## Цикл for

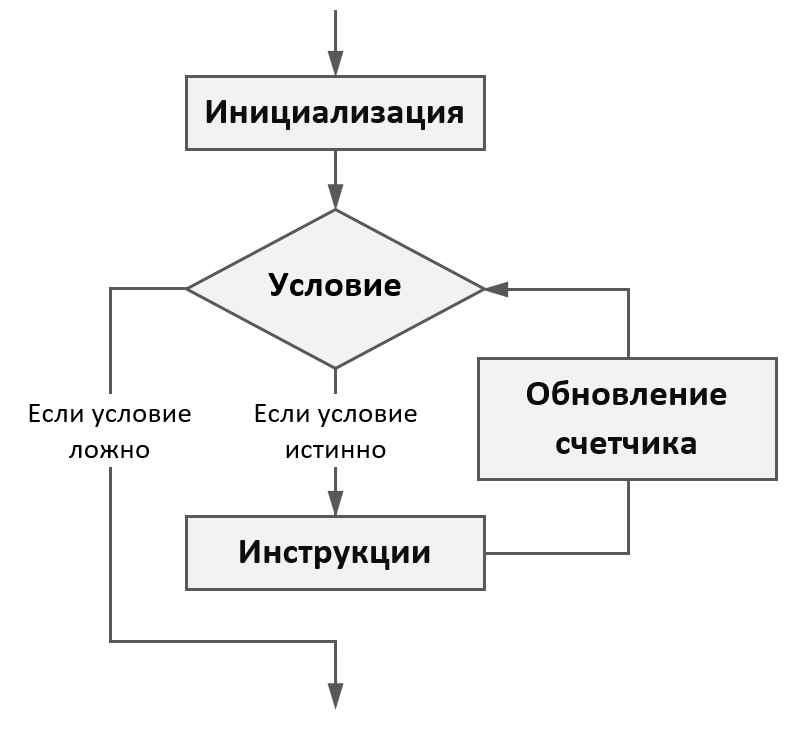
Данный цикл в основном используется когда **известно точное количество повторений**. Этот цикл ещё называют циклом со счётчиком.

Синтаксис цикла «for»:

for (инициализация; условие; финальное выражение) {

/\* тело цикла \*/

}



Основные части конструкции цикла «for»:

* **инициализация** - это выражение, которое выполняется один раз перед выполнением цикла; обычно используется для инициализации счётчика;
* **условие** - это выражение, истинность которого проверяется перед каждой итерацией; если выражение **вычисляется как истина**, то выполняется итерация; в противном случае цикл «for» завершает работу;
* **финальное выражение** - это выражение, которое выполняется в конце каждой итерации; обычно используется для изменения счетчика;
* **тело цикла** - инструкции, выполнение которых нужно повторять.

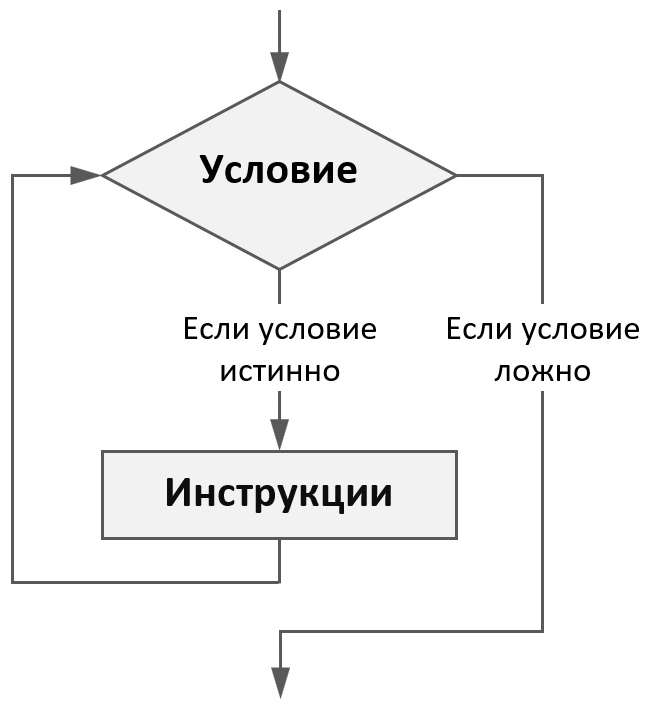
Цикл while

Данный цикл предназначен для многократного выполнения одних и тех же инструкций до тех пор, пока истинно некоторое условие. Цикл «while» в основном используется, когда **количество повторений заранее не известно.**

while (условие) {

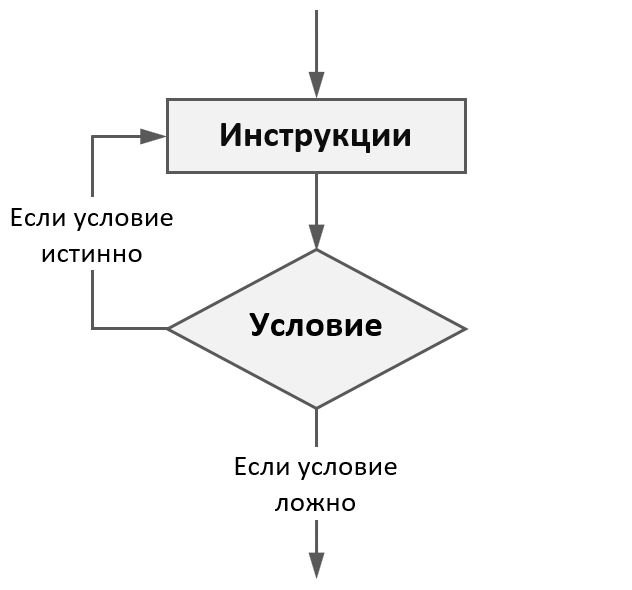
/\* тело цикла \*/

}



Цикл do...while

Цикл «do...while», также как и цикл «while», выполняет одни и те же инструкции до тех пор, пока указанное условие истинно. Но в отличие от «while» в «do...while» условие **проверяется после выполнения инструкций.** Поэтому цикл «do...while» в любом случае выполнится не меньше одного раза, даже если условие изначально ложно.



do {

/\* тело цикла \*/

} while (условие)

Цикл for...in

Цикл «for...in» предназначен для перебора перечисляемых имён свойств объекта. В JavaScript свойство является перечисляемым, если его внутренний флаг [[Enumerable]] равен true.

Свойства объекта, которые не относятся к перечисляемым, в цикле не участвуют.

Например, объект (массив) созданный с использованием функции-конструктора Array или его литеральной записи имеет не перечисляемые свойства от Array.prototype и Object.prototype, такие как indexOf(), some(), toString() и др. Они не будут участвовать в цикле.

Инструкции break и continue

Внутри тела цикла можно использовать специальные инструкции: break и continue.

Инструкция **«break» предназначена для прекращения выполнения текущего цикла**. Другими словами, она осуществляет выход и передачу управления инструкции, идущей после этого цикла.

Инструкция **«continue» предназначена для прекращения дальнейшего выполнения кода и перехода к следующей итерации цикла**.

### **Метки для break и continue**

Метка представляет собой идентификатором с двоеточием, который необходимо указать перед циклом.

someLabel: while (условие) {

// текло цикла

}

Далее после оператора break или continue необходимо указать эту метку:

someLabel: while (условие) {

if (условие) {

break someLabel;

}

}

Вызов break someLabel приведёт к переходу в **конец цикла, перед которым данная метка указана**.

Цикл for...of (новинка в ES6)

Цикл for...of появился в стандарте ES6. Предназначен он для перебора **итерируемых объектов**, т.е. объектов, в которых реализован метод Symbol.iterator. Этот метод ещё называют **итератором**. Именно его и использует цикл for...of для перебора объектов.

Метод Symbol.iterator имеется у String, Array, Map, Set, arguments, NodeList и других объекто