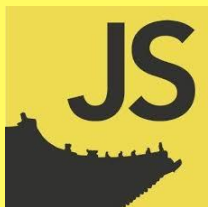


{ } + [ ]



# Javascript 10 실습

- 이벤트와 form -

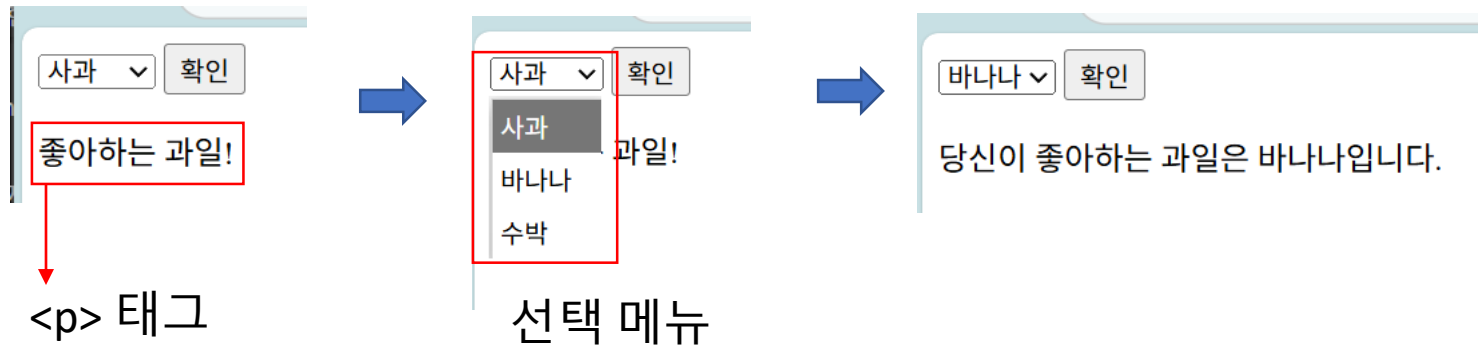
{ Na Daeyoung } + [ [soriru@handong.edu](mailto:soriru@handong.edu) ]

# Contents

- Form요소와 event
- 따라 만들기- progressbar

# P01. select 메뉴 다루기

- Form에서 select 요소의 선택값 읽어 오기
  - 좋아하는 과일을 선택하는 <select> 메뉴 생성
    - 과일 종류: 사과, 바나나, 수박
  - 확인 버튼을 누르면 “당신이 좋아하는 과일은 oo 입니다.”를
  - <p> 태그에 출력



기본 html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <form id="fruitForm">
    <select id="fruit">
      <option value="사과">사과</option>
      <option value="바나나">바나나</option>
      <option value="수박">수박</option>
    </select>
    <button type="submit">확인</button>
  </form>

  <p id="showFruit">좋아하는 과일!</p>

  <script>...
</script>

</body>
</html>
```

# P01. select 메뉴 다루기-답안

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <form id="fruitForm">
    <select id="fruit">
      <option value="사과">사과</option>
      <option value="바나나">바나나</option>
      <option value="수박">수박</option>
    </select>
    <button type="submit">확인</button>
  </form>

  <p id="showFruit">좋아하는 과일!</p>

  <script>
    const form = document.querySelector('#fruitForm');
    const select = document.querySelector('#fruit');
    const show = document.querySelector('#showFruit');

    form.addEventListener('submit', (e) => {
      e.preventDefault();

      show.textContent = `당신이 좋아하는 과일은 ${select.value}입니다.`;
    });
  </script>

</body>
</html>
```

# P01. select 메뉴 다루기 추가문제

- 기본 html 코드가 아래와 같이 변경되었을 때 같은 동작을 하도록 수정

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <form id="fruitForm">
    <select id="fruit">
      <option value="apple">사과</option>
      <option value="banana">바나나</option>
      <option value="waterMelon">수박</option>
    </select>
    <button type="submit">확인</button>
  </form>

  <p id="showFruit">좋아하는 과일!</p>
</body>
</html>
```

# P01. select 메뉴 다루기 추가문제-답안

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <form id="fruitForm">
    <select id="fruit">
      <option value="apple">사과</option>
      <option value="banana">바나나</option>
      <option value="waterMelon">수박</option>
    </select>
    <button type="submit">확인</button>
  </form>

  <p id="showFruit">좋아하는 과일!</p>
```

```
<script>
  const form = document.querySelector('#fruitForm');
  const select = document.querySelector('#fruit');
  const show = document.querySelector('#showFruit');
  let fruitName="";

  form.addEventListener('submit', (e) => {
    e.preventDefault();

    if(select.value=="apple"){
      fruitName="사과";
    }else if(select.value=="banana"){
      fruitName="바나나";
    }else if(select.value=="waterMelon"){
      fruitName="수박"
    }

    show.textContent = `당신이 좋아하는 과일은 ${fruitName}입니다.`;
  });
</script>

</body>
</html>
```

# P02. radio button으로 학년 선택하기

- **목표:** 여러 개의 라디오 버튼 중 하나를 선택하고, 선택된 값을 가져오는 방법 학습
  - 라디오 버튼을 이용해 학년을 선택하는 form 생성
  - "출력" 버튼을 누르면 선택한 학년을 <p> 태그에 출력
    - "선택한 학년: ○○학년"
  - 아무것도 선택 안 하면 "학년을 선택하세요" 출력

기본 html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <h1>학년 선택</h1>
  <form id="yearForm">
    <label><input type="radio" name="year" value="1"> 1학년 </label>
    <label><input type="radio" name="year" value="2"> 2학년 </label>
    <label><input type="radio" name="year" value="3"> 3학년 </label>
    <label><input type="radio" name="year" value="4"> 4학년 </label>
    <button type="submit">출력</button>
  </form>

  <p id="result">학년 선택 결과</p>

  <script>...
</script>
</body>
</html>
```

**학년 선택** 학년선택이 없는 상황

☐ 1학년 ☐ 2학년 ☐ 3학년 ☐ 4학년

학년 선택 결과

127.0.0.1:5500 says  
학년을 선택하세요.

**학년 선택** 학년 선택 이후 출력 버튼 클릭

☐ 1학년 ☒ 2학년 ☐ 3학년 ☐ 4학년

선택한 학년: 2학년

# P02. radio button으로 학년 선택하기-답안

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <h1>학년 선택</h1>
  <form id="yearForm">
    <label><input type="radio" name="year" value="1"> 1학년 </label>
    <label><input type="radio" name="year" value="2"> 2학년 </label>
    <label><input type="radio" name="year" value="3"> 3학년 </label>
    <label><input type="radio" name="year" value="4"> 4학년 </label>
    <button type="submit">출력</button>
  </form>

  <p id="result">학년 선택 결과</p>

  <script>
const form = document.querySelector('#yearForm');
const result = document.querySelector('#result');

form.addEventListener('submit', (e) => {
  e.preventDefault();

  const checked = form.year.value;

  if (!checked) {
    alert('학년을 선택하세요.');
```



## P02. radio button으로 학년 선택하기 추가문제

- 초기화 버튼 추가
  - 초기화 버튼은 선택시 form을 초기화 이벤트를 발생
  - radio의 선택 내용과 화면의 출력 결과
- 선택한 학년의 횟수를 카운트 하여 출력
  - 화면을 새로고침(refresh) 하기전까지 내용을 유지
    - 전역변수나 배열은 데이터를 유지할 수 있음
  - 출력 버튼을 클릭할 때 마다 아래와 같이 각 학년의 선택 결과를 출력

**학년 선택**

☐ 1학년 ☐ 2학년 ☐ 3학년 ☒ 4학년

선택한 학년

1학년:1번  
2학년:0번  
3학년:0번  
4학년:1번

**학년 선택**

☐ 1학년 ☐ 2학년 ☐ 3학년 ☐ 4학년

학년 선택 결과

# P02. radio button으로 학년 선택 추가문제-답안

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <h1>학년 선택</h1>
  <form id="yearForm">
    <label><input type="radio" name="year" value="1"> 1학년 </label>
    <label><input type="radio" name="year" value="2"> 2학년 </label>
    <label><input type="radio" name="year" value="3"> 3학년 </label>
    <label><input type="radio" name="year" value="4"> 4학년 </label>
    <br><button type="submit">출력</button>
    <button type="reset">초기화</button>
  </form>

  <p id="result">학년 선택 결과</p>
```

```
<script>
const form = document.querySelector('#yearForm');
const result = document.querySelector('#result');
let arrayResult=[0,0,0,0];

form.addEventListener('submit', (e) => {
  e.preventDefault();

  const checked = form.year.value;

  if (!checked) {
    alert('학년을 선택하세요. ');
  }else{
    if(checked==1){
      arrayResult[0]++;
    }else if(checked==2){
      arrayResult[1]++;
    }else if(checked==3){
      arrayResult[2]++;
    }else if(checked==4){
      arrayResult[3]++;
    }
    result.innerHTML = `선택한 학년<br>1학년:${arrayResult[0]}번<br>
2학년:${arrayResult[1]}번<br> 3학년:${arrayResult[2]}번<br> 4학년:${arrayResult[3]}번`;
  }

});
form.addEventListener('reset',()=>{
  arrayResult=[0,0,0,0];
  result.innerHTML = '학년 선택 결과';
});
</script>

</body>
</html>
```

# P03. checkbox 실습문제

- 문제: 관심 있는 과목 선택 후 화면에 출력하기
  - 체크박스를 가진 form을 만들고, “확인” 버튼을 누르면 사용자가 선택한 과목을 모두 화면에 출력하는 프로그램을 작성
  - <p>태그에 선택한 과목 이름은 쉼표로 연결하여 출력  
예) 선택한 과목: c언어, 소프트웨어입문
  - 아무것도 선택하지 않았다면
    - <p>태그에  
“과목을 하나 이상 선택하세요.” 출력

과목 선택 하기

☐ C언어 ☐ Python ☐ Javascript  
☐ 소프트웨어입문 ☐ 프론트엔드입문

확인

과목 선택 내용

과목 선택 하기

☐ C언어 ☐ Python ☐ Javascript  
☐ 소프트웨어입문 ☐ 프론트엔드입문

확인

학부를 하나 이상 선택하세요.

과목 선택 하기

☒ C언어 ☐ Python ☐ Javascript  
☐ 소프트웨어입문 ☒ 프론트엔드입문

확인

선택한 과목: C언어, 프론트엔드입문

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h2>과목 선택 하기</h2>
  <form id="subForm">
    <label><input type="checkbox" name="sub" value="clang"> C언어</label>
    <label><input type="checkbox" name="sub" value="python"> Python</label>
    <label><input type="checkbox" name="sub" value="javascript"> Javascript</label><br>
    <label><input type="checkbox" name="sub" value="processing"> 소프트웨어입문</label>
    <label><input type="checkbox" name="sub" value="frontend"> 프론트엔드입문</label><br>
    <button type="submit">확인</button>
  </form>

  <p id="result">과목 선택 내용</p>

  <script>...
</script>

</body>
</html>
```

# P03. checkbox 실습문제

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h2>과목 선택 하기</h2>
  <form id="subForm">
    <label><input type="checkbox" name="sub" value="clang"> C언어</label>
    <label><input type="checkbox" name="sub" value="python"> Python</label>
    <label><input type="checkbox" name="sub" value="javascript"> Javascript</label><br>
    <label><input type="checkbox" name="sub" value="processing"> 소프트웨어입문</label>
    <label><input type="checkbox" name="sub" value="frontend"> 프론트엔드입문</label><br>
    <button type="submit">확인</button>
  </form>

  <p id="result">과목 선택 내용</p>

  <script>
    const form = document.querySelector('#subForm');
    const result = document.querySelector('#result');

    function subName(sub){
      let name=null;
      switch(sub){
        case 'clang':
          name='C언어';
          break;
        case 'python':
          name='Python';
          break;
        case 'javascript':
          name='Javascript';
          break;
        case 'processing':
          name='소프트웨어입문';
          break;
        case 'frontend':
          name='프론트엔드입문';
          break;
      }
      return name;
    }
  </script>
</body>
</html>
```

```
form.addEventListener('submit', (e) => {
  e.preventDefault();

  // 체크된 체크박스들 가져오기
  const checked = document.querySelectorAll('input[name="sub"]:checked');

  // 선택 없음 처리
  if (checked.length === 0) {
    result.textContent = '학부를 하나 이상 선택하세요.';
    return;
  }

  // 선택된 value만 배열로 만들기
  const values = [];
  checked.forEach(item => values.push(subName(item.value)));

  // for(let i=0;i<checked.length;i++){
  //   console.log(checked[i].value);
  //   values.push(subName(checked[i].value));
  // }

  result.textContent = `선택한 과목: ${values.join(', ')}`;

});

</script>

</body>
</html>
```

# P04. 두 수의 최대공약수 구하기

- 화면에 2개의 값을 입력 후 [계산하기] 버튼을 클릭
  - 최대공약수를 구하는 함수- getGCD() 실행하고 구한 값을 반환
  - 반환한 값을 결과에 표현
- CSS: style.css, JS: script.js 로 저장
  - CSS파일은 css폴더에 저장
  - Js파일은 js폴더에 저장

### 최대공약수 구하기

두 수를 입력한 후 [계산하기]를 클릭하세요.

첫 번째 수

두 번째 수

계산하기

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>연습문제 2</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <h1>최대공약수 구하기</h1>
  <p>두 수를 입력한 후 [계산하기]를 클릭하세요.</p>
  <form>
    <label>첫 번째 수 : </label>
    <input type="text" id="number1" placeholder="첫 번째 수">
    <label>두 번째 수 : </label>
    <input type="text" id="number2" placeholder="두 번째 수">
    <button type="button" id="calc">계산하기</button>
  </form>
  <div id="result"></div>
  <script src="js/script.js"></script>
</body>
</html>
```

# P04. 두 수의 최대공약수 구하기

- Css 내용

```
* {
  box-sizing: border-box;
}

h1, p {
  text-align: center;
}

form {
  width: 400px;
  padding: 10px;
  margin: 20px auto;
}

label {
  display: none;
}

input {
  width: 100%;
  padding: 5px;
  margin: 5px 0;
}

button {
  width: 100%;
  height: 40px;
  padding: 5px;
  margin: 5px 0;
  border: 1px solid #222;
}

button:hover {
  background-color: #222;
  color: #fff;
  cursor: pointer;
}

#result {
  width: 150px;
  height: 60px;
  padding: 5px;
  margin: 15px auto;
  border: 1px solid #222;
  font-size: 2rem;
  text-align: center;
  line-height: 60px;
}
```

# P04. 두 수의 최대공약수 구하기

- getGCD() 함수

```
function getGCD(n1,n2){  
    let max = n1 > n2? n1: n2;  
    let GCD = 0;  
  
    for(let i=1 ; i<= max ; i++){  
        if(n1%i === 0 && n2%i === 0){  
            GCD=i;  
        }  
    }  
  
    return GCD;  
}
```

# P04. 두 수의 최대공약수 구하기-답안

- script.js

```
const num1=document.querySelector('#number1');
const num2=document.querySelector('#number2');
const btn=document.querySelector('#calc');
const result=document.querySelector('#result');

btn.addEventListener('click',()=>{
    result.innerHTML=getGCD(num1.value, num2.value);
});

function getGCD(n1,n2){
    let max = n1 > n2? n1: n2;
    let GCD = 0;

    for(let i=1 ; i<= max ; i++){
        if(n1%i === 0 && n2%i === 0){
            GCD=i;
        }
    }

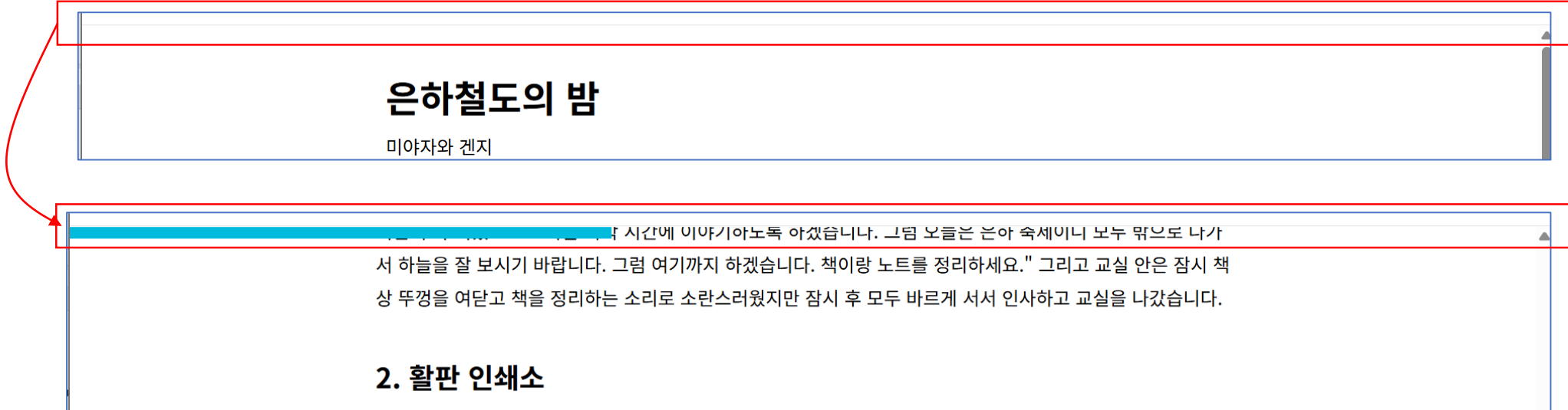
    return GCD;
}
```



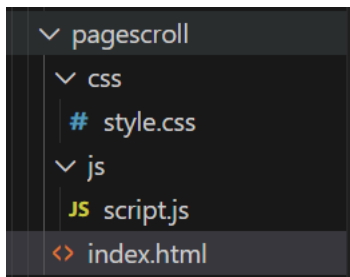
# 따라 만들기- 페이지의 스크롤 양 표현하기

- 웹 페이지의 콘텐츠 양이 많은 경우 세로로 긴 페이지로 만들어지게 되는데, 전체 페이지의 스크롤 양을 나타내는 '프로그레스 바' 구현

스크롤



폴더 구조



index.html에서 css폴더의 style.css, js폴더의 script.js 파일 연결

# 따라 만들기- 페이지의 스크롤 양 표현하기

- index.html 준비
  - 내용이 매우 많은 page

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>페이지의 스크롤 양 표시</title>
  <link rel="stylesheet" href="https://unpkg.com/ress/dist/ress.min.css"><!-- CSS 초기화 -->
  <link rel="stylesheet" href="css/style.css">
  <script src="js/script.js" defer></script>
</head>
<body> Progress bar 표현 <div> 태그
  <div id="bar"></div>
  <article>...
</article>
</body>
</html>
```

css와 js 파일을 external 연결

```
<article>
  <h1>은하철도의 방</h1>
  <p>미야자와 겐지</p>

  <h2>1. 오후 수업</h2>
  <p>"그럼 여러분은 이런 식으로 강이라고도 하고, 젖이 흐른 흔적이라고도 하는 이 희뿌연 것이 사
  캄파넬라가 손을 들었습니다. 그리고 네다섯 명이 손을 들었습니다. 조반니도 손을 들려다가 서둘러서 그만두었
  그런데 선생님은 이미 그것을 눈치 챈 것이었습니다.
  "조반니, 너는 알고 있었죠."
  조반니는 씩씩하게 일어났습니다만 일어서고 보니 이제는 확실하게 그것은 대답할 수 없었습니다. 자네리가 앞
  선생님은 다시 말했습니다.
  "커다란 망원경으로 은하를 잘 관찰하면 은하는 대체 무엇일까요?"
  역시 별이라고 조반니는 생각했습니다만 이번에도 바로 답할 수가 없었습니다.
  선생님은 잠시 난처한 기색이었습니다만 시선을 캄파넬라 쪽으로 돌리고 "그럼, 캄파넬라."하고 지명했습니다.
  선생님은 의외인 듯 한동안 꼼짝않고 캄파넬라를 보고 있었습니다만 서둘러 "그럼, 좋습니다."라고 말하며 직
  "이 희뿌연 은하를 크고 좋은 망원경을 보면 정말 수많은 작은 별로 보입니다. 조반니, 그러지요?"
  조반니는 새빨개져서 고개를 끄덕였습니다. 하지만 어느새 조반니의 눈에는 눈물이 가득 고였습니다. '그래, 나
  '그것을 캄파넬라가 잊었을리 없었을텐데 바로 대답하지 않은 것은 요즘 내가 아침에도 오후에도 일을 하느라
  선생님은 다시 말했습니다.
  "그러므로 만약 이 은하수가 정말 강이라고 생각한다면 하나하나의 작은 별은 모두 그 강바닥에 있는 모래나 거
  선생님은 빛나는 모래 알갱이가 많이 들어 있는 커다란 양면 볼록 렌즈를 가리켰습니다.
  "은하수 모양은 딱 이렇습니다. 이 하나하나 반짝이는 알갱이는 모두 우리 태양과 마찬가지로 스스로 빛을 내
  그리고 교실 안은 잠시 책상 뚜껑을 여닫고 책을 정리하는 소리로 소란스러웠지만 잠시 후 모두 바르게 서서
```

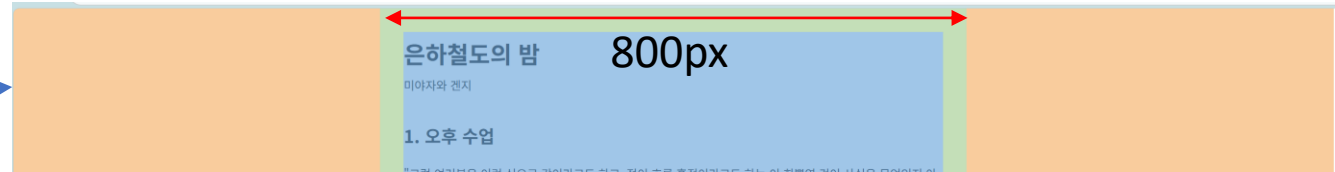
```
  <h2>2. 활판 인쇄소</h2>
  <p>조반니가 교문을 나설 때 같은 반 친구 예닐곱 명은 집으로 돌아가지 않고 캄파넬라를 한가운데에
  그렇지만 조반니는 팔을 크게 흔들면서 힘차게 교문을 나섰습니다. 나와서 보니 거리의 집들을 오늘 밤 은하
  집으로 돌아가지 않고 조반니는 시가지를 세 번 돌아선 곳에 있는 큰 활판 인쇄소에 들어갔습니다. 입구 바로
  조반니는 입구에서 세 번째에 있는 높은 테이블에 앉은 사람의 자리에 가서 인사했습니다. 그 사람은 잠시 선
  "어이! 돌보기 군, 안녕." 이라고 하자 주변의 네다섯 명의 사람들이 소리도 내지 않고 이쪽도 바라보지 않고
  조반니는 몇 번이나 눈을 비비면서 활자를 차례차례 주웠습니다.
  여섯시를 알리는 시계가 울리고 잠시 지났을 무렵, 조반니는 주문 활자를 가득 담은 넓적한 상자를 다시 한 번
  조반니는 인사를 한 다음 문을 열고 좀 전의 계산대가 있는 곳으로 갔습니다. 그러자 좀 전의 하얀 옷을 입은
```

# 따라 만들기- 페이지의 스크롤 양 표현하기

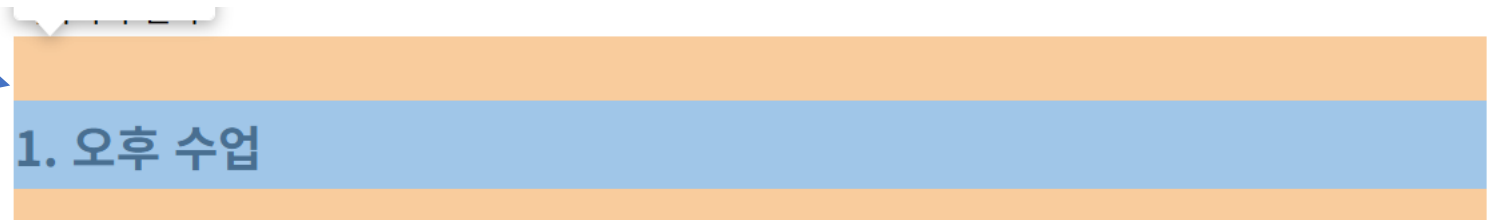
- style.css 파일

```
#bar {  
  background-color: #0bd;  
  position: fixed;  
  top: 0;  
  left: 0;  
  height: 10px;  
}  
  
article {  
  max-width: 800px;  
  margin: auto;  
  padding: 2rem;  
  line-height: 1.8;  
}  
  
h2 {  
  margin: 2rem 0 1rem;  
}
```

<div> 태그의 배경 색을 #0bd로 설정, position을 fixed로 설정하고 top: 0, left:0 으로 두어 왼쪽 상단에 위치를 고정, 10px의 높이를 주어 progressbar가 보이도록 설정



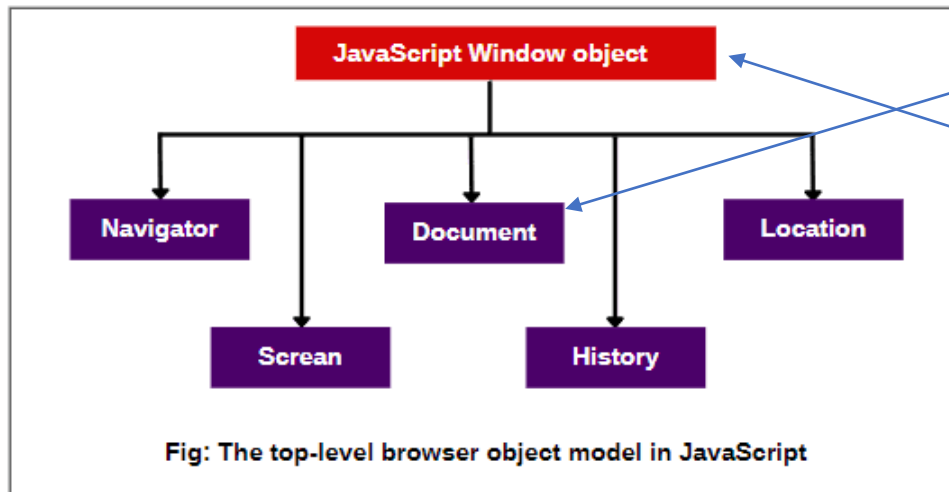
Article의 최대 너비: 800px, margin: auto로 설정하여 화면 가운데 정렬, 내부여백: 2rem 으로 설정된 것을 확인할 수 있음



<h2>태그의 좌우 외부여백이 0으로 설정되고, 위 여백은 2rem, 아래 여백은 1rem으로 설정된 결과

# 따라 만들기- 페이지의 스크롤 양 표현하기

- script.js 파일
  - Window 객체를 사용
    - 브라우저에서 실행되는 JavaScript의 전역 객체(global object)
    - 웹 페이지 하나당 하나의 window 객체가 생성되며, 이 객체 안에 해당 창(tab)의 문서(document), URL(location), 히스토리(history) 등 브라우저 관련 기능들이 포함



```
const getScrollPercent = () => {  
  // 스크롤 양  
  const scrolled = window.scrollY;  
  //console.log(`${scrolled} 스크롤 했습니다`);  
  
  // 페이지 전체 높이  
  const pageHeight = document.documentElement.scrollHeight;  
  
  // 표시 영역 높이  
  const viewHeight = document.documentElement.clientHeight;  
  //console.log(`페이지 높이 : ${pageHeight}, 표시 영역 높이 :  
  ${viewHeight}`)  
  
  // 스크롤한 비율  
  const percentage = scrolled / (pageHeight - viewHeight) * 100;  
  //console.log(percentage);  
  
  // 프로그레스 바의 폭 설정  
  document.querySelector('#bar').style.width = `${percentage}%`;  
}  
  
window.addEventListener('scroll', getScrollPercent);
```

# 따라 만들기- 페이지의 스크롤 양 표현하기

- script.js 파일

```
const getScrollPercent = () => {  
  // 스크롤 양  
  const scrolled = window.scrollY;  
  //console.log(`${scrolled} 스크롤 했습니다`);  
  
  // 페이지 전체 높이  
  const pageHeight = document.documentElement.scrollHeight;  
  
  // 표시 영역 높이  
  const viewHeight = document.documentElement.clientHeight;  
  //console.log(`페이지 높이 : ${pageHeight}, 표시 영역 높이 : ${viewHeight}`)  
  
  // 스크롤한 비율  
  const percentage = scrolled / (pageHeight - viewHeight) * 100;  
  //console.log(percentage);  
  
  // 프로그레스 바의 폭 설정  
  document.querySelector('#bar').style.width = `${percentage}%`;  
}
```

```
window.addEventListener('scroll', getScrollPercent);
```

1. window 객체에서 scroll 이라는 event가 발생하면  
getScrollPercent() 함수를 호출

2. getScrollPercent() 함수에서는  
window 객체가 가지고 있는 scrollY 속성의  
값(scroll된 양)을 읽어와서 scrolled 상수에 저장

**window.scrollY** 는  
웹 페이지에서 현재 얼마나 아래로 스크롤되었는지를  
픽셀(px) 단위로 알려주는 속성

**document.documentElement.scrollHeight**  
웹 페이지 전체 높이(스크롤까지 포함한 총 높이)\*\*를  
픽셀(px)로 알려주는 속성

**document.documentElement.clientHeight**  
지금 화면에 보이는 영역(뷰포트)의 높이를 픽셀(px) 단위로  
보여주는 속성

# { 감사합니다 } + [ Thankyou ]

{  
본 교재 또는 강좌는 2025년도 과학기술정보통신부 및 정보통신기획평가원에서 주관하여 진행하는 'SW중심대학사업'의  
결과물입니다.  
}