

ПОНЯТИЕ СЦЕНАРИЯ

Файлы Web-страниц, написанные на языке **HTML** (*Hypertext Markup Language* — *язык разметки гипертекста*), определяют внешний вид в окне браузера таких элементов, как текст, таблицы и изображения. При этом HTML работает только со статическими элементами.

JavaScript — это *язык сценариев* (*scripting language*), который может применяться в файлах Web-страниц наряду с HTML, например, для придания Web-страницам свойств диалоговых окон.

Эти интерактивные возможности позволяют разработчику динамически управлять элементами

Web-страниц. JavaScript можно также применять для изменения элементов Web-страницы при взаимодействии с ней пользователя. Например, можно предусмотреть, чтобы текст, отображаемый в текстовом поле формы, изменялся, когда пользователь щелкает на кнопке.

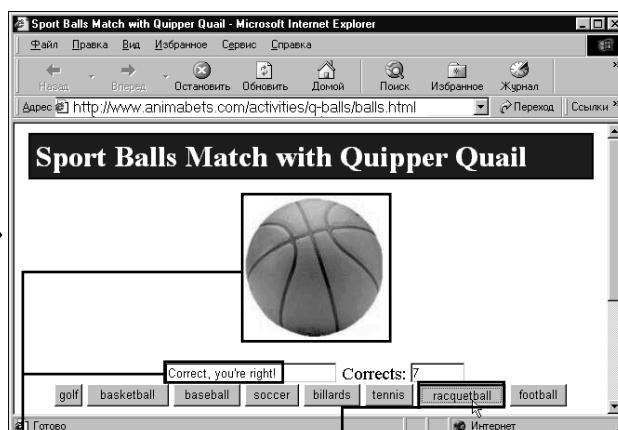
Сценарий JavaScript может выполняться в браузере без каких-либо дополнительных программных средств.

ПОНЯТИЕ СЦЕНАРИЯ



1 Откройте Web-страницу, например www.animabets.com/activities/q-balls/balls.html, окно браузера которой включает сценарий JavaScript.

2 Щелкните на одной из кнопок.



3 В текстовом окне отображается результат, и браузер выводит новое изображение.

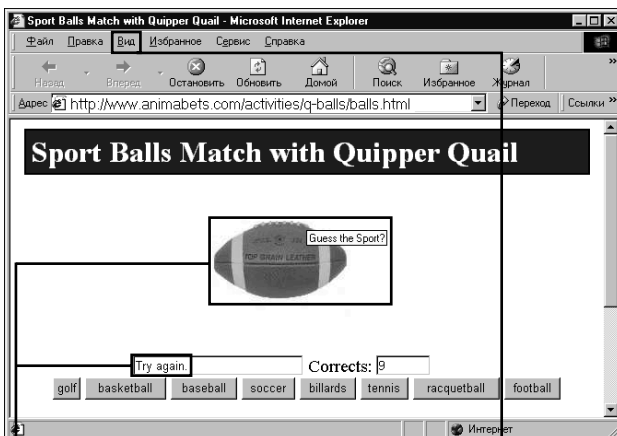
Щелкните на другой кнопке.

Замечание. Файл HTML для этой Web-страницы включает синтаксис JavaScript, который определяет способ обновления текстовых окон и изображений.

Что еще?

JavaScript — наиболее популярный язык сценариев для клиентов Web, поскольку он поддерживается браузерами Netscape Navigator и Internet Explorer, однако существуют и другие языки сценариев для обеспечения интерактивных возможностей Web-страниц. Если установить значение атрибута `language` дескриптора `<script>` равным VBScript, то в качестве языка сценариев будет использоваться VBScript.

Синтаксис VBScript подобен синтаксису языка Visual Basic, однако VBScript поддерживается только браузером Internet Explorer. Если вы знакомы с языком программирования Visual Basic, то вам может быть проще изучить VBScript, чем JavaScript. Однако при этом не стоит забывать, что VBScript работает только с браузером Microsoft Internet Explorer.



■ В текстовом окне вновь отображается результат, и браузер выводит новое изображение.

■ 4 Чтобы отобразить исходный текст, выберите команду Вид⇒Виде HTML (View⇒Source).



■ 5 Прокрутите текст на экране, пока не обнаружите дескриптор `<script>`.

■ Обратите внимание, что атрибут `language` установлен равным `javascript`.

Замечание. Логика обновления текстового окна зависит от ответа пользователя.

ВВОД ОПЕРАТОРОВ JAVASCRIPT В ПОЛЕ URL

Сценарии JavaScript состоят из операторов, интерпретация которых приводит к выполнению требуемой функции. Например, оператор JavaScript `document.write("hello")` отображает слово *hello* в окне браузера.

Интерпретация и выполнение операторов JavaScript осуществляются браузером. Эти операторы могут располагаться в файле Web-страницы или непосредственно вводиться в поле URL браузера.

Чтобы заставить браузер выполнить оператор JavaScript, введите ключевое слово `javascript:`

в поле Адрес (Address) браузера, после чего введите оператор JavaScript и нажмите клавишу `<Enter>`.

В конце оператора JavaScript необходимо поместить точку с запятой (;). В поле Адрес можно поместить несколько операторов JavaScript, разделив их точками с запятой.

Например, если ввести оператор `javascript:document.write("hello");`, после нажатия клавиши `<Enter>` браузер отобразит слово "hello".

ВВОД ОПЕРАТОРОВ JAVASCRIPT В ПОЛЕ URL



1 Откройте браузер.



2 В поле Адрес введите ключевое слово `javascript:.`

Что еще?

Обычно после ввода ключевого слова в поле Адрес браузер осуществляет поиск Web-страниц, соответствующих этому ключевому слову. Если необходимо, чтобы браузер нашел Web-страницы JavaScript, а не выполнил оператор JavaScript, просто опустите двоеточие после javascript. Если ввести в поле Адрес ключевое слово javascript: без двоеточия (:), браузер станет искать Web-страницы, соответствующие ключевому

слову javascript. Символ двоеточия сообщает браузеру, что следующий за ним текст необходимо интерпретировать, как оператор JavaScript.

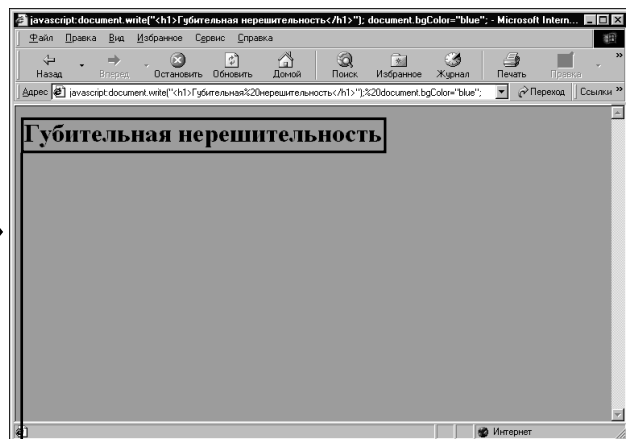
Если вы забыли добавить ключевое слово javascript:, браузер воспримет введенный оператор JavaScript как адрес Web-страницы. Он попытается отыскать по этому адресу Web-страницу, которой может не существовать.



3 После ключевого слова javascript: введите оператор document.write ("<h1>Губительная нерешительность</h1>");.

4 Введите также оператор document.bgColor="blue";.

5 Нажмите клавишу <Enter>.



Оператор JavaScript выполняется в браузере.

ВКЛЮЧЕНИЕ ОПЕРАТОРОВ JAVASCRIPT В ДОКУМЕНТ HTML

Сценарий JavaScript может быть добавлен в Web-страницу между дескрипторами `<script>`. Операторы JavaScript, включенные между дескрипторами `<script>`, выполняются при загрузке Web-страницы в браузер.

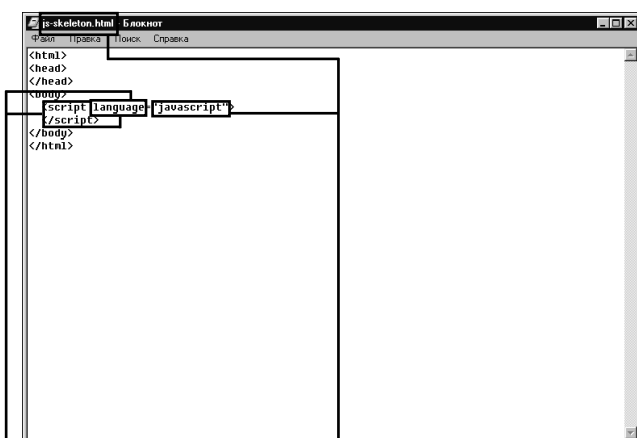
Дескрипторы `<script>` могут располагаться в любом месте между дескрипторами `<head>` или `<body>`. В файле одной Web-страницы может содержаться несколько наборов дескрипторов `<script>`.

Дескриптору `<script>` передаются два атрибута: `type` и `language`. Атрибут `type` может принимать одно из значений, соответствующих известным типам MIME, такое как `text/javascript`. (MIME — Multipurpose Internet Mail Extensions — многоцелевые расширения электронной почты в Internet.

Представляет собой набор стандартов для передачи мультимедийной информации посредством электронной почты. — Прим. ред.) Атрибут `language` — это еще один способ задания используемого языка сценариев. В случае JavaScript этот атрибут просто устанавливается равным `javascript`.

Если оператор JavaScript отображает в окне браузера какой-либо текст, то размещение текста соответствует размещению дескрипторов `<script>` в файле Web-страницы. Например, если HTML-файл содержит заголовок, за которым следует дескриптор `<script>`, предназначенный для вывода некоторого текста, сценарий JavaScript должен следовать непосредственно после заголовка HTML-файла.

ВКЛЮЧЕНИЕ ОПЕРАТОРОВ JAVASCRIPT В ДОКУМЕНТ HTML



1 Создайте файл `skeleton.html`, имеющий следующую структуру: `<html><head></head><body></body></html>`.

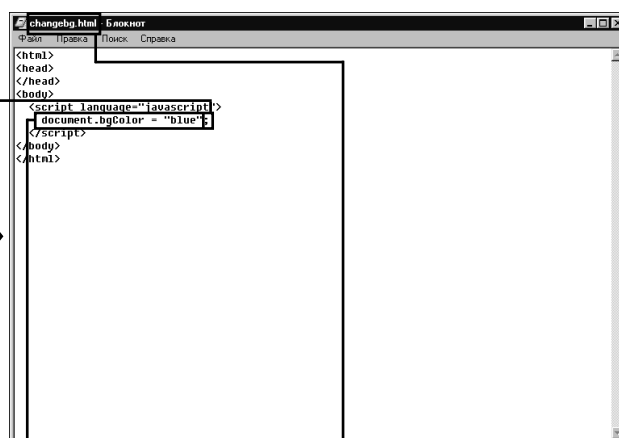
2 Между дескрипторами `<body>` поместите набор дескрипторов `<script>`.

3 К дескриптору `<script>` добавьте атрибут `language`.

4 Присвойте атрибуту `language` значение `javascript`.

5 Сохраните файл под именем `js-skeleton.html`.

Замечание. Эта схема может использоваться для быстрого создания Web-страницы, содержащей JavaScript.



6 Между дескрипторами `<script>` поместите оператор JavaScript для изменения цвета фона.

7 В конце оператора JavaScript добавьте точку с запятой (;).

8 Сохраните файл под именем `changebg.html`.

Замечание. Операторы JavaScript, размещенные между дескрипторами `<script>`, выполняются при загрузке файла.

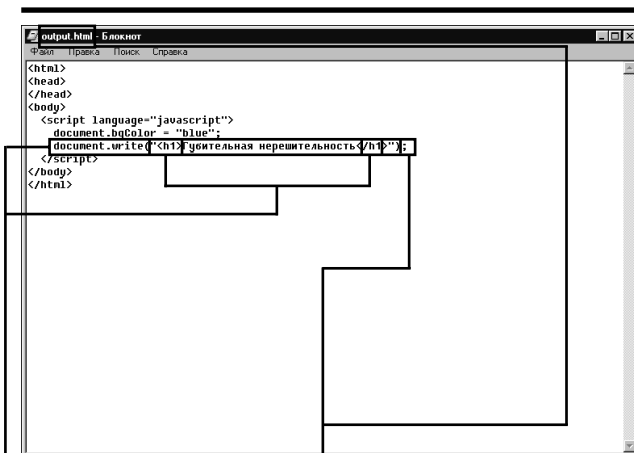
Попробуй!

С помощью JavaScript можно персонифицировать Web-страницу для посетителей вашего Web-узла. Приведенный ниже фрагмент программы предназначен для открытия диалогового окна, в которое пользователи могут ввести свое имя. Затем это имя отображается в Web-странице с помощью еще одного оператора JavaScript.

```
temp = window.prompt("Введите свое имя");

document.write("Добро пожаловать, " + temp
+ "на нашу Web-страницу. Надеемся,
Вам здесь понравится.");
```

Теперь, чтобы этот сценарий заработал, необходимо просто поместить его между дескрипторами `<script>`, а затем открыть файл в браузере.

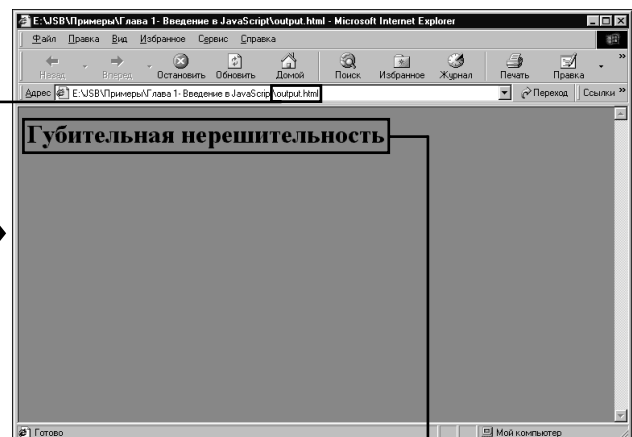


9 Для вывода на экран фразы добавьте еще один оператор JavaScript.

■ Оператор вывода может включать дескрипторы HTML.

10 В конец оператора JavaScript добавьте точку с запятой (;).

11 Сохраните файл под именем `output.html`.



12 Откройте файл `output.html` в браузере.

■ Обратите внимание, что цвет фона окна изменился и в нем отображился текст.

УСТАНОВЛЕНИЕ СВЯЗИ С ВНЕШНИМ ФАЙЛОМ JAVASCRIPT

С дескриптором `<script>` можно использовать еще один атрибут — `src`. Он может указывать на URL внешнего файла, содержащего сценарий JavaScript.

Внешний файл может не включать дескрипторов `<script>` и содержать только операторы JavaScript. Он может также не содержать расширения `html`. Обычно внешний файл JavaScript имеет расширение `js`.

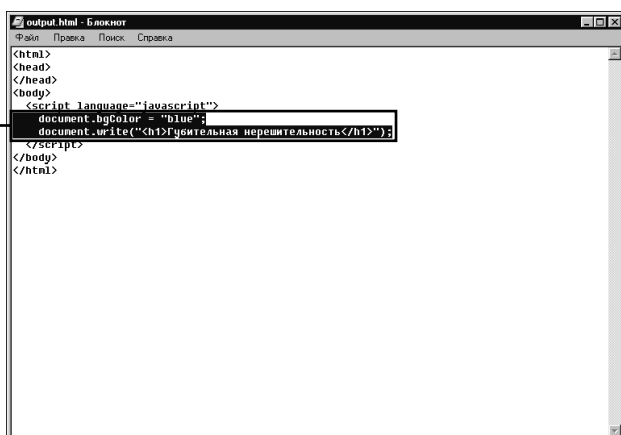
Внешний файл JavaScript может располагаться в любом месте сети Web, и на него можно сослаться

с помощью ключевого слова протокола `http://`, за которым следует Web-адрес внешнего файла.

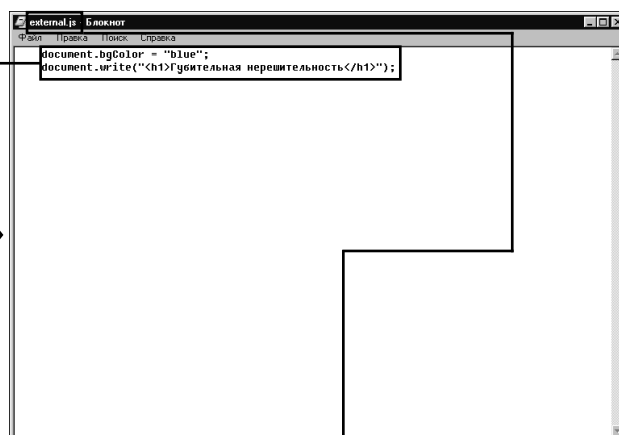
Например, с помощью оператора `<script src="myfile.js">` можно получить доступ к внешнему файлу JavaScript с именем `myfile.js`, расположенному в том же каталоге, что и HTML-файлу.

В приведенном ниже примере используется файл `output.html`, созданный в предыдущем разделе, “Включение операторов JavaScript в документ HTML”, однако соответствующие шаги можно применить к любому файлу.

УСТАНОВЛЕНИЕ СВЯЗИ С ВНЕШНИМ ФАЙЛОМ JAVASCRIPT



- 1 Откройте файл `output.html` с помощью программы Блокнот (Notepad).
- 2 Вырежьте содержимое дескрипторов `<script>` из файла.
- 3 Сохраните файл под именем `external.html`.



СОЗДАНИЕ ВНЕШНЕГО ФАЙЛА СЦЕНАРИЯ

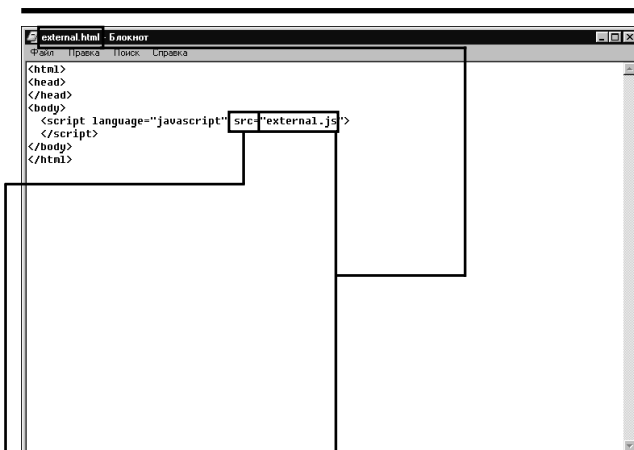
- 1 Откройте с помощью программы Блокнот новый файл.
- 2 Вставьте содержимое дескриптора `<script>`, вырезанное из файла `output.html`, в новый файл.
- 3 Сохраните файл под именем `external.js`.

Что еще?

Синтаксис JavaScript зависит от регистра клавиатуры. Если переменная JavaScript введена прописными буквами, то при обращении к переменной в соответствующих местах также необходимо использовать прописные буквы. Так, например, `hello`, `Hello` и `HELLO` — это разные переменные. Если введенный оператор JavaScript правильно не распознается, проверьте, нет ли орфографических ошибок в ключевых словах. Операторы JavaScript можно

выполнять многократно, поместив их внутрь структуры, которая называется *функцией*. Функции рассматриваются в главе 5.

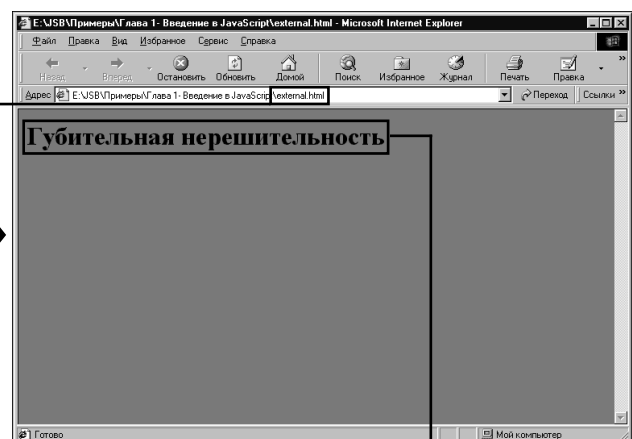
Операторы JavaScript, заключенные между дескрипторами `<head>`, выполняются раньше остальных дескрипторов HTML. Например, если внутри дескрипторов `<head>` расположены дескрипторы `<script>`, содержащие оператор JavaScript, предназначенный для вывода некоторого текста, этот текст отобразится вверху Web-страницы.



УСТАНОВЛЕНИЕ СВЯЗИ С ВНЕШНИМ ФАЙЛОМ

- 1 Откройте с помощью программы Блокнот внешний файл `external.html`.
- 2 Добавьте к дескриптору `<script>` атрибут `src`.

- 3 Установите значение атрибута `src` равным `external.js`.
- 4 Сохраните файл под именем `external.html`, записав его поверх исходного файла `external.html`.



- 5 Откройте файл `external.html` с помощью браузера.

- Операторы JavaScript по-прежнему работают (теперь, как внешний файл).

ПРЕДСТАВЛЕНИЕ СОДЕРЖИМОГО СЦЕНАРИЕВ ДЛЯ БРОУЗЕРОВ, НЕ ПОДДЕРЖИВАЮЩИХ JAVASCRIPT

Пользователи могут отключить поддержку JavaScript. Такая возможность предоставляется им в рамках управления конфигурацией браузера. Если пользователь отключит поддержку JavaScript, его браузер не будет отображать сценариев JavaScript, помещенных в Web-страницу.

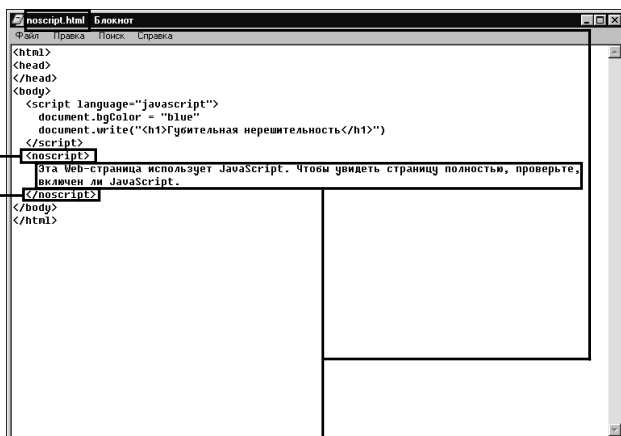
Браузером с отключенной поддержкой JavaScript можно управлять с помощью дескрипторов `<noscript>`. Если сценарий JavaScript не может быть обработан браузером, то в его окне отображается содержимое, заключенное внутри дескрипторов `<noscript>`.

Это содержимое может помочь пользователю, который отключил поддержку браузером JavaScript, понять, что отобразится в окне браузера при включенной поддержке JavaScript.

Например, если в файле Web-страницы содержится фраза “Эта страница использует JavaScript”, заключенная внутри дескрипторов `<noscript>`, она будет отображена (если JavaScript отключен).

В этом примере используется рассмотренный ранее файл `output.html`.

ПРЕДСТАВЛЕНИЕ СОДЕРЖИМОГО СЦЕНАРИЕВ ДЛЯ БРОУЗЕРОВ, НЕ ПОДДЕРЖИВАЮЩИХ JAVASCRIPT

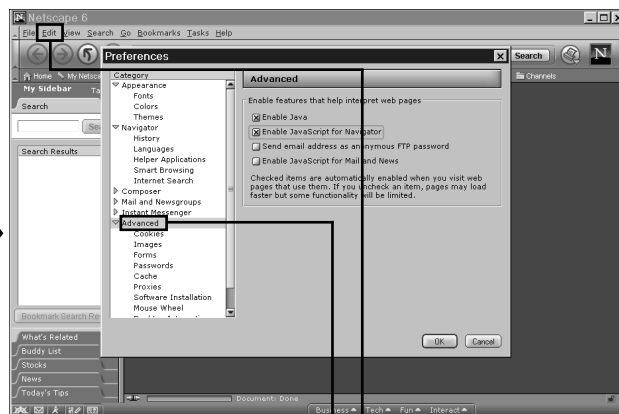


ДОБАВЛЕНИЕ ДЕСКРИПТОРОВ `<NOSCRIPT>` В WEB-СТРАНИЦУ

- 1 Откройте файл `output.html`.
- 2 После дескрипторов `<script>` добавьте набор дескрипторов `<noscript>`.

- 3 Между дескрипторами `<noscript>` разместите поясняющий текст.

- 4 Сохраните файл под именем `noscript.html`.



ОТКЛЮЧЕНИЕ ПОДДЕРЖКИ JAVASCRIPT В NETSCAPE NAVIGATOR

- 1 Откройте программу Netscape Navigator.

Замечание. В браузере Internet Explorer нет специальных средств отключения JavaScript.

- 2 Выберите команду `Edit → Preferences`, чтобы открыть диалоговое окно Preferences.

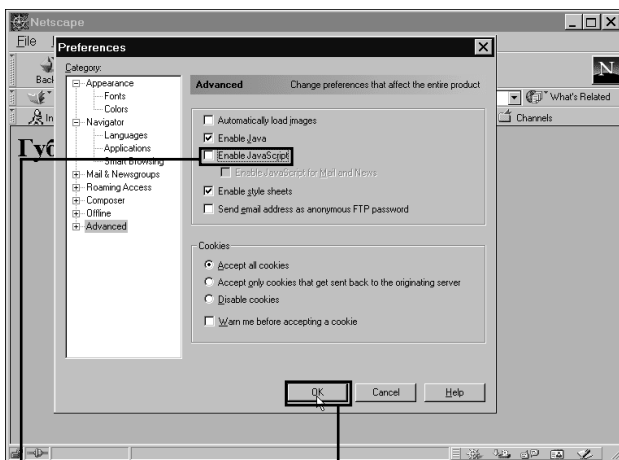
- 3 Из размещенного слева списка выберите категорию Advanced.

Что еще?

Поскольку конфигурацией браузера пользователя управлять нельзя, не существует способа ее изменения таким образом, чтобы включить поддержку JavaScript. Самое большее, что можно сделать, — сообщить пользователю, как включить эту поддержку. Можно, однако, определить, включена ли поддержка языка программирования Java.

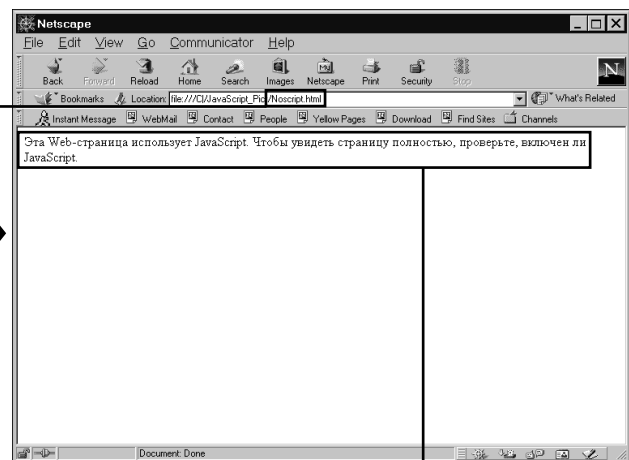
Многие современные возможности Web-страниц могут быть

включены или отключены с помощью параметров браузера. С помощью диалогового окна Preferences браузера Netscape Navigator можно включить поддержку языка Java, JavaScript, страниц стилей и cookies. С помощью диалогового окна Свойства обозревателя (Options) программы Internet Explorer можно отключить поддержку Java, элементов ActiveX, сценариев и файлов cookies.



4 Отключите опцию Enable JavaScript (☒ изменится на ☐).

5 Щелкните на кнопке OK.



ПРОСМОТР WEB-СТРАНИЦЫ С ПОМОЩЬЮ БРОУЗЕРА С ОТКЛЮЧЕННЫМ JAVASCRIPT

1 Откройте файл noscript.html в Netscape Navigator.

В окне браузера отображается текст, заключенный внутри дескрипторов <noscript>.

ДОБАВЛЕНИЕ КОММЕНТАРИЕВ В JAVASCRIPT

Синтаксис JavaScript может быть сложным для новичков. Один из способов документирования содержания каждого из разделов исходного текста заключается во включении в него блоков комментариев. Эти комментарии игнорируются браузером и могут быть просмотрены только в исходном файле.

Однострочные комментарии можно обозначить, предварив их текст двумя косыми чертами (//). Многострочные комментарии начинаются с косой черты (слэша) и звездочки (/*), а завершаются обратной комбинацией этих символов (*/). Наборы комментариев не могут быть вложены друг в друга. Все операторы внутри символов начала и конца комментариев игнорируются.

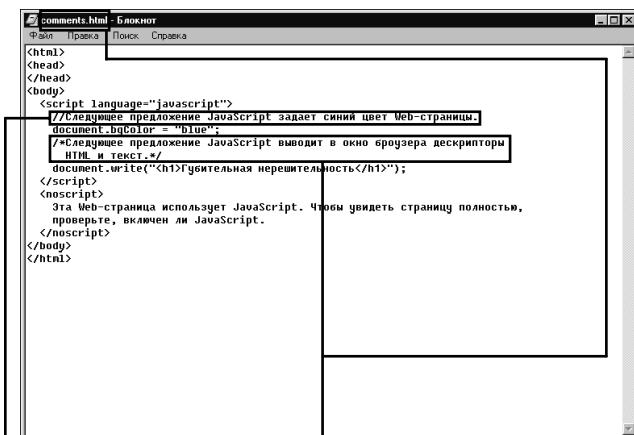
Комментарии могут встретиться в любом месте набора дескрипторов `<script>`.

Их можно использовать не только для документирования исходной программы, но и для отладки JavaScript.

Многострочные комментарии можно применять для быстрого отключения разделов текста программы без их удаления. Эта возможность особенно полезна, когда необходимо выполнить отладку разделов, которые работают неверно.

В приведенном ниже примере используется файл `noscript.html`, созданный в разделе “Представление содержимого сценариев для браузеров, не поддерживающих JavaScript”, однако соответствующие шаги можно применить к любому файлу.

ДОБАВЛЕНИЕ КОММЕНТАРИЕВ В JAVASCRIPT

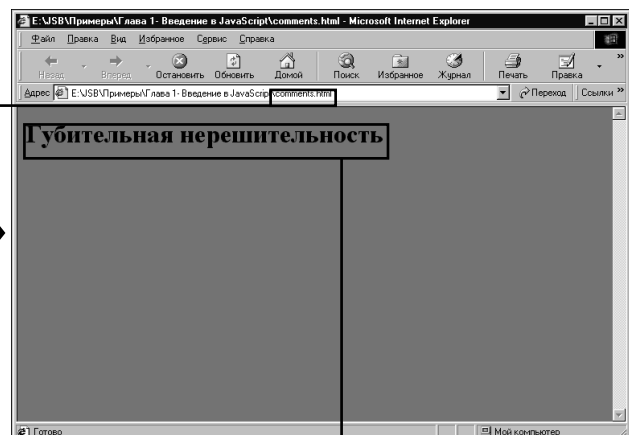


1 С помощью программы Блокнот откройте файл `noscript.html`.

2 Перед первым оператором JavaScript добавьте однострочный комментарий.

3 Перед вторым оператором JavaScript добавьте многострочный комментарий.

4 Сохраните файл под именем `comments.html`.



5 Откройте файл `comments.html` в браузере.

Обратите внимание, что комментарии, добавленные в файл, не отображаются браузером.

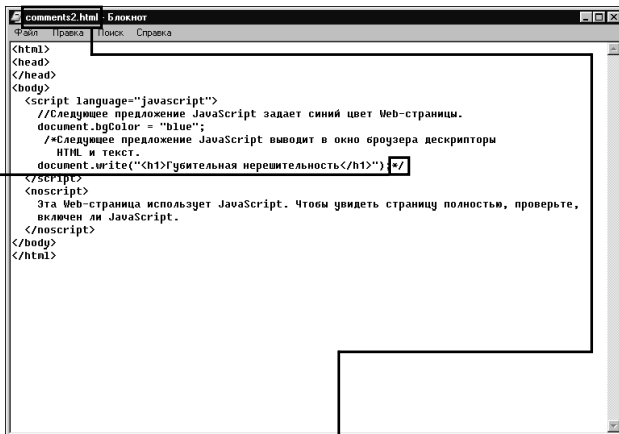
Попробуй!

Многие сценарии станут более понятными, если в их заголовок поместить комментарий. Этот заголовок-пояснение может содержать информацию о сценарии, включая сведения об авторе, дате создания и описание основных функций. Подобный заголовок может быть помечен как блок комментария следующим образом.

```
/*Сценарий персонификации

разработан КЛМ 16 мая 2000 года

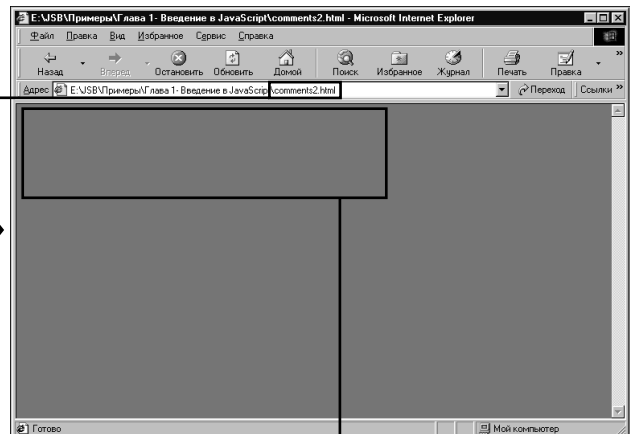
Описание. Этот сценарий предназначен
для ввода имени пользователя с помощью
диалогового окна приглашения,
сохранения имени в переменной
и его отображения в окне браузера
как части текста приветствия. */
```



6 Откройте с помощью программы Блокнот файл comments.html.

8 Сохраните файл под именем comments2.html.

7 Поместите закрывающий блок комментария после второго оператора JavaScript.



9 Откройте файл comments2.html в браузере.

Обратите внимание на то, что первый оператор JavaScript выполнялся, а второй — нет, поскольку последний теперь играет роль комментария.