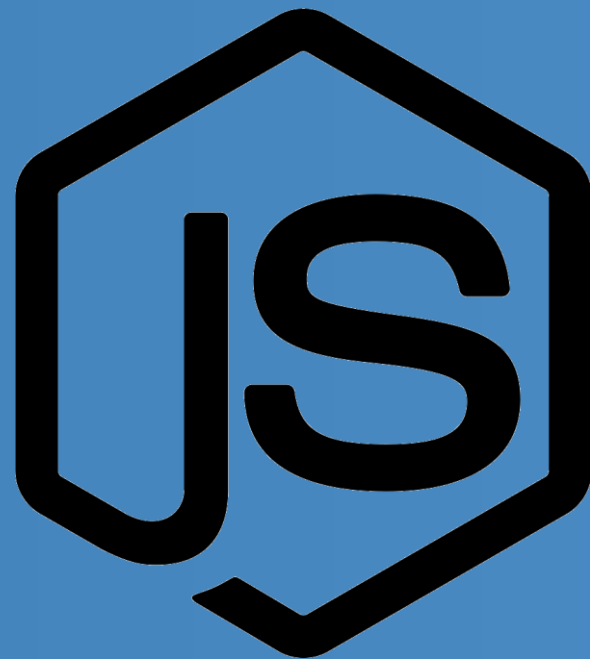




Занятие №6

# События, DOM



# BOM

**Основным объектом спецификации является объект окна.**

Многие вещи в браузере можно контролировать с помощью **javascript**, например, открытие окон, открытие вкладок, закрытие страниц, избранное и т. Д.

Эти функции не имеют ничего общего с веб-контентом. Поскольку стандарта нет, разные браузеры могут реализовывать одну и ту же функцию по-разному.

# DOM

**Самый фундаментальный объект DOM - это документ.**

Под документом здесь понимается веб-страница, которая представляет собой html-документ.

DOM не имеет ничего общего с браузером.

Его интересует содержимое самой веб-страницы.

# DOM – DOCUMENT OBJECT MODEL

Значения объектов DOM хранятся как структура кода HTML.

Корневой (самый первый) элемент документа находится в объекте «**documentElement**» главного объекта модели «**document**».

То есть полное имя «**document**»

Через это имя мы будем обращаться к элементам.

# Пишем

```
<body>
  <p id="p_one">This is a first paragraph</p>
  <p id="p_two">This is a second paragraph</p>

  <script>
    var pOne = document.getElementById("p_one")
    var pTwo = document.getElementById("p_two")

    pOne.innerText = "new text in first paragraph"
    pTwo.style.color = "red"
  </script>
</body>
```

# Пишем

```
<body>
  <p>This is a first paragraph</p>
  <p class="p_two">This is a second paragraph</p>
  <script>
    // возвращают список
    console.log(document.getElementsByTagName("p"))
    console.log(document.getElementsByClassName("p_two"))
  </script>
</body>
```

# Также можно в цикле пробежаться по массиву

```
<p>This is a first paragraph</p>
<p class="p_two">This is a second paragraph</p>
<script>
    // обращаемся по индексу к элементам
    document.getElementsByTagName("p")[0].innerText = "first paragraph"
    document.getElementsByClassName("p_two")[0].innerText = "second paragraph"
</script>
```

Дополнительно можете почитать

[https://www.w3schools.com/jsref/dom\\_obj\\_document.asp](https://www.w3schools.com/jsref/dom_obj_document.asp)

# События

Для взаимодействия с пользователем в JavaScript определен механизм событий. Например, когда пользователь нажимает кнопку, то возникает событие нажатия кнопки. В коде JavaScript мы можем определить возникновение события и как-то его обработать.

Для всех стандартных событий у элементов страницы предусмотрены обработчики. Традиционно их имена формируются из префикса «on» и имени события. Например, обработчик события щелчка мыши «click» будет иметь название «onclick».



```
<div onclick="myfunc()" style="background:red">div</div>  
<script>  
    function myfunc() {  
        alert("ALERT")  
    }  
</script>
```

ПЕРВЫЙ СПОСОБ ОПРЕДЕЛИТЬ  
Обработчик — это указать имя функции,  
которая сработает при нажатии

## ВТОРОЙ СПОСОБ ОПРЕДЕЛИТЬ

Обработчик — это обратиться к элементу и указать анонимную функцию

```
<div id="div_id" style="background:red">div</div>
<script>
    document.getElementById("div_id").onclick = function () {
        alert("ALERT")
    }
</script>
```

## ТРЕТИЙ СПОСОБ ОПРЕДЕЛИТЬ

Обработчик — это обратиться к элементу и повесить на него событие

```
<div id="div_id" style="background:red">div</div>
<script>
    document.getElementById("div_id")
        .addEventListener("click", function () {
            alert("ALERT")
        })
</script>
```

## Нельзя смешивать несколько способов

Если определить обработчик в HTML теге, а затем в скриптовой части снова указать функцию для того же события, то новое значение будет использовано вместо старого. Другими словами, новое тело обработчика сотрет старое и займет его место.

# Объект EVENT и его свойства

```
<div onclick="func(event) " style="background:red">div</div>  
<script>  
    function func(e) {  
        console.log(e)  
    }  
</script>
```

```
<head>
  <meta charset="utf-8" />
  <title>Events</title>
  <style>
    body {
      position: relative;
      height: 2000px;
    }
    div {
      background: navy;
      height: 50px;
      width: 100px;
      position: absolute;
    }
  </style>
</head>
<body onclick="moveStranger(event)">

  <div id="stranger"></div>

  <script>
    function moveStranger(e) {
      stranger.style.left=e.pageX+'px';
      stranger.style.top =e.pageY+'px'
    }
  </script>
</body>
```

```
<head>
  <meta charset="utf-8" />
  <title>Random color</title>
  <style>
    div {
      background: navy;
      height: 255px;
      width: 255px;
    }
  </style>
</head>
<body >
  <div id="colorBlock"></div>
  <script>
    colorBlock.addEventListener("click", function() {
      colorBlock.style.backgroundColor =
        "rgb(" +
        Math.round(255*Math.random()) +
        "," +
        Math.round(255*Math.random()) +
        "," +
        Math.round(255*Math.random()) +
        ")" +
        ";
    })
  </script>
</body>
```

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Random color</title>
    <style>
      body {
        margin: 0;
      }
      div {
        background: navy;
        height: 255px;
        width: 255px;
      }
    </style>
  </head>
  <body >
    <div id="colorBlock"></div>
    <script>
      colorBlock.addEventListener("mousemove",
        function(e) {
          colorBlock.style.backgroundColor =
            "rgb(127," + e.pageX + "," + e.pageY + ")" ;
        })
    </script>
  </body>
</html>
```



В следующей таблице приводится описание основных событий интерфейса пользователя.

**auxclick** - Щелчок дополнительной кнопки мыши (если есть)

**click** - Щелчок левой кнопкой

**dblclick** - Двойной щелчок

**Mousedown** - Нажата кнопка мыши

**keydown** - Нажата кнопка клавиатуры

**blur** - Элемент потерял фокус ввода

**Focus** - Элемент получил фокус ввода

**Wheel** - Повернуто колесо мыши

**Focusin** - Посылается перед получением фокуса

**Focusout** - Посылается перед потерей фокуса

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Defaults</title>
  </head>
  <body>
    <h1>Click on page and press a key</h1>
    <h2 id="out"></h2>
    <script>
      document.body.onkeydown = function(e) {
        var txt = "" ;
        if(e.ctrlKey) txt += "Ctrl - " ;
        if(e.altKey) txt += "Alt - " ;
        if(e.shiftKey) txt += "Shift - " ;
        txt += e.key ;
        out.innerText = txt ;
      }
    </script>
  </body>
</html>
```

ВОПРОСЫ

ПРАКТИКА



Давайте подведем итоги.