<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <meta http-equiv="X-UA-Compatible" content="ie=edge" />

    <title>Hello world</title>

  </head>

  <body>

    <div>

      <article>

        <p>Lorem ipsum dolor sit amet, odio omnesque ius cu, quo ex atqui antiopam. At detracto menandri eos. Duo in causae viderer, graeci <a href="#">reprehendunt</a> has in. Decore <mark>nemore</mark> philosophia te pro, nobis legere causae ex mei, odio putant mentitum ea ius. Vix nostro deserunt explicari eu.</p>

      </article>

    </div>

    <ul>

      <li><a href="#">Link1</a></li>

      <li><a href="#">Link2</a></li>

      <li><a href="#">Link3</a></li>

      <li><a href="#">Link4</a></li>

    </ul><span></span>

    <a href="#">Some link</a>

    <script src="app.js"></script>

  </body>

</html>

/\*//1. Найти параграф и получить его текстовое содержимое (только текст!)

let parag = document.querySelector('p');

let textP = parag.textContent;

console.log(textP);

//2.Создать функцию, которая принимает в качестве аргумента узел DOM и возвращает информацию (в виде объекта) о типе узла, об имени узла и о количестве дочерних узлов (если детей нет - 0).

function info (node){

    return{

        type: node.nodeType,

        name: node.nodeName,

        children: node.children.length,

    };

}

let res = info(parag);

console.log(res);

//3. Получить массив, который состоит из текстового содержимого ссылок внутри списка: getTextFromUl(ul) ---> ["Link1", "Link2", "Link3"]

let

let ul = document.getElementsByTagName('ul');

let newArr= [];

function getTextFromUl(ul){

    for(let i = 0; i<ul[0].children.length; i++){

        newArr[i] =  ul[0].children[i].textContent;

    }

    return newArr;

}

let res2 = getTextFromUl(ul);

console.log(res2);

let li = document.getElementsByTagName('li');

let newArr = [...li].map(element => element.textContent);

console.log(newArr);

//4. В параграфе заменить все дочерние текстовые узлы на “-text-” (вложенные теги должны остаться). Конечный результат:

//-text-<a href="#">reprehendunt</a>-text-<mark>nemore</mark>-text-

const p = document.getElementsByTagName('p');

const arrChildNodes = p[0].childNodes;

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

    if(arrChildNodes[i].nodeType == 3){

        arrChildNodes[i].textContent = '-text-';

        console.log(arrChildNodes[i]);

    }

}

console.log(arrChildNodes);

/\*

1. Найти в коде список ul и добавить класс “list”

2. Найти в коде ссылку, находящуюся после списка ul, и добавить id=link

3. На li через один (начиная с самого первого) установить класс “item”

4. На все ссылки в примере установить класс “custom-link”

\*/

//1. Найти в коде список ul и добавить класс “list”

/\*let ulList = document.querySelector('ul');

ulList.classList.add('list');

console.log(ulList);

//2. Найти в коде ссылку, находящуюся после списка ul, и добавить id=link

let linkAfterUl = ulList.nextElementSibling;

let nameA = document.getElementsByTagName('a');

while(linkAfterUl.tagName != 'A'){

     linkAfterUl = linkAfterUl.nextElementSibling;

}

linkAfterUl.setAttribute('id', 'link');

console.log(linkAfterUl);

//3.На li через один (начиная с самого первого) установить класс “item”

let allLinks = document.getElementsByTagName('li');

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

    if(i%2 !== 0){

        allLinks[i].classList.add('item');

    }

}

console.log(allLinks);

//4. На все ссылки в примере установить класс “custom-link”

let addClassLinks = document.links;

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

    addClassLinks[i].classList.add('custom-link');

}

console.log(addClassLinks); \*/

/\*

1. Не используя innerHTML, добавить в список несколько li с классом ‘new-item’ и текстом ‘item’ + номер li:

<ul>

<li><a href="#">Link1</a></li>

...

<li class=”new-item”>item 5</li>

<li class=”new-item”>item 6</li>

</ul>

Вручную номер li не ставить оно должно подставляться в зависимости от кол-ва лишек в списке.

function addLinks (amount){

    let ulArr = document.querySelector('ul');

    for(let i = 0; i<amount; i++){

        let newLink = document.createElement('li');

        ulArr.appendChild(newLink).classList.add('new-item');

        newLink.textContent = `item ${ulArr.children.length }`;

        console.dir(newLink);

   }

}

let res = addLinks(2);

//2. В каждую ссылку, которая находятся внутри списка ul  добавить по тегу strong (в каждую ссылку один - strong).

let liInUl = document.getElementsByTagName('li');

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

    let strong = document.createElement('strong');

    let a = liInUl[i].children;

     a[0] = a[0].appendChild(strong);

}

//3.В начало документа (в начало body) добавить картинку img с атрибутами src и alt (текст придумайте сами). В src добавьте реальный url к картинке. Для создания элемента используйте метод createElement.

let img = document.createElement('img');

img.setAttribute('src', 'https://avatars.mds.yandex.net/get-pdb/2304766/2abc8630-3f05-4a61-946f-6c45fcb558fa/s1200');

img.setAttribute('alt', 'picture');

document.body.insertAdjacentElement('beforebegin', img);

//4. Найти на странице элемент mark, добавить в конец содержимого текст “green” и на элемент установить класс green

let mark = document.querySelector('mark');

mark.classList.add('green');

let currText = mark.textContent += 'green';

//5.Отсортировать li внутри списка в обратном порядке (по тексту внутри)

let li = [...document.querySelectorAll('li')];

li.reverse();

\*/