

# JavaScript библиотеки



JavaScript  
Courses

[www.courses.dp.ua](http://www.courses.dp.ua)

# jQuery

# **jQuery – самая популярная JS библиотека**

*JavaScript библиотеки - подключаемые файлы с JavaScript-кодом.*

*Файл можно скачать и хранить рядом с другими файлами своего сайта, или подключить ссылкой на сайт разработчиков jQuery.*

**<https://code.jquery.com/>**

# jQuery любят за...

*Кроссбраузерность*

*Краткость*

*Спецэффекты*

**AJAX** (асинхронная  
загрузка/выгрузка данных)

# Как использовать jQuery

# Немного практики #1

Ut condimentum, purus non auctor dapibus, augue quam vestibulum lorem, id porta quam est at mauris. Mauris in iaculis neque. Aenean fringilla imperdiet justo ut gravida. Proin eget sollicitudin urna. Mauris faucibus non eros eget rhoncus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Phasellus sollicitudin ligula non augue elementum faucibus. Vivamus viverra sodales odio. Sed vulputate luctus dui, at luctus quam malesuada a. In hac habitasse platea dictumst. Sed suscipit nec magna eget sagittis. Curabitur eget diam eget arcu lobortis tempus. Suspendisse sed nunc scelerisque, pellentesque sem sed, mollis ligula. Vestibulum bibendum, justo eget finibus commodo, arcu velit condimentum sem, nec luctus risus tortor quis ipsum. Proin pulvinar, dolor a elementum imperdiet, tortor metus eleifend ex, at rhoncus elit ligula ut elit.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. In ex ante, elementum in felis vitae, vulputate congue mi. Fusce interdum nunc quis eros pulvinar, et auctor libero viverra. Nulla iaculis vulputate neque id finibus. Sed tincidunt feugiat odio auctor viverra. Suspendisse tincidunt enim ullamcorper, cursus sapien et, sagittis libero. Aliquam tristique molestie ligula ornare tempus.

## Photo Gallery



This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

Delete



This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

Delete



This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

Delete



This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

Delete

Воспользуйтесь заготовкой: [./source/ex01.html](http://source/ex01.html)

# Как использовать jQuery

***`$(function(){ ... });`** – выполнить описываемую функцию после загрузки документа (но не внешних файлов, стилей, изображений и т.п.); Такой подход решает ту же проблему что и событие **`window.onload`***

# Как использовать jQuery

**`$("selector").action()`**

***\$()** – главная (и по сути единственная) функция в библиотеке;*

***selector** – css-селектор, для выборки элемента, или можно сразу передать объект-тег (из дерева документа);*

***action()** – какое-либо действие над найденным элементом (если элементов много, то действие применится к каждому в этой коллекции).*



# jQuery и свойства CSS

```
44  
45 | $("div > span").css("background-color", "red");  
46
```

***.css(name, value)** – задаёт (или получает) значение (**value**) css-свойства (**name**).*

# jQuery и содержимое тегов

```
alert ($("#first_p").html());  
$("#first_p").html("<b>Some content</b>");
```

*.html()* – считывает или задёт содержимое тега (аналог *innerHTML*).

# jQuery и классы

```
$ ("p").addClass ("example");  
$ ("p").removeClass ("example");
```

***.addClass()** – добавляет к тегу класс;*

***.removeClass()** – удаляет класс у тега  
(если он есть).*

# jQuery и события

```
$("p").on("click", function(e) {  
    //....  
});  
  
$(".myclass").on("mouseenter", (function(e) {  
    //....  
}));
```

*Подписка на события в jQuery осуществляется методом **.on()**, которому передаётся имя события, и функция-обработчик. Если по селектору нашлось более одного элемента, то для каждого из них будет «оформлена» подписка. Для того, чтобы отписаться от события есть метод **.off()**.*

# Цепочки вызовов

```
$("p").on("click", function(){ /*...*/ }).on("dblclick", function(){ /*...*/  
}).text("Text").css("color", "red");
```

*Результат выполнения большинства функций jQuery это объект-обёртка jQuery, к которому по второму кругу можно применять какие-либо функции «улучшайзеры».*

# Удаление элементов с jQuery

```
$ ("p") .remove () ;
```

*В плане удаление ничего оригинального создатели jQuery не придумали))*

```
$ ("p") .empty () ;
```

*Но, придумали новый метод **.empty()** который очищает всё содержимое тега (делает операцию аналогичную **.innerHTML = ""**;)*

# Добавление элементов с jQuery

```
44      $ ("p") .append ("<b>Yahoo</b>") ;  
45  
46      $ ("p") .prepend ("<b>Yahoo</b>") ;  
47  
48      $ ("p") .before ("<b>Yahoo</b>") ;  
49  
50      $ ("p") .after ("<b>Yahoo</b>") ;
```

***.append()*** – добавляет тег в качестве последнего потомка найденного тега;

***.prepend()*** – добавляет тег в качестве первого потомка найденного тега;

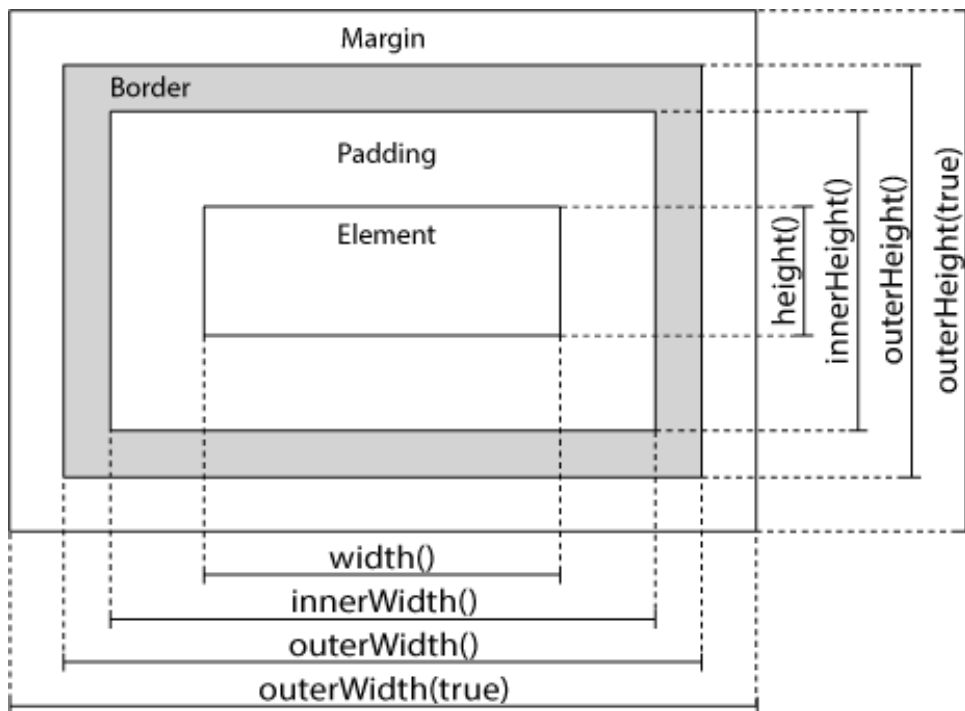
***.before()*** – добавляет перед найденным тегом;

***.after()*** – добавляет после найденного тега;

# jQuery и свойства CSS

```
$("p").css("color","red").css("background","yellow").css("font-size","22pt");
```

*Цепочки вызовов и функция `css()` позволяют в одну строчку установить любое количество свойств стиля.*



```
$("p").width(500);
```

*Для некоторых самых ходовый свойств есть даже отдельные методы: **width()**, **height()**, **innerWidth()**, **innerHeight()**, **outerWidth()**, **outerHeight()**.*



# Спецэффекты и jQuery

# Спецэффекты и jQuery

***.hide()/.show()*** – скрывает/отображает элемент на странице;

***.slideUp()/.slideDown()*** – сворачивает/разворачивает элемент на странице;

***.fadeOut()/.fadeIn()*** – «растворяет»/восстанавливает элемент на странице (работает со свойством *opacity*).

Перечисленные функции в качестве первого параметра могут получить время в миллисекундах, для задания продолжительности эффекта.

# jQuery и спецэффекты и callback

```
<script>

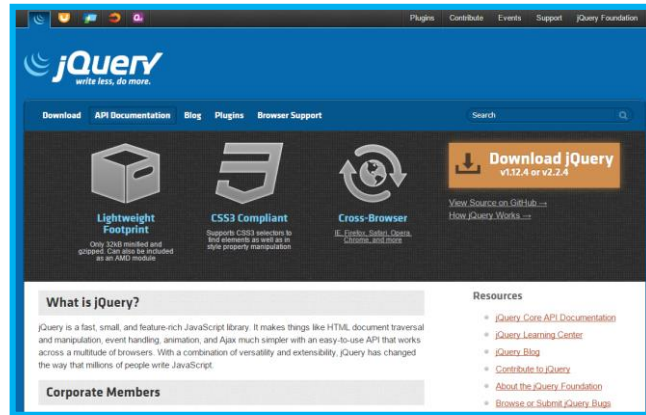
    $(function() {
        $("p").fadeOut(1000, function() { alert("animation finish!"); });
    });

</script>
```

*Поскольку анимация занимает какое-то время, то можно зарегистрировать функцию, которая будет вызвана сразу после того как анимация завершиться. Такие функции называют **callback-функциями**.*

# Документация по jQuery

# Документация по jQuery



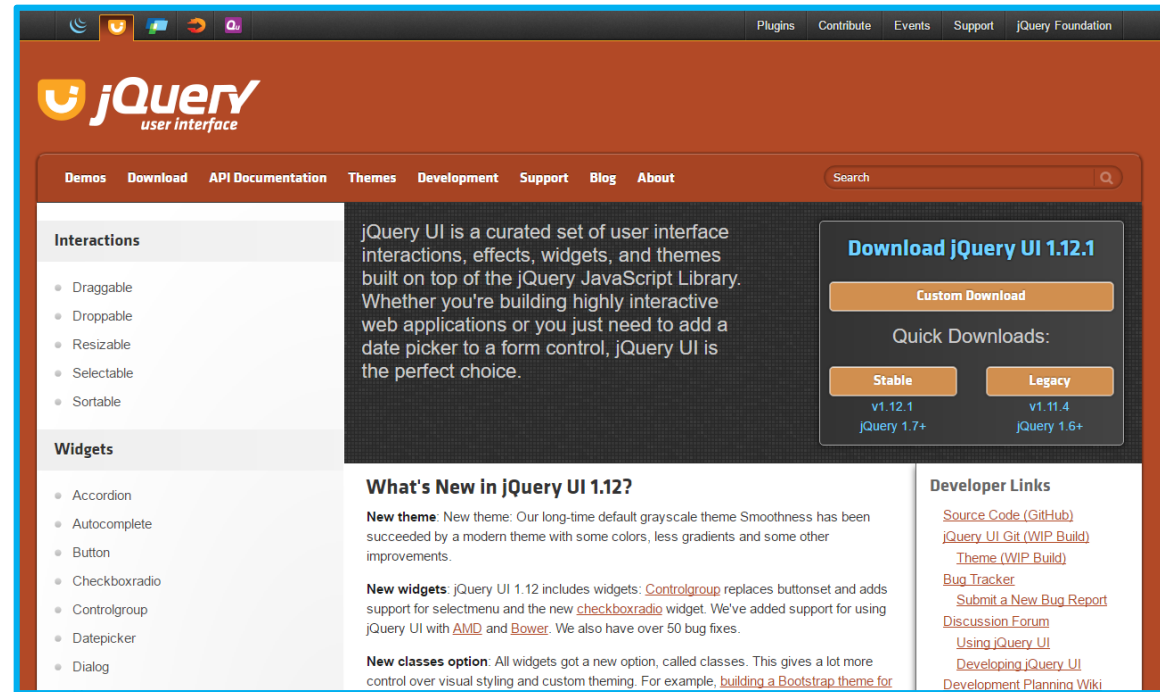
<https://jquery.com>



<https://www.w3schools.com/jquery/>

# jQuery плагины

# jQuery UI – набор элементов ввода



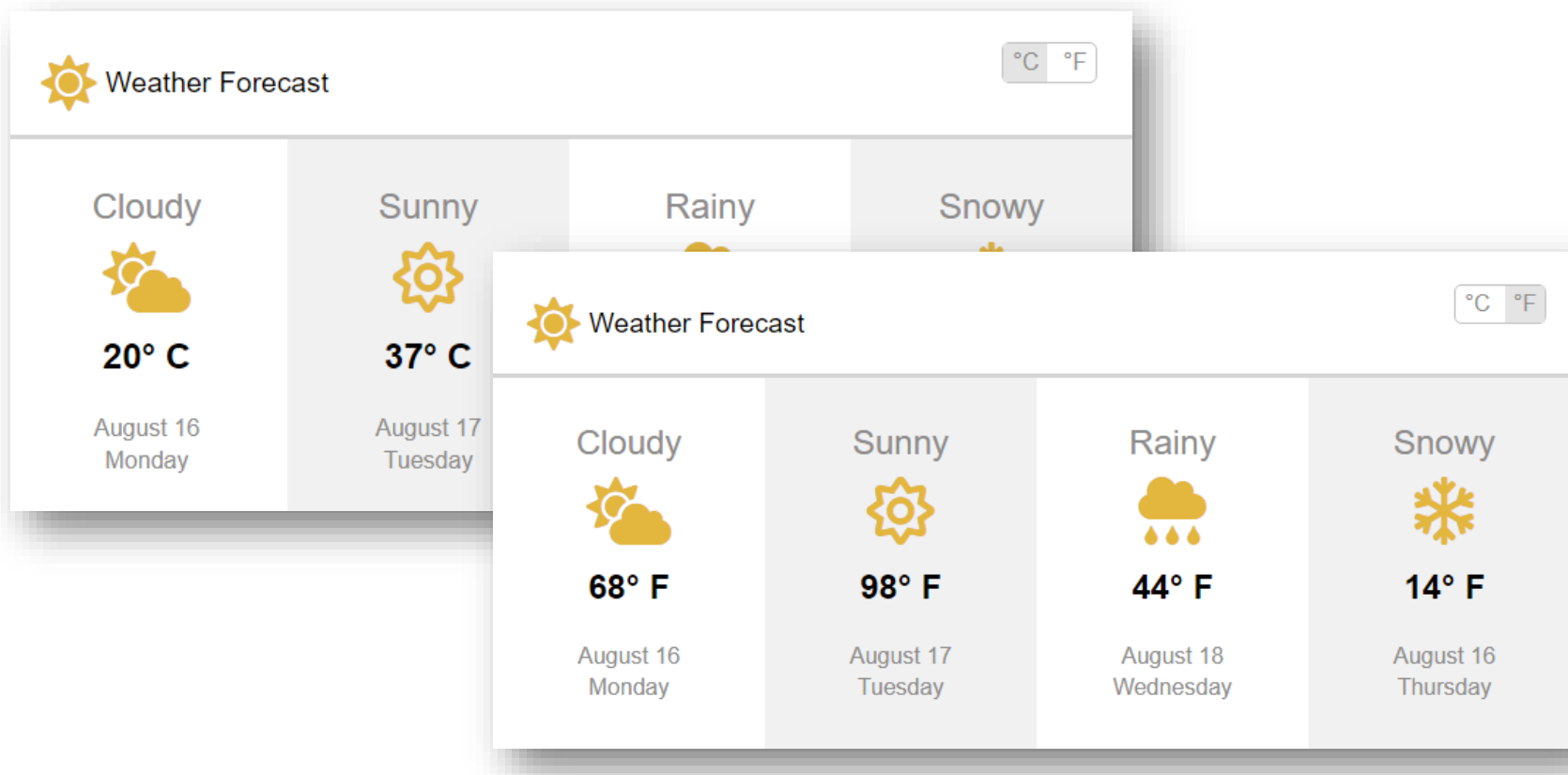
<https://jqueryui.com/>

*jQuery плагином называют JavaScript библиотеку которой для работы нужна jQuery.*

Немного практики #2



# Сделаем переключатель шкалы температуры



Воспользуйтесь заготовкой: [./source/ex02.html](https://source/ex02.html)

```

12  ($) => {
13
14      $("#c_button").on("click", () => {
15          if($("#c_button").hasClass("selected")){
16              return;
17          }
18          $("#c_button").addClass("selected");
19          $("#f_button").removeClass("selected");
20          $(".temperature").html( (i, item) => {
21              item = (parseInt(item) - 32) * ( 5 / 9 );
22              item = Math.round(item);
23              item += "° C";
24              return item;
25          });
26      });
27
28      $("#f_button").on("click", () => {
29          if($("#f_button").hasClass("selected")){
30              return;
31          }
32          $("#f_button").addClass("selected");
33          $("#c_button").removeClass("selected");
34          $(".temperature").html( (i, item) => {
35              item = parseInt(item) * ( 9 / 5 ) + 32;
36              item = Math.round(item);
37              item += "° F";
38              return item;
39          });
40      });
41
42  });

```

**Реализуем  
переключатель шкалы  
температуры (с jQuery)**

Немного практики #3

# Кредитный калькулятор 3.0

## Кредитный калькулятор

Сумма кредита (грн.)

Срок кредитования (мес.)

Ставка (% годовых)

7000

3

24

Рассчитать

#	Сумма платежа (грн.)	Часть тела кредита (грн.)	Проценты (грн.)	Остаток долга (грн.)
1	2473.33	2333.33	140.00	4666.67
2	2426.67	2333.33	93.33	2333.33
3	2380.00	2333.33	46.67	-0.00

Переплата: 280.00 грн.

Воспользуемся заготовкой: [./source/ex03.html](#)

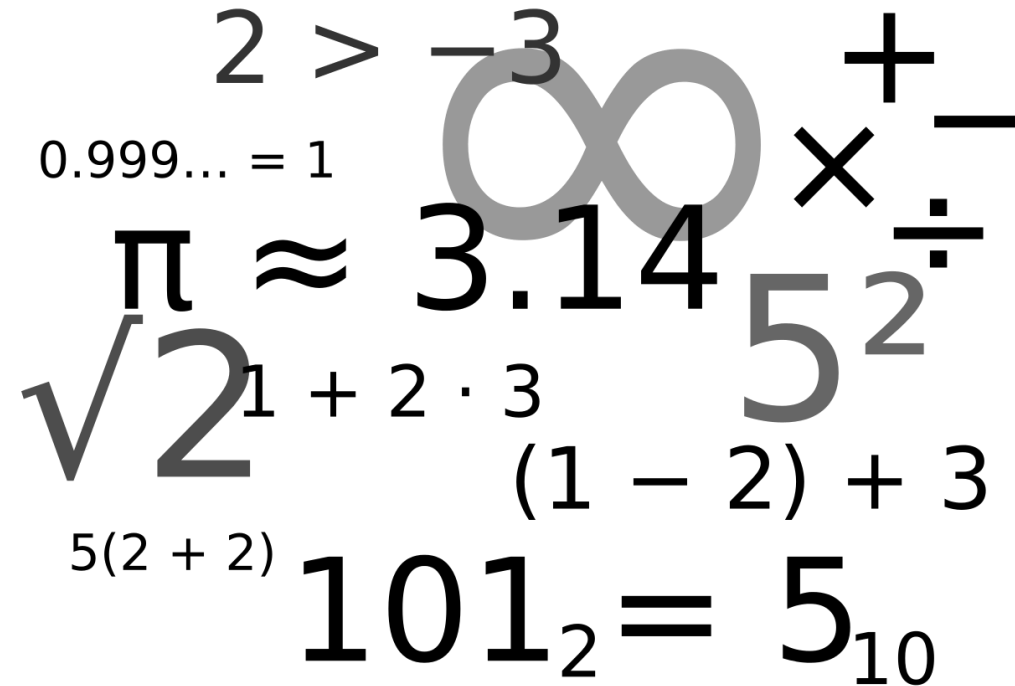
Реализуем кредитный калькулятор, который будет получать данные для расчёта из элементов ввода, и выводить план погашения кредита в разметку. Используем алгоритм классической схемы погашения.

# Кредитный калькулятор 3.0

```
14  $( () =>{
15      $("button").on("click", ()=>{
16          $('#results').empty();
17
18          var sum      = +sum_input.value;
19          var term      = +term_input.value;
20          var rate      = +rate_input.value;
21          var debt      = sum;
22
23          var rate_month = rate / 12 / 100;
24          var payment_body = +(sum / term).toFixed(2);
25
26          for(var i = 1; i <= term; i++){
27              var payment_percents = +(sum * rate_month).toFixed(2);
28              var payment = +(payment_body + payment_percents).toFixed(2);
29              if(i == term) {
30                  payment_body = sum;
31              }
32
33              sum -= payment_body;
34              sum = +sum.toFixed(2);
35
36              $('#results').append($('`
37                  <tr>
38                      <td>#${i}</td>
39                      <td>${payment}</td>
40                      <td>${payment_body}</td>
41                      <td>${payment_percents}</td>
42                      <td>${sum}</td>
43                  </tr>
44              `));
45          });
46      });
47  });
48  }
```

Математика в JS...

# Математика в JavaScript



A collage of mathematical symbols and expressions. At the top center is a large infinity symbol ( $\infty$ ). To its left is the expression  $2 > -3$ . Below that is  $0.999... = 1$ . To the right of the infinity symbol are the plus (+) and minus (-) signs. Below the infinity symbol is the expression  $\pi \approx 3.14$ . To the left of the infinity symbol is the square root symbol ( $\sqrt{2}$ ). Below the square root symbol is the expression  $1 + 2 \cdot 3$ . To the right of the infinity symbol is the multiplication (x) and division (÷) signs. Below the infinity symbol is the expression  $5^2$ . Below the square root symbol is the expression  $(1 - 2) + 3$ . At the bottom is the expression  $5(2 + 2)$  and the base conversion  $101_2 = 5_{10}$ .

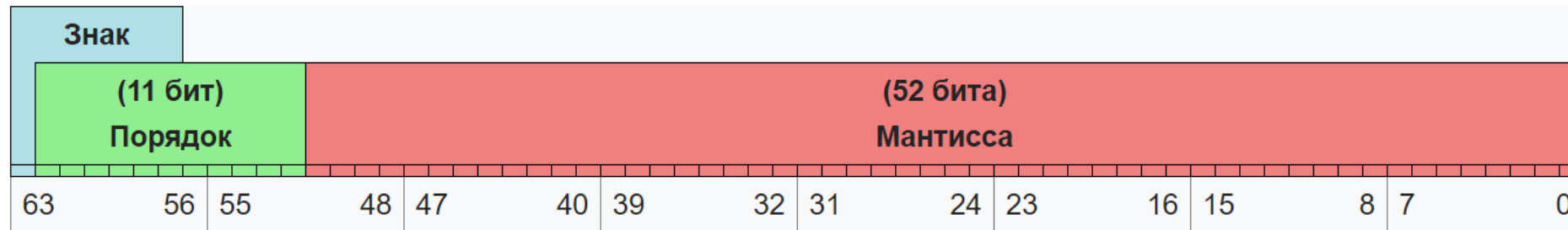
<https://habr.com/post/312880/>

<https://medium.com/nuances-of-programming/javascript-essentials-числа-и-математика-ccf6b582a79>

# Числа в JavaScript

Как хранятся числа в переменных типа **Number**:

[https://ru.wikipedia.org/wiki/Число\\_двойной\\_точности](https://ru.wikipedia.org/wiki/Число_двойной_точности)



Окончательное значение числа равняется  $\pm \text{знак} \cdot (1 + \text{мантисса} / 2^{52}) \times 2^{\text{порядок} - 1023}$ . Знак 0 соответствует положительным числам, знак 1 отрицательным. Старший бит мантиссы, который всегда равен единице, опускается. Порядок 0 ( $2^0 = 1$ ) записывается как 1023.



# Чисел в JavaScript

*Специальные константы (и методы) в JavaScript'е для определение «безопасных» чисел*

**Number.MIN\_VALUE**

**Number.MAX\_VALUE**

**Number.MIN\_SAFE\_INTEGER**

**Number.MAX\_SAFE\_INTEGER**

**Number.isSafeInteger()**

**Number.isInteger()**

# Округление числе в JavaScript

*Специальные Методы для округления:*

`Number.toFixed()`

`Number.toPrecision()`

`Math.round()`

`Math.floor()`

`Math.ceil()`

`Math.abs()`

`Math.sign()`

# Проблема сравнения чисел в JavaScript

*Сравнение чисел с поправкой на погрешность:*

```
function epsilon_equality(x, y) {  
    return Math.abs(x) - Math.abs(y) < Number.EPSILON;  
}  
  
var a = 0.1 + 0.2;  
var b = 0.3;  
  
console.log( a == b );  
console.log( epsilon_equality(a, b) );
```

**Number.EPSILON** – отражает разницу между 1 и наименьшим числом с плавающей точкой, что более 1.

Будет полезным

# BigInt - потенциальное решение проблемы

## BigInt: Integers as big as you want in JavaScript

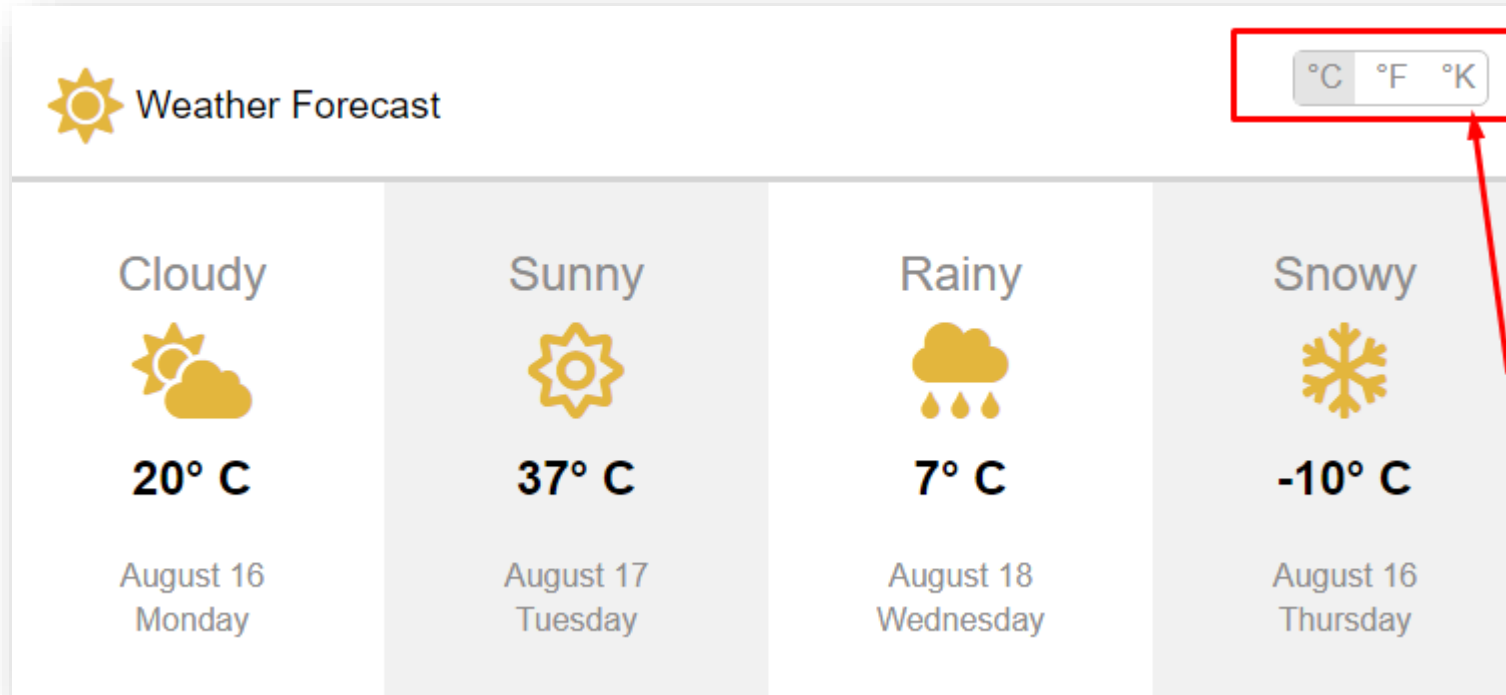
Daniel Ehrenberg  
Igalia  
In partnership with Bloomberg  
Web Engines Hackfest 2017

<https://developers.google.com/web/updates/2018/05/bigint>

Домашнее задание  
/сделать

# Домашнее задание #1.1

Реализуйте переключение между тремя шкалами  
(Цельсию, Фаренгейту и Кельвину)



Воспользуйтесь заготовкой [./homework/hw\\_i1.html](http://localhost/homework/hw_i1.html)

Домашнее задание  
/разобраться и сделать



# Домашнее задание #1.2: библиотека HighCharts

Библиотека **HighCharts** (<https://www.highcharts.com/docs/getting-started/installation>) – одно из лучших решений для построения графиков на веб-страницах. Ваша задача, разобраться как её использовать и создать страницу на которой библиотека «нарисует» вот такой график:



Данные для графика:

Дата	Температура
05.11.2018	10 ° C
06.11.2018	16.4 ° C
07.11.2018	3 ° C
08.11.2018	8.5 ° C
09.11.2018	10.4 ° C
10.11.2018	8.8 ° C
11.11.2018	7.5 ° C