**ЦИКЛЫ**

Цикл – инструкция, позволяющая повторно выполнять конкретный участок кода определённое количество раз.

Одно выполнение кода называется итерацией.

for (initialExpression; condition; incrementExpression) {

    statement

}

For

Порядок выполнения цикла for:

* • объявление и инициализация переменной-счетчика var i = 0; (initialExpression), которая вычисляется только один раз перед первой итерацией;
* • проверяется условие i < 5; (condition) и если оно возвращает true, то тело цикла (инструкция в фигурных скобках) будет выполнено, если же возвращается false, то тело цикла не выполняется и его выполнение прекращается (например, пока i меньше 10, цикл будет выполнятся);
* • выполнение тела цикла (statement);
* • выполнение выражения i++ (incrementExpression), которое изменяет значение переменной-счетчика, которая используется в проверке условия выполнения цикла, после этого вновь выполняется проверка условия (condition).

for (var i = 0; i < 5; i++) {

console.log("User-" + i);

}

"User-0"

"User-1"

"User-2"

"User-3"

"User-4"

Так же можно использовать декремент.

for (var i = 5; i>= 1; i--) {

console.log("User-" + i);

}

"User-5"

"User-4"

"User-4"

"User-2"

"User-1"

While

Принцип работы цикла while аналогичен циклу for: инициализация  
переменной-счетчика, проверка условия, выполнение тела цикла,  
изменение значения переменной-счетчика.

var i = 3;

while (i >= 0) {

console.log("Countdown: " + i);

i--;

}

"Countdown: 3"

"Countdown: 2"

"Countdown: 1"

"Countdown: 0"

Инструкция внутри цикла (блок кода в фигурных скобках) будет выполняться в случае, если условное выражение вычисляется в true. Если первая проверка даст результат false, блок инструкций не выполнится ни разу. Таким образом тело цикла будет выполнятся до тех пор покаcondition (условие) = true.

do … while

Отличие do … while от других циклических конструкций: тело цикла,  
выполнится минимум один раз, даже если условие будет false.

var i = 2;

do {

console.log("Even number: " + i);

i += 2;

} while (i < 10);

"Even number: 2"

"Even number: 4"

"Even number: 6"

"Even number: 8"

"Even number: 10"

do {

console.log("Loops.");

i++;

} while (i < 10);

"Loops."

Операторы break и continue

Иногда бывает необходимо выйти из цикла до его завершения.  
Оператор break прерывает работу цикла в указанный момент, даже если условие для выполнения следующей итерации true. Чаще всего оператор break; является частью конструкции if.

for (var i = 0; i < 10; i++) {

if (i == 6) {

break;

}

console.log(i);

}

Как только переменная i получит значение 6, выполнение цикла будет прервано.

Оператор continue прерывает выполнение текущей итерации, пропускает ее, и переходит к следующей.

var i = 0;

while (i < 10) {

i++;

if (i == 2 || i == 5 ) {

continue;

}

console.log(i);

}

Если переменная i получит значение 2 или 5, то тело цикла на этих итерациях выполнено не будет.

Вложенные циклы

Во вложенных циклах выполняется одна итерация внешнего, далее  
все итерации внутреннего. Так продолжается до полного выполнения  
цикла самого верхнего уровня.

for (var i = 1; i < 3; i++) {

console.log("Outside: " + i);

for (var j = 1; j < 3; j++) {

console.log("Inside: " + j);

}

}

"Outside: 1"

"Inside: 1"

"Inside: 2"

"Outside: 2"

"Inside: 1"

"Inside: 2"

Циклы для объектов и массивов

for … in

for (variable in object) {

statements

}

for … of

for (variable of array) {

statements

}

Циклы for…in используются для обхода свойств объектов, не являющихся массивами.  
Конструкция for…in перебирает ключи объекта.

var user = {

name: "John",

age: 25,

country: "USA"

};

for (var key in user) {

console.log(key);

}

"name"

"age"

"country"

Цикл for … of предназначен для итерации по элементам массива, но в отличие от цикла for … in при итерациях используется значение, а не ключ.  
Другими словами for … of перебирает значения массива.

var user = ["John", "25", "USA"];

for (var value of user) {

console.log(value);

}

"John"

"25"

"USA"

[ПРЕДЫДУЩАЯ ЧАСТЬ](https://www.jsexpert.net/course/clear-javascript-beginner/?course_type=content&course_page=9&lecture=1)[СЛЕДУЮЩАЯ ЧАСТЬ](https://www.jsexpert.net/course/clear-javascript-beginner/?course_type=content&course_page=9&lecture=3&section-quiz=1)