**[createTextNode vs innerHTML vs textContent](https://learn.javascript.ru/modifying-document" \l "createtextnode-vs-innerhtml-vs-textcontent)**

важность: 5

У нас есть пустой DOM-элемент elem и строка text.

Какие из этих 3-х команд работают одинаково?

1. elem.append(document.createTextNode(text))
2. elem.innerHTML = text
3. elem.textContent = text

Ответ: 1 и 3

### [Очистите элемент](https://learn.javascript.ru/modifying-document" \l "ochistite-element)

важность: 5

Создайте функцию clear(elem), которая удаляет всё содержимое из elem.

<ol id="elem">

<li>Привет</li>

<li>Мир</li>

</ol>

<script>

function clear(elem) { /\* ваш код \*/

elem.remove();

}

clear(elem); // очищает список

</script>

### [Почему остаётся "aaa"?](https://learn.javascript.ru/modifying-document" \l "pochemu-ostayotsya-aaa)

важность: 1

В примере ниже вызов table.remove() удаляет таблицу из документа.

Но если вы запустите его, вы увидите, что текст "aaa" все еще виден.

Почему так происходит?

<table id="table">

aaa

<tr>

<td>Тест</td>

</tr>

</table>

<script>

alert(table); // таблица, как и должно быть

table.remove();

// почему в документе остался текст "ааа"?

</script>

Ответ: потому что текст ааа находится без текстового тега.

**[Создайте список](https://learn.javascript.ru/modifying-document" \l "sozdayte-spisok)**

важность: 4

Напишите интерфейс для создания списка.

Для каждого пункта:

1. Запрашивайте содержимое пункта у пользователя с помощью prompt.
2. Создавайте элемент <li> и добавляйте его к <ul>.
3. Продолжайте до тех пор, пока пользователь не отменит ввод (нажатием клавиши Esc или введя пустую строку).

Все элементы должны создаваться динамически.

Если пользователь вводит HTML-теги, они должны обрабатываться как текст.

<script>

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

        document.body.append(ul);

        let text=prompt('Введите текст в пункт: ');

        while (text != ' ' && text!=null){

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

            elemText.textContent=text;

            ul.append(elemText);

            text=prompt('Введите текст в пункт: ');

        }

    </script>

**[Создайте дерево из объекта](https://learn.javascript.ru/modifying-document" \l "sozdayte-derevo-iz-obekta)**

важность: 5

Напишите функцию createTree, которая создаёт вложенный список ul/li из объекта.

Например:

let data = {

"Рыбы": {

"форель": {},

"лосось": {}

},

"Деревья": {

"Огромные": {

"секвойя": {},

"дуб": {}

},

"Цветковые": {

"яблоня": {},

"магнолия": {}

}

}

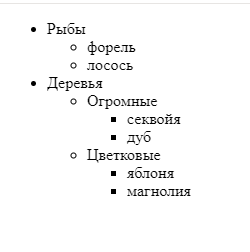
};

Синтаксис:

let container = document.getElementById('container');

createTree(container, data); // создаёт дерево в контейнере

Результат (дерево):



Выберите один из двух способов решения этой задачи:

1. Создать строку, а затем присвоить через container.innerHTML.
2. Создавать узлы через методы DOM.

Если получится – сделайте оба.

P.S. Желательно, чтобы в дереве не было лишних элементов, в частности -– пустых <ul></ul> на нижнем уровне.

<script>

    let data = {

      "Рыбы": {

        "форель": {},

        "лосось": {}

      },

      "Деревья": {

        "Огромные": {

          "секвойя": {},

          "дуб": {}

        },

        "Цветковые": {

          "яблоня": {},

          "магнолия": {}

        }

      }

    };

    function createTree(container, data) {

      /\* ваш код \*/

      container.append(createTreeDom(data));

    }

    function createTreeDom(data){

        if (!Object.keys(data).length) return;

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

        for (let key in data) {

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

            li.innerHTML = key;

            let childrenUl = createTreeDom(data[key]);

            if (childrenUl) {

            li.append(childrenUl);

            }

            ul.append(li);

        }

      return ul;

    }

    createTree(document.getElementById('tree'), data);

  </script>

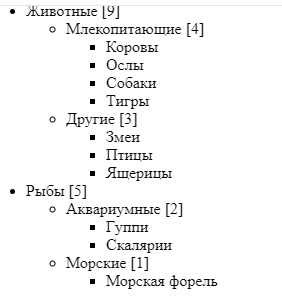
### [Выведите список потомков в дереве](https://learn.javascript.ru/modifying-document" \l "vyvedite-spisok-potomkov-v-dereve)

важность: 5

Есть дерево, организованное в виде вложенных списков ul/li.

Напишите код, который добавит каждому элементу списка <li> количество вложенных в него элементов. Узлы нижнего уровня, без детей – пропускайте.

Результат:



 <script>

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

    for (let li of lis) {

      let descendantsCount = li.getElementsByTagName('li').length;

      if (!descendantsCount) continue;

      li.firstChild.data += ' [' + descendantsCount + ']';

    }

  </script>

### [Создайте календарь в виде таблицы](https://learn.javascript.ru/modifying-document" \l "sozdayte-kalendar-v-vide-tablitsy)

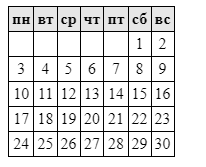
важность: 4

Напишите функцию createCalendar(elem, year, month).

Вызов функции должен создать календарь для заданного месяца month в году year и вставить его в elem.

Календарь должен быть таблицей, где неделя – это <tr>, а день – это <td>. У таблицы должен быть заголовок с названиями дней недели, каждый день – <th>, первым днём недели должен быть понедельник.

Например, createCalendar(cal, 2012, 9) сгенерирует в cal следующий календарь:



P.S. В этой задаче достаточно сгенерировать календарь, кликабельным его делать не нужно.

<body>

      <div id="calendar"></div>

    <script>const createCalendar = (cal, year, month) => {

        let inputDate = new Date(year, month, 0);

        let shiftDays = inputDate.getDay();

        let daysOfMonth = inputDate.getDate();

        let rows = Math.ceil((daysOfMonth + shiftDays) / 7);

        let table = document.createElement("table");

        table.style = "border: 0; margin: 10rem auto";

        table.innerHTML =

        "<tr><th>Пн</th><th>Вт</th><th>Ср</th><th>Чт</th><th>Пт</th><th>Сб</th><th>Вс</th></tr><tr>";

        for (let i = 0; i < rows; i++) {

        let row = document.createElement("tr");

        for (let j = 1; j <= 7; j++) {

        let col = document.createElement("td");

        col.style = "border: 1px solid; padding: 5px";

        row.append(col);

        }

        table.append(row);

        }

        for (let i = 1, days = 1; i < table.rows.length; i++) {

        let row = table.rows[i];

        for (let j = 0; j < row.cells.length; j++, days++) {

        if ((j < shiftDays && i === 2) || days > daysOfMonth) {

        days--;

        continue;

        }

        row.cells[j].textContent = days;

        }

        }

        cal.append(table);

        };

        createCalendar(calendar, 2012, 9);

        </script>

</body>

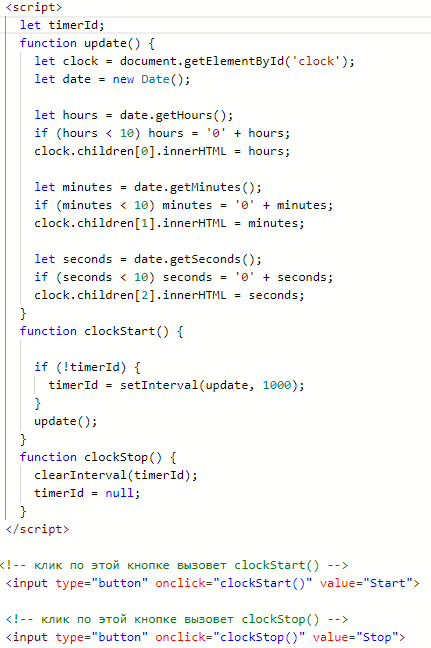
### [Цветные часы с использованием setInterval](https://learn.javascript.ru/modifying-document" \l "tsvetnye-chasy-s-ispolzovaniem-setinterval)

важность: 4

Создайте цветные часы как в примере ниже:



Для стилизации используйте HTML/CSS, JavaScript должен только обновлять время в элементах.



### [Вставьте HTML в список](https://learn.javascript.ru/modifying-document" \l "vstavte-html-v-spisok)

важность: 5

Напишите код для вставки <li>2</li><li>3</li> между этими двумя <li>:

<ul id="ul">

<li id="one">1</li>

<li id="two">4</li>

</ul>

Ответ: создаем скрипт с one.insertAdjacentHTML('afterend', '<li>2</li><li>3</li>');

### [Сортировка таблицы](https://learn.javascript.ru/modifying-document" \l "sortirovka-tablitsy)

важность: 5

Вот таблица:

<table>

<thead>

<tr>

<th>Name</th><th>Surname</th><th>Age</th>

</tr>

</thead>

<tbody>

<tr>

<td>John</td><td>Smith</td><td>10</td>

</tr>

<tr>

<td>Pete</td><td>Brown</td><td>15</td>

</tr>

<tr>

<td>Ann</td><td>Lee</td><td>5</td>

</tr>

<tr>

<td>...</td><td>...</td><td>...</td>

</tr>

</tbody>

</table>

В ней может быть больше строк.

Напишите код для сортировки по столбцу "name".

 let sortedRows = Array.from(table.rows)

          .slice(1)

          .sort((rowA, rowB) => rowA.cells[0].innerHTML > rowB.cells[0].innerHTML ? 1 : -1);

        table.tBodies[0].append(...sortedRows);