

Лабораторная работа 7. Первые скрипты

Язык JavaScript

JavaScript – одно из тех средств, которые нужны для динамики HTML-языка. С JavaScript гипертексты получают встроенную программу, запускающую HTML с новыми возможностями. Это язык программирования, который понятен браузеру. Это означает, что браузер умеет выполнять (интерпретировать) команды этого языка.

Программу на JavaScript (ее называют *скриптом*, или сценарием) можно помещать внутрь HTML-программы или держать в отдельном файле. Этот файл браузер прочитает (по специальной команде) во время интерпретации HTML-кода.

Коды JavaScript дополняют коды HTML и работают вместе с ними. Даже если они расположены в отдельном файле.

Скрипт размещается между двумя парными тегами <SCRIPT>. . .</SCRIPT> Обычно запись скрипта выглядит так, как показано в примере 1.

Пример 1

```
<HTML>
  <HEAD>
    <TITLE>Проверка браузера</TITLE>
  </HEAD>
  <BODY bgcolor=white text=black link=blue alink=red vlink=purple>
    <H1>Проверка браузера</H1>
    <HR>
    <SCRIPT language=JavaScript>
      <!--
        alert("Ваш браузер поддерживает язык JavaScript!");
      //-->
    </SCRIPT>
    <NOSCRIPT>
      <H2>Предупреждение</H2>
      <P>
        Ваш браузер не поддерживает JavaScript. Этот документ будет
        показан неверно, возможны сообщения об ошибках.
      </P>
      <P> Версия документа без скриптов расположена
        <A href=doc1.htm>Здесь</A>.
      </P>
    </NOSCRIPT>
    <P>
      Для возврата к основному тексту нажмите кнопку <EM>Назад</EM>
      на инструментальной панели браузера.
    </P>
  </BODY>
</HTML>
```

Скрипт в приведенном выше примере содержит всего одну команду на языке JavaScript – вызов функции alert. Функция alert создает информационное окно с единственной кнопкой ОК внутри (рис. 5.1). Пользователь читает сообщение, нажимает кнопку и окно убирается с экрана. Теперь браузер приступает к интерпретации команд, расположенных вслед за скриптом в HTML-коде – выводит в окошко текст, показанный на рис. 5.2.

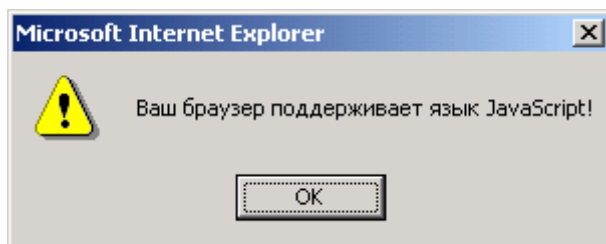


Рис. 5.1. Информационное окно alert

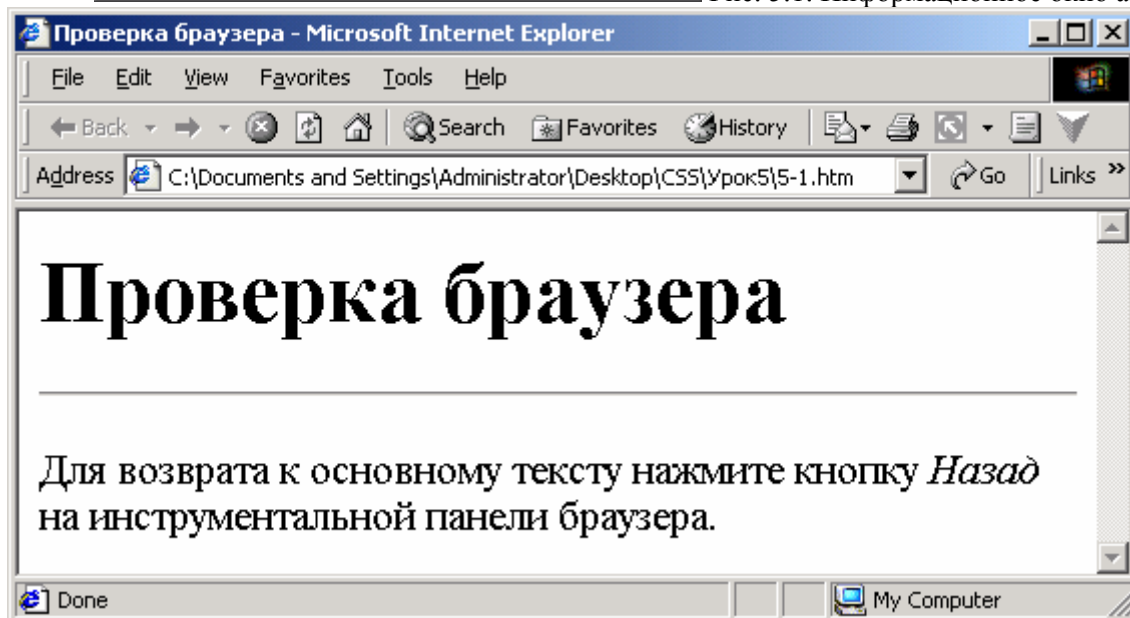


Рис. 5.2. Вид страницы в браузере, который поддерживает скрипты

Если браузер не поддерживает JavaScript или эта поддержка отключена в настройках, в окно документа выводится сообщение, показанное на рис. 5.3.

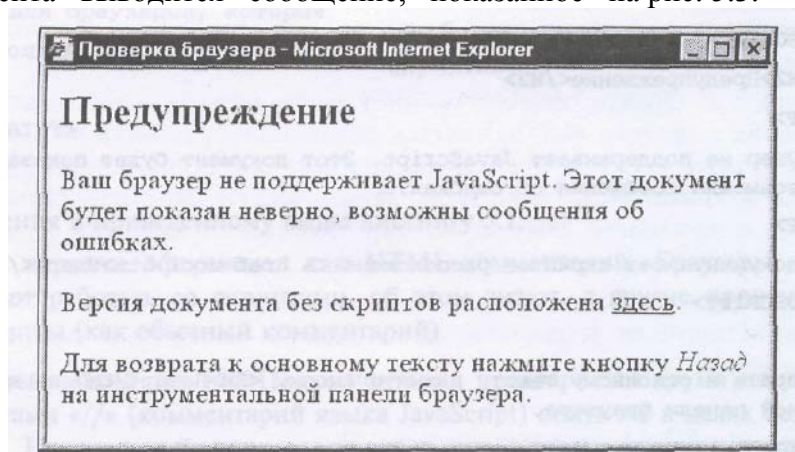


Рис. 5.3. Вид страницы в браузере, который не поддерживает скрипты

Если текст-аргумент функции alert слишком длинный, его можно записывать в нескольких строчках, используя знак «+» (операция соединения):

alert("Язык JavaScript был разработан в 1995 году фирмой"+ " Netscape для второй версии своего браузера");

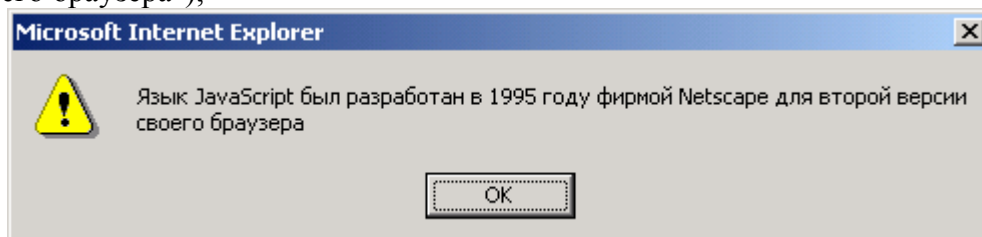


Рис. 5.4. Для записи длинной строки использована операция соединения

Операция соединения не означает разрыв строки на экране (рис. 5.4). Для перевода строк нужно использовать символы «\n»:

```
alert ("Первая строка.\nСтрока вторая.");
```

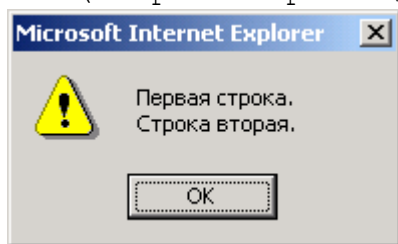


Рис. 5.5. Вывод текста в две строки

Задания

Проведите серию опытов с функцией alert. Задайте вывод сообщения (рис. 5.6).

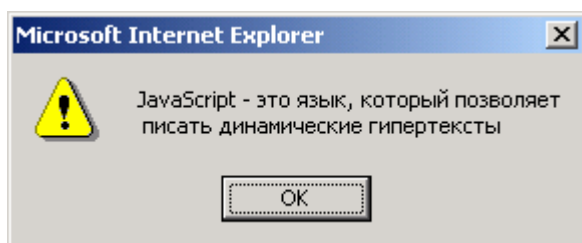


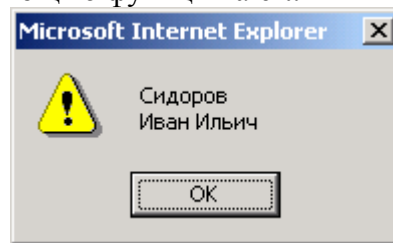
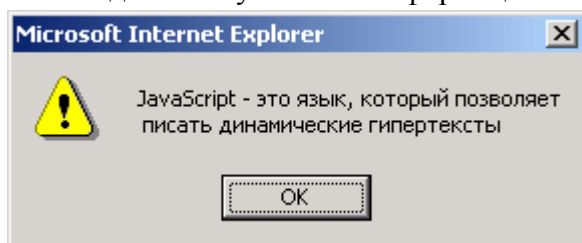
Рис. 5.6. Сообщение в информационном окне

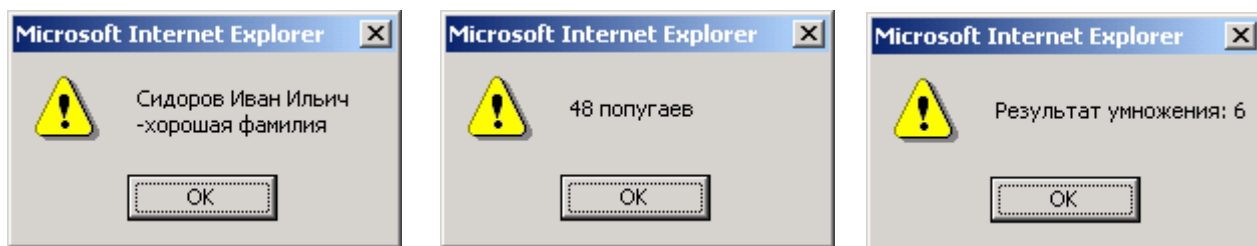
Попробуйте другие тексты, символы перевода строк, запись одной функции на нескольких строчках, запись нескольких функций. Для опытов используйте следующую заготовку (листинг 5.3).

Листинг 5.3

```
<HTML>
  <HEAD>
    <TITLE><Опыты с alert </TITLE>
  </HEAD>
  <BODY bgcolor=white text=black>
    <H2> Опыты с функцией alert </H2>
    <SCRIPT language=JavaScript>
      <! --
        alert ("! ! !") ;
      //-->
    </SCRIPT>
  </BODY>
</HTML>
```

Необходимо получить сл. информационные окна с помощью функции alert:





Переменные и константы

В JavaScript нет описателей типа, подобных `int`, `char` в языке Си или `integer`, `byte` в языке Паскаль. Тип переменной определяется ее текущим значением и может меняться при выполнении программы много раз.

Примеры:

```
x = "3";           // Сейчас переменная x строкового типа,
                   //ее значение — строка «3».
```

```
x = 3;             // Сейчас переменная x целого типа,
                   //ее значение — число 3.
```

```
x = "Оценка=" + x; //Сейчас переменная x строкового типа,
                   //ее значение — строка «Оценка=3».
```

В JavaScript можно использовать константы следующих видов.

Целое число

255 — десятичное целое.

0377 — восьмеричное целое (начинается с «0»).

0xFF — шестнадцатеричное целое (начинается с «0x»).

Дробное число

3.14 — обычная запись.

2.3e8 — «инженерная» запись ($2.3 \cdot 10^8$).

Строка

"Привет!"

'Привет!'

'It's Ok!'

— для кавычек внутри строки используют Указания `\'` и `\"`.

"\BA3 2104\""

"Перва\я строка.\nСтрока втора\я." — внутри записи можно использовать:

`\n` — перевод строки;

'Кинотеатр "Ударник"'

— внутри одинарных кавычек можно

"I'm programmer"

использовать двойные без символа `«\»` и наоборот.

Описание переменных

Переменная должна быть описана до ее использования. Для описаний берется ключевое слово `var`:

```
var x;           // Описание переменной с именем
"x".
```

```
var y = 5;       // Описание с присвоением числа.
```

```
var mes = "дядя Федор"; // Описание с присвоением строки.
```

Для образования имен переменных используются идентификаторы.

Идентификатор — это последовательность из латинских букв и арабских цифр, которая начинается с буквы. В записи разрешается использовать символ подчеркивания «_» наравне с латинскими буквами. Регистр буквы в имени переменной имеет значение. Переменные с именами «Counter» и «counter» считаются разными.

Выражения

Для арифметических операций используются сл. обозначения (табл. 6.1).

Таблица 6.1

Знак	Название	Пример
–	Унарный минус	–10
+	Сложение	X+Y
–	Вычитание	X–10
*	Умножение	2*x
/	Деление	Y/7
%	Остаток от деления	x % 3

Для строковых выражений используется операция соединения (конкатенации), обозначаемая знаком «+»:

person="Крокодил" + " Гена"; – Переменная person получает значение «Крокодил Гена»

x = "2" + 1; – Переменная x получает значение «21».

Заметьте, «21» здесь не число, а строка! Когда операция «+» связывает строку с данными другого типа, результат преобразуется в строку.

Команды

Команда завершается в JavaScript символом «;». Правила языка разрешают опускать этот знак в конце строки и перед закрывающей скобкой «}».

Комментарий

Наряду с уже известным строковым комментарием «//» можно использовать блочный комментарий «/* ... */»:

```
// Это строковый комментарий.  
/* А это  
    блочный комментарий. */
```

Вызов функции

Пока была показана только функция alert:

```
alert(выражение);
```

Аргументом этой функции может быть выражение, имеющее как текстовое, так и числовое значение.

```
var ex1 = "Ищите";  
var ex2 = "обращаете";  
alert(ex1+" и " + ex2);
```

Результат выполнения скрипта показан на рис. 6.1.

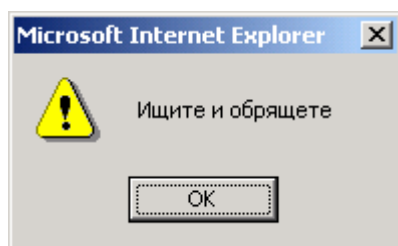
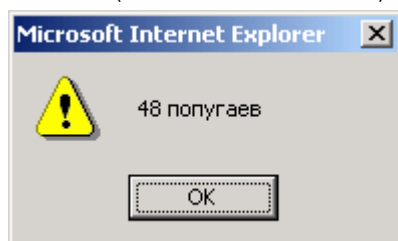


Рис. 6.1. Информационное окно alert

```
var x = 48;  
var ex1 = "попугаев";  
alert(x + " " + ex1);
```



Результат выполнения скрипта показан на рис. 6.2.

Рис. 6.2. Результат выполнения скрипта

```
var x = 2;
var y = 3;
alert("Результат умножения: " + x * y);
```

Результат выполнения скрипта показан на рис. 6.3.

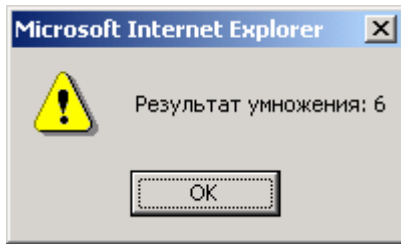


Рис. 6.3. Результат выполнения скрипта

Присваивание

Наряду с привычным присваиванием вида «переменная=выражение», можно использовать следующие записи (табл. 6.2):

Таблица 6.2

Общий вид	Пример	Что означает
переменная ++	x ++	x = x + 1
++ переменная	++ x	x = x + 1
переменная --	x --	x = x - 1
-- переменная	-- x	x = x - 1
переменная += выражение	x += "кот"	x = x + "кот"
переменная -= выражение	x -= 2	x = x - 2
переменная *= выражение	x *= 10	x = x * 10
переменная /= выражение	x /= y + z	x = x / (y+z)
переменная %= выражение	x %= 9	x = x % 9

Разница между командами ++ x и x ++ (-- x и x --) проявляется тогда, когда эти команды используются как составная часть других команд. В первом случае (знаки перед именем) операция выполняется до использования переменной, а во втором – после.

x = 5; — После выполнения этих команд: x и y равны 6.

y = ++x;

x = 5; — После выполнения этих команд: x равно 6, а y равно 5.

y = x++;

Задания

1. Что будет выведено в окно функции alert в результате выполнения следующих команд: число 10, число 35 или сообщение об ошибке?

а) // var x = 10;
var x = 35;
// var x = 8;
alert(x);

б) /* var x = 10;
var x = 35;
/* var x = 8;
alert (x) ;

в) /* var x = 10;
/var x = 35; /
var x = 8;
*/alert(x);

```
г) /* var x = 10;
    var x = 35;
    var x = 8;
    */alert(x);
```

2. Что будет выведено в окно alert в результате выполнения следующих команд?

```
а) var x = "60";
    var y = 20;
    alert (x + y)
```

```
б) var x = "60";
    var y = 20;
    alert (x - y);
```

```
в) var x = "60";
    var y = 20;
    alert (x / y);
```

```
г) var x = "с";
    var y = "ка";
    alert(x + "оба" + y);
```

```
д) var x = 13;
    var y = 3;
    y = ++ x / -- y;
    alert(y);
```

```
е) var x = 13;
    var y = 3;
    y = -- x / ++ y;
    alert (y);
```

```
ж) var x = 13;
    var y = 3;
    y = x ++ % y ++;
    alert(y);
```

```
з) var x = "";
    x += "обед";
    alert("п" + x + "а");
```

```
и) var x = "м";
    x += "а";
    x = "рам" + x;
    var y = "рог";
    alert("п" + y + x);
```

```
к) var str1 = "Все мы немножко лошади, \n";
    var str2 = "Каждый из нас по-своему лошадь.";
    alert(str1 + str2);
```

```
л) var str1 = "Результат вычислений:\n";
    var num1 = 35;
    var num2 = 8;
    alert(str1 + num1 / num2);
```

```
м) var str1 = "Результат вычислений:\n";
    var num1 = 35;
    var num2 = 8;
    alert(str1 + num1 % num2);
```

```
н) var str1 = "Результат вычислений:\n";
    var num1 = 35;
    var num2 = 8;
    alert(str1 + (num1 ++)(num2 ++));
```

```

o) var str1 = "Результат вычислений:\n";
   var num1 = 35;
   var num2 = 8;
   alert(str1 + (num1++) % (num2++));
п) var str1 = "Результат вычислений:\n";
   var num1 = 12;
   var num2 = 8;
   alert(str1 + num1 + "+" + num2 + "=" + num1 + num2);
р) var str1 = "Результат вычислений:\n";
   var num1 = 12;
   var num2 = 8;
   alert(str1 + num1 + "+" + num2 + "=" + (num1 + num2));
с) var str1 = "Результат вычислений:\n";
   var num1 = 12;
   var num2 = 8;
   alert(str1 + num1 + "*" + num2 + "=" + num1 * num2);
т) var str1 = "Результат вычислений:\n"; з
   var str2 = 4;
   var num1 = 12;
   var num2 = 8;
   alert(str1+num1+"*"+num2+"+"str2+"="+num1*num2+str2);
у) var str1 = "Результат вычислений:\n";
   var str2 = 4; var num1 = 12; var num2 = 8;
   alert(str1+num1+"*"+4-num2+"+"str2+"="+num1*num2+str2));
ф) var str1 = "10";
   var str2 = "5";
   alert(str1 / str2);
х) var str1 = "Этажерка";
   var str2 = 2;
   alert(str1 / str2);

```