

УРОК 6



Обращение к предыстории браузера

В HTML5 предоставляется возможность манипулировать предысторией браузера, контролируя его предыдущее и текущее состояние. В частности, объект `history` дает возможность перемещаться вперед и назад по веб-страницам, просматривавшимся в браузере. Это означает, что с помощью данного объекта предыстории можно, например, вернуться на три страницы назад.

Кроме того, в объекте состояния, помещаемом в предыстории браузера, могут быть сохранены данные. Таким образом, данные можно сначала ввести в объект состояния, а затем поместить объект состояния в предысторию браузера вместе с текущей веб-страницей. А для восстановления сохраненных данных достаточно извлечь объект состояния из предыстории браузера. Это дает возможность передавать данные из одной страницы на другую.

В этой главе будет показано, каким образом данный механизм работает на практике. А теперь перейдем непосредственно к рассмотрению того, что собой представляет предыстория браузера.

Общее представление о предыстории браузера

В этой главе будет рассмотрен пример HTML-документа `porphistory.html`, приведенного на рис. 6.1.

В данном примере демонстрируются различные свойства объекта `history`, доступного в HTML5. Так, если пользователь щелкнет на кнопке Назад, произойдет переход на одну просматривавшуюся страницу назад, что аналогично выбору кнопки Назад в самом браузере. А если пользователь щелкнет на кнопке Вперед,

то произойдет переход на одну просматривавшуюся страницу вперед. Если же следующая страница отсутствует, то ничего существенного не произойдет.

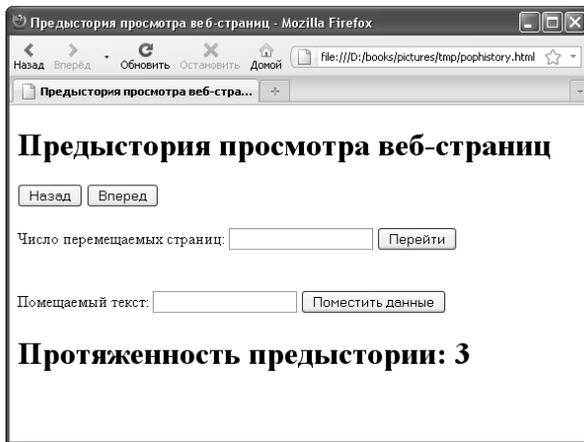


Рис. 6.1. Манипулирование предысторией браузера в примере HTML-документа pophistory.html

Кроме того, в текстовом поле можно ввести целое значение, например 5, а затем щелкнуть на кнопке Перейти, чтобы перейти на пять просматривавшихся страниц вперед. А если ввести отрицательное целое значение, то произойдет переход на соответствующее число просматривавшихся в браузере страниц назад.

Помимо этого, в рассматриваемом здесь примере отображается протяженность предыстории браузера, как показано на рис. 6.1. Доступность протяженности предыстории браузера является частью спецификации HTML5.

И наконец, в рассматриваемом примере пользователю предоставляется возможность ввести текст и поместить его в предысторию как данные, сохраненные в объекте текущего состояния, а затем перейти к другой странице и вернуться назад. В этом случае объект состояния будет извлечен из предыстории, и введенный пользователем текст появится в окне браузера. Подобным образом можно передавать данные с одной страницы на другую.

Механизм размещения и извлечения объекта состояния вместе с сохраненными данными должен поддерживаться браузером как часть его функций обращения к предыстории. К сожалению,

на момент написания этой книги отсутствовали какие-либо официальные сведения о подобной поддержке в браузерах, поэтому данный механизм будет рассмотрен только в коде. Но как только его поддержка появится в браузерах, вы уже будете готовы реализовать его на практике.

Ниже приведен краткий обзор программных средств, положенных в основу примера, рассматриваемого в данной главе.

Представление о прикладном программном интерфейсе предыстории браузера

Объект `history` является составной частью объекта `window` (т.е. его потомком) в браузере, и поэтому к нему можно обращаться по ссылке `window.history`. Ниже приведены атрибуты и функции, применяемые для реализации объекта `window.history` в коде HTML5.

- `window.history.length`
- `window.history.back()`
- `window.history.forward()`
- `window.history.go()`
- `window.history.pushState()`
- `window.history.replaceState()`
- `window.onpopstate`

Рассмотрим их по порядку и более подробно.

Атрибут `window.history.length`

Атрибут `window.history.length` предоставляет целый ряд элементов предыстории из сеанса просмотра веб-страниц в браузере. Более подробно этот атрибут будет рассмотрен далее по ходу создания примера HTML-документа `pophistory.html`.

Функция `window.history.back()`

Эта функция осуществляет продвижение на один шаг назад в предыстории браузера. Таким образом, она возвращает браузер

на предыдущую веб-страницу. Если же предыдущая страница отсутствует, эта функция не производит никаких действий.

Функция `window.history.forward()`

Данная функция осуществляет продвижение на один шаг вперед в предыстории браузера. Так, если пользователь, просматривающий страницу А, перейдет на страницу Б, а затем вернется к странице А, то для последующего перехода к странице Б, т.е. на одну просматривавшуюся страницу вперед, следует выполнить данную функцию. Если же следующая страница отсутствует, эта функция не производит никаких действий.

Функция `window.history.go([delta])`

Эта универсальная функция позволяет перемещаться вперед и назад на число страниц, указанное в ее аргументе `delta`. Так, если передать этой функции числовое значение `-2` в качестве ее аргумента `delta`, браузер осуществит возврат на две просматривавшиеся страницы назад. А если передать данной функции числовое значение `5`, то браузер осуществит переход на пять просматривавшихся страниц вперед.

В определении данной функции аргумент `delta` указывается в квадратных скобках потому, что он является необязательным. Поэтому если не предоставить ей аргумент `delta`, браузер просто перезагрузит текущую страницу. Как и следовало ожидать, данная функция не производит никаких действий, если аргумент `delta` оказывается за пределами предыстории браузера.

Функция `window.history.pushState(data, title [, url])`

Данная функция позволяет поместить данные в предысторию. Аргумент `data` обозначает объект, содержащий введенные данные, аргумент `title` — имя, по которому следует обращаться к данным, тогда как аргумент `url` позволяет связать данные с конкретной страницей, чтобы извлечь их из предыстории при перезагрузке этой страницы.

Функция `window.history.replaceState(data, title [, url])`

Эта функция осуществляет замену данных, имени и URL (т.е. веб-адреса) в текущем элементе предыстории браузера значениями, задаваемыми в аргументах `data`, `title` и `url`.

Атрибут `window.onpopstate`

Данный атрибут обозначает событие, происходящее при извлечении объекта состояния из предыстории. В итоге этот объект становится доступным вместе с хранящимися в нем данными. Обработка данного события будет продемонстрирована в коде по ходу создания примера HTML-документа `rophistory.html`.

Пример обращения к предыстории браузера

Для того чтобы приступить к созданию примера HTML-документа `rophistory.html`, демонстрирующего особенности обращения к предыстории браузера на практике, выполните следующие действия.

1. Создайте HTML-документ `rophistory.html` в любом редакторе текста, например в Windows WordPad.
2. Введите приведенный ниже код основного тела HTML-документа.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

  </head>
  <body>
    <h1>Предыстория просмотра веб-страниц</h1>
    .
    .
    .
  </body>
</html>
```

3. Введите выделенный ниже полужирным код для создания элементов `<div>`, используемых в данном примере для отображения результатов.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <br>
    <div id="length"></div>
    <br>
    <div id="state"></div>
    <br>
  </body>
</html>
```

4. Сохраните внесенные изменения в файле `rophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Итак, начало созданию примера HTML-документ `rophistory.html` положено. А теперь необходимо снабдить его функциональными возможностями для манипулирования предысторией браузера.

Ввод кнопки Назад

С помощью функции `window.history.back()` и кнопки Назад можно разрешить пользователю перемещаться назад по предыстории браузера. Для того чтобы ввести эту кнопку в рассматриваемый здесь пример HTML-документа `rophistory.html`, выполните следующие действия.

1. Откройте HTML-документ из файла `rophistory.html` в избранном вами редакторе текста, например в Windows WordPad.
2. Введите выделенный ниже полужирным код для создания новой кнопки Назад.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>
  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <input type="button" value="Назад"
      onclick="back();">
      .
      .
      .
    <br>
    <div id="length"></div>
    <br>
    <div id="state"></div>
    <br>
  </body>
</html>
```

3. Введите выделенный ниже полужирным код для создания функции `back`, вызывающей функцию `window.history.back()` и активизирующей связанную с ней кнопку Назад.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    <script type="text/javascript">
      function back()
      {
        window.history.back();
      }
    </script>
  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <input type="button" value="Назад"
      onclick="back();">
```

```
.  
. .  
<br>  
<div id="length"></div>  
<br>  
<div id="state"></div>  
<br>  
</body>  
</html>
```

4. Сохраните внесенные изменения в файле `rophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Теперь пользователь может щелкнуть на кнопке Назад, как показано на рис. 6.1, чтобы перейти на одну просматривавшуюся в браузере страницу назад. В частности, пользователь может перейти сначала на веб-сайт по адресу `www.usatoday.com`, как показано на рис. 6.2, а затем на страницу с примером HTML-документа `rophistory.html` (рис. 6.3). Если же он щелкнет на кнопке Назад, то произойдет возврат на начальную страницу веб-сайта по адресу `www.usatoday.com` (рис. 6.4).

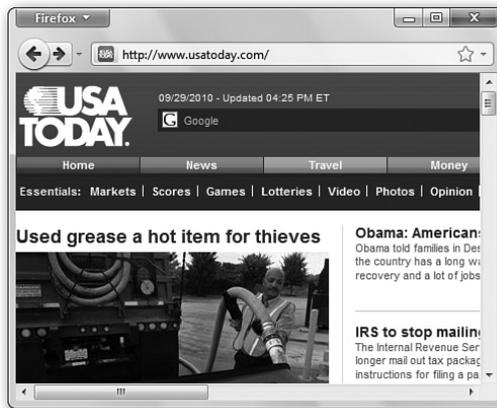


Рис. 6.2. Начальная страница веб-сайта периодического издания *USA Today*

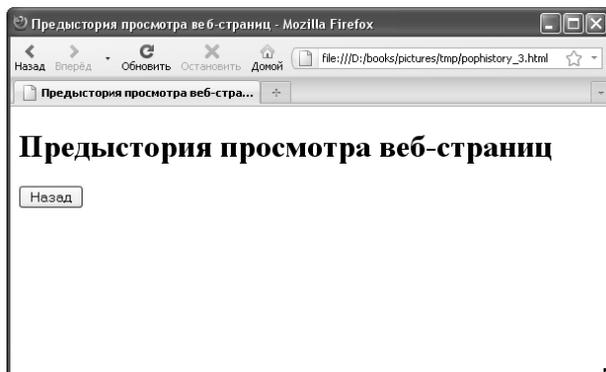


Рис. 6.3. Переход на страницу с примером HTML-документа pophistory.html

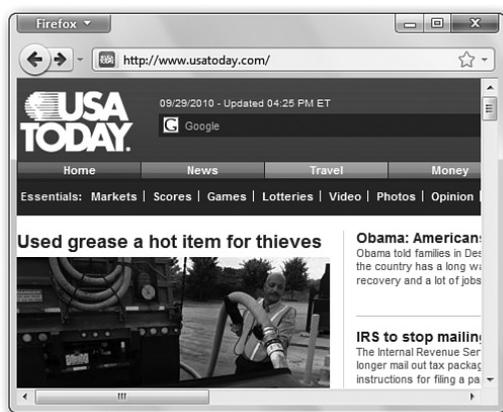


Рис. 6.4. Возврат на начальную страницу веб-сайта периодического издания USA Today

Ввод кнопки Вперед

С помощью функции `window.history.forward()` и кнопки Вперед можно также разрешить пользователю перемещаться вперед по предыстории браузера. Для того чтобы ввести эту кнопку в рассматриваемый здесь пример HTML-документа `pophistory.html`, выполните следующие действия.

1. Откройте HTML-документ из файла `rophistory.html` в избранном вами редакторе текста, например в Windows WordPad.
2. Введите выделенный ниже полужирным код для создания новой кнопки Вперед.

```

<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    .
    .
    .
  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <input type="button" value="Назад"
      onclick="back();">
    <input type="button" value="Вперед"
      onclick="forward();">
    .
    .
    .
    <br>
    <div id="length"></div>
    <br>
    <div id="state"></div>
    <br>
  </body>
</html>

```

3. Введите выделенный ниже полужирным код для создания функции `forward`, вызывающей функцию `window.history.forward()` и активизирующей связанную с ней кнопку Вперед.

```

<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    <script type="text/javascript">
      function back()

```

```
{
    window.history.back();
}

function forward()
{
    window.history.forward();
}

</script>
</head>

<body onload="loader()">
<h1>Предыстория просмотра веб-страниц</h1>
<input type="button" value="Назад"
    onclick="back();">
<input type="button" value="Вперед"
    onclick="forward();">
<br>
<div id="length"></div>
<br>
<div id="state"></div>
<br>
</body>
</html>
```

4. Сохраните внесенные изменения в файле `rophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Если теперь пользователь щелкнет на кнопке Вперед, в предыстории браузера произойдет переход на одну просматривавшуюся страницу вперед.

Ввод кнопки Перейти

С помощью текстового поля, кнопки Перейти и функции `window.history.go()` можно разрешить пользователю ввести конкретное число страниц для продвижения по предыстории браузера вперед (положительное целое значение) или назад (отрицательное целое значение).

Для того чтобы ввести кнопку Перейти и текстовое поле в рассматриваемый здесь пример HTML-документа `rophistory.html`, выполните следующие действия.

1. Откройте HTML-документ из файла `rophistory.html` в избранном вами редакторе текста, например в Windows WordPad.
2. Введите выделенный ниже полужирным код для создания нового текстового поля и кнопки Перейти.

```

<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    .
    .
    .
  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <input type="button" value="Назад"
      onclick="back();">
    <input type="button" value="Вперед"
      onclick="forward();">
    <br>
    Число перемещаемых страниц: <input id="amount"
      type="text">
    <input type="button" value="Перейти" onclick="go();">
    <br>
    .
    .
    .
    <br>
    <div id="length"></div>
    <br>
    <div id="state"></div>
    <br>
  </body>
</html>

```

3. Введите выделенный ниже полужирным код для создания функции `go`, вызывающей функцию `window.history.go()` и активизирующей связанную с ней кнопку Перейти.

```

<!DOCTYPE html>
<html>
  <head>
    <title>

```

```
    Предыстория просмотра веб-страниц
  </title>

  <script type="text/javascript">
    function back()
    {
      window.history.back();
    }

    function forward()
    {
      window.history.forward();
    }

    function go()
    {
      var amount = document.getElementById("amount").value;
      window.history.go(amount);
    }
  </script>
</head>

<body onload="loader()">
<h1>Предыстория просмотра веб-страниц</h1>
<input type="button" value="Назад"
  onclick="back();">
<input type="button" value="Вперед"
  onclick="forward();">
<br>
<div id="length"></div>
<br>
<div id="state"></div>
<br>
</body>
</html>
```

4. Сохраните внесенные изменения в файле `rophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Если теперь пользователь введет в текстовом поле число перемещаемых страниц и щелкнет на кнопке **Перейти**, произойдет переход вперед или назад по предыстории браузера на заданное число страниц.

Получение протяженности предыстории браузера

С помощью атрибута `window.history.length` можно определить общее число веб-страниц, хранящихся в предыстории браузера, т.е. количество элементов для продвижения по предыстории вперед или назад. Для того чтобы отобразить в рассматриваемом здесь примере HTML-документа `rophistory.html` протяженность предыстории браузера, выполните следующие действия.

1. Откройте HTML-документ из файла `rophistory.html` в избранном вами редакторе текста, например в Windows WordPad.
2. Введите выделенный ниже полужирным код для отображения количества элементов в предыстории браузера в элементе `<div>`.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>
    <script type="text/javascript">
      function back()
      {
        window.history.back();
      }

      function forward()
      {
        window.history.forward();
      }

      function go()
      {
        var amount = document.getElementById("amount").value;
        window.history.go(amount);
      }

      function loader()
      {
        var length = window.history.length;
        document.getElementById("length").innerHTML = "<h1> " +
          "Протяженность предыстории: " + length + "</h1>";
      }
    </script>
  </head>
  <div>
  </div>
</html>
```

```
</script>
</head>

<body onload="loader()">
<h1>Предыстория просмотра веб-страниц</h1>
<input type="button" value="Назад"
  onclick="back();">
<input type="button" value="Вперед"
  onclick="forward();">
<br>
Число перемещаемых страниц: <input id="amount"
  type="text">
<input type="button" value="Перейти" onclick="go();">
<br>
.
.
.
<br>
<div id="length"></div>
<br>
<div id="state"></div>
<br>
</body>
</html>
```

3. Сохраните внесенные изменения в файле `rophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Если теперь загрузить в браузер страницу с примером HTML-документа `rophistory.html`, на ней будет отображено количество элементов предыстории браузера, как показано на рис. 6.5.

Размещение данных в предыстории

Пользователю можно также разрешить помещать текстовые данные в предыстории текущей страницы для последующего их восстановления. Для этой цели выполните следующие действия.

1. Откройте HTML-документ из файла `rophistory.html` в избранном вами редакторе текста, например в Windows WordPad.
2. Введите выделенный ниже полужирным код для создания текстового поля и кнопки с меткой Поместить данные.

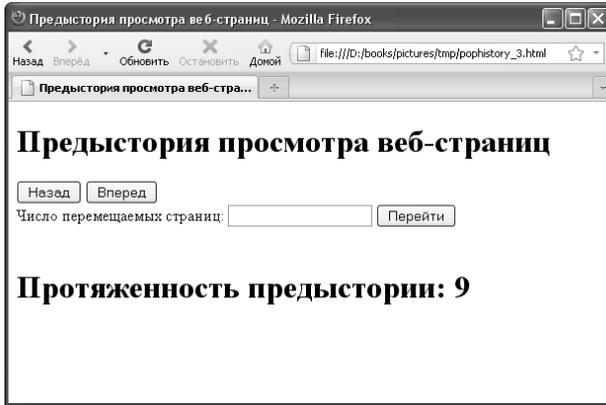


Рис. 6.5. Окно браузера с количеством элементов его предыстории

```

<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    <script type="text/javascript">
      function back()
      {
        window.history.back();
      }

      .
      .
      .
    </script>
  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <input type="button" value="Назад"
      onclick="back();">
    <input type="button" value="Вперед"
      onclick="forward();">
    <br>
    Число перемещаемых страниц: <input id="amount"
      type="text">
    <input type="button" value="Перейти" onclick="go();">
  </body>

```

```
<br>
<br>
<br>
Помещаемый текст: <input id="statedata"
  type="text">
<input type="button" value="Поместить данные"
  onclick="pushdata();">
<br>
<div id="length"></div>
<br>
<div id="state"></div>
<br>
</body>
</html>
```

3. Введите выделенный ниже полужирным код для сохранения введенного пользователем текста в объект и размещения этого объекта в предыстории текущей страницы. При возврате к этой странице сохраненный текст будет восстановлен.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    <script type="text/javascript">
      function back()
      {
        window.history.back();
      }
      .
      .
      function pushData()
      {
        var statedata =
          document.getElementById("statedata").value;
        var containerObject = {container: statedata};
        history.pushState(containerObject, "item",
          "pophistory.html");
      }
    </script>
  </head>

  <body onload="loader()">
```

```
<h1>Предыстория просмотра веб-страниц</h1>
<input type="button" value="Назад"
  onclick="back();">
<input type="button" value="Вперед"
  onclick="forward();">
<br>
Число перемещаемых страниц: <input id="amount"
  type="text">
<input type="button" value="Перейти" onclick="go();">
<br>
<br>
<br>
Помещаемый текст: <input id="statedata"
  type="text">
<input type="button" value="Поместить данные"
  onclick="pushdata();">
<br>
<div id="length"></div>
<br>
<div id="state"></div>
<br>
</body>
</html>
```

4. Сохраните внесенные изменения в файле `pophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Если теперь пользователь введет данные в текстовом поле и щелкнет на кнопке Поместить данные, введенный им текст будет помещен в предысторию текущей страницы. В следующем задании вам предстоит извлечь текст, помещенный в предысторию браузера.

Извлечение данных из предыстории

Итак, введенные данные сохранены в объекте состояния и помещены в предысторию текущей страницы, откуда они будут извлечены автоматически при перезагрузке текущей страницы. Таким образом, введенные однажды данные могут быть восстановлены из предыстории. Для того чтобы извлечь эти данные из предыстории и отобразить их в окне браузера, необходимо связать событие `onpopstate`, порождаемое для текущей страницы, с функцией его обработки.

1. Откройте HTML-документ из файла pophistory.html в избранном вами редакторе текста, например в Windows WordPad.
2. Введите выделенный ниже полужирным код для обработки события onpopstate, в результате чего текст, введенный пользователем, будет восстановлен и затем отображен в элементе <div>.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    <script type="text/javascript">
      function back()
      {
        window.history.back();
      }
      .
      .
      .
      function pushData()
      {
        var statedata = document.getElementById("statedata").value;
        var containerObject = {container: statedata};
        history.pushState(containerObject, "item",
          "pophistory.html");
      }

      function popData(event)
      {
        var state = "Страница: " + document.location + "
          Данные: " + event.state.container;
        document.getElementById("state").innerHTML = "<h1>" +
          state + "</h1>";
      };

      window.addEventListener("popstate", popData, false);
    </script>
  </head>

  <body onload="loader()">
    <h1>Предыстория просмотра веб-страниц</h1>
    <input type="button" value="Назад"
      onclick="back();">
    <input type="button" value="Вперед"
```

```

        onclick="forward();">
    <br>
    Число перемещаемых страниц: <input id="amount"
        type="text">
    <input type="button" value="Перейти" onclick="go();">
    <br>
    <br>
    <br>
    Помещаемый текст: <input id="statedata"
        type="text">
    <input type="button" value="Поместить данные"
        onclick="pushdata();">
    <br>
    <div id="length"></div>
    <br>
    <div id="state"></div>
    <br>
    </body>
</html>

```

3. Сохраните внесенные изменения в файле `rophistory.html`. Непременно сделайте это в текстовом формате. По умолчанию в текстовом редакторе WordPad выбирается расширенный текстовый формат RTF, непригодный для браузеров.

Если теперь пользователь введет текст на странице HTML-документа `rophistory.html`, щелкнет на кнопке Поместить данные, перейдет к следующей странице, а затем вернется к странице HTML-документа `rophistory.html`, то введенный им текст будет автоматически извлечен из предыстории и отображен на данной странице. (На момент написания этой книги отсутствовали какие-либо официальные сведения о поддержке механизма размещения и извлечения данных из предыстории в браузерах. Но как только его поддержка появится в браузерах, вы будете готовы реализовать его на практике.)

Полный исходный код примера HTML-документа `rophistory.html`

Ниже для справки приводится полностью исходный код рассмотренного в этой главе примера HTML-документа `rophistory.html`.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Предыстория просмотра веб-страниц
    </title>

    <script type="text/javascript">
      function back()
      {
        window.history.back();
      }
      function forward()
      {
        window.history.forward();
      }

      function go()
      {
        var amount = document.getElementById("amount").value;
        window.history.go(amount);
      }

      function loader()
      {
        var length = window.history.length;
        document.getElementById("length").innerHTML = "<h1>" +
          "Протяженность предыстории: " + length + "</h1>";
      }

      function pushData()
      {
        var statedata = document.getElementById("statedata").value;
        var containerObject = {container: statedata};
        history.pushState(containerObject, "item", "pophistory.html");
      }

      function popData(event)
      {
        var state = "Страница: " + document.location + " Данные: " +
          event.state.container;
        document.getElementById("state").innerHTML = "<h1>" + state +
          "</h1>";
      };

      window.addEventListener("popstate", popData, false);
    </script>
  </head>
  <body>
    <div id="amount">
      <input type="text" value="1" />
    </div>
    <div id="length">
      <input type="text" value="0" />
    </div>
    <div id="statedata">
      <input type="text" value="" />
    </div>
    <div id="state">
      <input type="text" value="" />
    </div>
  </body>
</html>
```

```
</head>
<body onload="loader()">
  <h1>Предыстория просмотра веб-страниц</h1>
  <input type="button" value="Назад"
    onclick="back();">
  <input type="button" value="Вперед"
    onclick="forward();">
  <br>
  <br>
  Число перемещаемых страниц: <input id="amount"
    type="text">
  <input type="button" value="Перейти" onclick="go();">
  <br>
  <br>
  <br>
  Помещаемый текст: <input id="statedata"
    type="text">
  <input type="button" value="Поместить данные"
    onclick="pushdata();">
  <br>
  <div id="length"></div>
  <br>
  <div id="state"></div>
  <br>
</body>
</html>
```