10.24

**DOM**

**얘를 알아야 하는 이유 : 절반 이상이 DOM을 다루기 때문에**

> TREE 형태의 html문서에 접근하기 위한 인터페이스

> 그냥 가져와서 style 적용하는 거라 생각하면 편함.

        // 태그 선택자, 클래스 선택자, 아이디 선택자 등등

        // document.querySelector('선택자') -> 해당 요소를 가져온다.

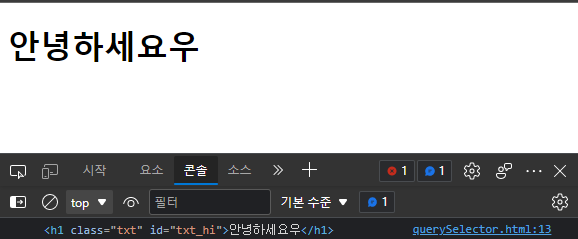
    <h1 class="txt" id="txt\_hi">안녕하세요우</h1>

    <script>

        const header = document.querySelector('h1');

        console.log();

    </script>



( ) 안에 선택자를 넣어서 애들을 가져올 수 있어진다.

// document.querySelectorAll('선택자') -> 해당 요소를 모두 가져온다.

        const headers = document.querySelectorAll('.txt')

        console.log(headers); //배열형태로 찍힘.

        for (let i = 0 ; i< headers.length; i++){

            headers[i].style.color = 'red'

            headers[i].style.backgroundColor = 'black'

            console.log(headers[i]);

        } //배열을 한개한개 따로따로 불러옴/ style을 따로따로 지정함.

    <div id="a"></div>

    <div id="b"></div>

    <script>

        const a = document.querySelector('#a')

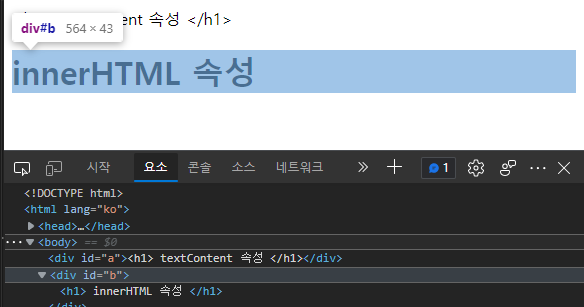
        const b = document.querySelector('#b')

        // 텍스트 형태 태그 사이에 넣어줌

        a.textContent = '<h1> textContent 속성 </h1>'

        // 태그 형태로 태그 사이에 넣어줌

        b.innerHTML = '<h1> innerHTML 속성 </h1>' //

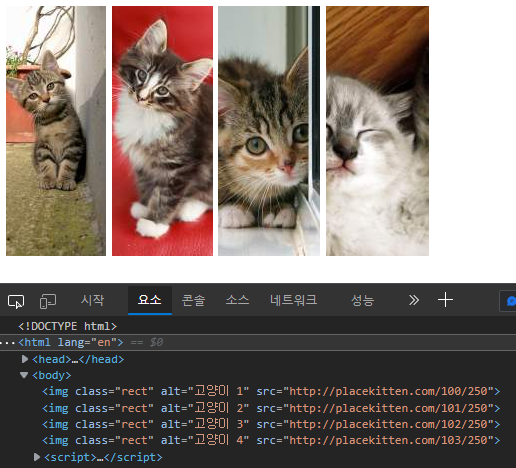


   <img class="rect" alt = "고양이 1">

   <img class="rect" alt = "고양이 2">

   <img class="rect" alt = "고양이 3">

   <img class="rect" alt = "고양이 4">



        const rects = document.querySelectorAll('.rect')

        rects.forEach(function(rect, index){

            const width = (index + 1 \* 100)

            const src = `http://placekitten.com/${width}/250` //src 선언

            // 요소. setAttribute (태그의 속성명, 속성값) - rect에 대응하는 src를 알아서 붙여줌.

            // 해당 속성명을 태그의 뒷쪽에 추가해준다.

            rect.setAttribute('src', src)

            // 값을 구하기 원하는 속성명을 괄호안에 적는다 (속성값을 가져다 주는 getAttribute)

             console.log(rect.getAttribute('alt'));

        });

        const header = document.createElement('h1');

        // 요소를 생성하는 createElement(생성하고 싶은 태그명) : 해당요소를 생성

        console.log(header);

        document.body.appendChild(header)

        //생성한 h1을 body안에 넣겠다 선언하는 appendChild (body에 child 요소로!)

        const box = document.querySelector('.box')

        // body가 아닌 box에 header을 넣고 싶은 경우, 일단 select를 먼저 하고

        box.appendChild(header)

        // 그 다음에 append태그로 header을 집어 넣는다

////////////////////////// 복습. setAttribute 열심히 써볼 것 .

    <div class="box">

        <class class="inner"> 테스트입니다 </class>

    </div>

    <script>

        const header = document.createElement('h1')

        const inner = document.querySelector('.inner')

        inner.parentNode.appendChild(header)

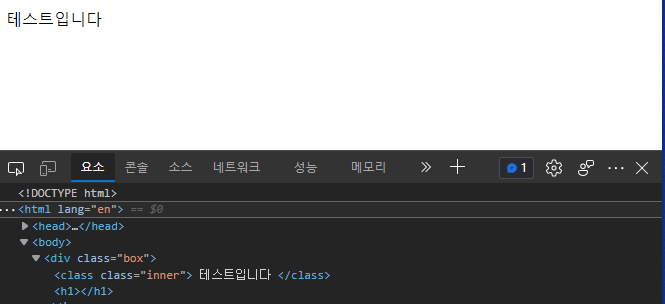
        // box태그를 선택하는 것과 비슷한 inner.parentNode.

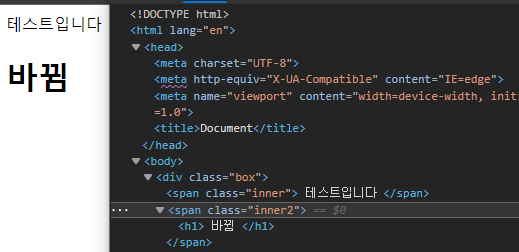
         // 그래서 box.appendChild와 동일함

         console.log(inner.nextSibling.nextSibling);

            // 다음에 오는 형제들을 찍어줌. (공백 다음 그 다음 inner2)

         inner.nextSibling.nextSibling.innerHTML = '<h1> 바뀜 </h1>'





- 이벤트? 사용자가 일으킬 수 있는 일 (클릭, 포커싱, hover, 탭 키 등)



- 이벤트 정의

// 1. 표준이벤트 모델

        button.addEventListener('click', function(){

            alert('클릭')

        })

        // 2. 고전이벤트 모델

        document.body.onclick = function(){ //-> body에 click 이벤트를 먹이는 코드

            alert('틀딱 클릭')

        }

        // 3. 인라인 이벤트 모델

        <button onclick="alertFunc()"> click me </button> << 스크립트 밖에 있는

        function alertFunc(){

            alert ('인라인 클릭')

        }

/// 키보드 이벤트

  const textarea = document.querySelector('textarea')

        const h1 = document.querySelector('h1')

        //keyup : 키보드에서 손을 땔 때

        //keypress : 키가 입력 되었을 떄 (아시아권 문자 잘 안됨)

        //keydown : 키보드가 눌릴 떄 (아시아권 문자 잘 안됨)

    textarea.addEventListener('keyup',function(){

        //const length = textarea.value //입력받는 곳 (textarea의 값을 넣을 수 있어짐)

        const length = textarea.value.length; //입력받는 곳 (textarea의 길이를 넣을 수 있어짐)

        h1.textContent = `글자수 : ${length}`

    })

       const input = document.querySelector('input')

        input.addEventListener('keyup', function(event){

            console.log(event); //눌린 키에 대한 정보가 들어가있음. (객체데이터형)

            console.log(event.keyCode); // keyCode: 입력한 키의 숫자를 나타냄

            console.log(event.code); // code : 입력한 키

            console.log(event.altKey); // 해당 키를 눌렀는지 true or false

            console.log(event.ctrlKey);

            console.log(event.shiftKey);

        })

\*아무것도 없는 p태그에 값 넣는 inner html - 공부하기\*

// input값 + button = p태그 적용

<body> <!-- 인치 변환 계산기 만들기 -->

    <input type="text">inch <br>

    <button>계산</button>

    <h1> </h1>

    <script>

        // 계산버튼이 클릭할 때 값이 변환되게 해야한다.

            const input = document.querySelector('input') //입력값 가져오기

            const calButton = document.querySelector('button') //계산버튼 가져오기

            const h1 = document.querySelector('h1')

            calButton.addEventListener('click',function(){ //이벤트 활성화

                const inch = Number(input.value) // input에서 입력받은 값 가져오기

                //check number or string?

                if(isNaN(inch)) { // isNAN : NAN값이 나온다면 = 숫자가 아닌 값이 나온다면

                    h1.textContent = `숫자가 아닌걸 입력했음`

                    return // 숫자가 아니면 더이상 진행시키지 않겠다는 뜻

                }

                const cm = inch \* 2.54; // 변환

                h1.textContent = `${cm} cm`

            })

Select – option 값에 따른 p태그 결과값

    <select>

        <option> 떡볶이 </option>

        <option> 순대 </option>

        <option> 튀김 </option>

        <option> 오뎅 </option>

    </select>

    <p>선택 : 떡볶이</p>

    <script>

        const select = document.querySelector('select')

        const p = document.querySelector('p')

        select.addEventListener('change',function(e){

            console.log(e);

            //select박스의 옵션을 다가져온다.

            const option = e.currentTarget.options

            console.log(option);

            const index = e.currentTarget.options.selectedIndex

            console.log(index);

                p.textContent = `선택 : ${option[index].textContent}`

                //.textContent를 써서 요소 안에 있는 글자만 떼어올 수 있음.

        })

Radio 선택에 따라 p값이 출력되는 함수

<body> <!-- radio의 값을 받는 p태그 -->

    <h1>★ 좋아하는 동물 선택 ★</h1>

        <input type="radio" name="pet" value="강아지">

        <span>강아지</span>

        <input type="radio" name="pet" value="고양이">

        <span>고양이</span>

        <input type="radio" name="pet" value="햄스터">

        <span>햄스터</span>

        <input type="radio" name="pet" value="기타">

        <span>기타</span>

        <hr>

        <h3 id="output"> </h3>

        <hr>

        <input type="checkbox">

        <h2 class="chk\_output"> </h2>

    <script>

        // 1. radio event

        const output = document.querySelector('#output')

        const radios = document.querySelectorAll('[name=pet]')

            // 4개를 한번에 선택해줘야 하기 때문에 ALL을 쓴다!

        // 클릭을 했을 때 뭔가가 떴으면 좋겠다! -> radio 버튼에 event를 걸어야 한다.

        // 근데 radio가 값이 여러개이기 때문에 for문을 써서 (배열로 값이 들어오기 때문에) 지정을 해준다

        radios.forEach(function(radio){ // radio에 대응하는 for문 실행

            radio.addEventListener('change',function(e){ //event 받아올 변수 지정. (각각의 radio 값이 변했을 때 대응하는!)

                const current = e.currentTarget

                if(current.checked){ //체크 여부

                    output.textContent = `좋아하는 동물은 ★${current.value}★`

                }

            })

        })

        // 2. checkbox event

        const box = document.querySelector('[type=checkbox]')

        const chk\_Out = document.querySelector('.chk\_output')

        box.addEventListener('change',function(e){

            if (e.currentTarget.checked){

                chk\_Out.textContent = `체크됨`

            } else{

                chk\_Out.textContent = `미체크됨`

            }

        })

    </script>

////////////////////////// 문제풀이

입력값을 선택한 단위에 따라 변환해주는 코드

// 진짜 개못한다 하나부터 열까지 다시 공부해라 – currentTarget – options, value, selectedIndex.

Options 값에 따른 다른결과 적용까지 공부하자 -> 레퍼런스 사이트 찾아보기

   <input type="text"> cm =

    <span></span>

    <select>

      <option value="10">mm</option>

      <option value="0.01">m</option>

      <option value="0.393701">inch</option>

    </select>

    <script>

        let currentVal // 현재값

        let transVal = 10; //변환상수

        const input = document.querySelector('input')

        const select = document.querySelector('select')

        const output = document.querySelector('span')

            function calculate(){

                output.textContent = (currentVal \* transVal).toFixed(2)

            }

            select.addEventListener('change',function(e){

                const options = e.currentTarget.options

                const index = e.currentTarget.options.selectedIndex

                transVal = (options[index].value)

                calculate()

            })

            input.addEventListener('keyup',function(e){

                currentVal = e.currentTarget.value

                 // 계산을 할 때 함수 이용해서 해라!

                calculate()

            })

// register