**Модальное окно на HTML5 и CSS3**

Модальные окна - часто используемый инструмент в арсенале веб мастера.  Он очень удачно подходит для решения большого количества задач, таких как вывод форм регистрации, рекламных блоков, сообщений и так далее.

Но, несмотря на важное место в информационном обеспечении проекта, модальные окна, как правило, реализуются на JavaScript, что может создавать проблемы при расширении функционала или обеспечении обратной совместимости.

HTML5 и CSS3 позволяют создавать модальные окна с необычайной легкостью.

**Разметка HTML**

Первый шаг на пути к созданию модального окна - простая и эффективная разметка:

|  |  |
| --- | --- |
| 1 | <a href="#openModal">Открыть модальное окно</a> |
| 2 |   |

|  |  |
| --- | --- |
| 3 | <div id="openModal" class="modalDialog"> |
| 4 |       |

|  |  |
| --- | --- |
| 5 | </div> |

Мы просто создаем ссылку “Открыть модальное окно”, которая указывает на элемент div *openModal,* который размещен ниже. Внешний вид формируется стилями в классе *modalDialog.*

Внутри элемента div размещается содержание модального окна. Также нужно организовать ссылку для закрытия окна. Для примера разместим простой заголовок и несколько параграфов. Полная разметка для нашего примера будет выглядеть следующим образом:

|  |  |
| --- | --- |
| 01 | <a href="#openModal">Открыть модальное окно</a> |
| 02 |   |

|  |  |
| --- | --- |
| 03 | <div id="openModal" class="modalDialog"> |
| 04 |     <div> |

|  |  |
| --- | --- |
| 05 |         <a href="#close" title="Закрыть" class="close">X</a> |
| 06 |         <h2>Модальное окно</h2> |

|  |  |
| --- | --- |
| 07 |         <p>Пример простого модального окна, которое может быть создано с использованием CSS3.</p> |
| 08 |         <p>Его можно использовать в широком диапазоне, начиная от вывода сообщений и заканчивая формой регистрации.</p> |

|  |  |
| --- | --- |
| 09 |     </div> |
| 10 | </div> |

**Стили**

Создаем класс  *modalDialog* и начинаем формировать внешний вид:

|  |  |
| --- | --- |
| 01 | .modalDialog { |
| 02 |     position: fixed; |

|  |  |
| --- | --- |
| 03 |     font-family: Arial, Helvetica, sans-serif; |
| 04 |     top: 0; |

|  |  |
| --- | --- |
| 05 |     right: 0; |
| 06 |     bottom: 0; |

|  |  |
| --- | --- |
| 07 |     left: 0; |
| 08 |     background: rgba(0,0,0,0.8); |

|  |  |
| --- | --- |
| 09 |     z-index: 99999; |
| 10 |     -webkit-transition: opacity 400ms ease-in; |

|  |  |
| --- | --- |
| 11 |     -moz-transition: opacity 400ms ease-in; |
| 12 |     transition: opacity 400ms ease-in; |

|  |  |
| --- | --- |
| 13 |     display: none; |
| 14 |     pointer-events: none; |

|  |  |
| --- | --- |
| 15 | } |

Наше окно будет иметь фиксированное положение, то есть смещаться вниз, если вы будете прокручивать страницу при открытом окне. Также наш фон черного цвета  будет расширяться на весь экран.

Фон будет иметь небольшую прозрачность, а также располагаться поверх всех остальных элементов за счет использования свойства *z-index*.

В завершении устанавливаем переходы для вывода нашего модального окна и скрываем его в неактивном состоянии.

Может быть вы не знаете свойство *pointer-events*, но оно позволяет контролировать как элементы будут реагировать на нажатие кнопки мыши. Мы устанавливаем значение его значение для класса *modalDialog*, так как ссылка не должна реагировать на нажатие кнопки мыши когда активен псевдо класс *“:target”*.

<http://paulradzkov.com/2012/pointer-events/> - о *pointer-events*

Теперь добавляем псевдо класс *:target* и стили для модального окна.

|  |  |
| --- | --- |
| 01 | .modalDialog:target { |
| 02 |     display: block; |

|  |  |
| --- | --- |
| 03 |     pointer-events: auto; |
| 04 | } |

|  |  |
| --- | --- |
| 05 |   |
| 06 | .modalDialog > div { |

|  |  |
| --- | --- |
| 07 |     width: 400px; |
| 08 |     position: relative; |

|  |  |
| --- | --- |
| 09 |     margin: 10% auto; |
| 10 |     padding: 5px 20px 13px 20px; |

|  |  |
| --- | --- |
| 11 |     border-radius: 10px; |
| 12 |     background: #fff; |

|  |  |
| --- | --- |
| 13 |     background: -moz-linear-gradient(#fff, #999); |
| 14 |     background: -webkit-linear-gradient(#fff, #999); |

|  |  |
| --- | --- |
| 15 |     background: -o-linear-gradient(#fff, #999); |
| 16 | } |

Псевдо класс *target* изменяет режим вывода элемента, поэтому наше модальное окно будет выводиться при нажатии на ссылку. Также мы изменяем значение свойства  *pointer-events*.

Мы определяем ширину модального окна и положение на странице. Также определяем градиент для фона и скругленные углы.

**Закрываем окно**

Теперь нужно реализовать функционал закрытия окна. Формируем внешний вид кнопки:

|  |  |
| --- | --- |
| 01 | .close { |
| 02 |     background: #606061; |

|  |  |
| --- | --- |
| 03 |     color: #FFFFFF; |
| 04 |     line-height: 25px; |

|  |  |
| --- | --- |
| 05 |     position: absolute; |
| 06 |     right: -12px; |

|  |  |
| --- | --- |
| 07 |     text-align: center; |
| 08 |     top: -10px; |

|  |  |
| --- | --- |
| 09 |     width: 24px; |
| 10 |     text-decoration: none; |

|  |  |
| --- | --- |
| 11 |     font-weight: bold; |
| 12 |     -webkit-border-radius: 12px; |

|  |  |
| --- | --- |
| 13 |     -moz-border-radius: 12px; |
| 14 |     border-radius: 12px; |

|  |  |
| --- | --- |
| 15 |     -moz-box-shadow: 1px 1px 3px #000; |
| 16 |     -webkit-box-shadow: 1px 1px 3px #000; |

|  |  |
| --- | --- |
| 17 |     box-shadow: 1px 1px 3px #000; |
| 18 | } |

|  |  |
| --- | --- |
| 19 |   |
| 20 | .close:hover { background: #00d9ff; } |

Для нашей кнопки устанавливаем фон и положение текста. Затем позиционируем кнопку, делаем ее круглой с помощью свойства скругления рамки и формируем легкую тень. При наведении курсора мыши на кнопку будем изменять цвет фона.

Полезные ссылки:

<http://shpargalkablog.ru/2011/06/css-box-shadow.html>

<http://shpargalkablog.ru/2011/02/zakruglennye-ugly-css.html>

<http://shpargalkablog.ru/2013/02/opacity-css.html>

<http://shpargalkablog.ru/2011/04/kak-sdelat-gradientes-na-css.html>

<https://webref.ru/css/target>