# Лабораторная работа № 5 Работа со строками в JavaScript

**Цель работы:** получить навыки обработки символьной информации в JavaScript.

**Краткие теоретические сведения:**

В JavaScript любые текстовые данные являются строками. Не существует отдельного типа «символ», который есть в ряде других языков.

Внутренний формат для строк — всегда UTF-16, вне зависимости от кодировки страницы.

Для задания строк в JavaScript есть разные типы кавычек. Строку можно создать с помощью одинарных, двойных либо обратных кавычек:

let single = 'single-quoted';

let double = "double-quoted";

let backticks = `backticks`;

Одинарные и двойные кавычки работают, по сути, одинаково, а если использовать обратные кавычки, то в такую строку мы сможем вставлять произвольные выражения, обернув их в ${…}:

function sum(a, b) {

return a + b;

}

console.log(`1 + 2 = ${sum(1, 2)}.`); // 1 + 2 = 3.

Для определения длины строки используется свойство length.

Чтобы получить символ, который занимает позицию pos, можно использовать квадратные скобки: [pos], а также можно использовать метод **charAt(pos)**. Первый символ занимает нулевую позицию:

let str = `Hello`;

console.log(str.length); //5

// получаем первый символ

console.log(str[0]); // H

console.log(str.charAt(0)); // H

Следует обратить внимание на то, что содержимое строки в JavaScript нельзя изменить.

let str = `Hello`;

console.log(str); // Hello

str[0] = "J";

console.log(str); // Hello

Методы **toLowerCase()** и **toUpperCase()** меняют регистр символов:

alert( 'Interface'.toUpperCase() ); // INTERFACE

alert( 'Interface'.toLowerCase() ); // interface

Для поиска подстроки используются следующие методы:

* **indexOf(substr, pos)** ищет подстроку substr в строке str, начиная с позиции pos, и возвращает позицию, на которой располагается совпадение, либо -1 при отсутствии совпадений;
* **lastIndexOf(substr, position)** ищет подстроку с конца строки к её началу;
* **includes(substr, pos)** возвращает true, если в строке str есть подстрока substr, либо false, если нет;
* **startsWith(substr)**, **endsWith(substr)** проверяют, соответственно, начинается ли и заканчивается ли строка определённой строкой.

Для получения подстроки в JavaScript есть 3 метода:

* **slice(start [, end])** возвращает часть строки от start до (не включая) end, если end отсутствует, slice возвращает символы до конца строки;
* **substring(start [, end])** возвращает часть строки между start и end;
* **substr(start [, length])** возвращает часть строки от start длины length.

Для сравнение строк используются стандартные операторы сравнения, строки сравниваются посимвольно в алфавитном порядке.

**Задание к лабораторной работе:**

**Задание 1.** В соответствии со своим вариантом написать Java-скрипт в файле **lab5-1.html**, выполняющий следующие действия

| Вариант | Задание |
| --- | --- |
| 01 | Подсчитать общее количество символов '+ ' и '-' и заменить каждый символ ';' на ',' и '.' |
| 02 | После каждого символа ',' вставить пробел и подсчитать количество букв 'A' и 'B' отдельно |
| 03 | Заменить символ '\*' на '++' и подсчитать общее количество букв 'F' и 'D' |
| 04 | Подсчитать количество букв 'C' и 'D' отдельно и заменить каждую пару символов '\*\*' на символ ' ' |
| 05 | После каждого символa '!' вставить символ 'I' и подсчитать общее количество цифр в строке |
| 06 | Удалить каждую пару символов 'PQ' и подсчитать общее количество символов '.' и ',' в строке |
| 07 | Подсчитать количество пар символов '+ -' и заменить каждый символ '\*' на '/-' |
| 08 | После каждой цифры вставить такую же цифру и подсчитать количество пар 'AC' в строке |
| 09 | Удалить каждый символ 'A', стоящий после ',' и подсчитать количество пар 'BC' |
| 10 | Подсчитать количество символов '.' , стоящих перед пробелом, и заменить каждую пару символов 'ST' на символ 'P' |
| 11 | После каждого символа 'A' вставить пробел и подсчитать количество символов 'B', стоящих между знаками '+' и '-' |
| 12 | Удалить каждый символ '?', стоящий после ';', и подсчитать общее количество символов 'о и 'O' |
| 13 | Подсчитать количество символов '+', стоящих между'A' и 'B', заменить каждый символ '0' на 'OO' |
| 14 | В каждую пару символов 'AB' вставить символ '\*', подсчитать, сколько раз в строке символ 'I' стоит перед '2' |
| 15 | Вставить символ ';' после каждого символа 'A' и после каждого 'B', подсчитать, сколько раз символ'C' встречается между символами '\*' и '/' |

**Задание 2.** В соответствии со своим вариантом написать Java-скрипт в файле **lab5-2.html**, выполняющий следующие действия

| Вариант | Задание |
| --- | --- |
| 01 | После каждого слова поставить запятую. Подсчитать количество слов, в которых есть буква 'п' |
| 02 | Подсчитать количество букв в третьем слове. Поменять местами первое и последнее слова |
| 03 | Во втором слове после каждой буквы вставить пробел. Определить количество слов, которые заканчиваются на 'е' |
| 04 | Перед первой буквой каждого слова вставить символ '\*'. Определить количество слов, в которых нет ни одной буквы 'г' |
| 05 | Для первого слова указать, сколько букв 'и' в нем содержится. Переставить первое слово в конец строки |
| 06 | Определить количество слов, начинающихся с буквы 'A'. После каждой буквы предпоследнего слова вставить символ '\*' |
| 07 | Подсчитать количество букв во втором слове. Каждое слово заключить в кавычки. |
| 08 | Подсчитать количество слов, длина которых больше 5. Удалить пробелы, стоящие между первым и вторым словом. |
| 09 | Определить количество слов, в которых буква 'и' встречается хотя бы один раз. Поменять местами первое и второе слово |
| 10 | Третье слово строки поставить после первого. Определить количество слов, в которых первая и последняя буквы совпадают |
| 11 | Определить количество слов, вторая буква которых 'р'. Удалить последнюю букву в каждом слове |
| 12 | Подсчитать количество букв в предпоследнем слове. В каждом слове поменять местами первую и последнюю буквы |
| 13 | Перед каждой буквой третьего слова поставить '-'. Определить количество слов, после которых один пробел |
| 14 | После последней буквы каждого слова вставить точку. Для пятого слова указать, сколько букв 'И' в нем содержится. |
| 15 | Удалить все пробелы из строки, кроме тех, которые стоят между первым и вторым словом. Определить количество слов, которые по длине меньше 3 |

**Задание 3.** Написать простейшую программу шифрования. Программа каждую букву заменяет на следующую за ней в алфавите («я» переходит в «а»).

**Задание 4.** Написать программу, которая в каждом слове перемешивает буквы местами за исключением первой и последней буквы слова.

**Требования по содержанию отчета:**

Отчет должен включать следующие пункты:

1. Титульный лист.
2. Цель работы.
3. Задания на лабораторную работу.
4. Листинги программного кода.
5. Результаты выполнения скриптов.
6. Выводы.

**Контрольные вопросы для защиты:**

1. Как выводится текстовая информация с использованием JavaScript?
2. Как осуществитьт конкатенация строк?
3. Какая функция осуществляет поиск заданного элемента в строке?
4. Как определить длину строки?
5. Как из исходной строки выделить подстроку?
6. Как выполнить замену подстроки в исходной строке?
7. Для чего используются регулярные выражения?
8. Какие функции используются в JavaScript для работы с регулярными выражениями?
9. Как вывести подстроки в соответствии с шаблоном?