

Объект math в javascript

метод	описание
abs	абсолютное значение
sin, cos, tan	тригонометрические функции
log	натуральный логарифм
exp	экспонента
pow	показательная функция
sqrt	квадратный корень
min	наименьшее значение
max	наибольшее значение

Формула Герона

$$S = \sqrt{p(p-a)(p-b)(p-c)}, \text{ где } p = \frac{a+b+c}{2}$$

```
function areaOfTriangle(obj){  
    var a=1*obj.st1.value;  
    var b=1*obj.st2.value;  
    var c=1*obj.st3.value;  
    var p=(a+b+c)/2;  
    var s=Math.sqrt(p*(p-a)*(p-b)*(p-c));  
    obj.res.value=s;  
}
```

Сторона 1	<input type="text"/>	Сторона 2	<input type="text"/>	Сторона 3	<input type="text"/>
<input type="button" value="Вычислить"/>	<input type="button" value="Отменить"/>	Результат	<input type="text"/>		

Округление

Метод *toFixed* объекта *Number*. Синтаксис записи следующий:

`(x).toFixed(N)`

где *x* - число, которое надо округлить, а *N* - число знаков после запятой.

.....

```
var s=Math.sqrt(p*(p-a)*(p-b)*(p-c));
```

```
s=s.toFixed(2);
```

```
obj.res.value=s;
```

```
}
```

Selection Statements

(Условные конструкции)

Оператор ветвления (условный оператор) – конструкция языка программирования, обеспечивающая выполнение определённой команды (набора команд) только при условии истинности некоторого логического выражения.

ЕСЛИ: (*Условие удовлетворяет истинности*)

ТО: Выполнить эту серию инструкций

ИНАЧЕ: Выполнить эту серию инструкций

Selection Statements (Условные конструкции)

- Условный оператор: `if...else`
- Тернарный оператор: `... ? ... : ...`
- Оператор многозначного выбора: `switch – case`

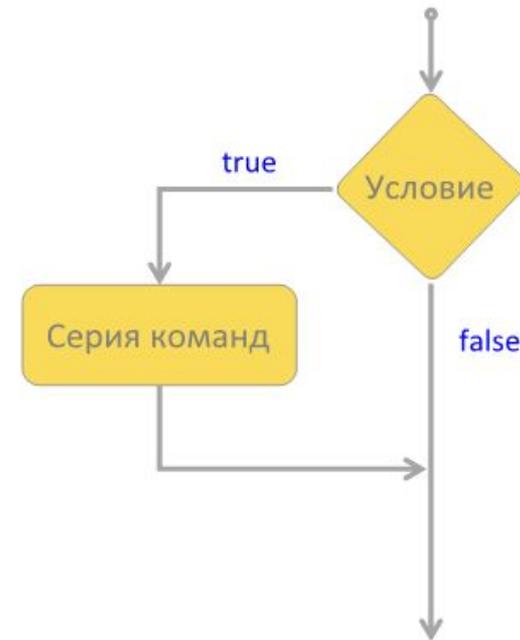
Условный оператор *if*

Синтаксис у него следующий:

```
if B {S1}  
else {S2}
```

где *B* – выражение логического типа, а *S1* и *S2* – операторы.

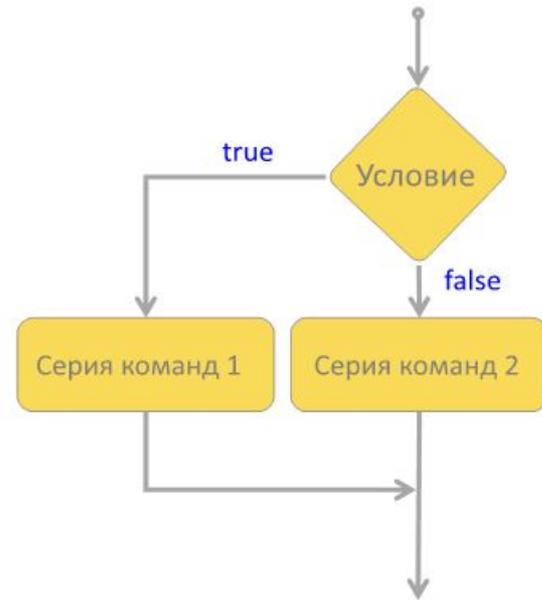
Реализует выполнение определённых команд при условии, что используемое логическое выражение в условии удовлетворяет истинности. Результатом вычисления логического выражения должно быть *true* или *false*.



```
var a = 10;  
if (a == 10) {  
  alert("a=10");  
};
```

if-else

В зависимости от условия выполняется только одна из двух серий команд, входящих в команду ветвления. Если условие соблюдено, то надо выполнить серию 1, а если нет – то серию 2.



```
var a =10;  
if (a ==10){  
  alert("a=10");  
}  
else {  
  alert("a!=10");  
};
```

Тернарный (третичный)

оператор

Тернарный оператор – операция, возвращающая свой второй или третий операнд в зависимости от значения логического выражения, заданного первым ОП



```
var a =10;
```

```
var msg = (a == 10) ? "a = 10": "a != 10";
```

```
alert(msg);
```

Поиск максимального числа

Пусть у нас есть форма, в которую пользователь вводит 3 значения. Напишем сценарий, который определит максимальное из введенных чисел.

```
function maxZnach(obj){  
    var a=1*obj.zn1.value;  
    var b=1*obj.zn2.value;  
    var c=1*obj.zn3.value;  
    var m=a;  
    if (b>m) m=b;  
    if (c>m) m=c;  
    obj.res.value=m;  
}
```

Значение 1	<input type="text"/>	Значение 2	<input type="text"/>	Значение 3	<input type="text"/>
<input type="button" value="Максимальное значение"/>		<input type="text"/>	<input type="button" value="Отменить"/>		

Поиск максимального числа используя метод *max* объекта

```
Math  
function maxZnach(obj){  
    var a=1*obj.zn1.value;  
    var b=1*obj.zn2.value;  
    var c=1*obj.zn3.value;  
    obj.res.value=Math.max(Math.max(a,b),c);  
}
```

Значение 1

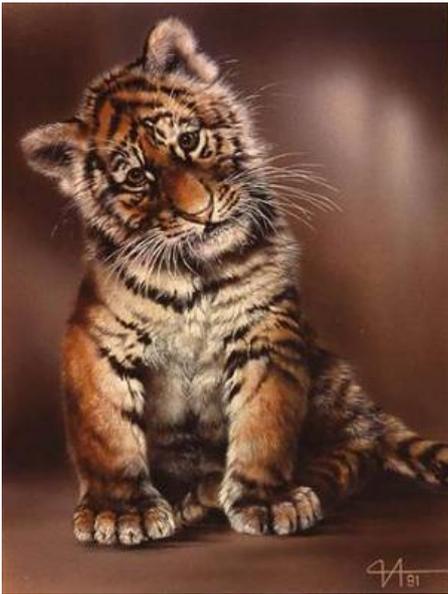
Значение 2

Значение 3

Максимальное значение

Отменить

Эффект приближения



```

```

```
function bigPict(){  
    var w=document.tigr.width;  
    if (w<302){  
        document.tigr.width=w+10;  
        document.tigr.src="images/tigrenok.jpg"  
        setTimeout("bigPict()", 500)  
    }  
}
```

Эффект приближения



- ✓ Допишите наш сценарий так, чтобы картинка при выходе курсора мыши принимала исходные значения (т.е. 102 пиксела).

Задание для самостоятельной работы

- ✓ В поле для введения задано число. При нажатии на кнопку определить парное оно или нет. Предусмотреть введение только положительных значений.