

Лабораторная работа №9 Работа со строками

Строковые литералы или строковые переменные являются в языке JavaScript объектом типа `string`, к которому могут быть применены методы, определенные в языке. Создание нового объекта требует вызова функции-конструктора объекта. Для того чтобы создать строковый объект, надо применить конструктор `newString`, например:

```
s=newString("результат=")
```

Объект `string` имеет единственное свойство `length` (длина_строки). Выражение `s.length` выдает значение 10, равное длине строки, содержащейся в строковом объекте `s`. Объект `string` имеет два типа методов. С методами, непосредственно влияющими на саму строку, мы сейчас и познакомимся, рассматривая примеры обработки текстовой информации.

Одним из часто используемых методов является метод выделения из строки отдельного символа. Метод `charAt(ni)` возвращает символ, позицию которого определяет параметр `ni`. Символы в строке перенумерованы, начиная с 0.

Пример 1. Вывод символов строки в "столбик"

Напишем сценарий, при выполнении которого заданный текст выводится в "столбик", т.е. на каждой строке размещается по одному символу.

При решении задачи из заданной строки последовательно выбираются символы. Формируется новая строка, в которой за каждым символом ставится последовательность символов, обеспечивающая переход на новую строку. Когда строка результата сформирована, то она размещается в текстовом поле формы, тем самым для исходной строки осуществляется вывод в "столбик". Сценарий, осуществляющий обработку строки, приведен в листинге 1.

Листинг 1. Вывод символов строки в "столбик"

```
<HTML>
<HEAD>
<TITLE>Вывод символов строки в "столбик"</TITLE>
<script language="JavaScript">
<!-- //
function ttest(s)
{ var sres="Прочитанный текст:"+" \r\n"+s+"\r\n"+
"Текст в "столбик":'+"\r\n"
var cur=""
for ( var i=0; i <= s.length-1; i += 1)
```

```

    {c=s.charAt(i); cur +=c+"\r\n" }
    sres+=cur
    return sres
  }
  //-->
</script>
</HEAD>
<BODY bgcolor="#FFFFCC">
<H4>Символы текущей строки в столбик</H4>
<FORM name="form1">
Введите строку: <input type="text" size=20 name="st1"><hr>
<input
                    type="button"
                    value=Выполнить
onClick="form1.res.value=ttest(form1.st1.value)">
<input type="reset" value=Очистить><hr>
<textarea cols=20 rows=7 name= res></textarea>
</FORM>
</BODY>
</HTML>

```

Метод `substr (n1,n2)` позволяет выделять из строки подстроку. Параметр `n1` задает позицию первого символа подстроки; параметр `n2` определяет количество символов в подстроке. Например, если `строка="сборник"`, то в результате выполнения `substr (0,4)` будет выделена подстрока "сбор".

Пример 2. Вычисление количества повторений строки в тексте

Напишем программу, которая определяет, сколько раз заданное слово встречается в определенном тексте.

В тексте слова разделяются пробелами. После того как очередное слово найдено, просмотр продолжается с символа, следующего за найденным словом.

HTML-код документа представлен в листинге 2.

Листинг 2. Количество заданных слов в тексте

```

<HTML>
<HEAD>
<TITLE>Количество заданных слов в тексте</TITLE>
<script language="JavaScript">
<!-- //
function numword(obj)
{ var h=obj.data.value
var s=obj.textin.value
s=' '+s+' '

```

```

h=' '+h+' '
var
m=h.length
var res=0
var i=0
while (i <= s.length-1)
{ ch=s.substr(i,m)
if (ch==h) {res+=1; i = i+m-
1}else
i++
}
obj.result.value=res
}
//-->
</script>
</HEAD>
<BODY bgcolor="#FFFFCC">
<H4>Количество заданных слов в тексте</H4>
<FORM
name="form1">
Введите текст:<br>
<textarea name="textin" rows=4 cols=20></textarea><hr>
Введите слово: <input type="text" name="data"
size="8"><hr>
<input
type="button"
value="Определить"
onClick="numword(form1)"><hr>
Количество слов в тексте: <input type="text" name="result"
size=8><hr>
<input type="reset" value="Отменить">
</FORM>
</BODY>
</HTML>

```

Упражнения

1. Проверить примеры из лабораторной работы.
2. Слова в заданном тексте разделяются пробелами.
Напишите программу, которая определяет количество слов в тексте.
3. Напишите программу, в которой все слова А заменены словом В, где А и В - заданные слова, возможно, различной длины.
4. Напишите программу, которая "сжимает" заданный текст, т. е. заменяет все подряд идущие пробелы на один.