

Создание таймеров и интервалов

В языке JavaScript можно вызывать методы для выполнения отложенного кода. За урок мы научимся создавать интервалы, а также таймеры в языке Js. Нами будут изучены методы «setTimeout» и «setInterval».

В **JavaScript** есть две функции, которые позволяют вызывать другие функции через определенное количество времени.

Использование интервалов

Интервалы записываются через метод `setInterval()`. Отличительная черта интервалов от таймеров заключается в способе вызова функции. Интервалы вызывают функцию постоянно через определенное количество секунд. Таймер вызывает функцию всего один раз, после чего прекращает свою работу.

Для создания интервалов используйте конструкцию:

```
// Будет вызвана функция someFunction через 1.5 секунды  
setInterval("someFunction()", 1500);
```

```
function someFunction() {  
    console.log("Функция срабатывает каждые 1,5 секунды");  
}
```

Чтобы остановить интервал его необходимо записать в переменную, после чего очистить данные:

```
// Ссылка на интервал записывается в переменную  
var linkInterval = setInterval("someFunction()", 1500);
```

```
function someFunction() {  
    console.log("Функция срабатывает каждые 1,5 секунды");  
    // Для остановки используйте метод clearInterval  
    clearInterval(linkInterval);  
}
```

Использование таймеров

Здесь схожая структура, только вот таймер не нужно очищать, так как они срабатывают лишь один раз:

```
setTimeout("simple()", 1500);
```

```
function simple() {  
    console.log("Функция будет вызвана лишь один раз через 1.5 секунды после старта программы");  
}
```

За счёт этих функций вы можете откладывать выполнение других функций на определенное количество времени.

Метод «setInterval»:

```
var counter = 0;

var id = setInterval(my_func, 1000);

function my_func() {
  counter++;
  console.log("Counter: " + counter);

  if(counter == 3)
    clearInterval(id);
}

// setInterval(function() {
//   counter++;
//   console.log("Прошло секунд: " + counter);
// }, 1000);
```

Метод «setTimeout»:

```
setTimeout(my_func, 2000);

function my_func() {
  console.log("Timer is working!");
}
```

Задание 1:

Постепенный вывод чисел

Напишите функцию, что будет последовательно выводит в консоль числа от 10 до 32, с интервалом между числами 120 мс. То есть, каждые 120 мс в консоли должно появляться очередное число.

Для реализации используйте интервалы.

Решение задания:

```
// Можно прописать лишь название функции, но без скобок  
var intervalNumbers = setInterval(numbersToConsole, 120);  
var numbers = 10;
```

```
function numbersToConsole() {  
    // Выводим число  
    console.log("Число - " + numbers);  
  
    numbers++; // Увеличиваем число на 1  
    if(numbers > 32) // Если число больше 32, тогда останавливаем интервал  
        clearInterval(intervalNumbers);  
}
```

Использование таймеров

Выполните предыдущее задание, но вместо интервалов используйте таймеры.

Решение задания:

```
// Вызов функции  
var numbers = 10;  
numbersToConsole();
```

```
function numbersToConsole() {  
    // Выводим число  
    console.log("Число - " + numbers);  
  
    numbers++; // Увеличиваем число на 1  
    // Если число меньше или равно 32, то вызываем функцию снова  
    if(numbers <= 32)  
        // Можно прописать просто название функции, но без скобок  
        setTimeout(numbersToConsole, 120);  
}
```