



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-02-19

Миронов А.Д.

Принял
Ассистент

Меркулов Е.С.

Практические работы выполнены

«__» _____ 2020 г.

(подпись студента)

«Зачтено»

«__» _____ 2020 г.

(подпись руководителя)

Москва 2020

Оглавление

Практическая работа №6: «JavaScript. Математические функции. Строковые функции. Функции для массивов»	3
Практическое задание №1: «Работа с %»	3
Практическое задание №2: «Работа со степенью и корнем»	5
Практическое задание №3: «Работа с функциями округления»	7
Практическое задание №4: «Нахождение минимального и максимального числа»	8
Практическое задание №5: «Работа с рандомом»	9
Практическое задание №6: «Работа с модулем»	10
Практическое задание №7: «Задачи на математические функции»	11
Практическое задание №8: «Работа с регистром символов»	12
Практическое задание №9: «Работа с length, substr, substring, slice. Работа с indexOf»	13
Практическое задание №10: «Работа с replace»	15
Практическое задание №11: «Работа с split»	16
Практическое задание №12: «Работа с join»	18
Практическое задание №13: «Работа с concat»	19
Практическое задание №14: «Работа с reverse»	20
Практическое задание №15: «Работа с push, unshift»	21
Практическое задание №16: «Работа с shift, pop»	22
Практическое задание №17: «Работа со slice»	23
Практическое задание №18: «Работа со splice»	24
Практическое задание №19: «Работа с sort»	26
Практическое задание №20: «Работа с Object.keys»	27

Практическая работа №6: «JavaScript. Математические функции. Строковые функции. Функции для массивов»

Практическое задание №1: «Работа с %»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.1).

Листинг 1.1 – Создание .js файла

```
var a=10, b=3;
alert(a%b);

if (a%b==0)
    alert("Делится");
else alert("Делится с остатком: "+a%b);
```

Результат выполнения в консоли (Рисунок 1.1-1.3):

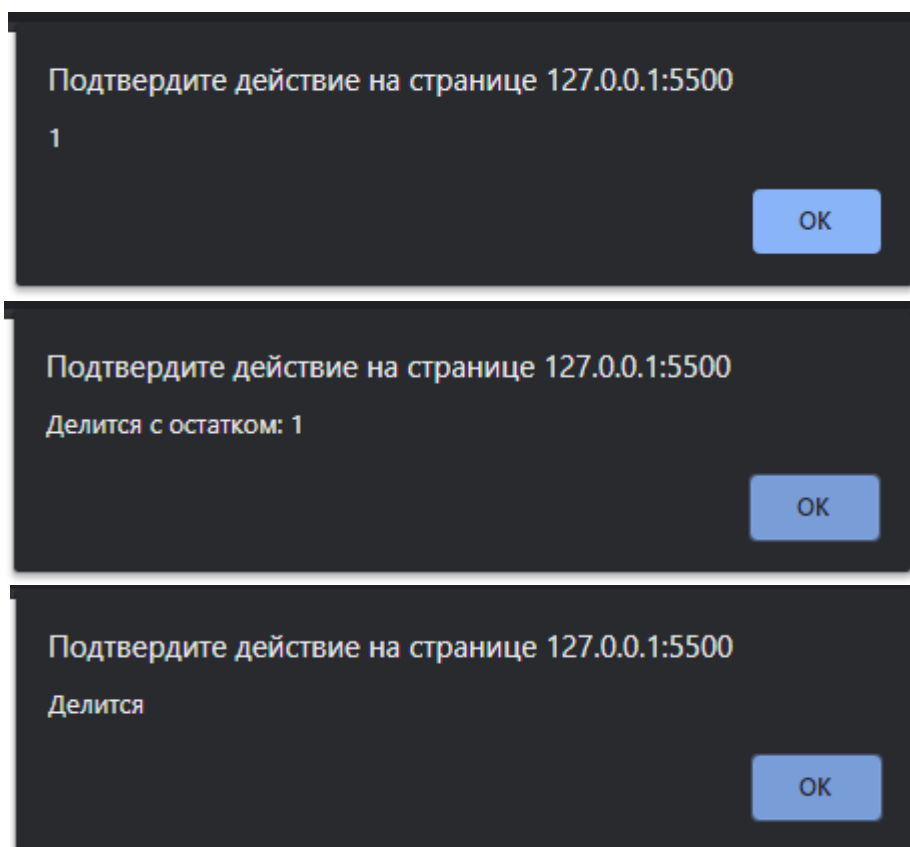


Рисунок 1.1-1.3 – Результат выполнения кода

Практическое задание №2: «Работа со степенью и корнем»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.2).

Листинг 1.2 – Создание .js файла

```
var st=Math.pow(2, 10);  
alert(st);  
  
alert(Math.sqrt(245));  
  
var m=[4, 2, 5, 19, 13, 0, 10], res=0;  
for (var i=0; i<m.length; i++)  
    res+=Math.pow(m[i], 3)  
alert(Math.sqrt(res));
```

Результат выполнения кода в консоли (Рисунок 1.4-1.6):

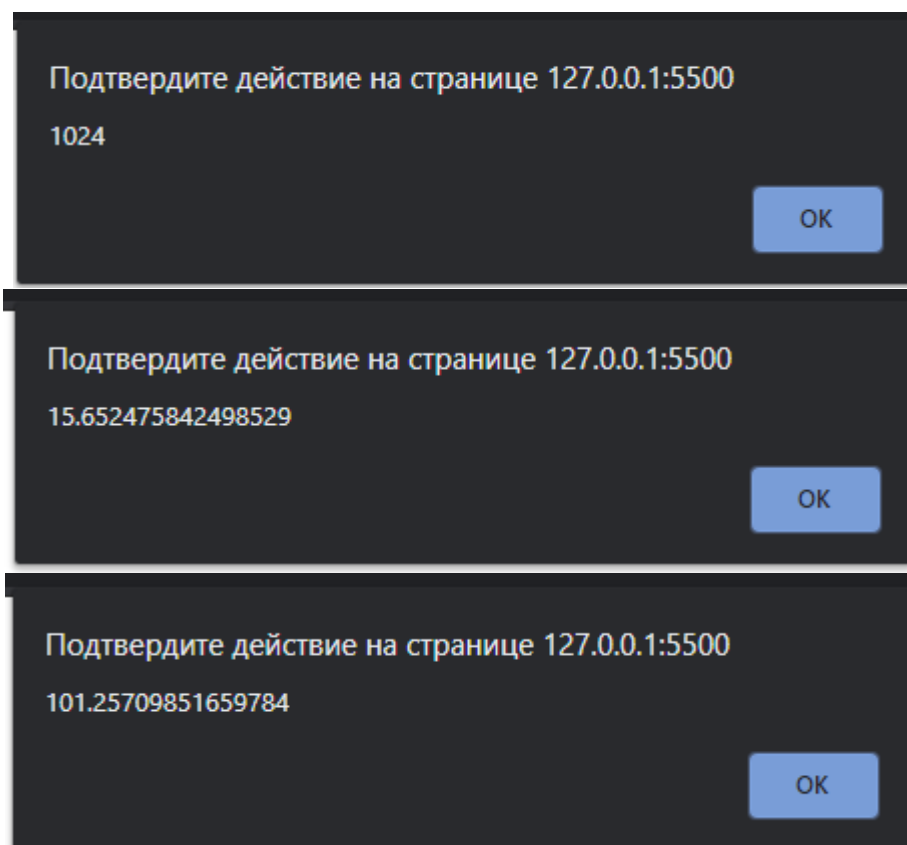


Рисунок 1.4-1.6 – Результат выполнения кода

Практическое задание №3: «Работа с функциями округления»

Задание:

1. Найти квадратный корень из 379. Результат 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'.

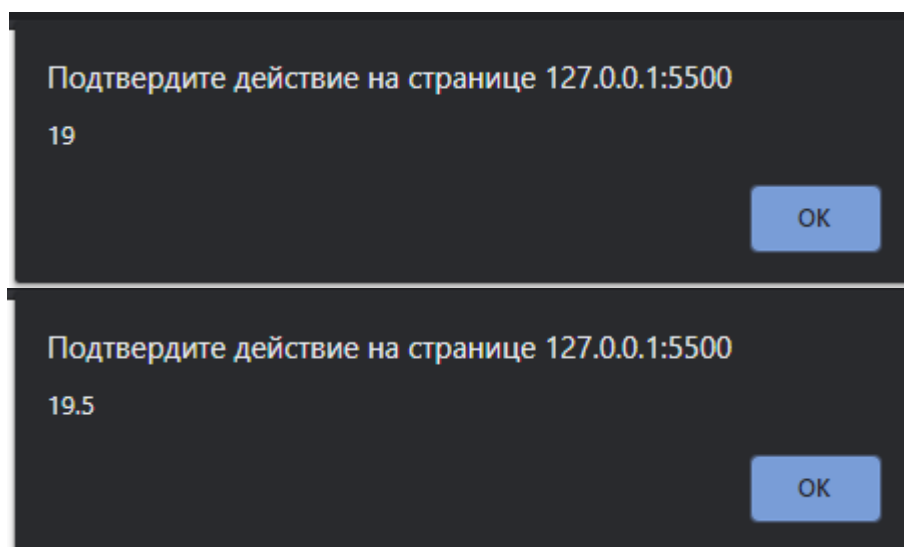
Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.3).

Листинг 1.3 – Создание .js файла

```
alert(Math.sqrt(379).toFixed(0));  
alert(Math.sqrt(379).toFixed(1));  
alert(Math.sqrt(379).toFixed(2));  
  
var obj={  
    floor:0,  
    ceil:0  
};  
obj.floor=Math.floor(Math.sqrt(587));  
obj.ceil=Math.ceil(Math.sqrt(587));  
alert(obj.floor+" "+obj.ceil);
```

Результат выполнения кода в консоли (Рисунок 1.7-1.10):



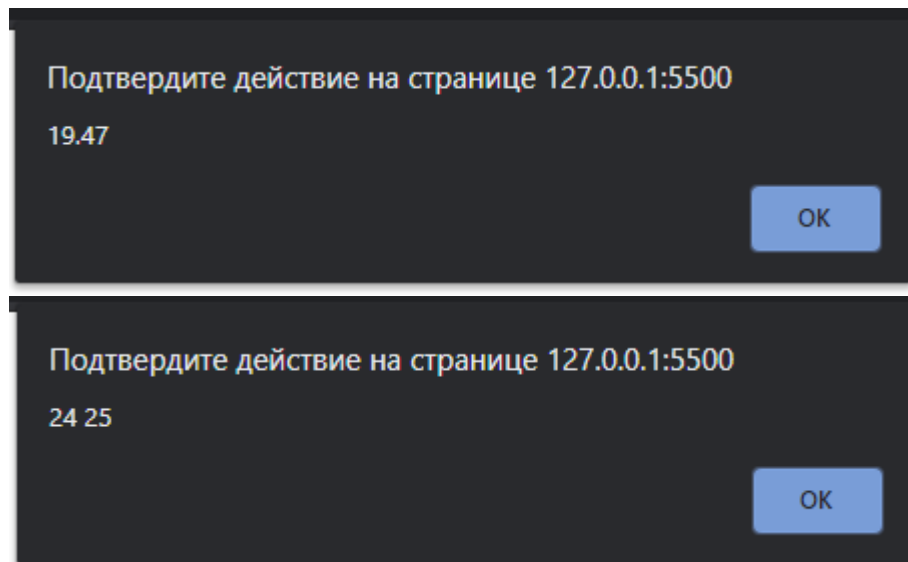


Рисунок 1.7-1.10 – Результат выполнения кода

Практическое задание №4: «Нахождение минимального и максимального числа»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.4).

Листинг 1.4 – Создание .js файла

```
alert(Math.max(4, -2, 5, 19, -130, 0, 10));  
alert(Math.min(4, -2, 5, 19, -130, 0, 10));
```

Результат выполнения кода в консоли (Рисунок 1.11-1.12):

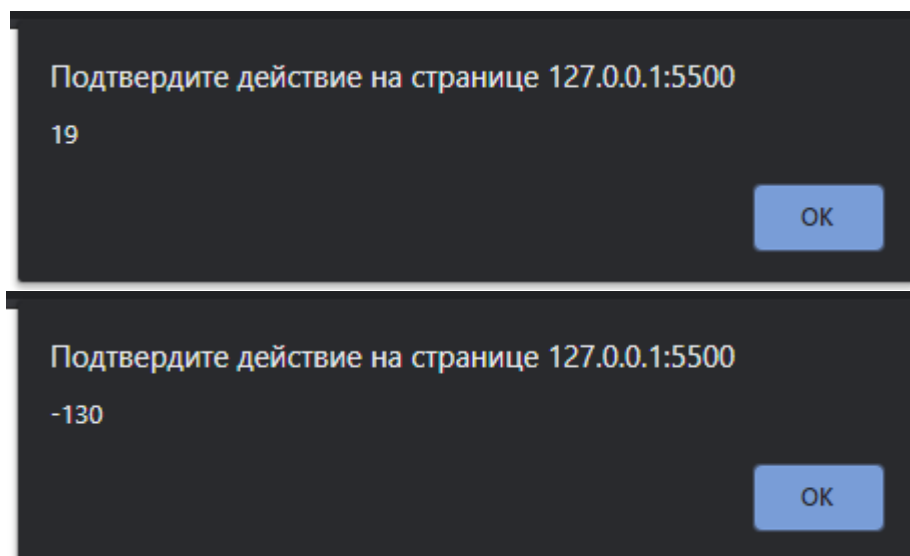


Рисунок 1.11-1.12 – Результат выполнения кода

Практическое задание №5: «Работа с рандомом»

Задание:

1. Вывести на экран случайное целое число от 1 до 100.
2. Заполнить массив 10-ю случайными целыми числами.

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.5).

Листинг 1.5 – Создание .js файла

```
var temp=Math.floor(Math.random()*100+1);  
alert(temp);  
  
var arr=[];  
for (var i=0; i<10; i++)  
    arr.push(Math.floor(Math.random()*100+1));  
alert(arr);
```

Результат выполнения кода в консоли (Рисунок 1.13-1.14):

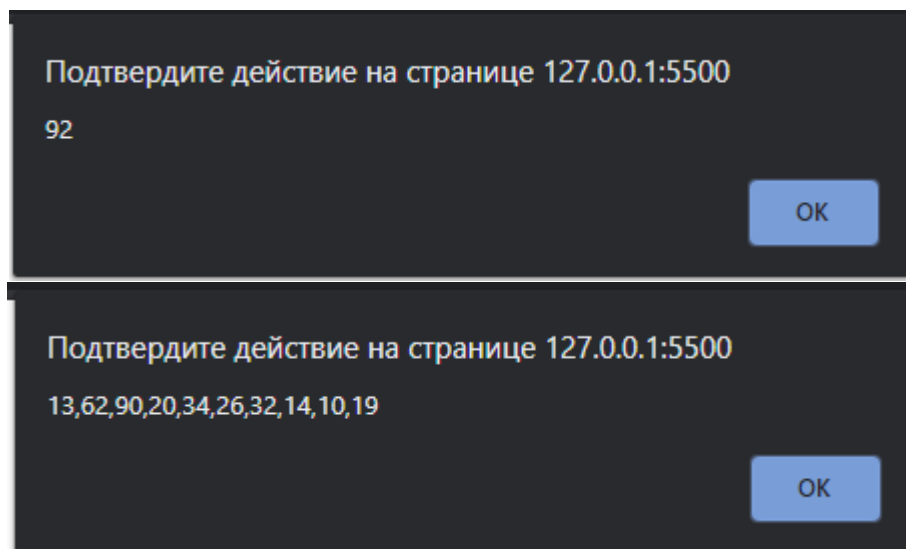


Рисунок 1.13-1.14 – Результат выполнения кода

Практическое задание №6: «Работа с модулем»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.6).

Листинг 1.6 – Создание .js файла

```
var a=0, b=1;  
alert(Math.abs(a-b));  
  
var c=Math.abs(a-b);  
alert(c);
```

Результат выполнения кода в консоли (Рисунок 1.15-1.17):

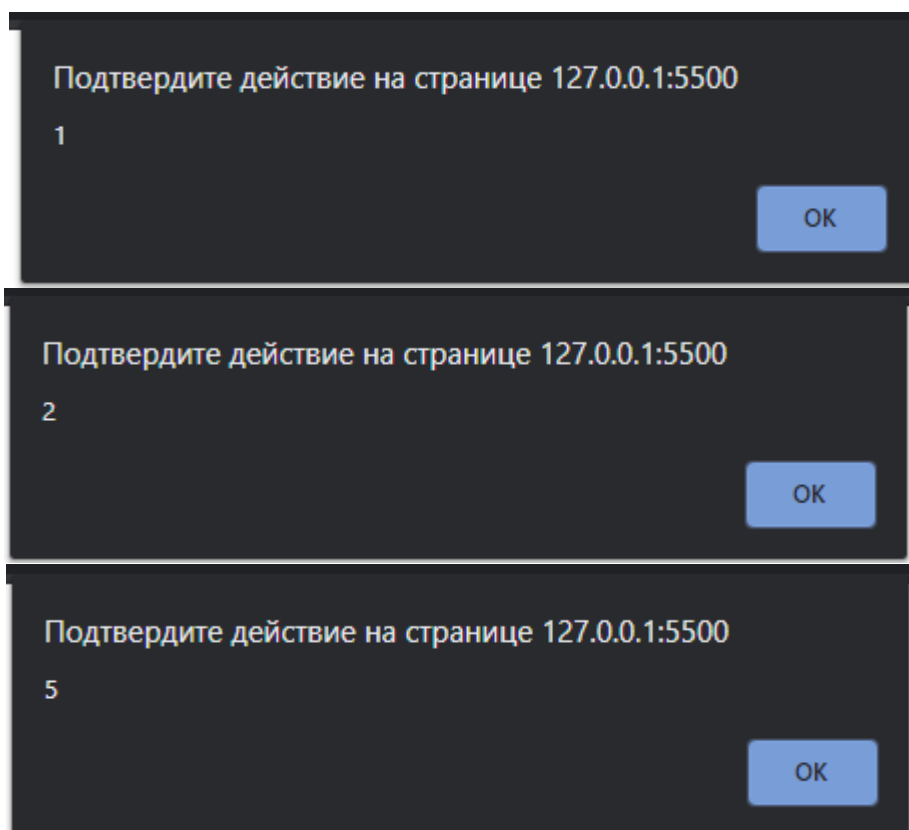


Рисунок 1.15-1.17 – Результат выполнения кода

Практическое задание №7: «Задачи на математические функции»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.7).

Листинг 1.7 – Создание .js файла

```
var arr=[12, 15, 20, 25, 59, 79], result=0;
for (var i=0; i<arr.length; i++)
    result+=arr[i];
alert(result/arr.length);

var num=5;
result=1;
while(num>=1)
{
    result*=num;
    num--;
}
alert(result);
```

Результат выполнения кода в консоли (Рисунок 1.18-1.19):

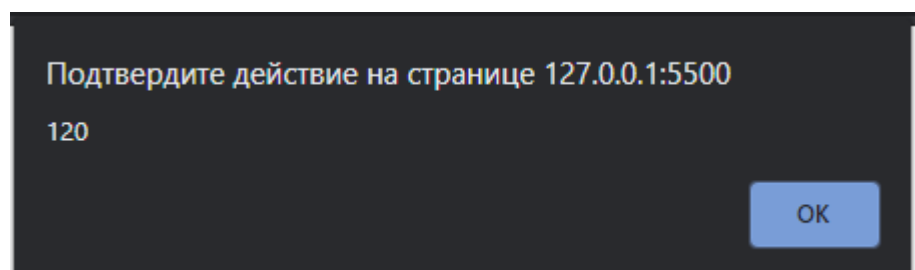
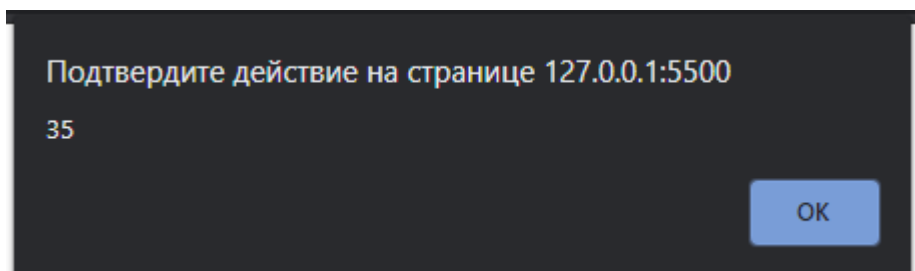


Рисунок 1.18-1.19 – Результат выполнения кода

Практическое задание №8: «Работа с регистром символов»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.8).

Листинг 1.8 – Создание .js файла

```
var str="js";  
alert(str.toUpperCase());  
  
var str="JS";  
alert(str.toLowerCase());
```

Результат выполнения кода в консоли (Рисунок 1.20-1.21):

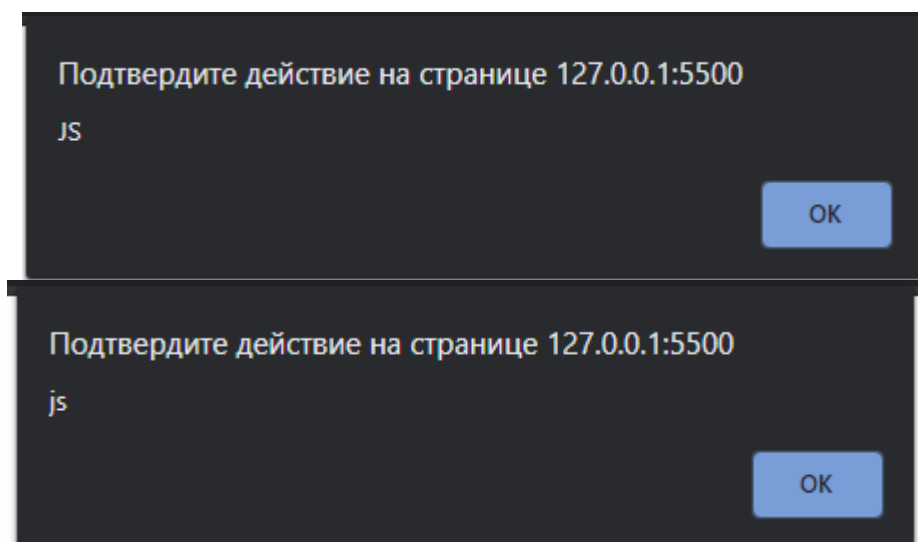


Рисунок 1.20-1.21 – Результат выполнения кода

Практическое задание №9: «Работа с length, substr, substring, slice. Работа с indexOf»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.9).

Листинг 1.9 – Создание .js файла

```
var str="я учу javascript!";
alert(str.length);

alert(str.substr(0, 1));
alert(str.substring(0, 1));
alert(str.slice(0, 1));

alert(str.indexOf("учу"));

var length=20;
if (str.length>length){
    str=str.substr(0, length);
    str+="...";
}
alert(str);
```

Результат выполнения кода в консоли (Рисунок 1.22-1.27):

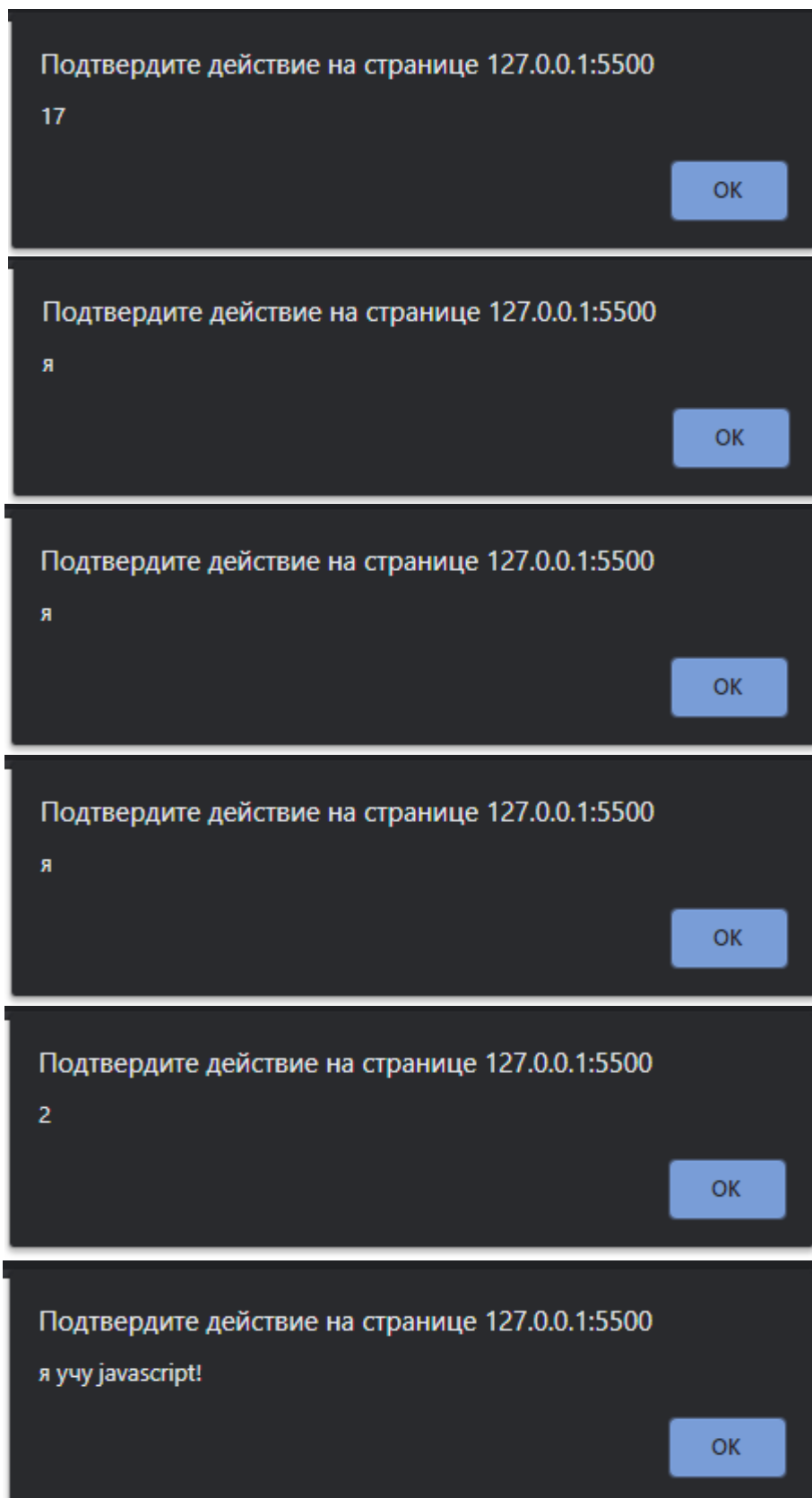


Рисунок 1.22-1.27– Результат выполнения кода

Практическое задание №10: «Работа с replace»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.10).

Листинг 1.10 – Создание .js файла

```
var str="Я-учу-javascript!";  
alert(str.replace(/-/g, '!'));
```

Результат выполнения кода в консоли (Рисунок 1.28):

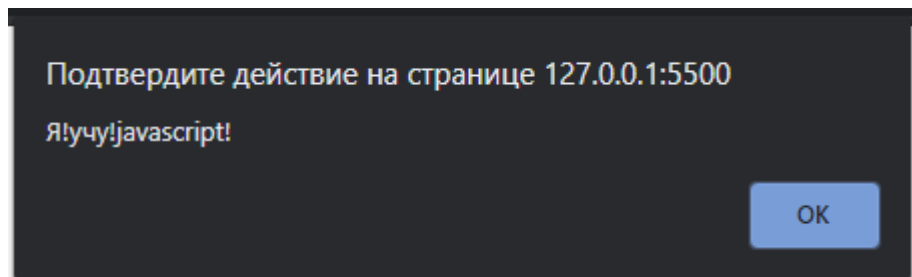


Рисунок 1.28 – Результат выполнения кода

Практическое задание №11: «Работа с split»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.11).

Листинг 1.11 – Создание .js файла

```
var str="я учу javascript!";
alert(str.split(' '));

alert(str.split(''));

str="2020-11-18";
var temp, result;
temp=str.substr(8);
result=temp+ ".";
temp=str.substr(5,2);
result+=temp+ ".";
temp=str.substr(0,4);
result+=temp;
alert(result);
```

Результат выполнения кода в консоли (Рисунок 1.29-1.31):

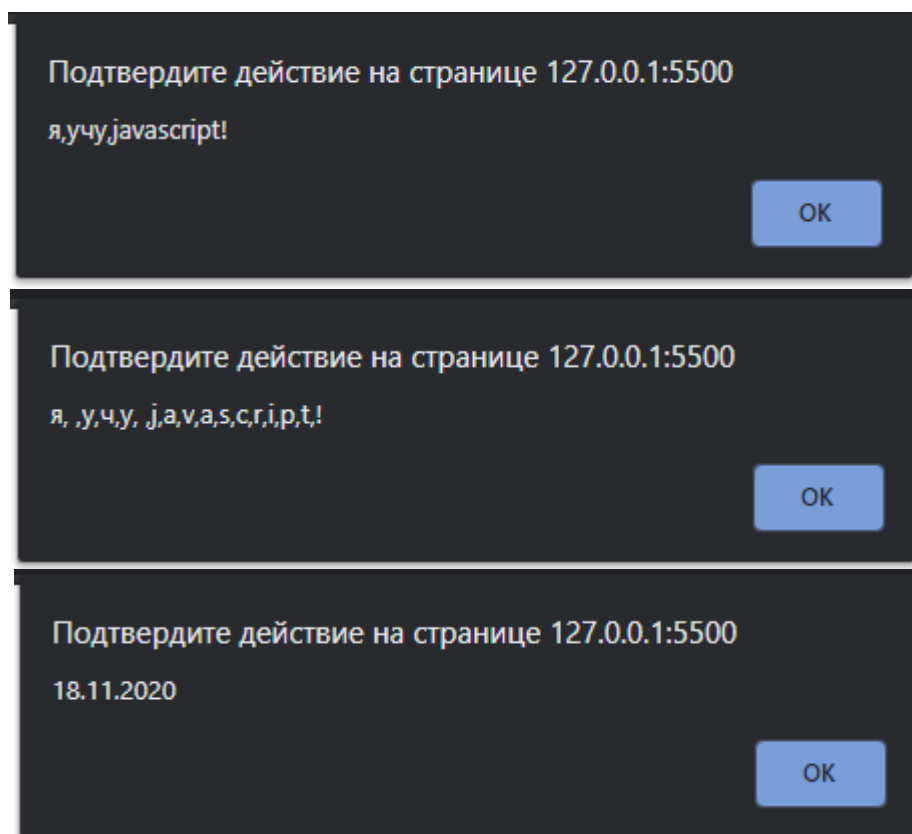


Рисунок 1.29-1.31 – Результат выполнения кода

Практическое задание №12: «Работа с join»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.12).

Листинг 1.12 – Создание .js файла

```
var arr=['я', 'учу', 'javascript', '!'];  
alert(arr.join('+'));
```

Результат выполнения кода в консоли (Рисунок 1.32):

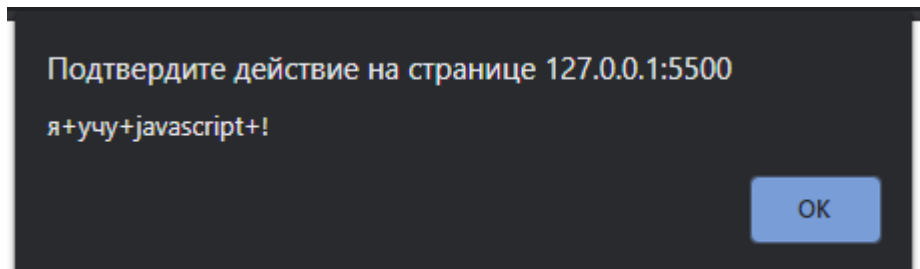


Рисунок 1.32 – Результат выполнения кода

Практическое задание №13: «Работа с concat»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.13).

Листинг 1.13 – Создание .js файла

```
var arr1=[1, 2, 3], arr2=[4, 5, 6];  
var arr3=arr1.concat(arr2);  
alert(arr3);
```

Результат выполнения кода в консоли (Рисунок 1.33):

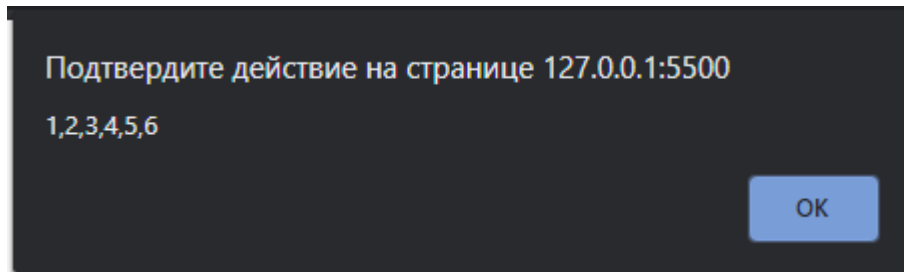


Рисунок 1.33 – Результат выполнения кода

Практическое задание №14: «Работа с reverse»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.14).

Листинг 1.14 – Создание .js файла

```
var arr1=[1, 2, 3];  
var arr2=arr1.reverse();  
alert(arr2);
```

Результат выполнения кода в консоли (Рисунок 1.34):

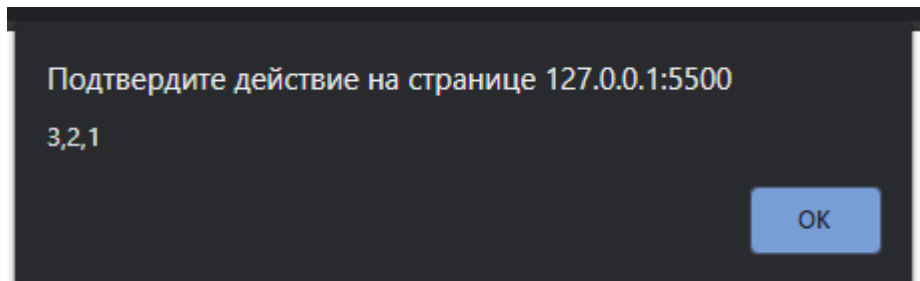


Рисунок 1.34 – Результат выполнения кода

Практическое задание №15: «Работа с push, unshift»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.15).

Листинг 1.15– Создание .js файла

```
var arr1=[1, 2, 3];  
arr1.push(4);  
arr1.push(5);  
arr1.push(6);  
alert(arr1);  
  
var arr2=[1, 2, 3];  
arr2.unshift(4);  
arr2.unshift(5);  
arr2.unshift(6);  
alert(arr2);
```

Результат выполнения кода в консоли (Рисунок 1.35):

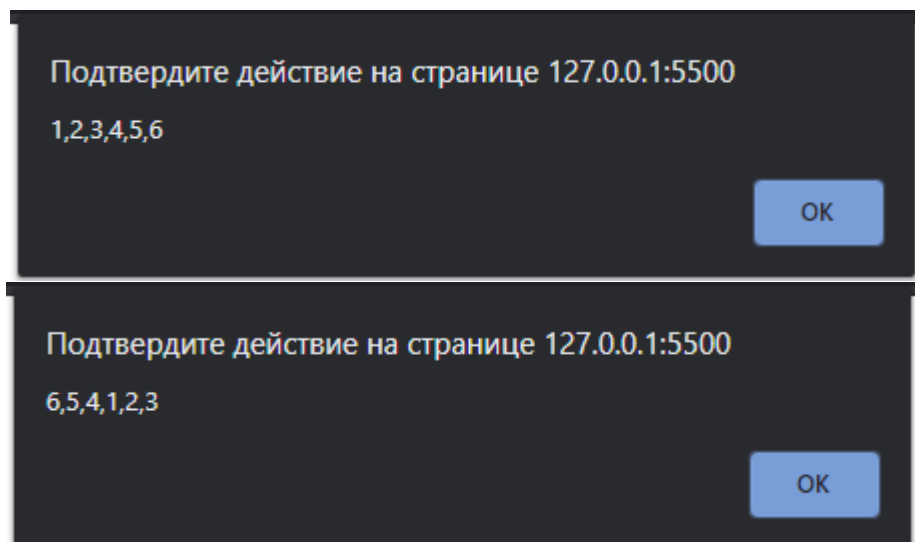


Рисунок 1.35 – Результат выполнения кода

Практическое задание №16: «Работа с shift, pop»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.16).

Листинг 1.16 – Создание .js файла

```
var arr1=['js', 'css', 'jq'];  
alert(arr1.shift());  
  
alert(arr1.pop());  
var arr1=['js', 'css', 'jq'];  
alert(arr1.shift());  
  
alert(arr1.pop());
```

Результат выполнения кода в консоли (Рисунок 1.36):

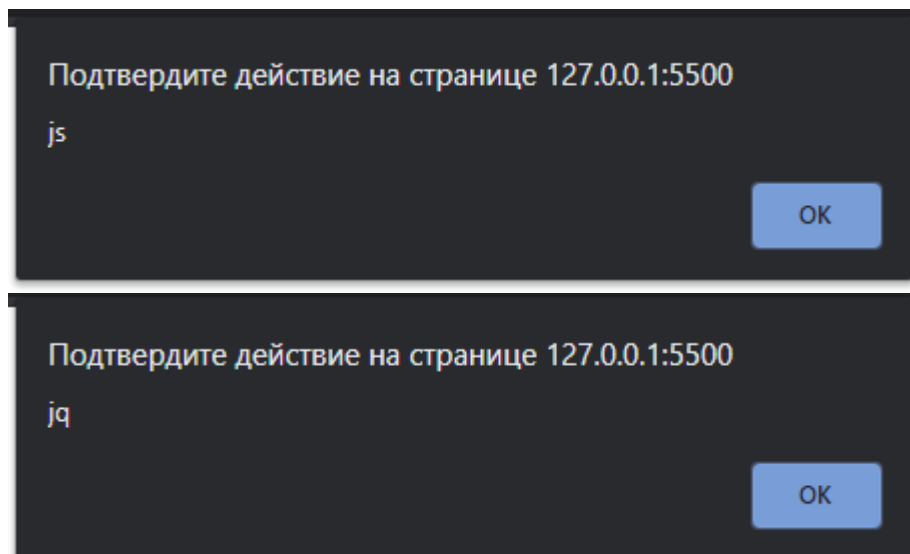


Рисунок 1.36 – Результат выполнения кода

Практическое задание №17: «Работа со slice»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.17).

Листинг 1.17 – Создание .js файла

```
var arr1=[1, 2, 3, 4, 5];  
alert(arr1.slice(0, 3));  
  
alert(arr1.slice(3));
```

Результат выполнения кода в консоли (Рисунок 1.37-1.38):

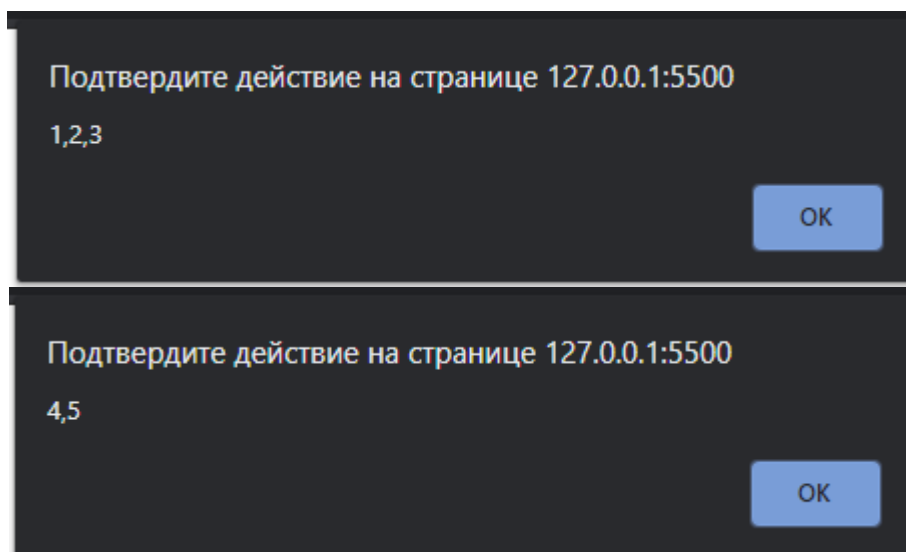


Рисунок 1.37-1.38 – Результат выполнения кода

Практическое задание №18: «Работа со splice»

Задание:

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

1. Создание .js файла с необходимой функциональностью (Листинг 1.18).

Листинг 1.18 – Создание .js файла

```
var arr1=[1, 2, 3, 4, 5];
arr1.splice(1, 2);
alert(arr1);

arr1=[1, 2, 3, 4, 5];
arr1.splice(0, 1);
arr1.splice(3, 1);
alert(arr1);

arr1=[1, 2, 3, 4, 5];
arr1.splice(3,0, 'a');
arr1.splice(4,0, 'b');
arr1.splice(5,0, 'c');
alert(arr1);

arr1=[1, 2, 3, 4, 5];
arr1.splice(1,0, 'a');
arr1.splice(2,0, 'b');
arr1.splice(6,0, 'c');
arr1.splice(8,0, 'e');
alert(arr1);
```

Результат выполнения кода в консоли (Рисунок 1.39-1.42):

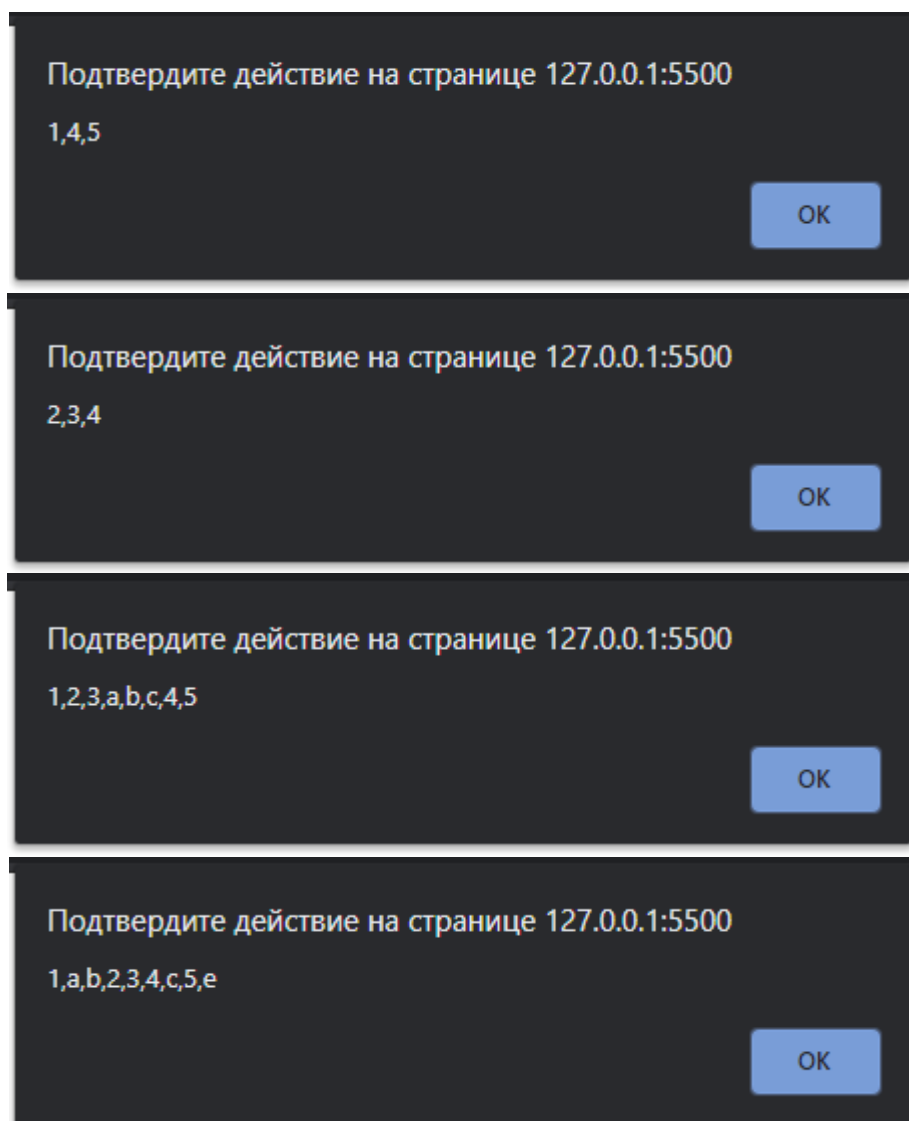


Рисунок 1.39-1.42 – Результат выполнения кода

Практическое задание №19: «Работа с sort»

Задание:

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

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.19).

Листинг 1.19 – Создание .js файла

```
var arr=[3, 4, 1, 2, 7];  
alert(arr.sort());
```

Результат выполнения кода в консоли (Рисунок 1.43):

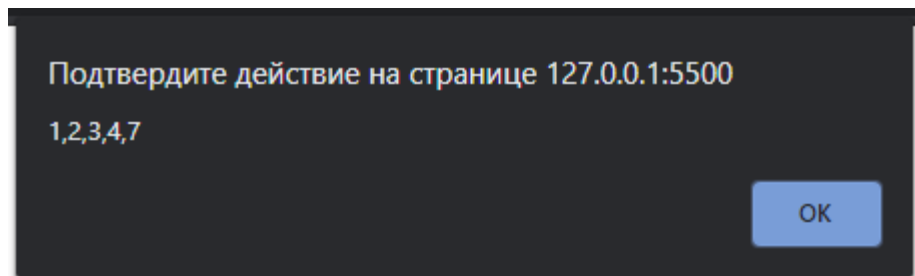


Рисунок 1.43 – Результат выполнения кода

Практическое задание №20: «Работа с Object.keys»

Задание:

1. Дан объект {js:'test', jq: 'hello', css: 'world'}. Получить массив его ключей.

Описание выполнения задания

1. Создание .js файла с необходимой функциональностью (Листинг 1.20).

Листинг 1.20 – Создание .js файла

```
var obj={js:'test', jq: 'hello', css: 'world'};  
alert(Object.keys(obj));
```

Результат выполнения кода в консоли (Рисунок 1.44):

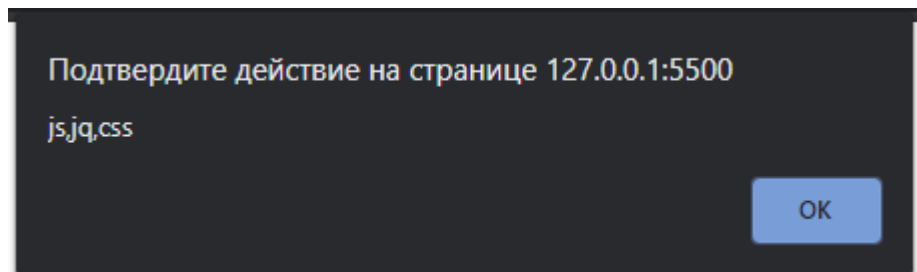


Рисунок 1.44 – Результат выполнения кода

Вывод: в ходе выполнения практической работы были освоены различные функции, такие как: функция нахождения максимального и минимального числа, строковые функции и функции для массивов в JavaScript: изучена работа с оператором деления с остатком, функциями степени и корня, округления до ближайшего целого, округление до большего и меньшего целого, работа с рандомом, модулем числа, регистрами символов, различными методами строк, в том числе length, substr, slice, indexOf, replace, методами массивов, в том числе sort, splice, shift, unshift, pop, push, sort и работа с методом Object.keys, возвращающим массив ключей данного объекта.