


<b>Школа:</b>		
<b>Дата:</b>	<b>ФИО учителя:</b>	
<b>Класс:</b>	<b>Участвовали:</b>	<b>Не участвовали:</b>
<b>Тема урока:</b> Ввод и вывод данных, операторы управления.		
<b>Цели обучения, которые достигаются на данном уроке</b>	знать и использовать операторов управления, ввод и вывод данных	
<b>Воспитание ценностей</b>	Воспитание эмоционально-положительной направленности на практическую деятельность, интереса к информатике, личной ответственности за результаты своей работы	
<b>Предварительные знания</b>	Учащиеся уже разрабатывали проекты в IDE и на этом уроке повторяют ее возможности. Активизация уже имеющихся знаний осуществляется через групповую работу	
<b>Межпредметные связи</b>	Программирование	
<b>Запланированные этапы урока</b>	<b>Запланированная деятельность на уроке</b>	<b>Ресурсы</b>
<b>Начало урока</b> <u>5</u> мин	<p>Приветствие, проверка присутствующих. Объявление темы и целей урока. Делим детей на 3 группы при помощи игры Random</p>  <p>Актуализация знаний «Мозговой штурм». (5 минут). Метод «Толстые и тонкие вопросы» (для начала беседы по изучаемой теме).</p>	Презентация, учебник Слайд 1-3
<b>Середина урока</b> <u>30</u> мин	<p>Для взаимодействия с пользователем в JavaScript предусмотрено несколько простейших возможностей в виде диалоговых модальных окон, которые выводятся на экран пользователя при помощи методов <code>alert()</code>, <code>confirm()</code> и <code>prompt()</code> объекта <code>Window</code> (более подробно о понятиях 'метод' и 'объект' мы поговорим позже).</p> <p><b>Модальное окно</b> – это окно, которое приостанавливает выполнение сценария и дальнейшее взаимодействие со страницей до тех пор, пока оно не будет закрыто пользователем.</p> <p style="text-align: center;"><b>Метод <code>alert()</code></b></p> <p>С методом <code>alert()</code>, который принимает в качестве аргумента строку, мы уже познакомились в предыдущих пунктах данного параграфа. Метод выводит на экран модальное окно с каким-либо сообщением. При этом выполнение скрипта будет приостановлено до тех пор, пока пользователь не нажмет на кнопку 'ОК' (см. пример №1).</p>	

```
1
2
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <!-- Сообщаем браузеру кодировку документа -->
7   <meta charset="utf-8">
8   <!-- Не забываем про заголовок документа -->
9   <title>Вывод сообщения при помощи метода alert()</title>
10 </head>
11 <body>
12   <!-- Используем встроенный сценарий -->
13   <script>
14     alert('Нажмите "ОК", чтобы продолжить!');
15   </script>
16   <p>
17     <dfn>Модальное окно</dfn> – это окно, которое пр
18     выполнение сценария и <br>дальнейшее взаимодейст
19     до тех пор, пока оно не будет закрыто пользовате
20   </p>
21 </body>
22 </html>
```

**Пример №1. Вывод сообщения при помощи метода alert()**

### Метод confirm()

Если необходимо, чтобы пользователь не просто увидел сообщение или вопрос, но и мог как-то на него отреагировать, используется метод `confirm()`, который дополнительно имеет две кнопки 'OK' и 'Cancel' (см. пример №2).

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <!-- Сообщаем браузеру кодировку документа -->
5   <meta charset="utf-8">
6   <!-- Не забываем про заголовок документа -->
7   <title>Вывод модального окна при помощи метода con
8 </head>
9 <body>
10   <script>
11     //Передаем методу в качестве аргумента строку,
12     //а результат присвоим переменной
13     var answer = confirm('Хотите изучить JavaScript?');
14
15     //Если окно вернет true, то
16     if(answer==true){
17       //выведем сообщение,
18       alert('Так держать!');
19     }else{
20       //выведем такое сообщение
```

```

19     alert('Ну, тогда флаг вам в руки и пока!');
20     }
21     </script>
22 </body>
23 </html>
24
25
26
27
28

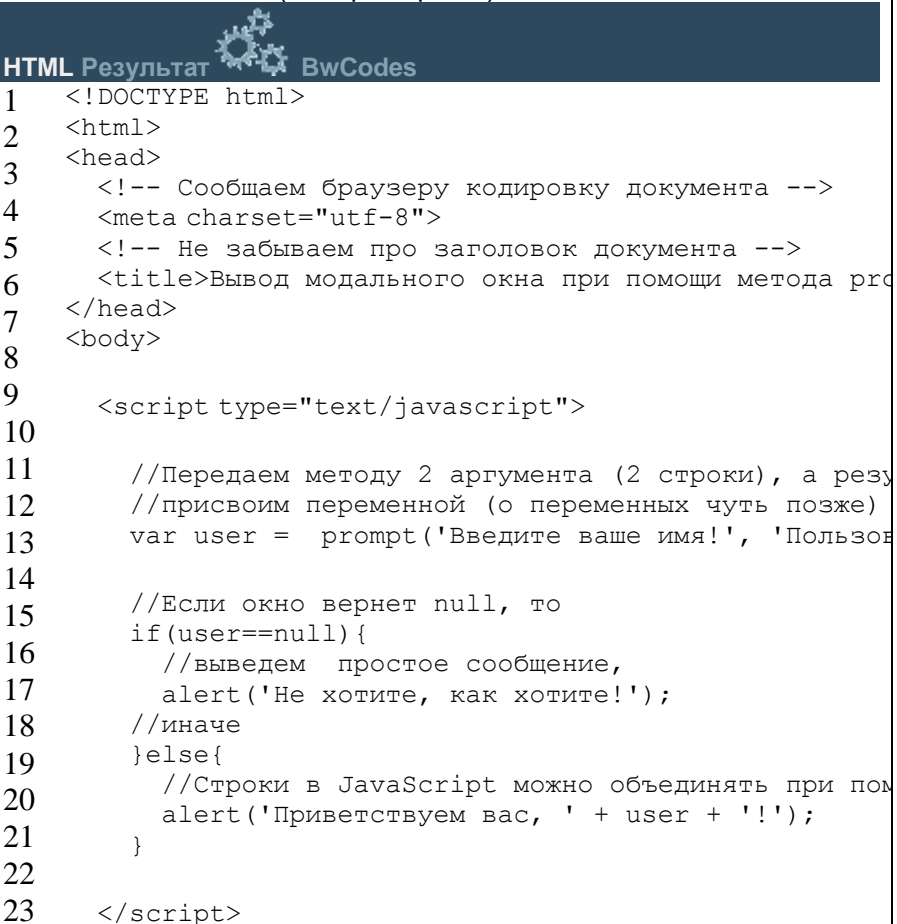
```

### **Пример №2. Вывод модального окна при помощи метода confirm()**


В качестве аргумента метод `confirm()` принимает строку, например, вопрос, на который нужно ответить утвердительно или отрицательно (можно вообще аргумента не передавать, но тогда будет выведено пустое модальное окно без заголовка с двумя вышеуказанными кнопками). Если пользователь нажмет кнопку 'OK', метод вернет логическое значение `true`, а если воспользуется кнопкой 'Cancel', то будет возвращено логическое значение `false` (о логических значениях мы поговорим позже, пока же будем понимать их как 'да' и 'нет').

### **Метод prompt()**

Иногда возникает необходимость, чтобы пользователь ввел в окне какие-либо данные. Для таких случаев предусмотрен метод `prompt()`, создающий модальное окно, которое не только выводит сообщение (в виде заголовка этого окна), но также имеет поле для ввода текста и две кнопки 'OK' и 'Cancel' (см. пример №3).



```

HTML Результат  BwCodes
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <!-- Сообщаем браузеру кодировку документа -->
5   <meta charset="utf-8">
6   <!-- Не забываем про заголовок документа -->
7   <title>Вывод модального окна при помощи метода prompt()</title>
8 </head>
9 <body>
10
11   <script type="text/javascript">
12     //Передаем методу 2 аргумента (2 строки), а резу
13     //присвоим переменной (о переменных чуть позже)
14     var user = prompt('Введите ваше имя!', 'Пользов
15
16     //Если окно вернет null, то
17     if(user==null){
18       //выведем простое сообщение,
19       alert('Не хотите, как хотите!');
20     }
21     //иначе
22     else{
23       //Строки в JavaScript можно объединять при пом
24       alert('Приветствуем вас, ' + user + '!');
25     }
26   }
27 </script>

```

	<pre> 24 25 &lt;/body&gt; 26 &lt;/html&gt; 27 28 29 30 </pre> <p><b>Пример №3. Вывод модального окна при помощи метода <code>prompt()</code></b>  В качестве аргументов метод <code>prompt()</code> принимает через запятую две строки: первая - служит заголовком окна, а вторая - задает начальное содержимое поля ввода (значение по умолчанию). Если их не указать, то будет выведено модальное окно без заголовка и с пустым полем ввода (запустите пример и попробуйте различные варианты). Далее, если пользователь нажмет кнопку 'OK', то метод вернет значение, которое указано в данный момент в поле ввода (это может быть как введенное пользователем значение, так и значение по умолчанию). Если же пользователь воспользуется кнопкой 'Cancel', то будет возвращено специальное значение <code>null</code> (само значение <code>null</code> мы рассмотрим позже, пока же будем понимать его как 'пустое значение').</p> <p>Из описанных выше методов отдельно хотелось бы обратить внимание на метод <code>alert()</code>, поскольку его мы будем довольно часто использовать при написании кода благодаря его особенности приостанавливать дальнейшее выполнение скрипта (что может быть очень полезным при его отладке). В общем же случае мы не будем использовать данные встроенные методы. В место этого мы будем создавать собственные диалоговые окна нужной нам конфигурации и вида.</p>	
<p><b>Конец урока</b> <b>__5__ мин</b></p>	<p>Домашнее задание:</p> <p>С помощью метода модального окна написать программу</p>	
<p><b>Дифференциация – каким образом Вы планируете оказать больше поддержки? Какие задачи Вы планируете поставить перед более способными учащимися?</b></p>	<p><b>Оценивание – как Вы планируете проверить уровень усвоения материала учащимися?</b></p>	<p><b>Охрана здоровья и соблюдение техники безопасности</b></p>
<p>На этапе закрепления менее способным учащимся предлагаются различные подмости.</p>	<p>Формативное оценивание ранее полученных знаний. На этапе выполнения практической работы учащиеся оцениваются по критериям.</p>	<p>Соблюдение техники безопасности при передвижении учащихся по классу во время групповой работы «Карусель», ТБ при работе за компьютером.</p>