|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | | | |
| Институт информационных технологий (ИТ) | |
| Кафедра инструментального и прикладного программного обеспечения (ИиППО) | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ** | | | |
| **по дисциплине** | | | |
| **«Разработка клиентских частей интернет-ресурсов»** | | | |
| Выполнил студент группы ИКБО-01-19 | | Иноземцев Н.С. | |
|  | |  | |
| Принял  *Ассистент* | | Коваленко М.А. | |
| Практические работы выполнены | «\_\_\_»\_\_\_\_\_\_\_2020 г. | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись студента) | |
| «Зачтено» | «\_\_\_»\_\_\_\_\_\_\_2020 г. | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись руководителя) | |
|  |  | |  | |

Москва 2020

**Оглавление**

[Практическая работа №5: «JavaScript. Математические функции. Строковые функции. Функции для массивов» 3](#_Toc57573953)

[**Задача 1.** 3](#_Toc57573954)

[**Задача 2.** 4](#_Toc57573955)

[**Задача 3.** 5](#_Toc57573956)

[**Задача 4.** 7](#_Toc57573957)

[**Задача 5.** 8](#_Toc57573958)

[**Задача 6.** 9](#_Toc57573959)

[**Задача 7.** 10](#_Toc57573960)

[**Задача 8.** 12](#_Toc57573961)

[**Задача 9.** 13](#_Toc57573962)

[**Задача 10.** 15](#_Toc57573963)

[**Задача 11.** 16](#_Toc57573964)

[**Задача 12.** 17](#_Toc57573965)

[**Задача 13.** 18](#_Toc57573966)

[**Задача 14.** 18](#_Toc57573967)

[**Задача 15.** 19](#_Toc57573968)

[**Задача 16.** 20](#_Toc57573969)

[**Задача 17.** 21](#_Toc57573970)

[**Задача 18.** 22](#_Toc57573971)

[**Задача 19.** 24](#_Toc57573972)

[**Задача 20.** 25](#_Toc57573973)

# Практическая работа №5: «JavaScript. Математические функции. Строковые функции. Функции для массивов»

**Цель работы:** Изучить принцип работы с математическими функциями, строковыми функциями и функциями массивов.

**Задача 1.**

Работа с %:

* 1. Даны переменные a = 10 и b = 3. Найти остаток от деления a на b*;*
  2. Даны переменные a и b. Проверить, что a делится без остатка на b. Если это так - выведите 'Делится' и результат деления, иначе вывести 'Делится с остатком' и остаток от деления.

**Описание выполнения задания:**

Листинг 1.1 – создать переменные a и b присвоить им значения 10 и 3 соответственно и вывести остаток от деления а на b.

let a = 10;

let b = 3;

alert(a%b);

Результат выполнения кода на странице (см. Рисунок 1.1):

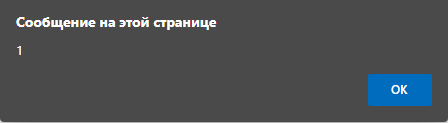


Рисунок 1.1

Листинг 1.2 - проверить делится ли число нацело, вывести в соответствие с этим текст и значение.

let a = 10;

let b = 3;

if(a%b == 0) {

alert("Делится " + a/b);

}

else {

alert('Делится с остатком ' + a%b);

}

Результат выполнения кода на странице (см. Рисунок 1.2):

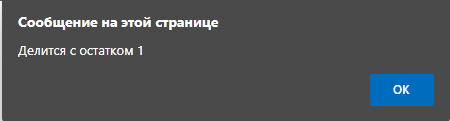


Рисунок 1.2

**Задача 2.**

Работа со степенью и корнем:

1. Возвести 2 в 10 степень. Результат записать в переменную st.
2. Найти квадратный корень из 245.
3. Дан массив с элементами 4, 2, 5, 19, 13, 0, 10. Найти квадратный корень из суммы кубов его элементов. Для решения воспользоваться циклом for.

**Описание выполнения задания:**

Листинг 2.1 – вывести 2 в 10 степени, записав в переменную в st.

let st = Math.pow(2, 10);

alert(st);

Результат выполнения кода на странице (Рисунок 2.1):

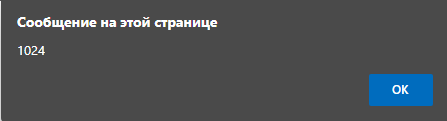


Рисунок 2.1

Листинг 2.2 – найти квадратный корень из 245.

alert(Math.sqrt(245));

Результат выполнения кода на странице (Рисунок 2.2):

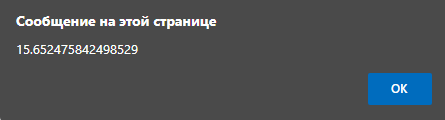


Рисунок 2.2

Листинг 2.3 –найти найти квадратный корень из суммы кубов элементов заданного массива.

let a = [4, 2, 5, 19, 13, 0, 10];

let sum = 0;

for (let i = 0; i < a.length; i++) {

sum += Math.pow(a[i], 3);

}

alert(Math.sqrt(sum));

Результат выполнения кода на странице (Рисунок 2.3):

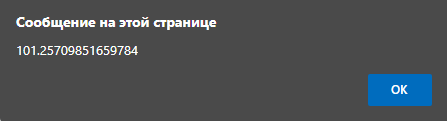


Рисунок 2.3

**Задача 3.**

Работа с функциями округления:

* 1. Найти квадратный корень из 379. Резуfl4льтат округлить до целых, до десятых, до сотых.
  2. Найти квадратный корень из 587. Округлить результат в большую и меньшую стороны, записать результаты округления в объект с ключами 'floor' и 'ceil'.

**Описание выполнения задания:**

Листинг 3.1 – найти квадратный корень заданного числа и округлить его до целых, десятых, сотых.

let res = Math.sqrt(379);

alert(Math.round(res) + ' ' + Math.round(res \* 10)/10 + ' ' + Math.round(res \* 100)/100);

Результат выполнения кода на странице (Рисунок 3.1):

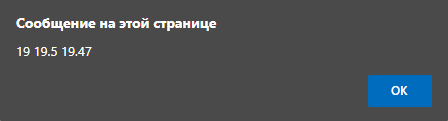


Рисунок 3.1

Листинг 3.2 – найти квадратный корень заданного числа и округлить его в большую и нижнюю сторону, записал как значение для ключа в объкте.

let res = Math.sqrt(587);

let result = {floor: Math.floor(res), ceil: Math.ceil(res)};

alert(result.floor + ' ' + result.ceil);

Результат выполнения кода на странице (Рисунок 3.2):

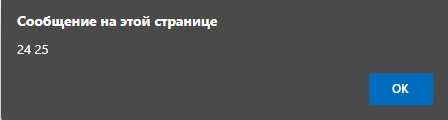


Рисунок 3.2

**Задача 4.**

Нахождение максимального и минимального числа:

1. Даны числа 4, -2, 5, 19, -130, 0, 10. Найти минимальное и максимальное число.

**Описание выполнения задания:**

Листинг 4.1 – найти минимальное и максимальное число в заданном наборе чисел.

let a = [4, -2, 5, 19, -130, 0, 10];

alert(Math.max(a[0], a[1], a[2], a[3], a[4], a[5], a[6]) + ' ' + Math.min(a[0], a[1], a[2], a[3], a[4], a[5], a[6]));

Результат выполнения кода на странице (Рисунок 4.1):

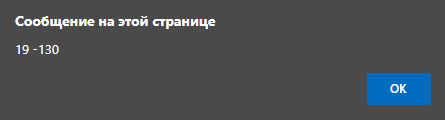
****

Рисунок 4.1

**Задача 5.**

Работа с рандомом:

1. Вывести на экран случайное целое число от 1 до 100.
2. Заполнить массив 10-ю случайными целыми числами. *(Подсказка: нужно воспользоваться циклами for или while).*

**Описание выполнения задания:**

Листинг 5.1 – вывести на экран случайное целое число от 1 до 100.

alert(Math.round(Math.random() \* 101));

Результат выполнения кода на странице (Рисунки 5.1):

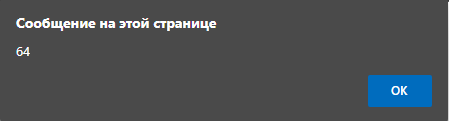


Рисунок 5.1

Листинг 5.2 – заполнить массив 10-ю разными случайными числами.

let a = [];

for(let i = 0; i < 10; i++) {

a[i] = Math.round(Math.random() \* 101)

}

alert(a);

Результат выполнения кода на странице (Рисунки 5.2):

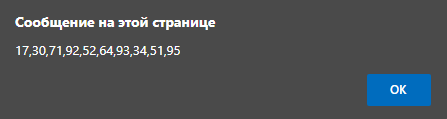


Рисунок 5.2

**Задача 6.**

Работа с модулем:

1. Даны переменные a и b. Найти модуль разности a и b. Проверить работу скрипта для различных a и b.
2. Даны переменные a и b. Отнять от a переменную b и результат присвоить переменной c. Сделать так, чтобы в любом случае в переменную c записалось положительное значение. Проверить работу скрипта при a и b, равных соответственно 3 и 5, 6 и 1.

**Описание выполнения задания:**

Листинг 6.1 – найти модуль разности двух переменных.

let a = -10;

let b = 23;

alert(Math.abs(a-b));

Результат выполнения кода на странице (Рисунок 6.1):

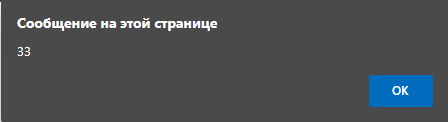


Рисунок 6.1

Листинг 6.2 – записать в переменную модуль разности двух других переменных.

let a = 6;

let b = 1;

let c = Math.abs(a-b);

alert(c);

Результат выполнения кода на странице (Рисунок 6.2):

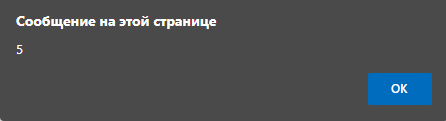


Рисунок 6.2

**Задача 7.**

Работа с массивами:

1. Дан массив arr. Найти среднее арифметическое его элементов. Проверить задачу на массиве с элементами 12, 15, 20, 25, 59, 79.
2. Написать скрипт, который будет находить факториал числа. Факториал (обозначается !) - это произведение (умножение) всех целых чисел, меньше данного, и его самого. Например, 4! = 1\*2\*3\*4.

**Описание выполнения задания:**

Листинг 7.1 - найти среднее арифметическое элементов заданного массива.

let arr = [12, 15, 20, 25, 59, 79];

let sum = 0;

for(let i = 0; i < arr.length; i++) {

sum += arr[i];

}

sum /= arr.length;

alert(sum);

Результат выполнения кода на странице (Рисунок 7.1):

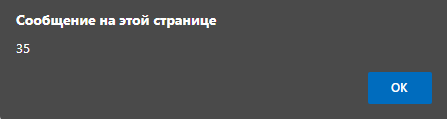


Рисунок 7.1

Листинг 7.2 - создать скрипт для нахождения факториала числа

let a = 5;

let res = 1;

for(let i = 2; i <= a; i++) {

res \*= i;

}

alert(res);

Результат выполнения кода на странице (Рисунок 7.2):

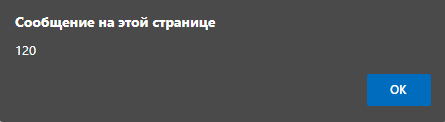


Рисунок 7.2

**Задача 8.**

Работа с регистром символов:

1. Дана строка 'js'. Сделать из нее строку 'JS'.
2. Дана строка 'JS'. Сделать из нее строку 'js'.

**Описание выполнения задания:**

Листинг 8.1 – преобразовать строку к верхнему регистру.

let a = 'js';

alert(a.toUpperCase());

Результат выполнения кода на странице (Рисунок 8.1):

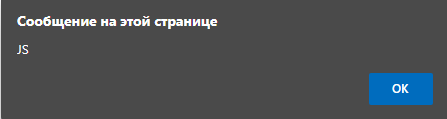


Рисунок 8.1

Листинг 8.2 –привести строку к нижнему регистру.

let a = 'JS';

alert(a.toLowerCase());

Результат выполнения кода на странице (Рисунок 8.2):

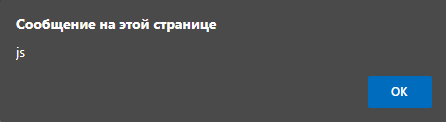


Рисунок 8.2

**Задача 9.**

Работа с length, substr, substring, slice. Работа с indexOf:

* 1. Дана строка 'я учу javascript!'. Найти количество символов в этой строке.
  2. Дана строка 'я учу javascript!'. Вырезать из нее слово 'учу' и слово 'javascript' тремя разными способами (через substr, substring, slice).
  3. Дана строка 'я учу javascript!'. Найти позицию подстроки 'учу'.
  4. Дана переменная str, в которой хранится какой-либо текст. Реализовать обрезание длинного текста по следующему принципу: если количество символов этого текста больше заданного в переменной n, то в переменную result записать первые n символов строки str и добавить в конец троеточие '...'. В противном случае в переменную result записать содержимое переменной str.

**Описание выполнения задания:**

Листинг 9.1 - найти количество символов в данной строке.

let a = 'я учу javascript!';

alert(a.length);

Результат выполнения кода на странице (Рисунок 9.1):

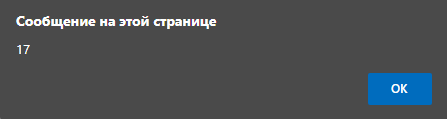


Рисунок 9.1

Листинг 9.2 -вырезать слова из строки с помощью методов subsrt, substring, slice.

let a = 'я учу javascript!';

alert(a.substr(2, 3) + a.substr(6));

alert(a.substring(2, 5) + a.substring(6));

alert(a.slice(1, 2) + a.slice(2));

Результат выполнения кода на странице (Рисунок 9.2):

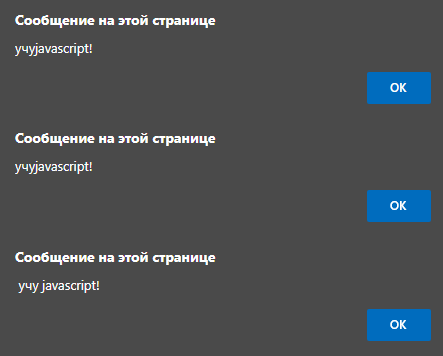


Рисунок 9.2

Листинг 9.3 - найти позицию подстроки в строке.

let a = 'я учу javascript!';

alert(a.indexOf('учу'));

Результат выполнения кода на странице (Рисунок 9.3):

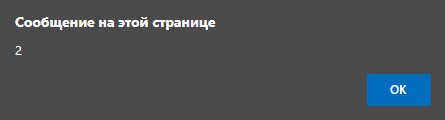


Рисунок 9.3

Листинг 9.4 - в зависимости от размера строки обрезать ее или присвоить полностью переменной и вывести ее.

let str = 'я учу javascript!';

let n = 5;

let result;

if(str.length > n) {

result = str.substr(0, n);

}

else {

result = str;

}

alert(result);

Результат выполнения кода на странице (Рисунок 9.4):

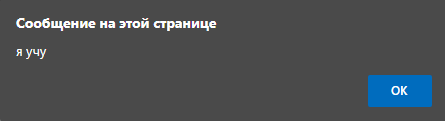


Рисунок 9.4

**Задача 10.**

Работа с replace:

1. Дана строка 'Я-учу-javascript!'. Заменить все дефисы на '!' с помощью глобального поиска и замены..

**Описание выполнения задания:**

Листинг 10.1 – проверить работу условия равенства нулю.

let a = 'Я-учу-javascript!';

a = a.replace(/-/g, '!');

alert(a);

Результат выполнения кода на странице (Рисунок 10.1):

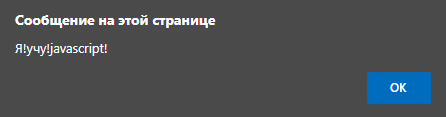


Рисунок 10.1

**Задача 11.**

Работа с split:

* 1. Дана строка 'я учу javascript!'. С помощью метода split записать каждое слово этой строки в отдельный элемент массива.
  2. Дана строка 'я учу javascript!'. С помощью метода split записать каждый символ этой строки в отдельный элемент массива.
  3. В переменной date лежит дата в формате '2020-11-18'. Преобразовать эту дату в формат '18.11.2020'.

**Описание выполнения задания:**

Листинг 11.1 – записать каждое отдельное слово строки в элемент массива.

let a = 'я учу javascript!';

let arr = a.split(' ');

alert(arr);

Результат выполнения кода на странице (Рисунок 11.1):

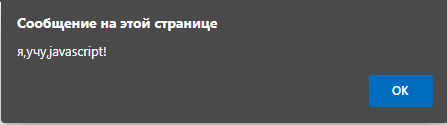


Рисунок 11.1

Листинг 11.2 – записать каждый символ строки в элемент массива.

let a = 'я учу javascript!';

let arr = a.split('');

alert(arr);

Результат выполнения кода на странице (Рисунок 11.2):

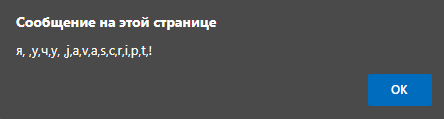


Рисунок 11.2

Листинг 11.3 – преобразовать строку даты в другой формат.

let date = '2020-11-18';

let res = date.split('-').reverse().join('.');

alert(res);

Результат выполнения кода на странице (Рисунок 11.2):

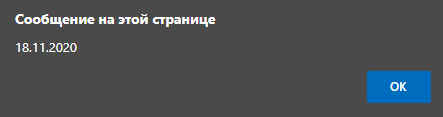


Рисунок 11.2

**Задача 12.**

Работа с join:

1. Дан массив ['я', 'учу', 'javascript', '!']. С помощью метода join преобразовать массив в строку 'я+учу+javascript+!'.

**Описание выполнения задания:**

Листинг 12.1 –создать строку из элементов массива с помощью метода join.

let a = ['я', 'учу', 'javascript', '!'];

alert(a.join('+'));

Результат выполнения кода на странице (Рисунок 12.1):

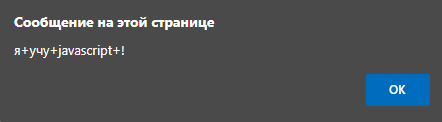


Рисунок 12.1

**Задача 13.**

Работа с concat. Даны два массива: [1, 2, 3] и [4, 5, 6]. Объединить их вместе.

**Описание выполнения задания:**

Листинг 13 – объединить два массива.

let a = [1, 2, 3];

let b = [4, 5, 6];

a = a.concat(b);

alert(a);

Результат выполнения кода на странице (Рисунок 13):

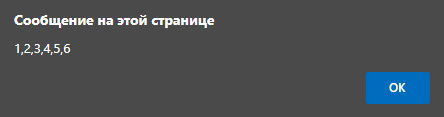


Рисунок 13

**Задача 14.**

Работа с reverse. Дан массив [1, 2, 3]. Сделать из него массив [3, 2, 1].

**Описание выполнения задания:**

Листинг 14.1 – вывести с помощью конструкций цикла for и while числа от 1 до 100.

let a = [1, 2, 3];

alert(a.reverse());

Результат выполнения кода на странице(первые 50 цифр) (Рисунок 14.1):

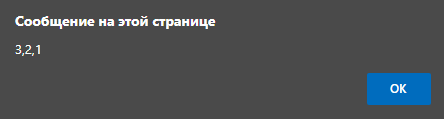


Рисунок 11.1

**Задача 15.**

Работа с push, unshift:

1. Дан массив [1, 2, 3]. Добавить ему в конец элементы 4, 5, 6.
2. Дан массив [1, 2, 3]. Добавить ему в начало элементы 4, 5, 6.

**Описание выполнения задания:**

Листинг 15.1 – добавить в конец заданного массива элементы.

let a = [1, 2, 3];

a.push(4,5,6);

alert(a);

Результат выполнения кода на странице (Рисунок 15.1):

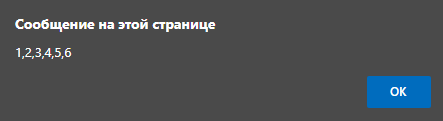


Рисунок 15.1

Листинг 15.2 – добавить в начало заданного массива элементы.

let a = [1, 2, 3];

a.unshift(4,5,6);

alert(a);

Результат выполнения кода на странице (Рисунок 15.2):

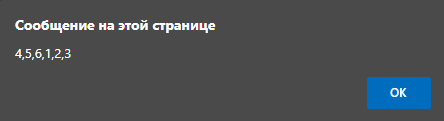


Рисунок 15.2

**Задача 16.**

Работа с shift, pop:

1. Дан массив ['js', 'css', 'jq']. Вывести на экран первый элемент.
2. Дан массив ['js', 'css', 'jq']. Вывести на экран последний элемент.

**Описание выполнения задания:**

Листинг 16.1 – вывести на экран первый элемент массива.

let a = ['js', 'css', 'jq'];

alert(a.shift());

Результат выполнения кода на странице (Рисунок 16.1):

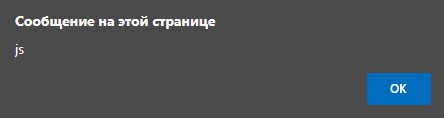


Рисунок 16.1

Листинг 16.2 – вывести с помощью цикла for-in ключи и элементы заданного объекта в определенном формате.

let a = ['js', 'css', 'jq'];

alert(a.pop());

Результат выполнения кода на странице (Рисунок 16.2):

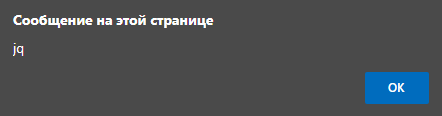


Рисунок 16.2

**Задача 17.**

Работа со slice:

1. Дан массив [1, 2, 3, 4, 5]. С помощью метода slice записать в новый элементы [1, 2, 3].
2. Дан массив [1, 2, 3, 4, 5]. С помощью метода slice записать в новый элементы [4, 5].

**Описание выполнения задания:**

Листинг 17.1 – записать срез массива в переменную и вывести ее.

let a = [1, 2, 3, 4, 5];

let b = a.slice(0, 3);

alert(b);

Результат выполнения кода на странице (Рисунок 17.1):

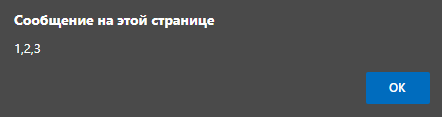


Рисунок 17.1

Листинг 17.2 – записать срез массива с определенного индекса в переменную и вывести ее.

let a = [1, 2, 3, 4, 5];

let b = a.slice(3);

alert(b);

Результат выполнения кода на странице (Рисунок 17.2):

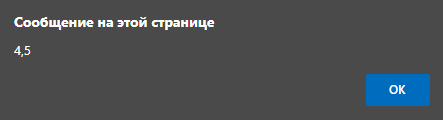


Рисунок 17.2

**Задача 18.**

Работа со splice:

1. Дан массив [1, 2, 3, 4, 5]. С помощью метода splice преобразовать массив в [1, 4, 5].
2. Дан массив [1, 2, 3, 4, 5]. С помощью метода splice записать в новый массив элементы [2, 3, 4].
3. Дан массив [1, 2, 3, 4, 5]. С помощью метода splice сделать из него массив [1, 2, 3, 'a', 'b', 'c', 4, 5].
4. Дан массив [1, 2, 3, 4, 5]. С помощью метода splice сделать из него массив [1, 'a', 'b', 2, 3, 4, 'c', 5, 'e'].

**Описание выполнения задания:**

Листинг 18.1 – преобразовать массив заданным образом с помощью метода splice.

let a = [1, 2, 3, 4, 5];

a.splice(1, 2);

alert(a);

Результат выполнения кода на странице (Рисунок 18.1):

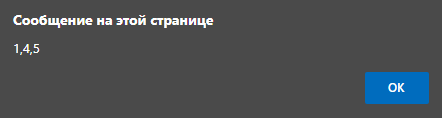


Рисунок 18.1

Листинг 18.2 – преобразовать массив заданным образом с помощью метода splice.

let a = [1, 2, 3, 4, 5];

let b = a.splice(1,3);

alert(b);

Результат выполнения кода на странице (Рисунок 18.2):

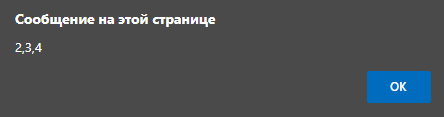


Рисунок 18.2

Листинг 18.3 – преобразовать массив заданным образом с помощью метода splice.

let a = [1, 2, 3, 4, 5];

a.splice(3, 0, 'a', 'b', 'c');

alert(a);

Результат выполнения кода на странице (Рисунок 18.3):

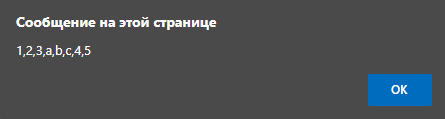


Рисунок 18.3

Листинг 18.4 – преобразовать массив заданным образом с помощью метода splice.

let a = [1, 2, 3, 4, 5];

a.splice(1, 0, 'a', 'b');

a.splice(6, 0, 'c');

a.splice(8, 0, 'e');

alert(a);

Результат выполнения кода на странице (Рисунок 18.2):

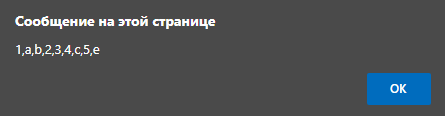


Рисунок 18.2

**Задача 19.**

Работа с sort. Дан массив [3, 4, 1, 2, 7]. Отсортировать его.

**Описание выполнения задания:**

Листинг 19 – отсортировать заданный массив.

let a = [3, 4, 1, 2, 7];

a.sort();

alert(a);

Результат выполнения кода на странице (Рисунок 19):

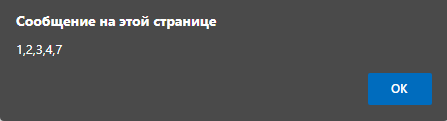


Рисунок 19

**Задача 20.**

Работа с Object.key. Дан объект {js:'test', jq: 'hello', css: 'world'}. Получить массив его ключей.

**Описание выполнения задания:**

Листинг 20 – вывести массив ключей заданного объекта.

let a = {js:'test', jq: 'hello', css: 'world'};

let b = Object.keys(a);

alert(b);

Результат выполнения кода на странице (Рисунок 20):

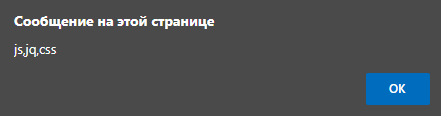


Рисунок 20