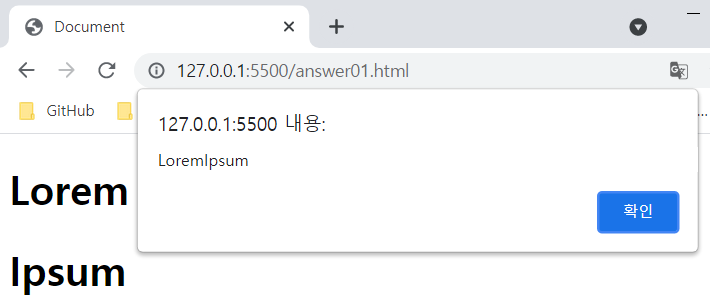
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2021\_1\_WEB\_13\_과제 | 학번 : | 20175105 | 이름 : | 곽영주 |

* 개념 확인

1. 빈 괄호를 채워넣으시오
2. jQuery로 속성을 조작할 때는 ( attr ) 메서드를 사용합니다.
3. jQuery로 스타일 속성을 조작할 때는 ( css ) 메서드를 사용합니다.
4. jQuery로 내부의 글자를 조작할 때는 ( html ) 메서드와 ( text ) 메서드를 사용합니다.
5. jQuery로 클래스 속성을 추가할 때는 ( addClass ) 메서드를 사용합니다.
6. 웹 브라우저의 ( location ) 객체는 현재 문서의 URL과 관련된 정보를 가지고 있습니다
7. ( navigator )객체는 현재 작동중인 브라우저에 대한 다양한 정보를 가지고 있습니다
8. 다음 질문에 OX를 표시하시오.
9. $() 함수는 querySelector() 메서드처럼 한 번에 하나의 문서 객체만 선택합니다. ( O )
10. 문서 객체의 속성을 조작할 때는 attr() 메서드를 사용합니다. ( O )
11. 문서 객체의 class 속성을 조작할 때는 class() 메서드를 사용합니다. ( X )
12. jQuery는 무료로 사용할 수 있는 자바스크립트 라이브러리입니다. ( O )
13. $() 함수의 리턴값은 jQuery 객체 입니다. ( O )
14. 다 음 코드의 실행 결과를 제시하시오.

|  |
| --- |
| <h1>Lorem</h1>  <h1>Ipsum</h1>  <script src="https://code.jquery.com/jquery-3.4.1.js"></script>  <script> alert($('h1').text())  </script> |



1. 다음과 같이 메서드를 연속해서 사용하는 것을 무엇이라고 하는가?

$('h1').attr('data-index', function (index) {

return index; }).css('color', 'red').addClass('header')

|  |  |
| --- | --- |
| ① 메서드 체이닝 | ② 메서드 커넥션 콜 |
| ③ 메서드 오버라이 | ④ 메서드 오버로딩 |

1. jQuery 이용방법 두 가지를 설명하시오
2. 직접 jQuery 파일을 다운받아 PC에 설치 후 연결
3. jQuery 라이브러리 서비스 제공. CDN 호스트 서비스 이용 연결
4. 제시된 HTML 코드에서 <p> 태그 요소의 배경색을 임의의 색상으로 적용하고 클래스 속성을 “pitem”으로 추가하는 jQuery 코드를 두 가지 방법으로 작성하시오

<body>

    <p>영역1</p>

    <p>영역2</p>

    <p>영역3</p>

</body>

$(document).ready(function(){

            $("p").css("background-color", "yellow");

            $("p").addClass("pitem");

        });

        $(document).ready(function(){

            $("p").addClass("pitem");

            $(".pitem").css("background-color", "yellow");

        });

1. jQurey not() 메소드를 이용하여 첫번째 <li>요소만 제외시킨 <li>의 배경과 글자색을 임의로 적용하는 jQurey 코드를 작성하시오



<body>

  <ul id="myList">

    <li>내용1</li>

    <li>내용2</li>

    <li>내용3</li>

    <li>내용4</li>

    <li>내용5</li>

  </ul>

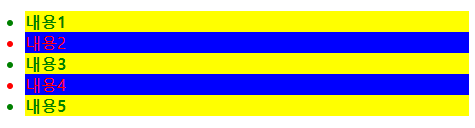
</body>

$(document).ready(function(){

            $("#myList li").not(":first").css({"background-color":"blue", "color":"red"});

        });

1. jQurey 위치 필터 선택자를 사용하여 제시된 결과처럼 <li> 요소에 스타일을 적용하는 jQurey 코드를 작성하시오



<body>

  <ul id="myList">

    <li>내용1</li>

    <li>내용2</li>

    <li>내용3</li>

    <li>내용4</li>

    <li>내용5</li>

  </ul>

</body>

$(document).ready(function(){

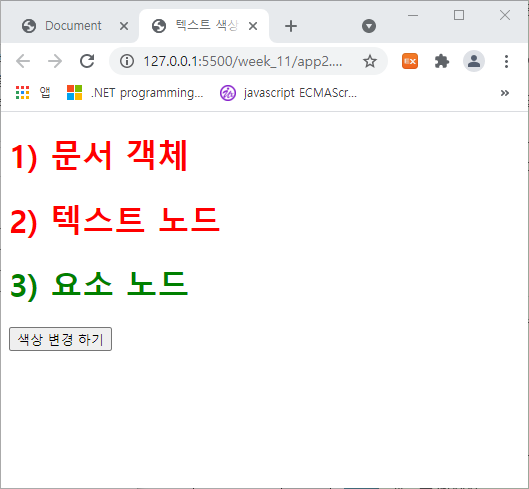
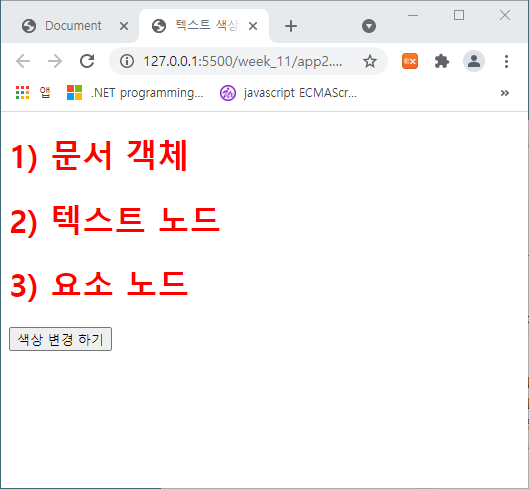
           $("#myList li:even").css({"background-color":"yellow", "color":"green"});

           $("#myList li:odd").css({"background-color":"blue", "color":"red"});

        });

* 개념 활용

1. 버튼을 클릭하면 임의의 순서로 텍스트 색상을 변경하는 스크립트를 작성하시오



<body>

  <h1>1) 문서 객체</h1>

  <h1>2) 텍스트 노드</h1>

  <h1>3) 요소 노드</h1>

  <input type="button" id="tagN" value="색상 변경 하기">

</body>

|  |
| --- |
| [소스]   <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <script>          window.onload = function(){              let h = document.querySelectorAll("h1");              let i = 0;              document.getElementById("tagN").addEventListener("click", function(event){                  if(i === 3){                      i = 0;                  }                    if(i === 0){                      h[i].style.color = "green";                      h[i+2].style.color = "black";                  }                  else{                      h[i].style.color = "green";                      h[i-1].style.color = "black";                  }                  i++;              });          }      </script>  </head>  <body>      <h1>1) 문서 객체</h1>      <h1>2) 텍스트 노드</h1>      <h1>3) 요소 노드</h1>      <input type="button" id="tagN" value="색상 변경 하기">  </body>  </html> |
| [실행결과] |

1. 웹 문서에 새로운 요소를 추가하고 삭제와 위치변경을 수행하는 스크립트를 작성하시오. – 이벤트 핸들러는 **이벤트 핸들러 프로퍼티를 사용할 것**
2. ‘자식 요소 순서 변경’ 버튼을 클릭하면 웹브라우저로 출력된 요소의 순서를 임의로 변경
3. ‘자식 요소 삭제’ 버튼을 클릭하면 임의의 요소를 삭제한다
4. ‘자식 요소 추가’ 버튼을 클릭하면 div 자식 요소로 <p>태그 요소를 추가한다.

<body>

  <div id="container">

    <p class="item">1) 사과</p>

    <p class="item">2) 레몬</p>

    <p class="item">3) 자두</p>

    <p class="item">4) 키위</p>

    <p class="item">5) 포도</p>

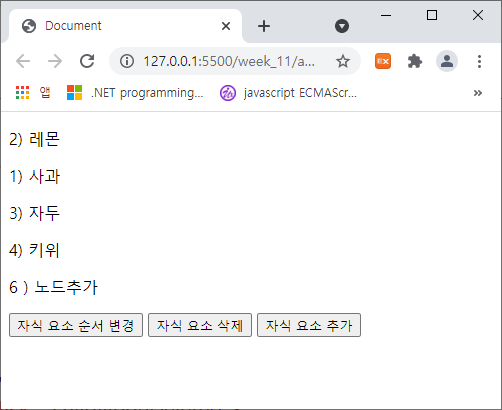
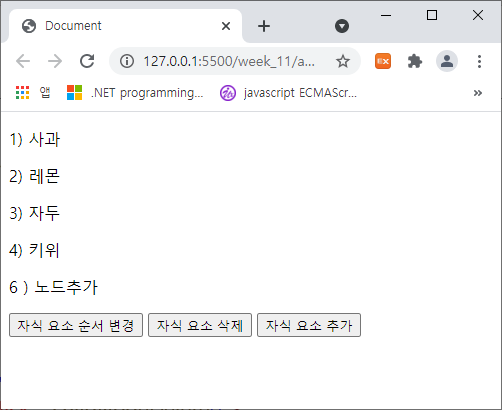
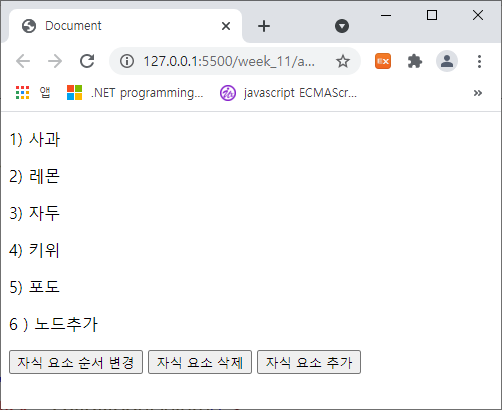
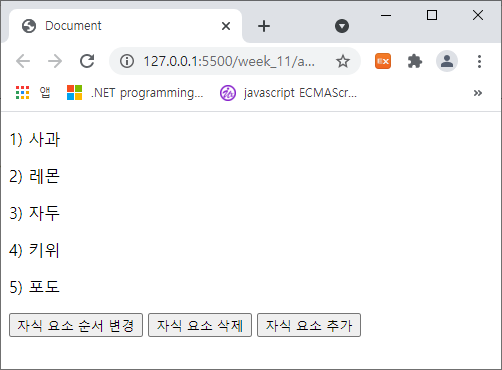
  </div>

  <input type="button" value="자식 요소 순서 변경" onclick="childNodeChange()">

  <input type="button" value="자식 요소 삭제" onclick="childNodeDelete()">

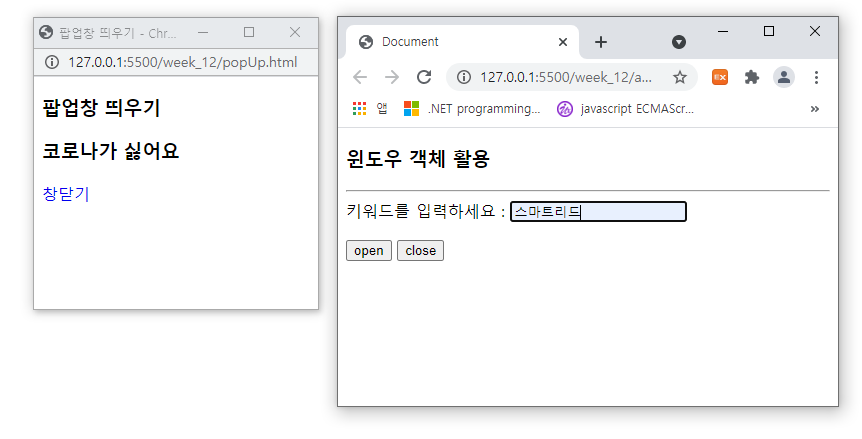
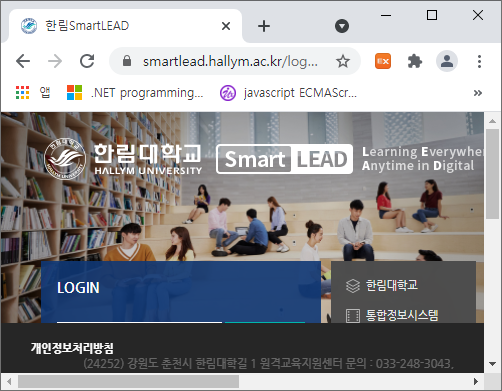
  <input type="button" value="자식 요소 추가" onclick="childNodeAdd()">

</body>



|  |
| --- |
| [소스]  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <script>          window.onload = function(){              let childNodeChange = document.getElementById("childNodeChange");              let childNodeDelete = document.getElementById("childNodeDelete");              let childNodeAdd = document.getElementById("childNodeAdd");              childNodeChange.onclick = function(){                  let divchild = document.querySelector("#container");                  let pchild = document.querySelectorAll(".item");                  let index01 = Math.floor(Math.random() \* pchild.length);                  let index02 = Math.floor(Math.random() \* pchild.length);                  divchild.insertBefore(divchild.children[index02], divchild.children[index01]);              }              childNodeDelete.onclick = function(){                  let divchild = document.querySelector("#container");                  let pchild = document.querySelectorAll(".item");                  let index01 = Math.floor(Math.random() \* pchild.length);                  divchild.removeChild(pchild[index01]);              }              childNodeAdd.onclick = function(){                  let pchild = document.querySelectorAll(".item");                  const newElem = document.createElement("p");                  const newText = document.createTextNode(`${pchild.length + 1}) 노드추가`);                  newElem.appendChild(newText);                  newElem.className = "item";                  document.querySelector("div").appendChild(newElem);              }          }      </script>  </head>  <body>      <div id="container">          <p class="item">1) 사과</p>          <p class="item">2) 레몬</p>          <p class="item">3) 자두</p>          <p class="item">4) 키위</p>          <p class="item">5) 포도</p>      </div>      <input type="button" value="자식 요소 순서 변경" id="childNodeChange">      <input type="button" value="자식 요소 삭제" id="childNodeDelete">      <input type="button" value="자식 요소 추가" id="childNodeAdd">  </body>  </html> |
| [실행결과] |

1. 웹 문서 로드 후 팝업 창을 실행 하며, 텍스트 박스로 입력된 키워드에 해당하는 웹 페이지를 새로운 창에서 출력하는 스크립트를 작성하시오.
2. 팝업창에서 ‘창닫기’ 텍스트를 클릭하면 창을 close한다.
3. ‘open’ 버튼을 클릭하면 텍스트박스로 입력된 페이지로 이동
4. ‘close’ 버튼을 클릭하면 새로 오픈한 창을 close한다



<body>

  <h3>윈도우 객체 활용</h3>

  <hr>

  <form>

    키워드를 입력하세요 : <input type="text" id="keyword"><p></p>

    <input type="button" value="open" onclick="winOpen()">

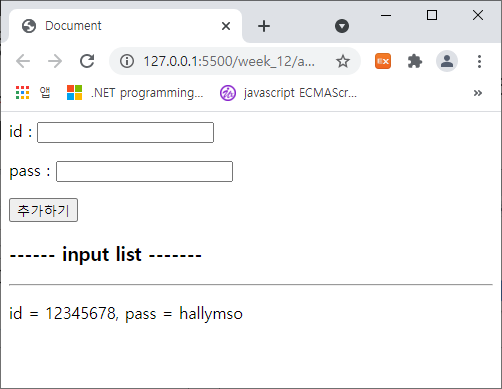
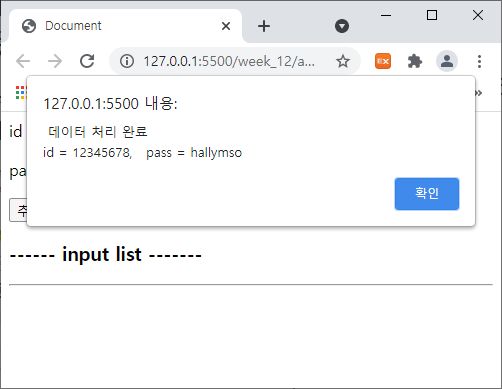
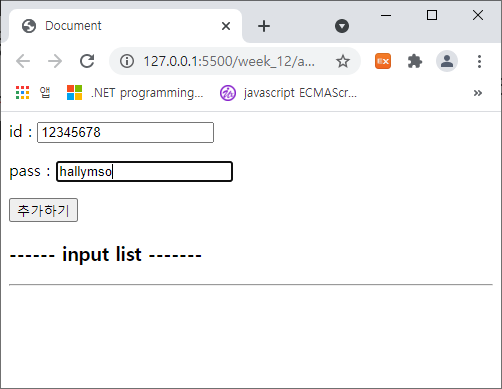
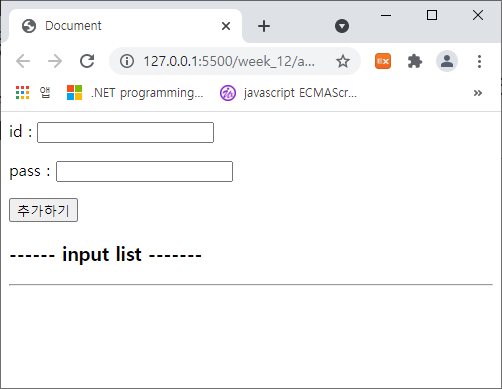
    <input type="button" value="close" onclick="winClose()">

  </form>

</body>

|  |
| --- |
| [소스]  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <script>          let win = null;          window.onload = function(){              win = window.open("popUp.html", "", "width=300, height=300");              if(win == null){                  alert("팝업 차단을 해제하세요.");              }              win.moveTo(300, 100);          }          function winOpen(){              let keyword = document.getElementById("keyword");              let place = "width=500, height=500, top=300, left=300";              switch(keyword.value){                  case "스마트리드":                      win = window.open("https://smartlead.hallym.ac.kr", "\_blank", place);                      win.moveBy(100, 100);              }          }          function winClose(){              win.close();          }      </script>  </head>  <body>      <h3>윈도우 객체 활용</h3>      <hr>      <form>          키워드를 입력하세요 : <input type="text" id="keyword"><p></p>          <input type="button" value="open" onclick="winOpen()">          <input type="button" value="close" onclick="winClose()">      </form>  </body>  </html>  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <style>          p{              font-size: 18px;              font-weight: bold;          }      </style>      <script>          function winClose(){              this.close();          }      </script>  </head>  <body>      <P>팝업창 띄우기</P>      <p>코로나가 싫어요</p>      <button onclick="winClose()">창닫기</button>  </body>  </html> |
| [실행결과] |

1. 제시된 결과처럼 텍스트박스로 입력된 데이터를 웹 문서에 추가는 스크립트를 작성하시오 – **이벤트 핸들러는 addEventListener 메소드를 사용할 것**
2. id에 입력된 글자가 8이상이면 자동으로 포커스가 pass로 이동
3. pass에서 백스페이스이고 입력된 글자가 없으면 id로 포커스 이동
4. 추가하기 버튼을 클릭하면 경고창으로 데이터 출력 후 웹 브라우저로 출력
5. 텍스트 박스에 입력된 값 추가하기 버튼을 클릭하면 삭제



<body>

  <form>

    id : <input type="text" maxlength="8">

    <p></p>

    pass : <input type="text" maxlength="8">

    <p></p>

    <input type="button" value="추가하기" id="add">

  </form>

  <h3>------ input list -------</h3>

  <hr>

  <div id='container'></div>

</body>

|  |
| --- |
| [소스]  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <script>          function Mem(id, pass){              this.id = id;              this.pass = pass;              this.toString = function(){                  return `id = ${this.id.value}, pass = ${this.pass.value}`;              }          }          const member = () => {              let id = document.querySelectorAll("input")[0];              let pass = document.querySelectorAll("input")[1];              return new Mem(id, pass);          }          const ch\_focus = (id, pass) => {              id.addEventListener("keydown", function(){                  if(8 <= id.value.length){                      pass.focus();                  }              });              pass.addEventListener("keydown", function(event){                  if(event.keyCode == 8 && pass.value.length == 0){                      id.focus();                  }              });          }          const check = () => {              let data = member();              ch\_focus(data.id, data.pass);          }          const clear = (data) => {              data.id.value = "";              data.pass.value = "";          }          const process = () => {              let con = document.getElementById("container");              let newP = document.createElement("p");              let data = member();              let newText = document.createTextNode(`${data.toString()}`);              newP.appendChild(newText);              con.appendChild(newP);              alert(` 데이터 처리 완료\n${data.toString()}`);              clear(data);          }          window.onload = function(){              let btn = document.getElementById("add");              btn.addEventListener("click", function(){                  process();              });              check();          };      </script>  </head>  <body>      <form>          id : <input type="text" maxlength="8">          <p></p>          pass : <input type="text" maxlength="8">          <p></p>          <input type="button" value="추가하기" id="add">      </form>      <h3>------ input list -------</h3>      <hr>      <div id='container'></div>  </body>  </html> |
| [실행결과] |