니다.

```
span {
  font-weight: bold;
  color: #0e9bdc;
  text-decoration: underline;
}
```

<span> </span>으로 둘러싸인 콘텐츠를 꾸며주는 부분입니다.

```
</style>
</head>
```



# HTML5

```
<body>
<h2>세렝게티 국립공원</h2>
<p><span>탄자니아의 킬리만자로산 서쪽</span>에 위치한 세렝게티(Serengeti)의 광활한 평원은 면적
이 1,500,000ha이며, 사바나 지역에 있습니다.<span>사자, 코끼리, 들소, 얼룩말</span> 등 약 300
만 마리의 대형 포유류가 살고 있습니다.</p>
</body>
</html>
```

# HTML5

예제 5-3 목록 항목 앞에 글머리 지정하기

05\list-style-type.html

```
<head>
<meta charset='utf-8'>
<style>
  li {
    list-style-type: square;
  }
</style>
</head>
<body>
<h3>축제명 : 제주 마을박람회 축제</h3>
<ul>
  <li>일 시 : 2018년 9월 중</li>
  <li>장 소 : 월대천 및 외도동 일대</li>
  <li>주요 프로그램 : 어린이 사생대회, 토크콘서트, 은어 낚시, 소원 빌기</li>
</ul>
</body>
```

- list-style-type
  - 각 항목에 붙는 글머리 형태 지정
  - square 속성값
    - 각 항목 앞에 정사각형 포인트





# HTML5

예제 5-5 CSS에 설명 글 달기

05\list-style-image-comment.html

```
<head>
<meta charset='utf-8'>
<style>
/* 목록의 글머리 삽입하기
   list-style-image 속성을 사용
*/
li {
  list-style-image: url('img/arrow.gif'); /* 글머리에 arrow.gif 이미지를 삽입 */
}
</style>
</head>
<body>
<h3>제주 여행 정보</h3>
<ul>
  <li>제주 축제 및 행사</li> <!-- 앞에 글머리가 삽입됨 -->
  <li>추천 여행 코스</li> <!-- 앞에 글머리가 삽입됨 -->
  <li>제주 맛집</li> <!-- 앞에 글머리가 삽입됨 -->
</ul>
</body>
```



# HTML5

예제 6-1 태그 선택자, id 선택자, 클래스 선택자의 사용법

06\selector.html

## style

```
④p {  
  font-size: 14px;  
  line-height: 180%;  
}  
⑤#fly {  
  color: red;  
  font-weight: bold;  
}  
⑥.blue {  
  color: blue;  
}
```

## body

```
<h3>잠자리란?</h3>  
①<p><span ②id='fly'>잠자리</span>는 잠자리목에 속  
하는 곤충으로 전 세계적으로 분포하는 포식성 곤충이며 여러  
가지 해충을 잡아먹는 유익한 곤충입니다.</p>  
  
③<p>앞머리에 커다란 <span ④class='blue'>한 쌍의  
겹눈</span>을 가지고 있습니다. 또한 <span ⑤class=  
'blue'>날카로운 턱</span>을 가지고 있으며, 이빨이 튼튼  
합니다. 파리, 모기, 나비 등의 살아있는 곤충을 잡아먹고  
섭니다.</p>
```



# HTML5

- 태그 선택자
  - 태그의 영역 선택하고 이후에 오는 CSS 명령을 해당 영역에 적용
  - p
- id 선택자
  - 웹 페이지에서 유일무이한 단 하나의 특정 영역 지정하여 CSS 명령 적용
  - id명 앞에 샵(#) 붙여야
- 클래스 선택자
  - 두 군데 이상의 특정 영역 지정하여 동일한 CSS 적용
  - 클래스명 앞에 점(.) 붙여야



# HTML5

예제 6-2 태그 선택자

06\tag-selector.html

## style

```
❶ body {  
    font-family: '돋움';  
}  
❷ h3 {  
    font-family: '맑은고딕';  
    color: blue;  
}  
❸ p {  
    font-size: 14px;  
    line-height: 150%;  
}  
❹ li {  
    list-style-type: square;  
    font-size: 16px;  
}  
❺ span {  
    font-weight: bold;  
}
```

## body

```
<h3> - 배낭여행이란?</h3>  
<p>여권, 항공권 등 여행 시 필요한 것만을 준비하고 현지에서  
숙박, 식사 등을 해결하는 자유여행을 말합니다.</p>  
  
<h3>- 배낭여행의 종류</h3>  
<p>배낭여행에는 모든 일정을 자신이 정하는 자유 배낭, 여  
러 명이 같이 출발 전 숙소와 교통편 등을 미리 예약하고 여행  
하는 단체 배낭, 자유 배낭과 단체 배낭의 중간 형태인 패키지  
배낭여행 등이 있습니다.</p>  
  
<h3>- 배낭여행 준비</h3>  
<ul>  
    <li><span>여권 준비</span> : 여권이 없으면 신청하고  
    여권 유효기간을 반드시 체크.</li>  
    <li><span>비행기 예약</span> : 항공사의 예매 사이트  
    나 예약 대행 사이트 이용.</li>  
    <li><span>여행 스케줄</span> : 스케줄은 가능한 세부  
    적으로 잘 짜야 함.</li>  
    <li><span>짐싸기</span> : 꼭 필요한 물품만으로 최대  
    한 간단하게 짐 준비.</li>  
</ul>
```

# HTML5

예제 6-3 특정 단락에 id 선택자 사용

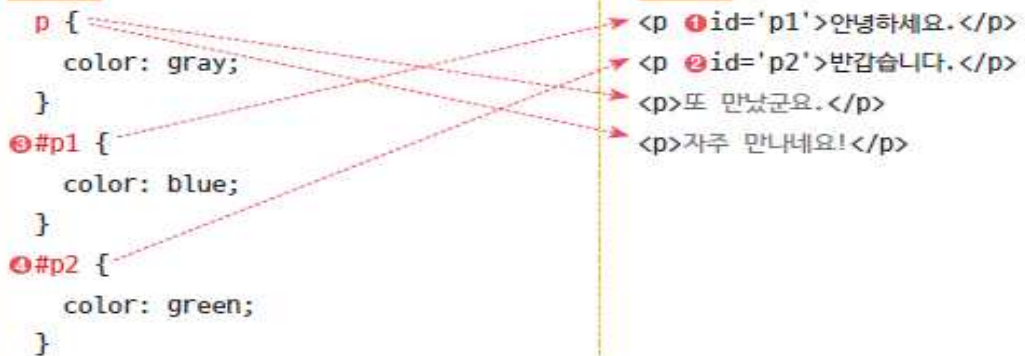
06\id-selector.html

## style

```
p {  
  color: gray;  
}  
③ #p1 {  
  color: blue;  
}  
④ #p2 {  
  color: green;  
}
```

## body

```
<p ① id='p1'>안녕하세요.</p>  
<p ② id='p2'>반갑습니다.</p>  
<p>또 만났군요.</p>  
<p>자주 만나네요!</p>
```



# HTML5

예제 6-4 목록에 사용된 클래스 선택자

06\shopping-news.html

## style

```
① .green {  
    color: green;  
    font-weight: bold;  
}
```

```
② .blue {  
    color: blue;  
    font-weight: bold;  
}
```

## body

```
<h2>쇼핑뉴스</h2>  
<ul>  
  <li><span ① class='green'>루바트</span> 봄맞이 세  
트상품 65% 할인 판매!</li>  
  <li><span ① class='green'>라리아네</span> 봄나들이  
여성룩 최대 50% 할인!</li>  
  <li><span ① class='green'>우리마트</span> 우수 농  
특산물 직거래 장터</li>  
  <li><span ② class='blue'>우리물</span> 소상공인과  
함께하는 경제적인 아이템</li>  
  <li><span ② class='blue'>골드스타</span> 14K 금값  
폭락! 최대 35% 할인!</li>  
</ul>
```

# HTML5

예제 6-4 목록에 사용된 클래스 선택자

06\shopping-news.html

## style

```
① .green {  
    color: green;  
    font-weight: bold;  
}
```

```
② .blue {  
    color: blue;  
    font-weight: bold;  
}
```

## body

```
<h2>쇼핑뉴스</h2>  
<ul>  
  <li><span ① class='green'>루바트</span> 봄맞이 세  
트상품 65% 할인 판매!</li>  
  <li><span ① class='green'>라리아네</span> 봄나들이  
여성복 최대 50% 할인!</li>  
  <li><span ① class='green'>우리마트</span> 우수 농  
특산물 직거래 장터</li>  
  <li><span ② class='blue'>우리물</span> 소상공인과  
함께하는 경제적인 아이템</li>  
  <li><span ② class='blue'>골드스타</span> 14K 금값  
폭락! 최대 35% 할인!</li>  
</ul>
```

# >>>> HTML5

## 새로 추가된 레이아웃 태그

- `<header>`
- `<nav>`
- `<section>`
- `<aside>`
- `<footer>`







# HTML5

예제 11-1 [그림 11-4] 전체 웹 페이지의 레이아웃

11\layout.html

## style

```
⑨ .clear {  
  ⑩ clear: both;  
}  
header {  
  width: 995px;  
  height: 100px;  
  margin-top: 10px;  
  border: solid 1px green;  
}  
nav {  
  width: 995px;  
  height: 70px;  
  margin-top: 10px;  
  border: solid 1px green;  
}  
section {  
  width: 674px;  
  height: 240px;  
  ⑥ float: left;  
  margin-top: 10px;  
  border: solid 1px green;  
}
```

## body

```
① <header>  
  상단 헤더  
</header>  
  
② <nav>  
  내비게이션 메뉴  
</nav>  
  
③ <section>  
  메인 콘텐츠  
</section>
```



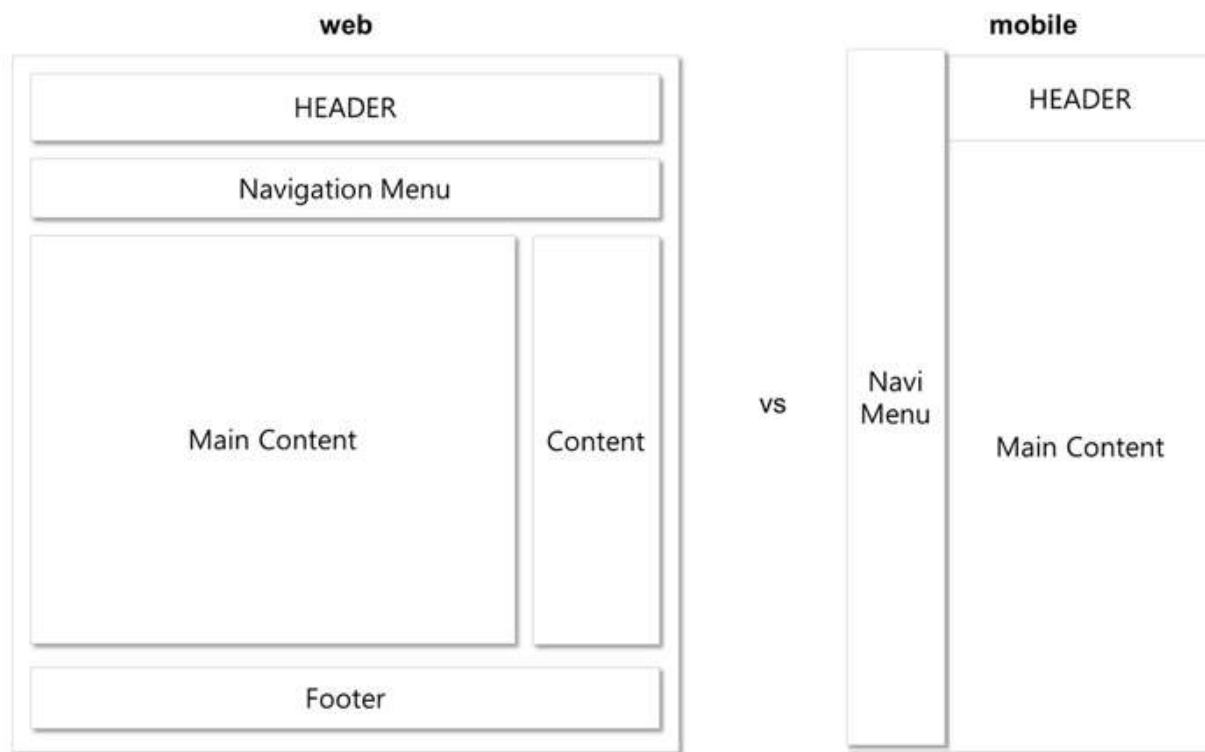
# HTML5

```
aside {  
  width: 290px;  
  height: 240px;  
  7 float: left;  
  margin-top: 10px;  
  margin-left: 29px;  
  border: solid 1px green;  
}  
footer {  
  width: 995px;  
  height: 130px;  
  margin-top: 10px;  
  border: solid 1px green;  
}
```

```
4 <aside>  
  사이드바  
</aside>  
8 <div class='clear'></div>  
  
5 <footer>  
  하단 푸터  
</footer>
```



# HTML5





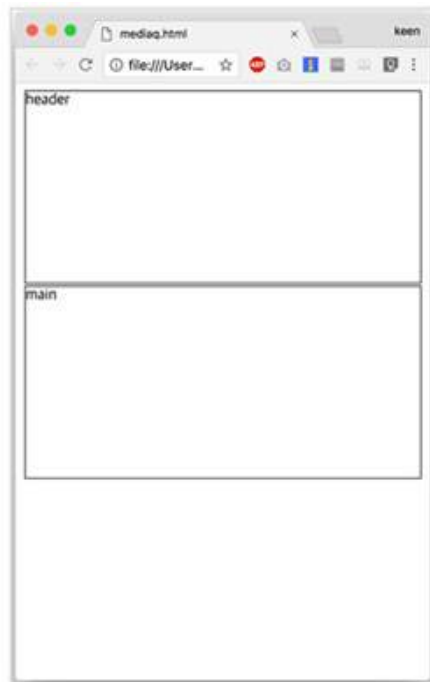
# HTML5

```
<style>
div{
border : 1px solid black;
}
@media (max-width: 400px) { //‘최대 너비 400px가 참’일 경우에 아래의 스타일을 적용합니다.
  .header {
    display: none; // 즉, 브라우저의 너비가 400px 이하인 경우는 사라집니다.
  }
}
</style>
```

```
<div class="header">
header
</div>
<div class="main">
main
</div>
```



# HTML5





# HTML5

```
<link rel="stylesheet" media="(max-width: 320px)" href="mobile.css" />
```

