



JS

JavaScript

лекція 7

Робота з DOM, об'єкт document, базова модель подій

Властивості і методи об'єкта document

- document - відповідає за взаємодію JavaScript з об'єктом документа(сторінки). Являється кореневим елементом DOM-дерева (об'єктом типу DOMDocument згідно специфікації DOM)

Властивості і методи об'єкта document

- Методи

- clear() - призначений для очистки поточного документа з вікна браузера
- close() - закриває документ і змушує негайно оновити свій вміст після використання write
- write() - записує текст або елемент переданий як параметр
- writeln() - записує строку і добавляє символ повернення каретки і
- createAttribute() - генерує об'єкт атрибута і повертає посилання на нього
- setAttributeNode() - добавляє новий атрибут
- createElement() - повертає посилання на щойно створений елемент
- elementFromPoint() - повертає посилання на елемент, що знаходиться за координатами X,Y
- getElementById() - повертає елемент для котрого ім'я передано в якості параметра
- getSelection() - повертає виділений текст на сторінці

Властивості і методи об'єкта document

- Властивості
 - document.inputEncoding - повертає кодування, що й під час синтаксичного розбору (парсинга) документа.
 - document.lastModified - повертає строку, що містить дату і час останньої зміни поточного документа
 - document.readyState - повертає рядок, що містить статус поточного документа.
 - document.referrer - повертає рядок, що містить адресу (URL) сторінки, з якої користувач прийшов на цю сторінку
 - document.cookie - дозволяє отримати або встановити cookie, які будуть пов'язані з поточним документом.
 - document.domain - повертає рядок, що містить доменне ім'я сервера, з якого завантажений поточний документ.
 - document.URL - повертає рядок, що містить повний URL адреса поточного HTML документа
 - forms, frames, images, links, scripts, plugins, embeds, applets
 - width, height, fileSize
 - title, body

Доступ до окремих елементів сторінки за допомогою властивостей об'єкта document

- body, title, forms, images, links, anchors, applets, embeds

Доступ до атрибутів елементів сторінки

- `elem.hasAttribute (name)` - перевіряє наявність атрибута
- `elem.getAttribute (name)` - отримує значення атрибута
- `elem.setAttribute (name, value)` - встановлює атрибут
- `elem.removeAttribute (name)` - видаляє атрибут

- Відмінності від властивостей
 - Завжди являються строками
 - Ім'я не чутливе до регістру

Базова модель подій

- `onblur` - Виникає при втраті елементом фокусу введення (коли користувач активізує інший елемент, зазвичай або за допомогою клацання на ньому, або за допомогою клавіші табуляції)
- `onchange` - не сфокусоване введення полем форми, коли значення поля було модифіковано під час доступу до нього
- `onclick` - Виконання клацання на елементі
- `ondblclick` - Виконання подвійного клацання на елементі
- `onfocus` - Отримання фокусу введення елементом (наприклад, при виборі елемента користувачем для маніпуляцій або введення даних)
- `onkeydown` - Натискання клавіші в той час, коли елемент має фокус введення
- `onkeyup` - Звільнення раніше натиснутою клавіші в той час, коли елемент має фокус введення
- `onload` - Закінчення завантаження об'єкта (як правило, вікна або фрейму) в браузер
- `onmousedown` - Натискання кнопки миші в той час, коли елемент має фокус введення
- `onmousemove` - Рух покажчика миші, коли він знаходиться на елементі
- `onmouseout` - Зсув покажчика миші з елемента (в результаті чого покажчик виявляється за межами даного елементу)
- `onmouseover` - Приміщення покажчика миші на елемент
- `onmouseup` - Звільнення раніше утримуючи кнопки миші в той час, коли елемент має фокус введення
- `onreset` - Повернення форми в початковий стан (наприклад, за допомогою натискання кнопки "Скидання") `<form>`
- `onselect` - Вибір тексту користувачем (як правило, шляхом підсвічування тексту за допомогою миші) `<input>`, `<textarea>`
- `onsubmit` - Відправка форми сервера (зазвичай в результаті клацання на кнопці "Подача запиту") `<form>`
- `onunload` - Вивантаження браузером поточного документа з вікна або фрейму `<body>`, `<frameset>`

Нестандартні обробники подій

- http://js-help.net/text/obrabotka_sobitij/bazovaya_model/nestandardnie_sobitiya.php
- Нестандартна прив'язка подій
- `<p id="myParagraph">Указатель попадает на этот текст!</p>
<script type="text/jscript" for="myParagraph" event="onmouseover">
 alert("Нестандартная разметка документа создает проблемы \n
и для разработчика, и для пользователя!");
</script>`

Застосування коду в якості обробника подій

- Призначення обробників
 - Через HTML-атрибут
 - Через властивість об'єкта
 - Спеціальні методи
 - `element.addEventListener(имя_события, обработчик, фаза)`
 - `element.removeEventListener(имя_события, обработчик, фаза)`

Реєстрація функцій-обробників в якості властивості елемента, ключове слово `this`

- `link`

Передача посилань на подію та елемент, в котрому воно виникло

- Об'єкт event
 - Передається як перший параметр
 - window.event

```
function doSomething(event) {  
    event = event || window.event  
}  
element.onclick = doSomething
```

Завдання

- Створити HTML сторінку та призначити обробник події завершення завантаження цієї сторінки
- На сторінці створити кнопку та призначити обробник події onclick трьома різними способами