 «функции-стрелки» или «стрелочные функции» (arrow functions), т.к. выглядит следующим образом:

let func = (arg1, arg2, ...argN) => expression;

Это создаёт функцию func, которая принимает аргументы arg1..argN, затем вычисляет expression в правой части с их использованием и возвращает результат.

let sum = (a, b) => a + b;

/\* Эта стрелочная функция представляет собой более короткую форму:

let sum = function(a, b) {

return a + b;

};

\*/

* Если у нас только один аргумент, то круглые скобки вокруг параметров можно опустить, сделав запись ещё короче:
* let double = n => n \* 2;
* // примерно тоже что и: let double = function(n) { return n \* 2 }

alert( double(3) ); // 6

* Если аргументов нет, круглые скобки будут пустыми, но они должны присутствовать:
* let sayHi = () => alert("Hello!");

sayHi();

Стрелочные функции можно использовать так же, как и Function Expression.

Например, для динамического создания функции:

let age = prompt("Сколько Вам лет?", 18);

let welcome = (age < 18) ?

() => alert('Привет!') :

() => alert("Здравствуйте!");

welcome();

Они очень удобны для простых однострочных действий, когда лень писать много слов.

**[Многострочные стрелочные функции](https://learn.javascript.ru/arrow-functions-basics" \l "mnogostrochnye-strelochnye-funktsii)**

Если в теле функции несколько выражений или инструкций необходимы фигурные скобки и return