



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 №5: «JavaScript. Работа с переменными, строками, массивами, циклами и условиями» .....	3
Практическое задание №1: «Работа с переменными» .....	3
Практическое задание №2: «Работа со строками» .....	6
Практическое задание №3: «Функция prompt» .....	8
Практическое задание №4: «Обращение к символам строки» .....	10
Практическое задание №5: «Работа со временем» .....	12
Практическое задание №6: «Работа с присваиванием и декрементами» .....	13
Практическое задание №7: «Работа с массивами» .....	15
Практическое задание №8: «Объекты (ассоциативные массивы)» .....	17
Практическое задание №9: «Многомерные массивы» .....	19
Практическое задание №10: «Работа с if-else» .....	21
Практическое задание №11: «Работа с логическими переменными» .....	27
Практическое задание №12: «Работа с && (и) и    (или)» .....	29
Практическое задание №13: «Работа со switch-case» .....	33
Практическое задание №14: «Циклы while и for» .....	35
Практическое задание №15: «Работа с for для массивов» .....	40
Практическое задание №16: «Работа с for-in» .....	42
Практическое задание №17: «Задачи с циклами» .....	45

## Практическая работа №5: «JavaScript. Работа с переменными, строками, массивами, циклами и условиями»

### Практическое задание №1: «Работа с переменными»

#### Задание:

1. Создать переменную num и присвоить ей значение 3. Вывести значение этой переменной на экран с помощью метода alert.
2. Создать переменные a=10 и b=2. Вывести на экран их сумму, разность, произведение и частное (результат деления).
3. Создать переменные c=15 и d=2. Просуммировать их, а результат присвоить переменной result. Вывести на экран значение переменной result.
4. Создать переменные a=10, b=2 и c=5. Вывести на экран их сумму.
5. Создать переменные a=17 и b=10. Отнять от a переменную b и результат присвоить переменной c. Затем создать переменную d, присвоить ей значение 7. Сложить переменные c и d, а результат записать в переменную result. Вывести на экран значение переменной result.

#### Описание выполнения задания

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

#### Листинг 1.1 – Создаём .js файл

```
let num=3;
alert(num);

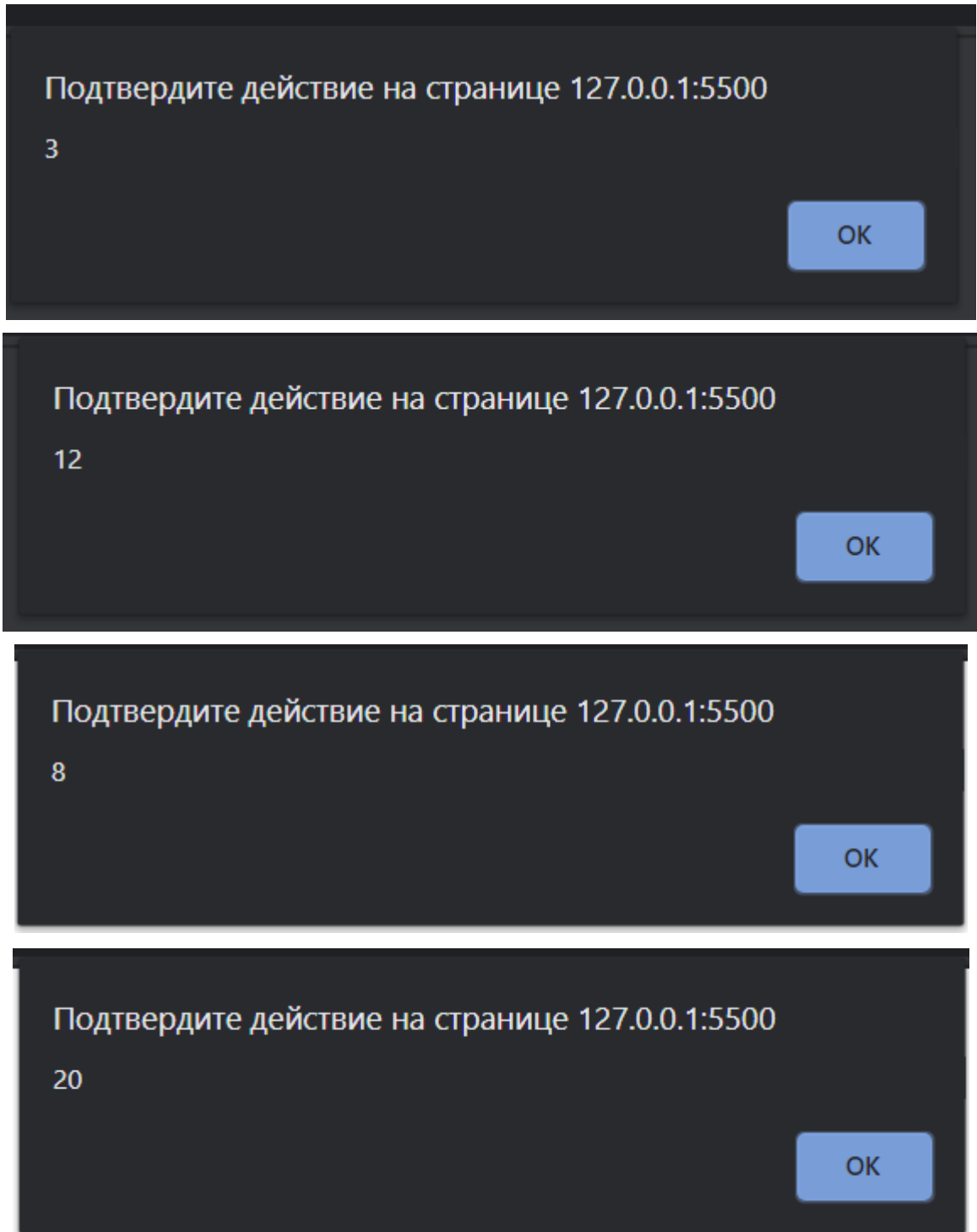
var a=10, b=2;
alert(a+b);
alert(a-b);
alert(a*b);
alert(a/b);

var c=15, d=2;
var result = c+d;
alert(result);

var a=10, b=2, c=5;
```

```
alert(a+b+c);  
var a=17, b=10;  
var c=a-b;  
var d=7;  
var result = c+d;  
alert(result);
```

Результат выполнения кода на странице (Рисунки 1.1 – 1.8):



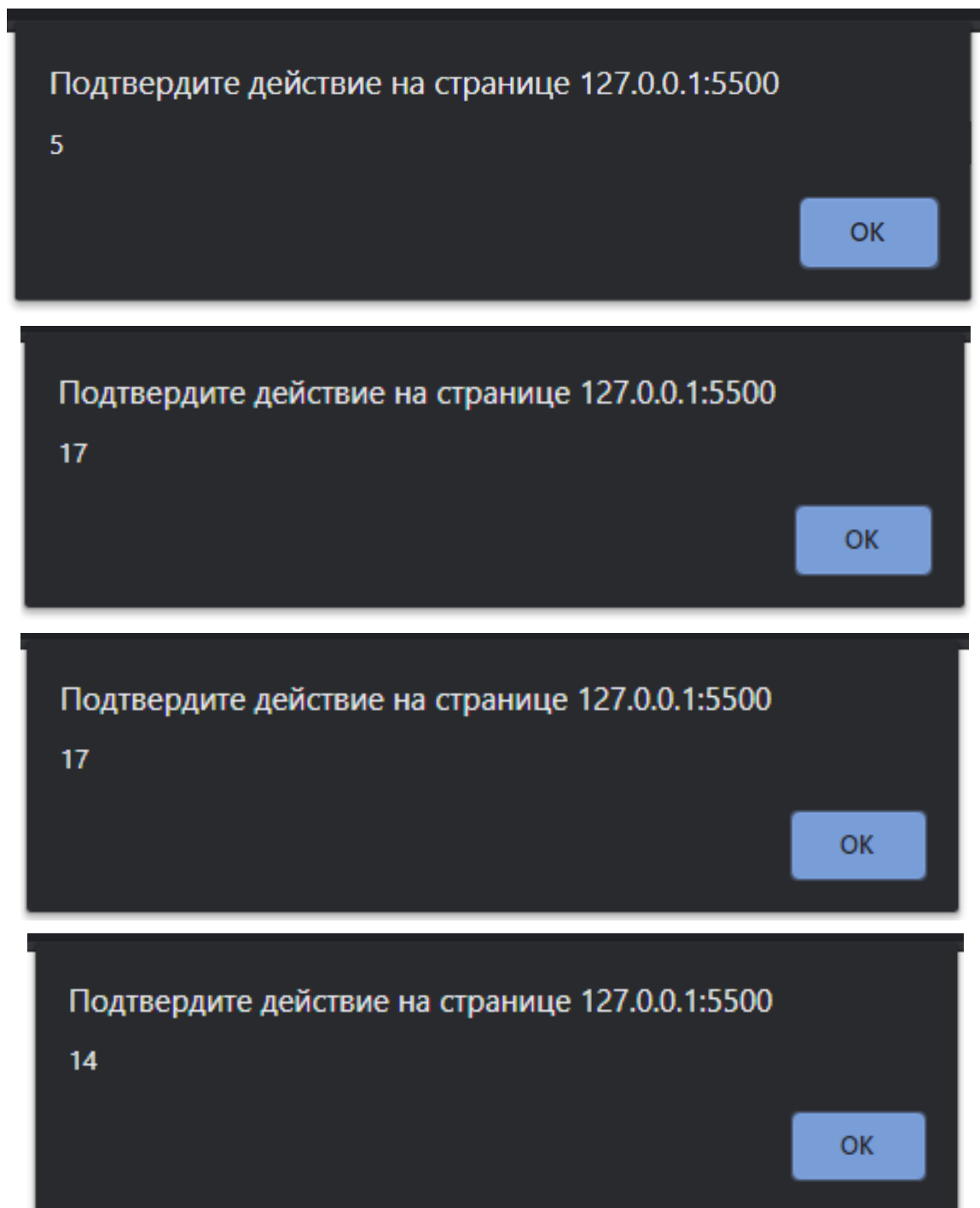


Рисунок 1.1-1.8 – Результат выполнения кода



Результат выполнения кода в консоли (Рисунок 1.9-1.12):

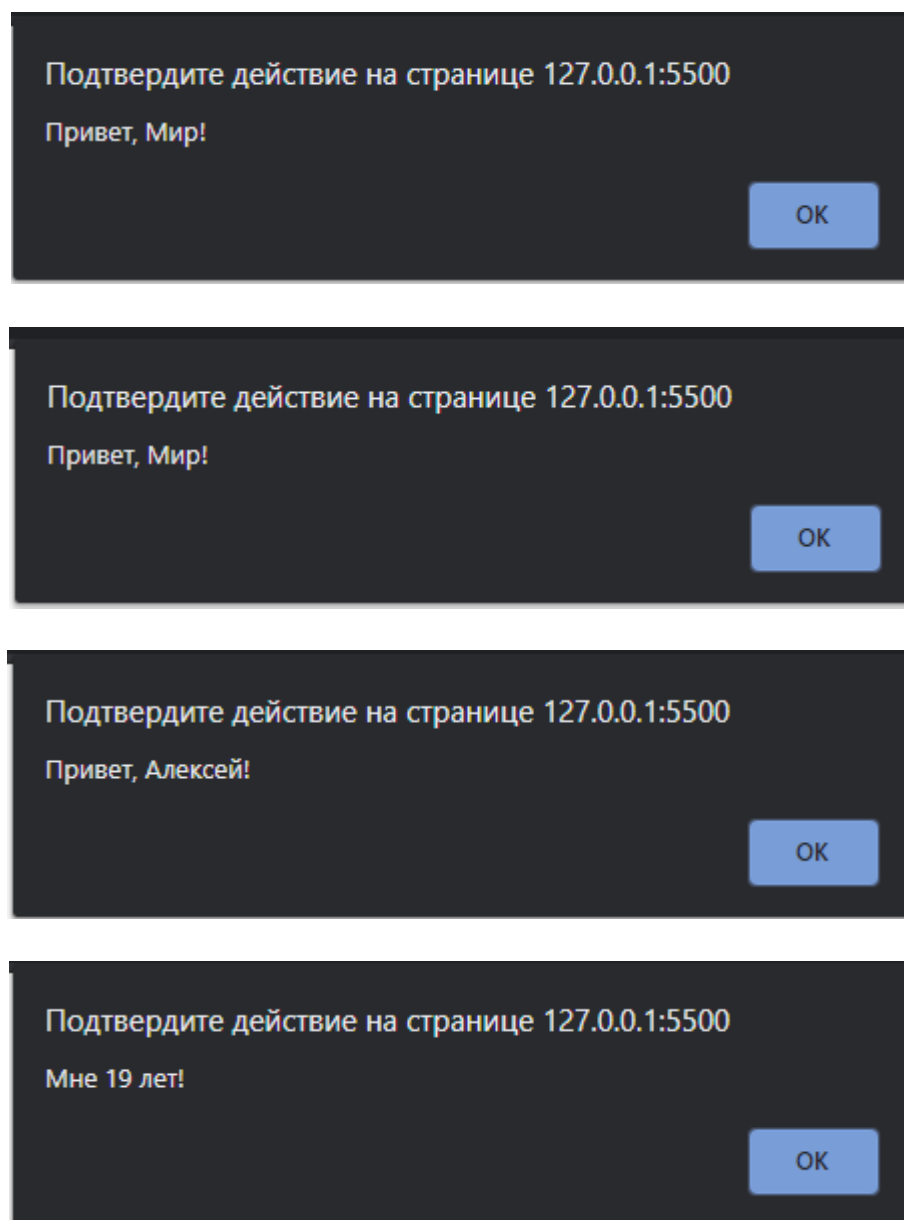


Рисунок 1.9-1.12– Результат выполнения кода

## Практическое задание №3: «Функция prompt»

### Задание:

1. Спросить имя пользователя с помощью метода `prompt`. Вывести с помощью `alert` сообщение 'Ваше имя %имя%'.
2. Спросить у пользователя число. Выведите с помощью `alert` квадрат этого числа.

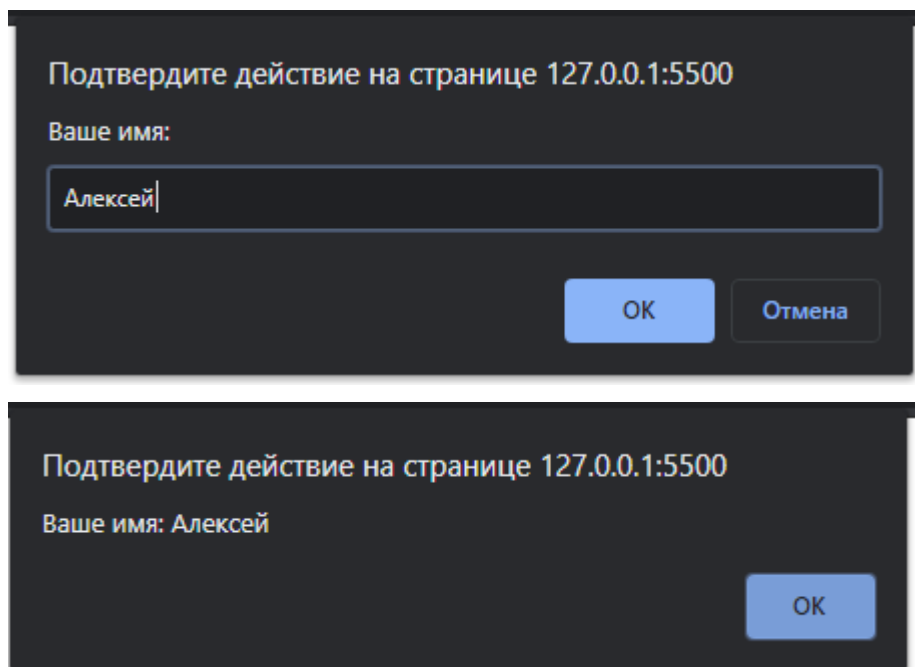
### Описание выполнения задания

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

### Листинг 1.3 – Создание `.js` файла

```
var name = prompt("Ваше имя:");  
alert("Ваше имя: "+name);  
  
var x = prompt("Введите число");  
alert(x*x);
```

Результат выполнения кода на странице (Рисунки 1.13-1.16):





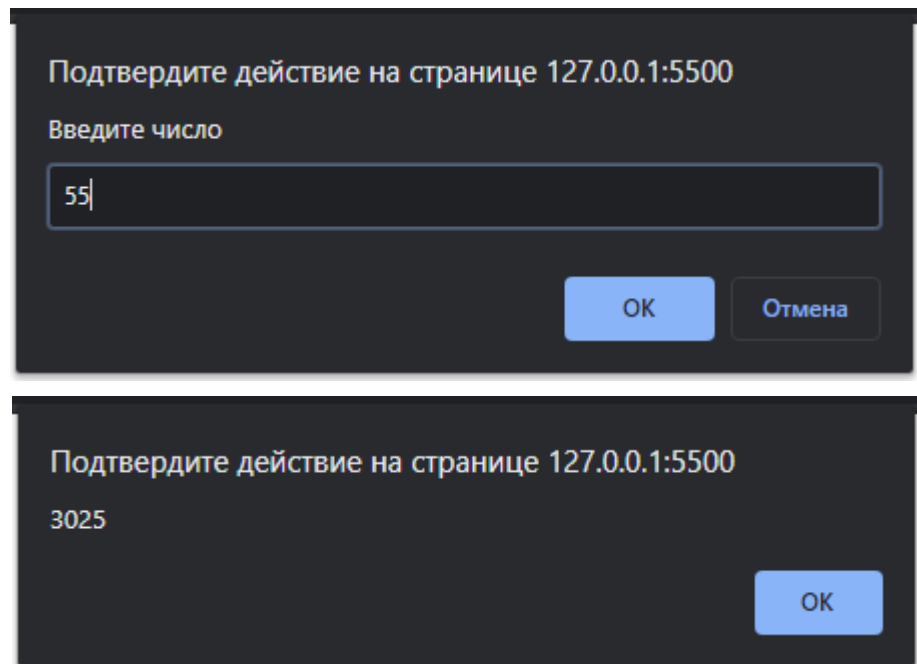


Рисунок 1.13-1.16– Результат выполнения кода

## Практическое задание №4: «Обращение к символам строки»

### Задание:

1. Создать переменную `str` и присвоить ей значение `'abcde'`. Обращаясь к отдельным символам этой строки вывести на экран символ `'a'`, символ `'c'`, символ `'e'`
2. Создать переменную `num` и присвоить ей значение `'12345'`. Найти произведение (умножение) цифр этого числа.

### Описание выполнения задания

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

#### Листинг 1.4 – Создание `.js` файла

```
var str = "abcde";  
alert(str[0]+str[2]+str[4]);  
  
var num="12345";  
alert(num[0]*num[1]*num[2]*num[3]*num[4]);
```

Результат выполнения кода в консоли (Рисунок 1.17-1.18):

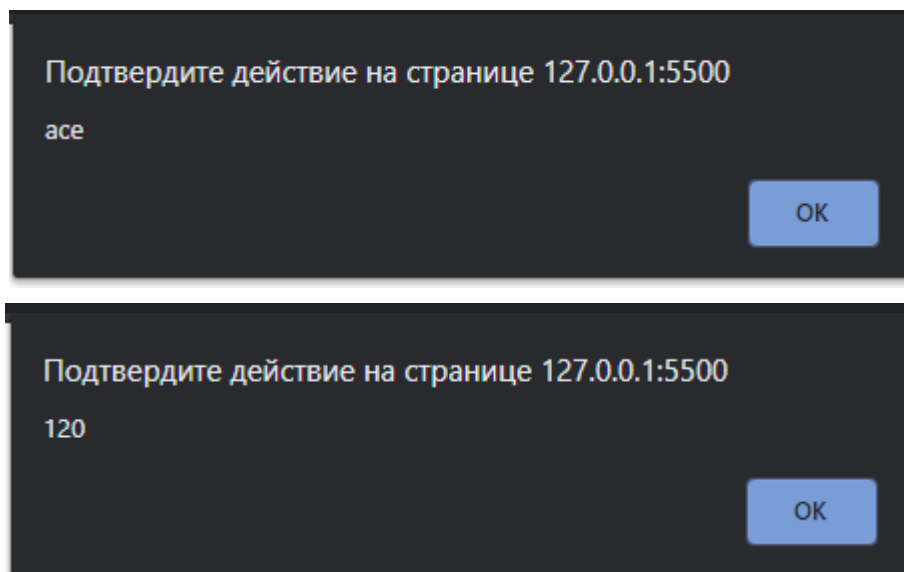


Рисунок 1.17-1.18– Результат выполнения кода

## Практическое задание №5: «Работа со временем»

### Задание:

1. Написать скрипт, который считает количество секунд в часе, сутках, в месяце.
2. Создать три переменные – час, минута, секунда. С их помощью вывести текущее время в формате ‘час:минута:секунда’.

### Описание выполнения задания

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

#### Листинг 1.5 – Создание .js файла

```
alert("Секунд в часе: "+60*60+"\nВ сутках: "+60*60*24+"\nВ месяце: "+60*60*24*31)
;
var hours=8, minutes=57, seconds=11;
alert(hours+":"+minutes+":"+seconds);
```

Результат выполнения кода в консоли (Рисунки 1.19-1.20):

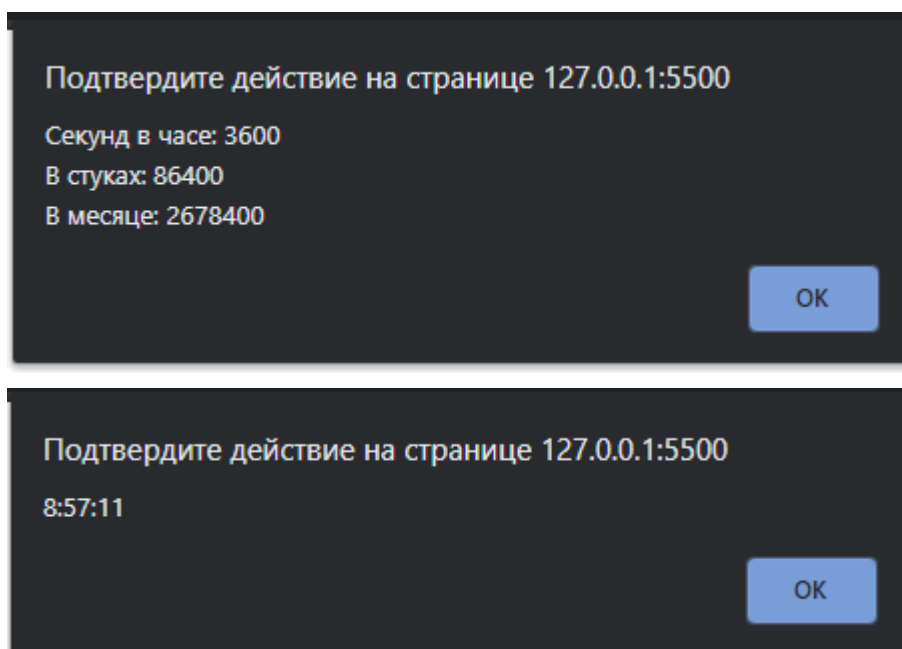


Рисунок 1.19-1.20– Результат выполнения кода

## Практическое задание №6: «Работа с присваиванием и декрементами»

### Задание:

1. Переделать ниже представленный код так, чтобы в нём использовались операции `+=`, `-=`, `*=`, `/=`. Количество строк кода при этом не должно измениться.

```
var num = 47;  
num = num + 7;  
num = num - 18;  
num = num * 10;  
num = num / 15;  
alert(num);
```

2. Переделать ниже представленный код так, чтобы в нём использовались операции `++`, `--`. Количество строк кода при этом не должно измениться.

```
var num = 10;  
num = num + 1;  
num = num + 1;  
num = num - 1;  
alert(num);
```

### Описание выполнения задания

1. Создание .js файла с функциями (Листинг 1.6).

Листинг 1.6 – Создание .js файла

```
var num = 47;  
num += 7;  
num -= 18;  
num *= 10;  
num /= 15;  
alert(num);  
var num = 10;  
num++;  
num++;  
num--;  
alert(num);
```

Результат выполнения кода в консоли (Рисунок 1.21-1.22):

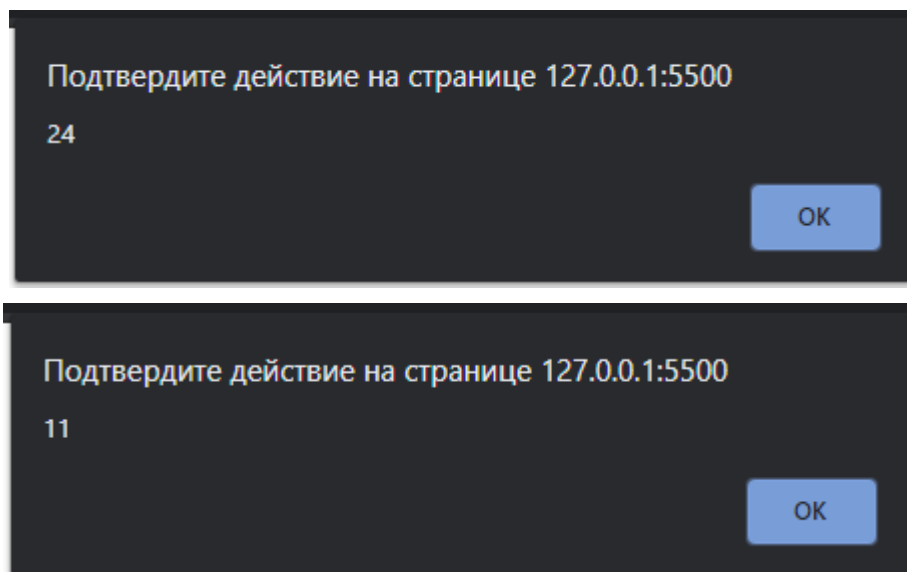


Рисунок 1.21-1.22– Результат выполнения кода

## Практическое задание №7: «Работа с массивами»

### Задание:

1. Создать массив `arr = ['a', 'b', 'c']`. Вывести его на экран с помощью функции `alert`.
2. С помощью массива `arr` из предыдущего номера вывести на экран содержимое первого, второго и третьего элементов.
3. Создать массив `arr = ['a', 'b', 'c', 'd']` и с его помощью вывести на экран строку `'a+b, c+d'`.
4. Создать массив `arr` с элементами 2, 5, 3, 9. Умножить первый элемент массива на второй, а третий элемент на четвертый. Результаты сложить, присвоить переменной `result`. Вывести на экран значение этой переменной.

### Описание выполнения задания

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

#### Листинг 1.7 – Создание `.js` файла

```
var arr = ['a', 'b', 'c'];
alert(arr);

alert(arr[0]+arr[1]+arr[2]);

var arr = ['a', 'b', 'c', 'd'];
alert(arr[0]+" "+arr[1]+", "+arr[2]+" "+arr[3]);

var arr=[2, 5, 3, 9];
var result=arr[0]*arr[1]+arr[2]*arr[3];
alert(result);
```

Результат выполнения кода в консоли (Рисунок 1.23-1.26):

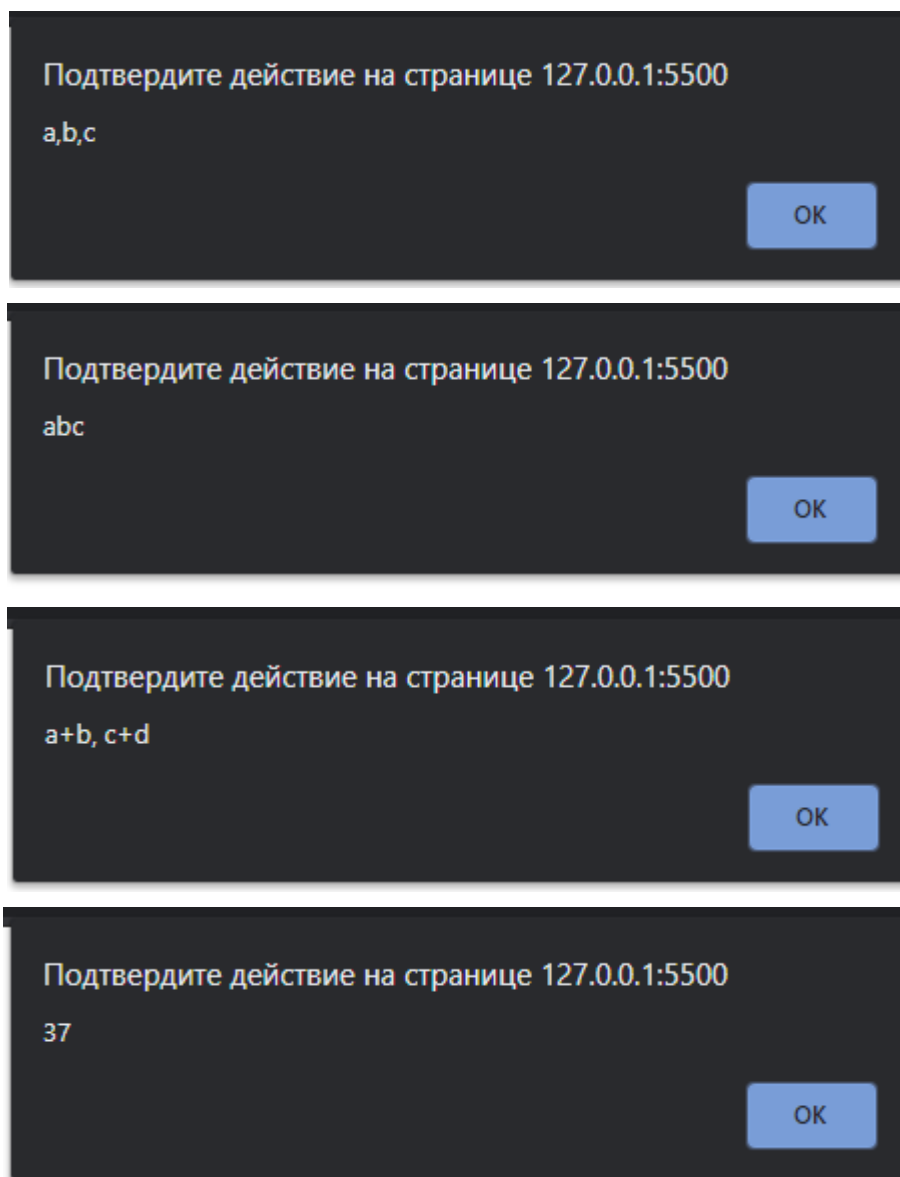


Рисунок 1.23-1.26 – Результат выполнения кода



## Практическое задание №8: «Объекты (ассоциативные массивы)»

### Задание:

1. Создать объект obj. Вывести на экран элемент с ключом 'с' двумя способами: через квадратные скобки и как свойство объекта:  
`var obj = {a: 1, b: 2, c: 3};`
2. Создать массив заработных плат obj. Вывести на экран зарплату Пети и Коли. `var obj = {Коля: '1000', Вася: '500', Петя: '200'};`
3. Создать объект с днями недели. Ключами в нем должны служить номера дней от начала недели (понедельник - первый и т.д.). Вывести на экран текущий день недели.
4. Пусть теперь номер дня недели хранится в переменной day, например, там лежит число 3. Вывести день недели, соответствующий значению переменной day.

### Описание выполнения задания

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

#### Листинг 1.8 – Создание .js файла

```
let obj = { a:1, b:2, c:3};

alert(obj.c);
alert(obj["c"]);

var obj2 = {Коля: '1000', Вася: '500', Петя: '200'};
alert(obj2.Петя);
alert(obj2.Коля);

var week= {1: "понедельник", 2: "вторник", 3: "среда", 4: "четверг", 5: "пятница",
, 6: "суббота", 7: "воскресенье"};
alert(week[5]);

var day = 5;
alert(week[day]);
```

Результат выполнения кода в консоли (Рисунок 1.27-1.32):

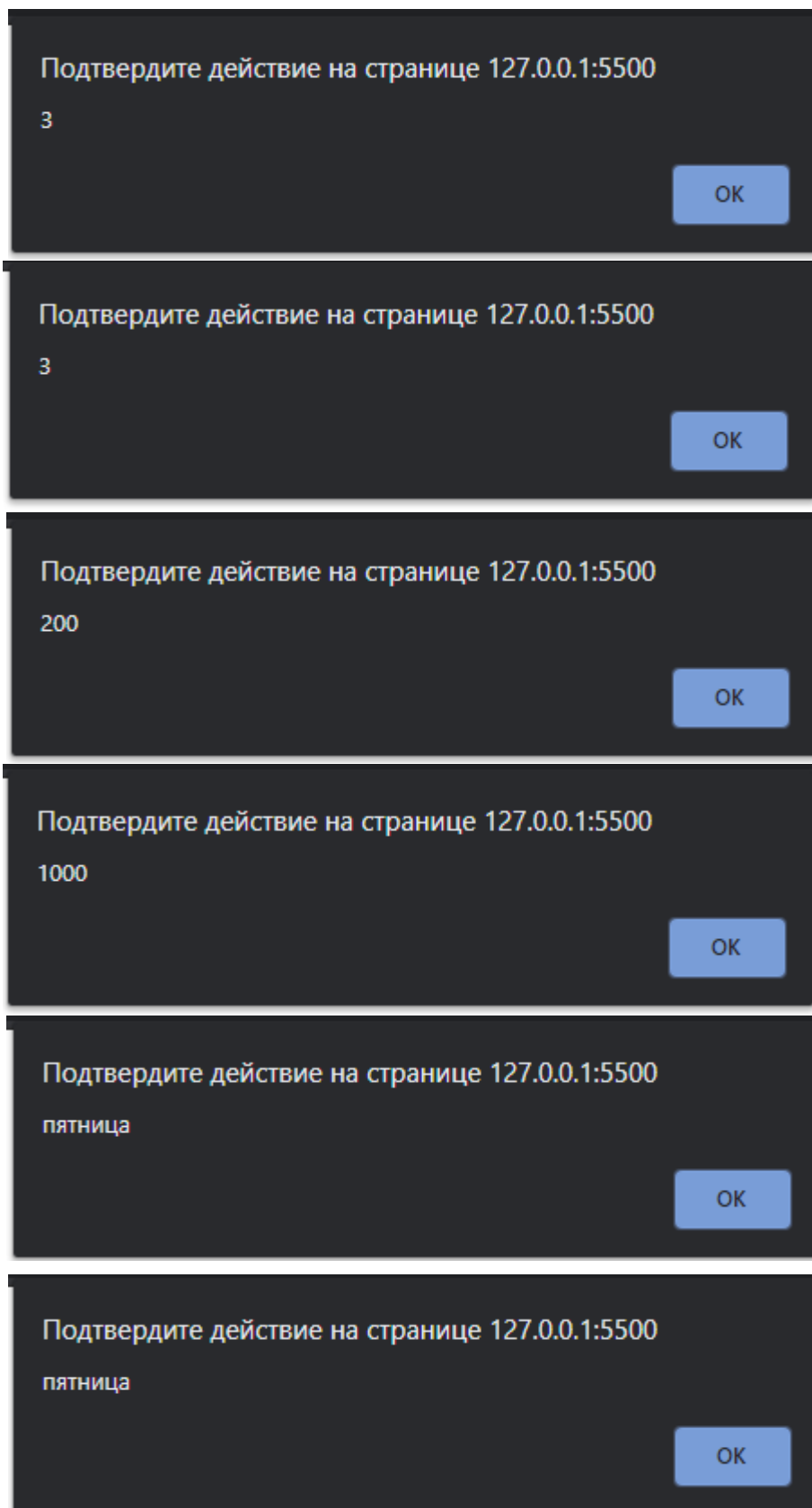


Рисунок 1.27-1.32– Результат выполнения кода

## Практическое задание №9: «Многомерные массивы»

### Задание:

1. Дан массив `[ [1, 2, 3], [4, 5, 6], [7,8,9] ]`. Вывести на экран цифру 4 из этого массива.
2. Дан объект `{js:['jQuery', 'Angular'], php: 'hello', css: 'world'}`. Вывести с его помощью слово 'jQuery'.
3. Создать двухмерный массив. Первые два ключа - это 'ru' и 'en'. Пусть первый ключ содержит элемент, являющийся массивом названий дней недели по-русски, а второй - по-английски. Вывести с помощью этого массива понедельник по-русски и среду по-английски (пусть понедельник - это нулевой день).
4. Пусть теперь в переменной `lang` хранится язык (она принимает одно из значений или 'ru', или 'en' - либо то, либо то), а в переменной `day` - номер дня. Вывести словом день недели, соответствующий переменным `lang` и `day`. То есть: если, к примеру, `lang = 'ru'` и `day = 3` - то вывести 'среда'.

### Описание выполнения задания

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

#### Листинг 1.9 – Создание .js файла

```
let matrix=[[1,2,3],[4,5,6],[7,8,9]];
alert(matrix[1][0]);

let mat={js:['jQuery', 'Angular'], php: 'hello', css: 'world'};
alert(mat["js"][0]);

let week={ru:["понедельник", "вторник", "среда", "четверг", "пятница", "суббота",
"воскресенье"], en:["monday", "tuesday", "wednesday", "thursday", "friday", "sat
urday", "sunday"]};
alert(week["ru"][0]+" "+week["en"][2]);

let lang="ru", day=3;
alert(week[lang][day-1]);
```

Результат выполнения кода в консоли (Рисунок 1.33-1.36):

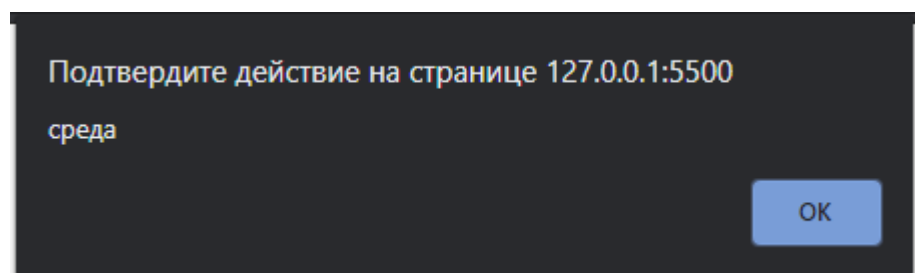
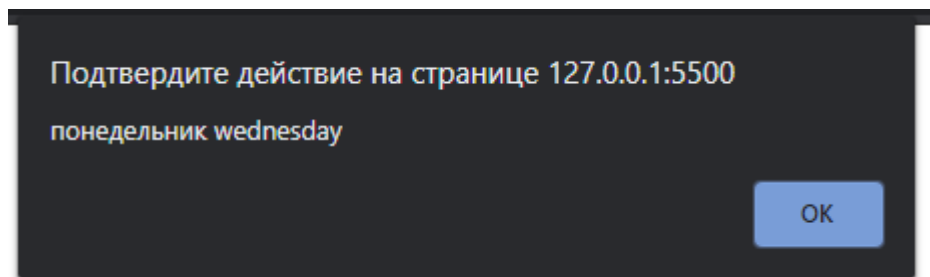
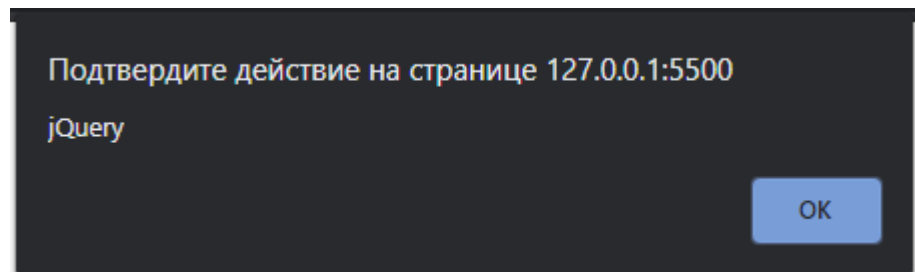
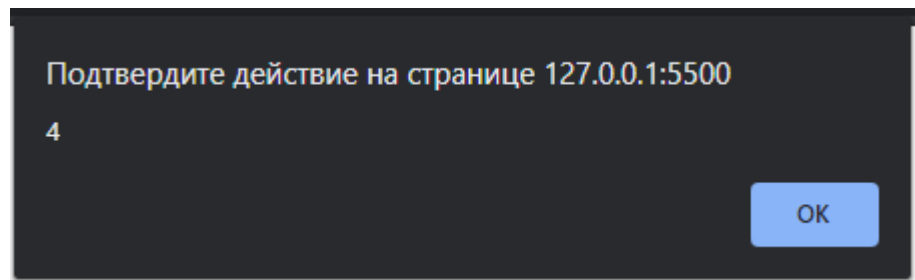


Рисунок 1.33-1.36– Результат выполнения кода

## Практическое задание №10: «Работа с if-else»

### Задание:

1. Если переменная *a* равна нулю, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 1, 0, -3.
2. Если переменная *a* больше нуля, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 1, 0, -3.
3. Если переменная *a* меньше нуля, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 1, 0, -3.
4. Если переменная *a* больше или равна нулю, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 1, 0, -3.
5. Если переменная *a* меньше или равна нулю, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 1, 0, -3.
6. Если переменная *a* не равна нулю, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 1, 0, -3.
7. Если переменная *a* равна 'test', то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном 'test', 'тест', 3.
8. Если переменная *a* равна '1' и по значению и по типу, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при *a*, равном '1', 1, 3.

### Описание выполнения задания

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

#### Листинг 1.10 – Создание .js файла

```
var a=-1;
if (a==0)
    alert("Верно");
else alert("Неверно");

if (a>0)
    alert("Верно");
else alert("Неверно");

if (a<0)
    alert("Верно");
else alert("Неверно");
```

```
if (a>=0)
    alert("Верно");
else alert("Неверно");

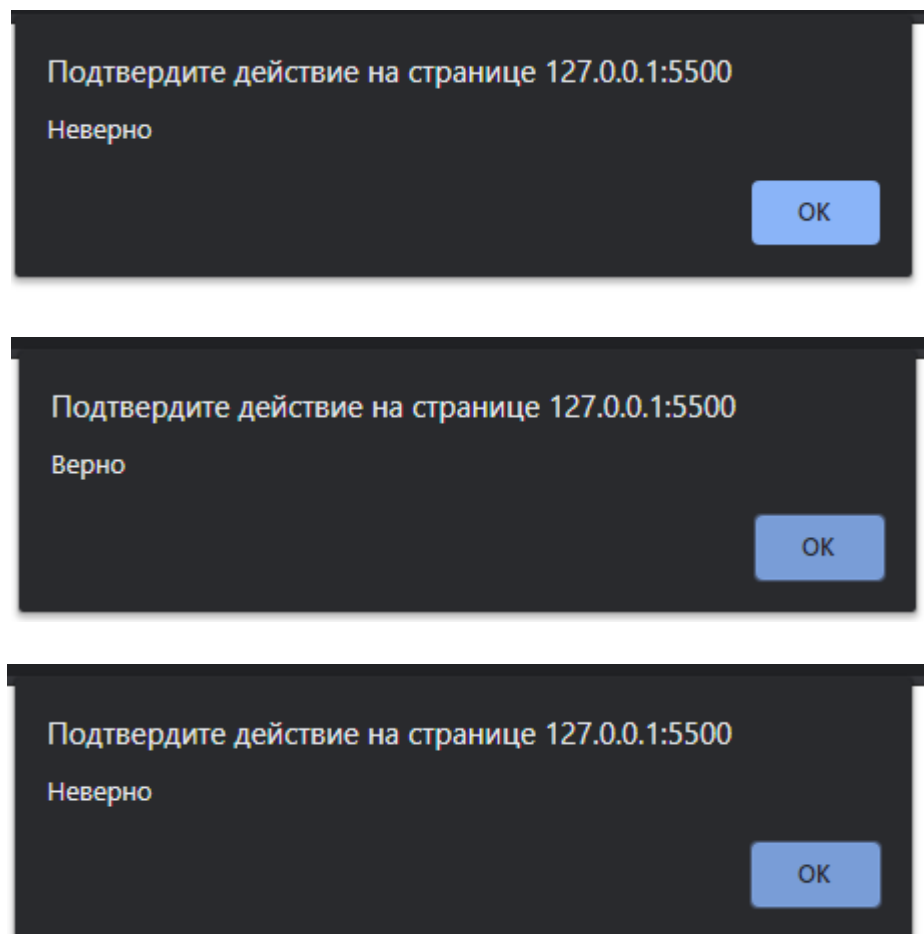
if (a<=0)
    alert("Верно");
else alert("Неверно");

if (a!=0)
    alert("Верно");
else alert("Неверно");

if (a=="test")
    alert("Верно");
else alert("Неверно");

if (a=="1" && typeof(a)=="string")
    alert("Верно");
else alert("Неверно");
```

Результат выполнения кода в консоли (Рисунок 1.37-1.60):



Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK



Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Верно

OK

Подтвердите действие на странице 127.0.0.1:5500

Неверно

OK

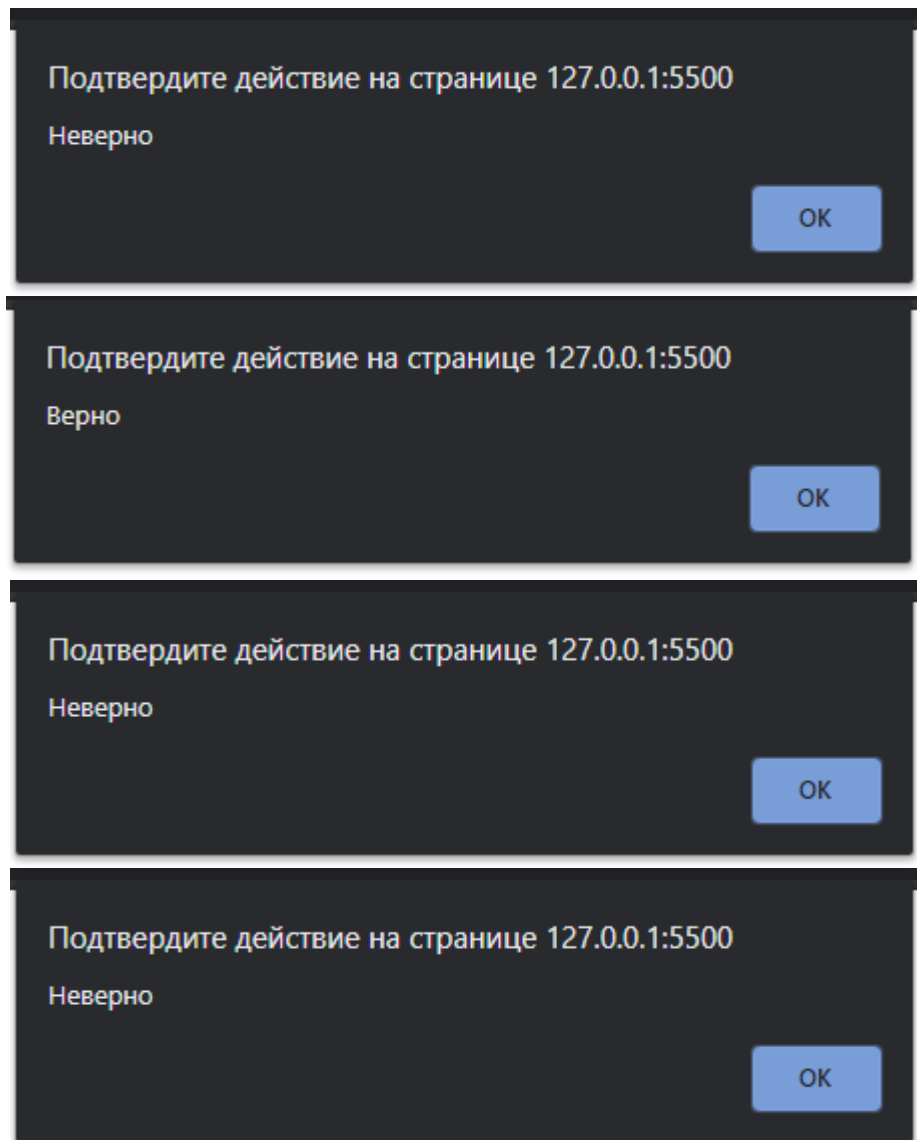


Рисунок 1.37-1.60– Результат выполнения кода

## Практическое задание №11: «Работа с логическими переменными»

### Задание:

1. Если переменная `test` равна `true`, то вывести 'Верно', иначе вывести 'Неверно'. Проверьте работу скрипта при `test`, равном `true`, `false`. Написать два варианта скрипта - с короткой записью и с длинной.
2. Если переменная `test` не равна `true`, то вывести 'Верно', иначе вывести 'Неверно'. Проверить работу скрипта при `test`, равном `true`, `false`. Написать два варианта скрипта - с короткой записью и с длинной.

### Описание выполнения задания

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

Листинг 1.11 – Создание `.js` файла

```
var test=true;
if (test==true)
    alert("Верно");
else alert("Неверно");
if (test)
    alert("Верно");
else alert("Неверно");

if (test==false)
    alert("Верно");
else alert("Неверно");
if (!test)
    alert("Верно");
else alert("Неверно");
```

Результат выполнения кода в консоли (Рисунок 1.62-1.65):

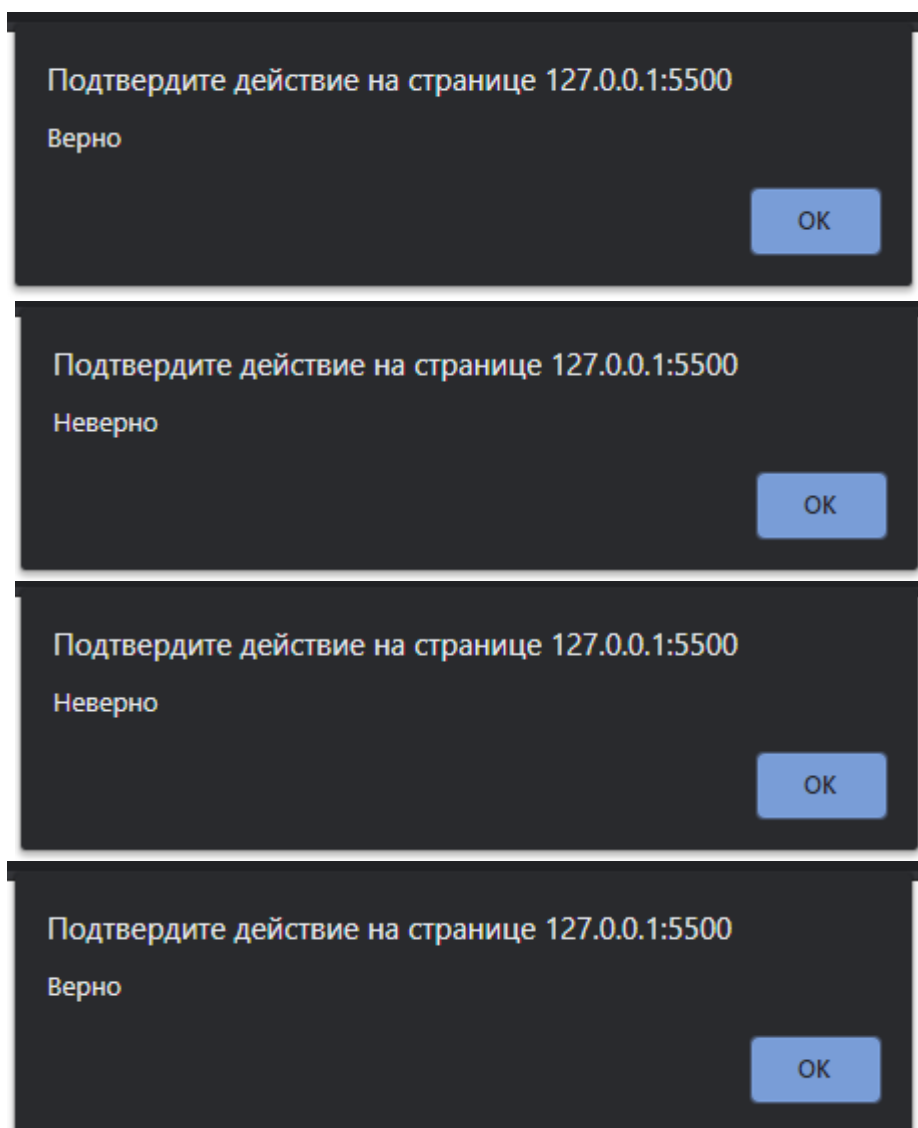


Рисунок 1.62-1.65– Результат выполнения кода

## Практическое задание №12: «Работа с && (и) и || (или)»

### Задание:

1. Если переменная *a* больше нуля и меньше 5-ти, то вывести строку 'Верно', иначе вывести на экран 'Неверно'. Проверить работу скрипта при *a*, равном 5, 0, -3, 2.
2. Если переменная *a* равна нулю или равна двум, то прибавить к ней 7, иначе поделить ее на 10. Вывести новое значение переменной на экран. Проверить работу скрипта при *a*, равном 5, 0, -3, 2.
3. Если переменная *a* равна или меньше 1, а переменная *b* больше или равна 3, то вывести сумму этих переменных, иначе вывести их разность (результат вычитания). Проверить работу скрипта при *a* и *b*, равном 1 и 3, 0 и 6, 3 и 5.
4. Если переменная *a* больше 2-х и меньше 11-ти, или переменная *b* больше или равна 6-ти и меньше 14-ти, то вывести 'Верно', в противном случае вывести 'Неверно'.

### Описание выполнения задания

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

#### Листинг 1.12 – Создание .js файла

```
var a=0;
if (a>0 && a<5)
    alert("Верно");
else alert("Неверно");

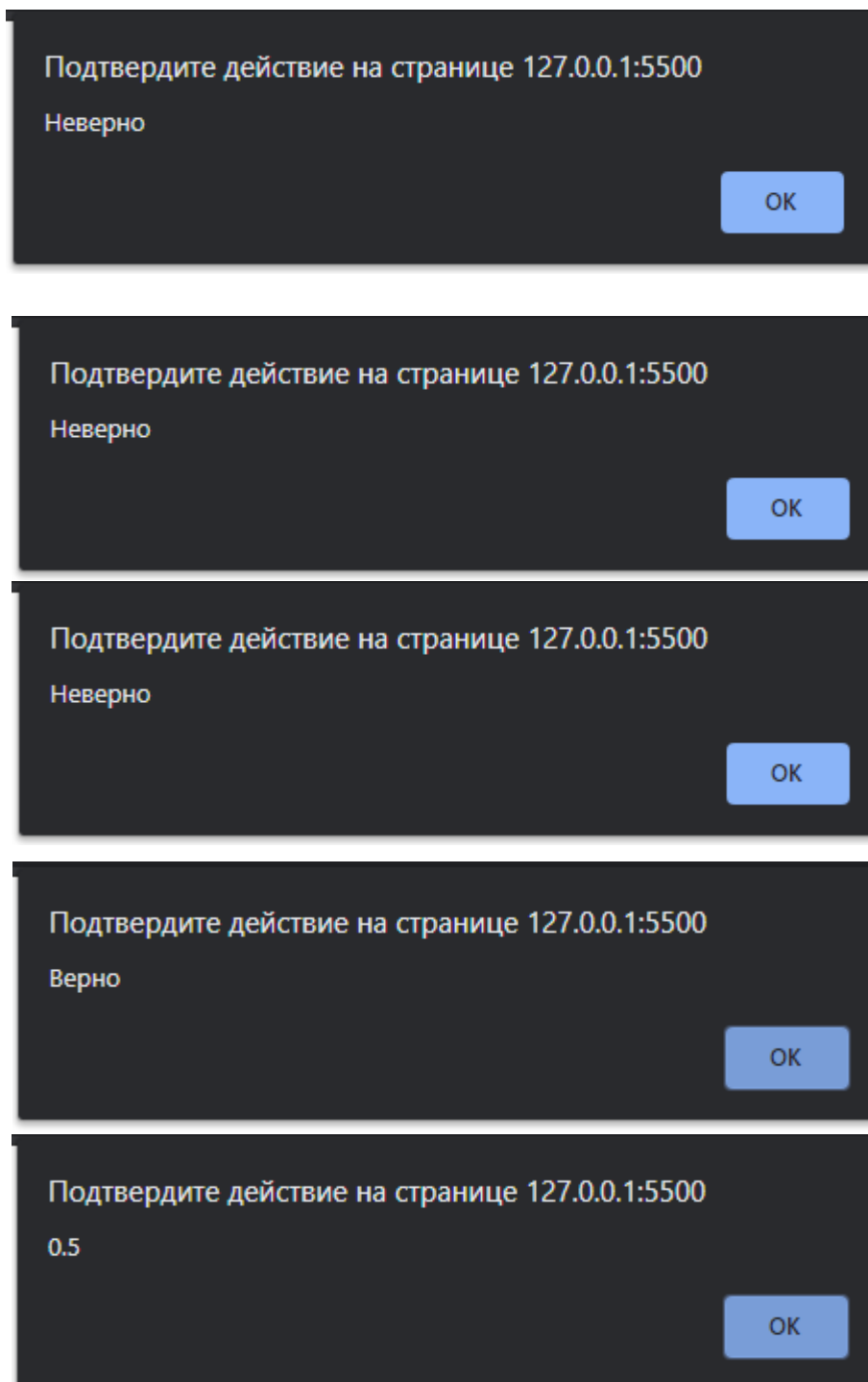
if (a==0 || a==2){
    a+=7;
    alert(a);
}
else{
    a/=10;
    alert(a);
}

var b=0;
if (a<=1 && b>=3){
    alert(a+b);
}
```

```
else{
    alert(a-b);
}

if ((a>2 && a<11) || (b>=6 && b<14))
    alert("Верно");
else alert("Неверно");
```

Результат выполнения кода в консоли (Рисунок 1.66-1.77):



Подтвердите действие на странице 127.0.0.1:5500

7

OK

Подтвердите действие на странице 127.0.0.1:5500

-0.3

OK

Подтвердите действие на странице 127.0.0.1:5500

9

OK

Подтвердите действие на странице 127.0.0.1:5500

4

OK

Подтвердите действие на странице 127.0.0.1:5500

6

OK

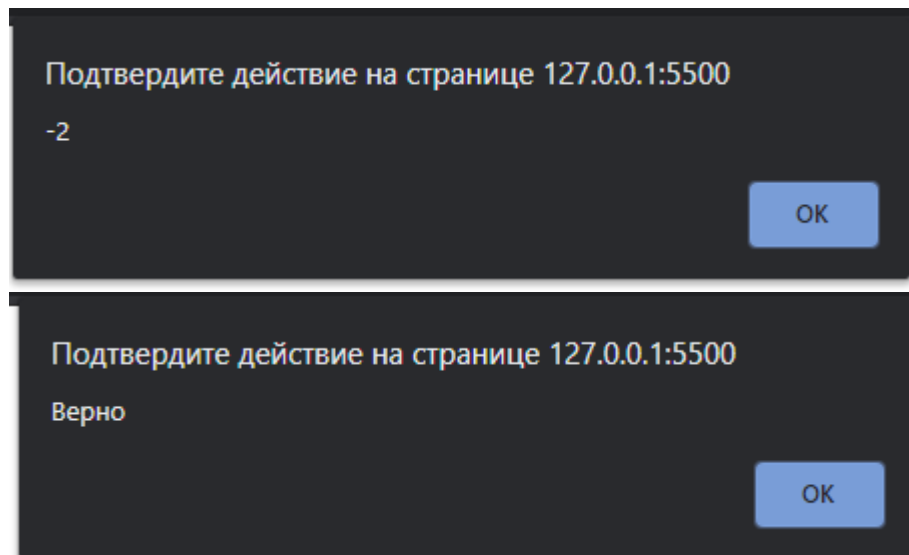


Рисунок 1.66-1.77– Результат выполнения кода



## Практическое задание №13: «Работа со switch-case»

### Задание:

1. Переменная num может принимать 4 значения: 1, 2, 3 или 4. Если она имеет значение '1', то в переменную result записать 'зима', если имеет значение '2' – 'весна' и так далее. Решить задачу через switch-case.

### Описание выполнения задания

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

Листинг 1.13 – Создание .js файла

```
var a=1;
switch(a){
  case 1:{
    alert("Зима");
    break;
  }
  case 2:{
    alert("Весна");
    break;
  }
  case 3:{
    alert("Лето");
    break;
  }
  case 4:{
    alert("Осень");
    break;
  }
  default:{
    alert("");
    break;
  }
}
```

Результат выполнения кода в консоли (Рисунок 1.78-1.81):

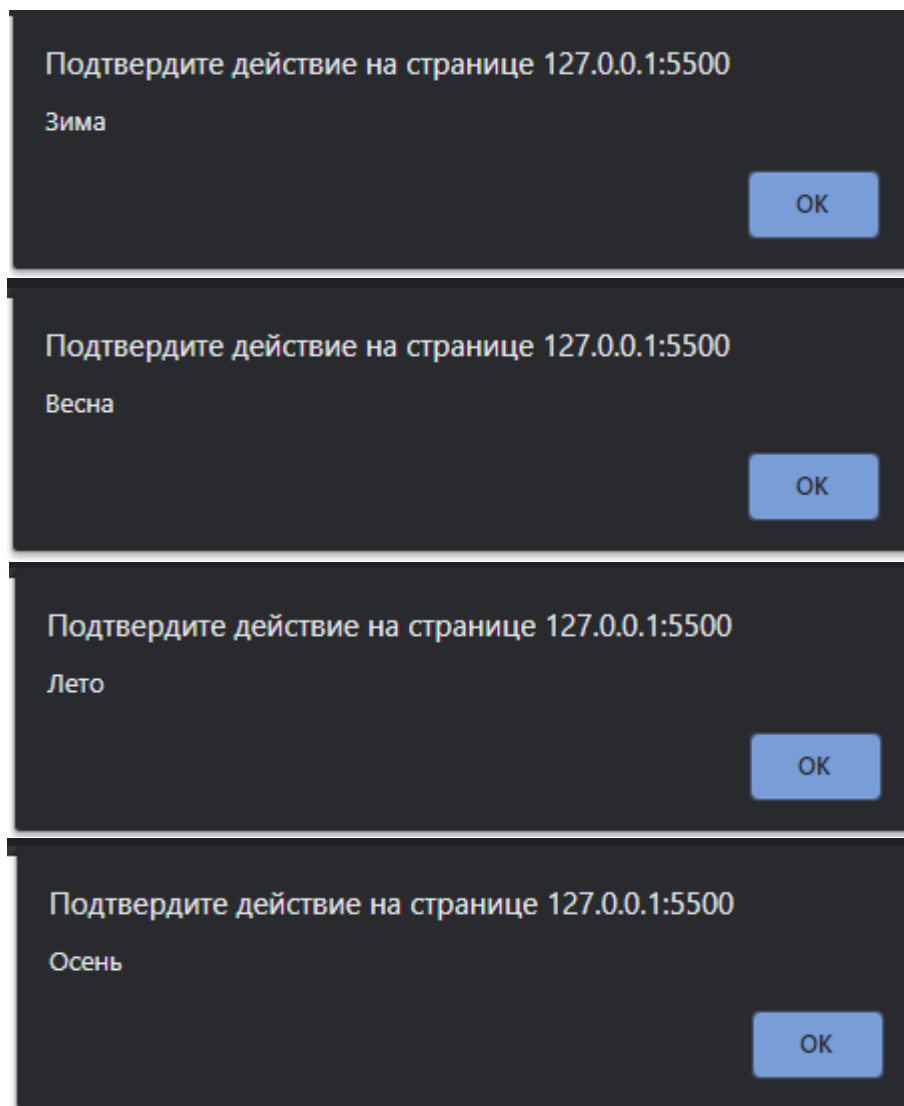


Рисунок 1.78-1.81– Результат выполнения кода

## Практическое задание №14: «Циклы while и for»

### Задание:

1. Вывести столбец чисел от 1 до 100.
2. Вывести столбец чисел от 11 до 33.
3. Вывести столбец четных чисел в промежутке от 0 до 100.
4. С помощью цикла найти сумму чисел от 1 до 100.

### Описание выполнения задания

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

Листинг 1.14 – Создание .js файла

```
var aW=1, strW="";
while (aW<101)
{
    strW+=aW+"\n";
    aW++;
}
alert(strW);
var strF="";
for (var i=1; i<101; i++){
    strF+=i+"\n";
}
alert(strF);

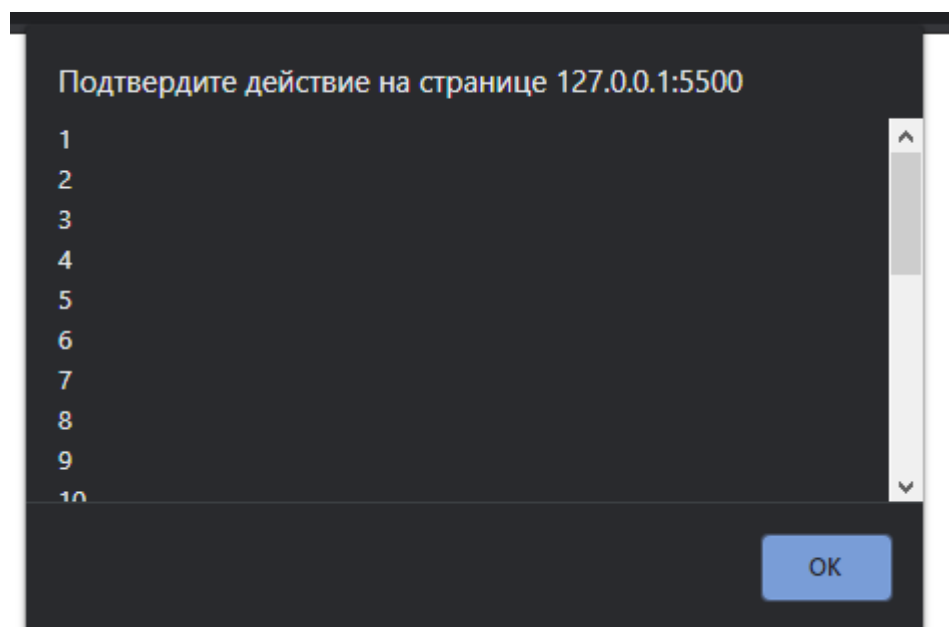
var aW=11, strW="";
while (aW<34)
{
    strW+=aW+"\n";
    aW++;
}
alert(strW);
var strF="";
for (var i=11; i<34; i++){
    strF+=i+"\n";
}
alert(strF);

var aW=0, strW="";
while (aW<101)
{
    strW+=aW+"\n";
    aW+=2;
}
alert(strW);
```

```
var strF="";
for (var i=0; i<101; i+=2){
    strF+=i+"\n";
}
alert(strF);

var aW=1, result=0;
while (aW<101)
{
    result+=aW;
    aW++;
}
alert(result);
result=0;
for (var i=1; i<101; i++){
    result+=i;
}
alert(result);
```

Результат выполнения кода в консоли (Рисунок 1.82-1.89):



Подтвердите действие на странице 127.0.0.1:5500

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

OK

Подтвердите действие на странице 127.0.0.1:5500

- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

OK

Подтвердите действие на странице 127.0.0.1:5500

- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

OK

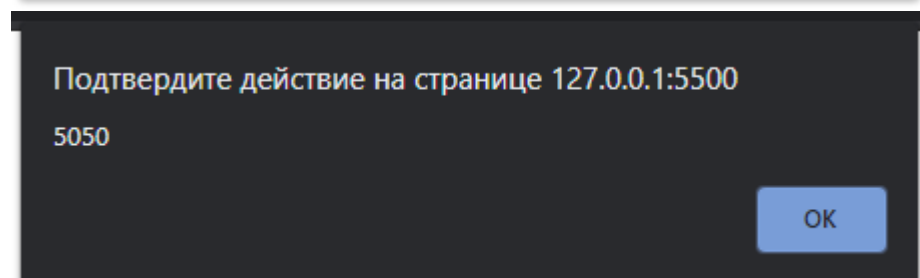
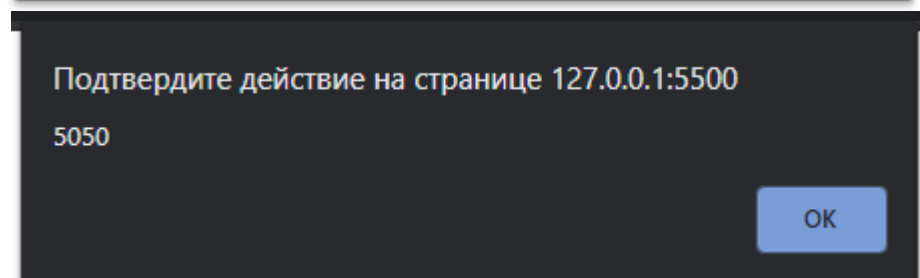
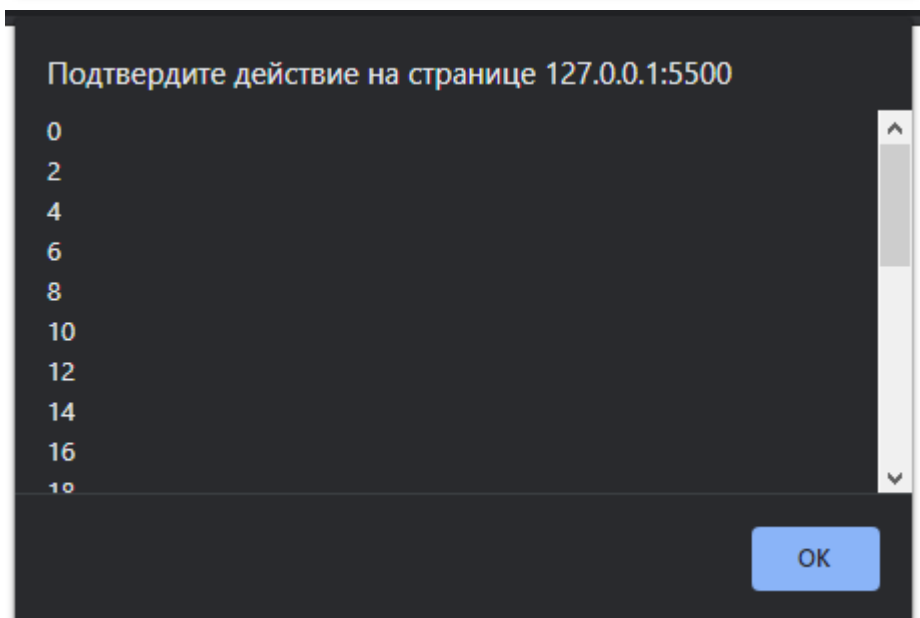
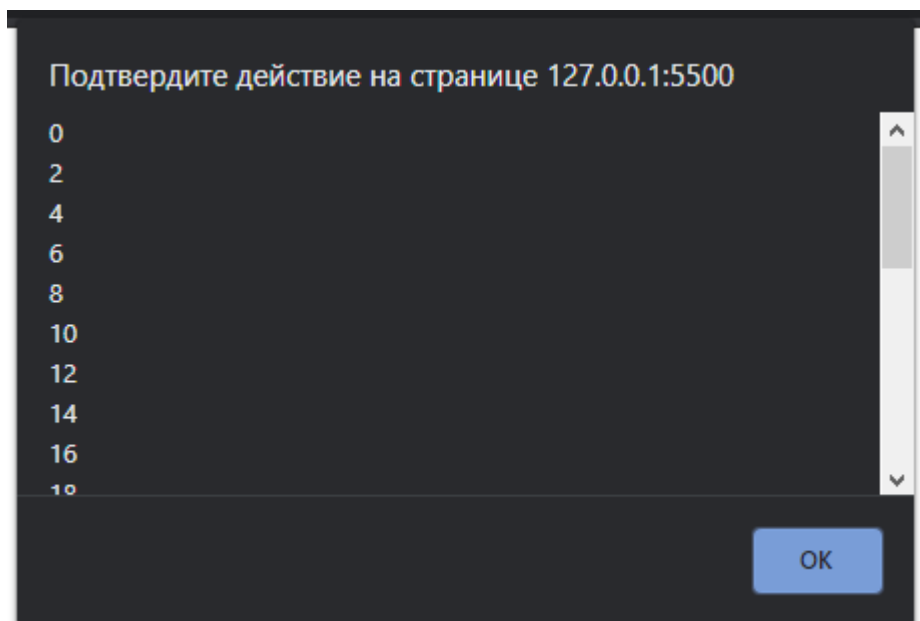


Рисунок 1.82-1.89– Результат выполнения кода

## Практическое задание №15: «Работа с for для массивов»

### Задание:

1. Дан массив с элементами [1, 2, 3, 4, 5]. С помощью цикла for вывести все эти элементы на экран.
2. Дан массив с элементами [1, 2, 3, 4, 5]. С помощью цикла for найти сумму элементов этого массива. Записать ее в переменную result.

### Описание выполнения задания

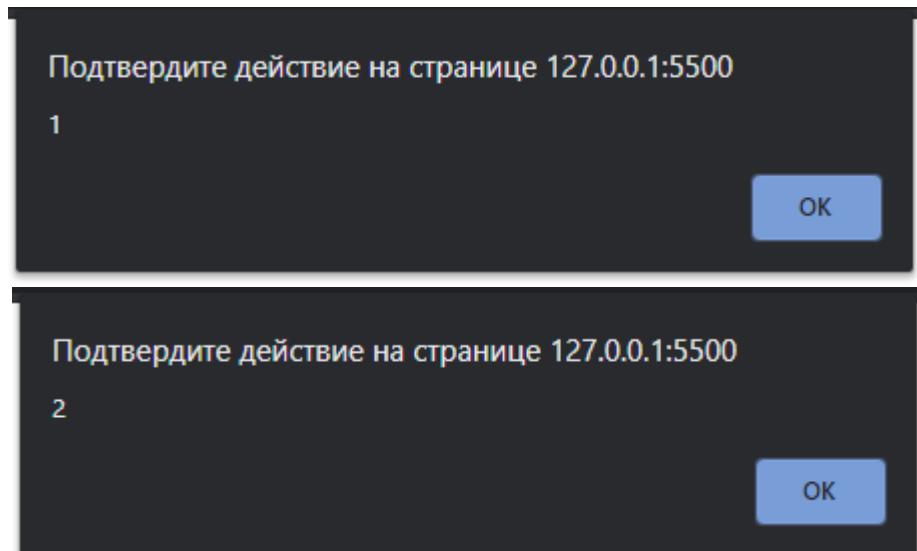
1. Создание .js файла с необходимой функциональностью (Листинг 1.15).

#### Листинг 1.15 – Создание .js файла

```
var arr=[1, 2, 3, 4, 5];
for (var i=0; i<arr.length; i++){
    alert(arr[i]);
}

var result=0;
for (var i=0; i<arr.length; i++){
    result+=arr[i];
}
```

Результат выполнения кода в консоли (Рисунок 1.90-1.95):





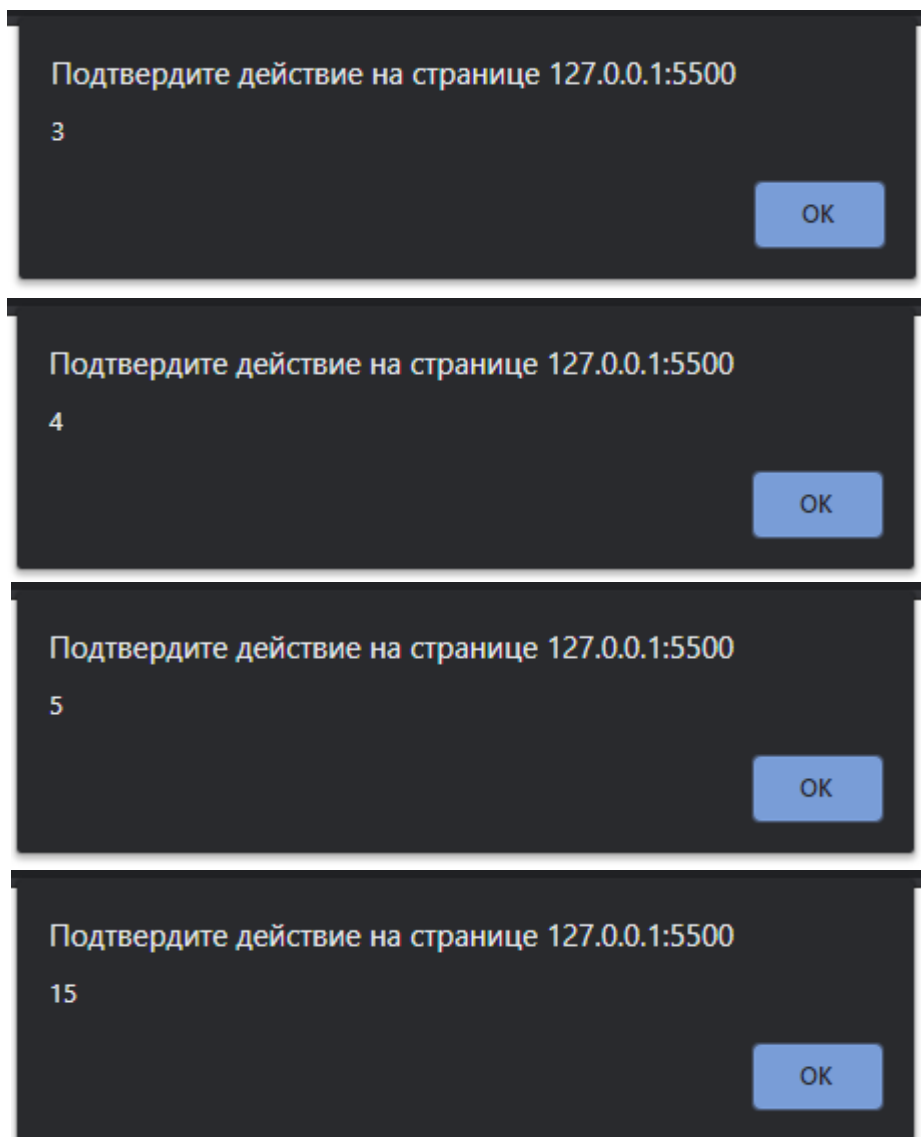


Рисунок 1.90-1.95 – Результат выполнения кода

## Практическое задание №16: «Работа с for-in»

### Задание:

1. Дан объект obj. С помощью цикла for-in вывести на экран ключи и элементы этого объекта.

var obj = {green: 'зеленый', red: 'красный', blue: 'голубой'};

2. Дан объект obj с ключами Коля, Вася, Петя с элементами '200', '300', '400'.  
С помощью цикла for-in вывести на экран строки такого формата: 'Коля - зарплата 200 долларов'.

### Описание выполнения задания

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

Листинг 1.16 – Создание .js файла

```
var obj = {green: 'зеленый', red: 'красный', blue: 'голубой'};

for (let key in obj){
    alert(key+" "+obj[key]+'\\n');
}

var obj = {Коля: 200, Вася: 300, Петя: 400};

for (let key in obj){
    alert(key+" - зарплата "+obj[key]+" долларов\\n");
}
```

Результат выполнения кода в консоли (Рисунок 1.96-1.101):

Подтвердите действие на странице 127.0.0.1:5500  
green зеленый

OK

Подтвердите действие на странице 127.0.0.1:5500  
red красный

OK

Подтвердите действие на странице 127.0.0.1:5500  
blue голубой

OK

Подтвердите действие на странице 127.0.0.1:5500  
Коля - зарплата 200 долларов

OK

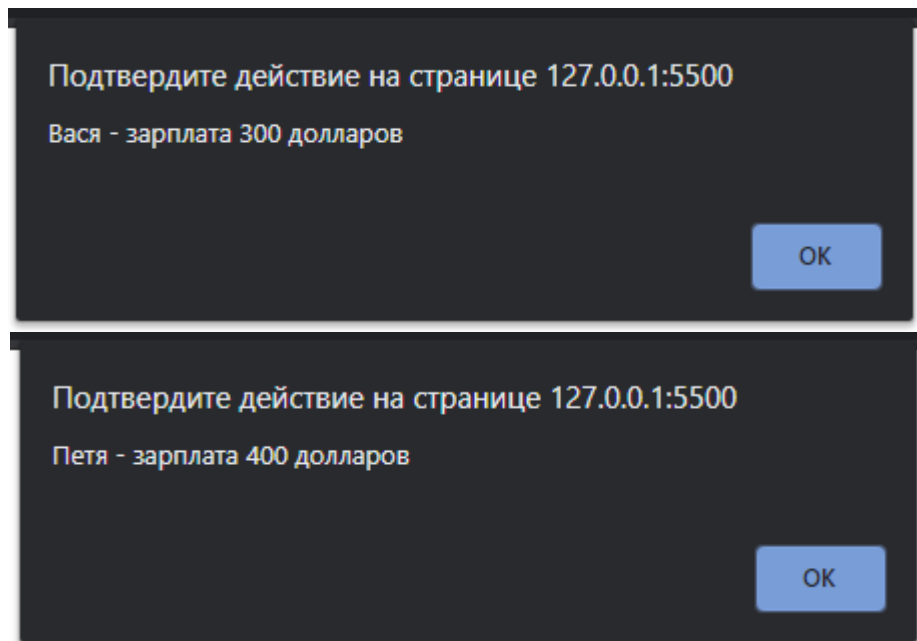


Рисунок 1.96-1.101– Результат выполнения кода

## Практическое задание №17: «Задачи с циклами»

### Задание:

1. Дан массив с элементами 2, 5, 9, 15, 0, 4. С помощью цикла for и оператора if вывести на экран столбец тех элементов массива, которые больше 3-х, но меньше 10.
2. Дан массив с числами. Числа могут быть положительными и отрицательными. Найти сумму положительных элементов массива.
3. Дан массив с элементами 1, 2, 5, 9, 4, 13, 4, 10. С помощью цикла for и оператора if проверить есть ли в массиве элемент со значением, равным числу 4. Если есть - вывести на экран 'Есть!' и выйти из цикла. Если нет - ничего делать не надо.
4. Дан массив числами, например: [10, 20, 30, 50, 235, 3000]. Вывести на экран только те числа из массива, которые начинаются на цифры 1, 2, 5.
5. Дан массив с элементами 1, 2, 3, 4, 5, 6, 7, 8, 9. С помощью цикла for создать строку '-1-2-3-4-5-6-7-8-9-'.
6. Составить массив дней недели. С помощью цикла for вывести все дни недели, а выходные дни вывести жирным.
7. Составить массив дней недели. С помощью цикла for вывести все дни недели, а текущий день вывести курсивом. Текущий день должен храниться в переменной day.
8. Дано число n=1000. Делите его на 2 столько раз, пока результат деления не станет меньше 50. Какое число получится? Посчитать количество итераций, необходимых для этого (итерация - это проход цикла), и записать его в переменную num.

### Описание выполнения задания

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

Листинг 1.17 – Создание .js файла

```
var m=[2, 5, 9, 15, 0, 4];  
for (var i=0; i<m.length; i++)  
    if (m[i]>3 && m[i]<10)
```

```

        alert(m[i]);

var res=0;
for (var i=0; i<m.length; i++)
    if (m[i]>0)
        res+=m[i];
alert(res);

var m2=[1, 2, 5, 9, 4, 13, 4, 10];
for (var i=0; i<m2.length; i++)
    if (m2[i]==4)
    {
        alert("Есть!");
        break;
    }

var m3=[10, 20, 30, 50, 235, 3000];
for (var i=0; i<m3.length; i++)
    if (((m3[i].toString())[0]==1) || ((m3[i].toString())[0]==2) || ((m3[i].toString())[0]==5))
        alert(m3[i]);

var str="-";
var m4=[1, 2, 3, 4, 5, 6, 7, 8, 9];
for (var i=0; i<m4.length; i++)
    str+=m4[i]+"-";
alert(str);

var m5=["понедельник", "вторник", "среда", "четверг", "пятница", "суббота", "воскресенье"];
for (var i=0; i<m5.length; i++)
    if (i==5 || i==6)
        alert(m5[i].bold());
    else alert(m5[i]);

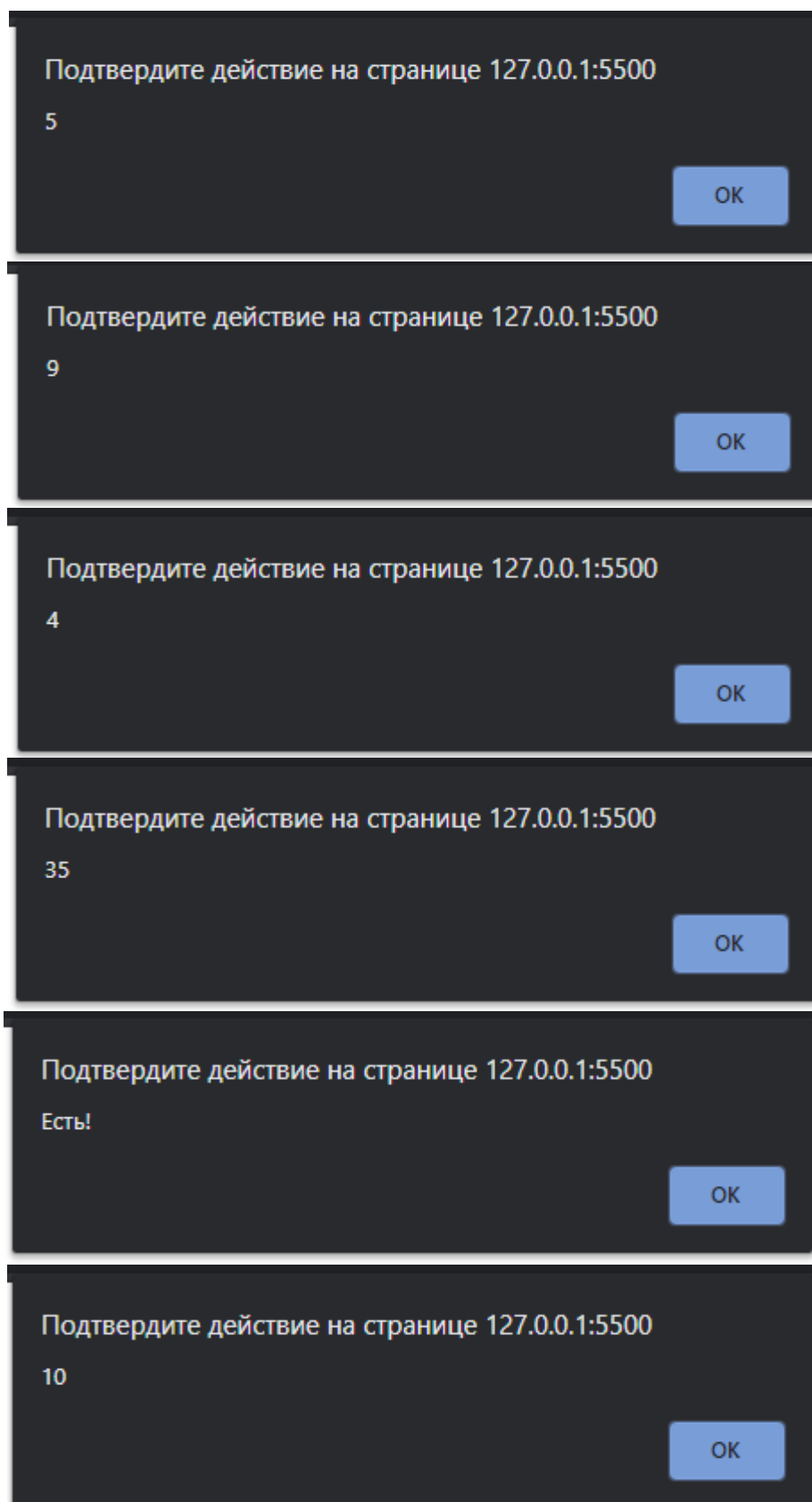
var m6=["понедельник", "вторник", "среда", "четверг", "пятница", "суббота", "воскресенье"];
for (var i=0; i<m6.length; i++)
    if (i==5)
    {
        var day=m6[i];
        alert(m6[i].italics());
    }
    else alert(m6[i]);

var n=1000, num=0;
while(n>50)
{
    n/=2;
    num++;
}

```

```
}  
alert(n+" "+num);
```

Результат выполнения кода в консоли (Рисунок 1.102-1.126):





Подтвердите действие на странице 127.0.0.1:5500

20

OK

Подтвердите действие на странице 127.0.0.1:5500

50

OK

Подтвердите действие на странице 127.0.0.1:5500

235

OK

Подтвердите действие на странице 127.0.0.1:5500

вторник

OK

Подтвердите действие на странице 127.0.0.1:5500

среда

OK

Подтвердите действие на странице 127.0.0.1:5500

четверг

OK

Подтвердите действие на странице 127.0.0.1:5500

пятница

OK

Подтвердите действие на странице 127.0.0.1:5500

**суббота**

OK

Подтвердите действие на странице 127.0.0.1:5500

**воскресенье**

OK

Подтвердите действие на странице 127.0.0.1:5500

понедельник

OK

Подтвердите действие на странице 127.0.0.1:5500

вторник

OK

Подтвердите действие на странице 127.0.0.1:5500

среда

OK

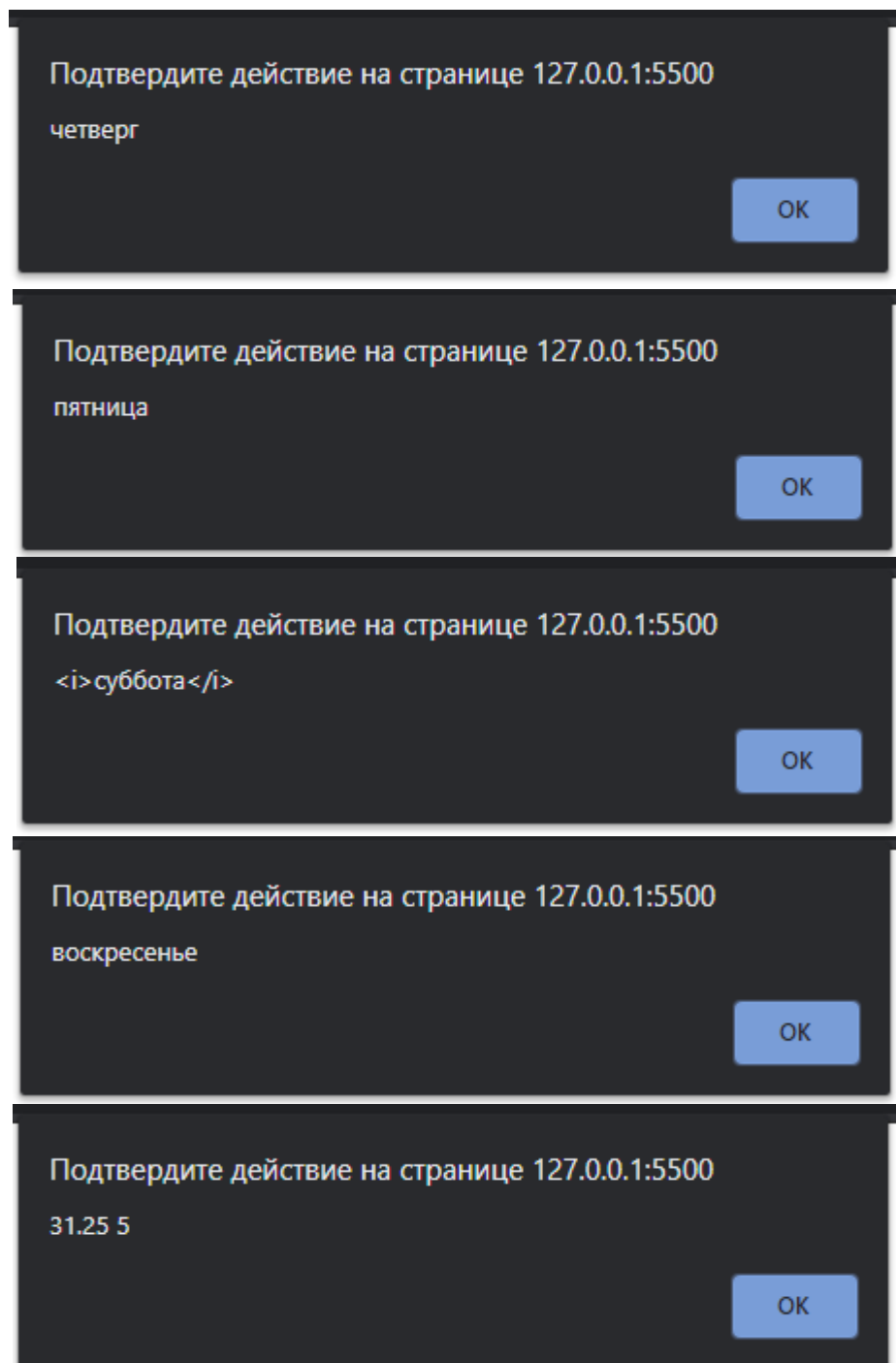


Рисунок 1.102-1.126 – Результат выполнения кода

**Вывод:** в ходе выполнения практической работы были изучены основные принципы работы с JavaScript, рассмотрена работа с переменными, строками, функция `prompt`, обращение к символам строки, работа со временем, работа с присваиванием и декрементами, работа с массивами, объекты (ассоциативные массивы), многомерные массивы, работа с `if-else`, работа с логическими переменными, работа с логическими операторами, работа со `switch-case`, циклами `while` и `for`, работа с `for` для массивов, с `for-in`, а также решены задачи с циклами.