

CS101 Thesaurus

	ключевое слово		
	произвольное имя		
Term	Eng Term	Description	Code example
Переменная	variable	хранилище с именем. Может хранить только значение	<code>let varX = 10;</code> // это инициализация и объявление переменной
Объявление переменной	variable declaration	выделение памяти под значение	<code>let varX;</code>
Инициализация переменной (или присвоение значения)	variable init	начальное присвоение значения	<code>let varX; varX = 10;</code>
Строка	string	строковое значение	<code>let str = 'I am a string';</code>
Тип данных	data type	тип данных, хранящихся в переменной или массиве	<code>let varX = 10; // тип данных: число let str = 'I am a string'; // тип данных: строка let str = '10'; // тип данных: строка let status = true; // тип данных: boolean (логический)</code>
Оператор	operator	математические, логические	<code>a = 3; // оператор присваивания a == 3; // оператор сравнения a === 3; // строгое сравнение (в т.ч. по типу) i++; // ++ - оператор</code>
Операнд	operand	-	<code>a > b; // a и b - операнды i++; // i - операнд</code>
Выражение	expression	-	<code>x = a + b * 2; x > 10 x > 10 && y < 12</code>
Тело функции/цикла/условного оператора	function/loop/conditional body	код между фигурными скобками	<code>... { // все от скобки до скобки - тело функции/цикла/условного оператора }</code>
Условный оператор	conditional statement	конструкция для осуществления ветвления	<code>if (x > 10) { // body }</code>
Массив	array	множество значений, объединенных одним именем	<code>let ages = [77, 12, 55, 4];</code>
Элемент массива	array item/element	-	<code>ages[0] // один элемент массива</code>
Значение элемента	item/element value	-	<code>let ages = [77, 12, 55, 4]; // 77 - значение эл-ма ages[0]</code>
Индекс элемента	index	порядковый номер элемента	<code>ages[0] // 0 - индекс элемента</code>
Длина массива	length	количество элементов в массиве	<code>ages.length // возвращает кол-во элементов массива</code>
Цикл	loop	повторяющийся блок кода	<code>for (let i = 0; i < 10; i++) { // повторяется то, что внутри фигурных скобок (т.е. тело цикла) }</code>
Итерация	iteration	одно повторение цикла	<code>for (let i = 0; i < 3; i++) { console.log('это итерация №' + i); console.log('это все еще итерация №' + i); } // вывод в консоль: // 0 // 0 // 1 // 1 // 2 // 2</code>
Функция	function	блок кода, имеющий имя, входные параметры (опционально) и возвращаемое значение (опционально)	-
Объявление функции / Определение функции	function declaration / function definition	создание функции без запуска	<code>function hello() { console.log("Hello!"); }</code>
Вызов функции	function call	чтобы запустить функцию, ее нужно вызвать по имени	<code>hello();</code>
Точка вызова функции	function entry point	строка, в которой расположен вызов функции	<code>function hello() { console.log("Hello!"); } hello(); // это точка вызова ф-ии</code>
Параметры функции	function parameters	имена переменных, которые поступают на вход функции	<code>// a и b - параметры ф-ии function summation(a, b) { console.log(a + b); }</code>
Аргументы функции	function arguments	значения, которые присваиваются параметрам функции в порядке упоминания	<code>hello(3, 7); // 3 и 7 - аргументы ф-ии</code>
Возвращаемое значение	return value	то, что поступает на выход из функции (все, что стоит после return)	<code>function summation(a, b) { return (a + b); // результат a+b - это возвращаемое значение }</code>
Встроенная функция	incorporated function	функции, которые уже определены кем-то за вас (разработчиками языка или библиотеки), и которые можно вызывать	<code>confirm() // это вызов встроенной в JavaScript функции. Ее определение скрыто от разработчика prompt() // то же самое</code>
Метод	method	функция объекта (класса)	