Типы функций в JavaScript

Основные виды функций

1. Function Declaration (Объявление функции)

javascript

Copy

Download

function sum(a, b) {

return a + b;

}

**Особенности:**

* Создается до выполнения кода (поднимается - hoisting)
* Можно вызывать до объявления
* Имеет собственное имя

2. Function Expression (Функциональное выражение)

javascript

Copy

Download

const sum = function(a, b) {

return a + b;

};

**Особенности:**

* Создается в момент выполнения
* Нельзя вызвать до объявления
* Может быть анонимной

3. Named Function Expression (Именованное функциональное выражение)

javascript

Copy

Download

const factorial = function calc(n) {

return n <= 1 ? 1 : n \* calc(n - 1);

};

**Особенности:**

* Имя доступно только внутри функции
* Полезно для рекурсивных вызовов
* Удобно при отладке (имя отображается в стеке вызовов)

Сравнение типов функций

| **Характеристика** | **Declaration** | **Expression** | **Named Expression** |
| --- | --- | --- | --- |
| Hoisting | ✅ Да | ❌ Нет | ❌ Нет |
| Доступность имени | Глобально | Нет имени | Только внутри |
| Рекурсивные вызовы | По имени | Через переменную | По внутреннему имени |
| Отладка | Хорошо | Плохо | Хорошо |

Практические примеры

Разница в hoisting

javascript

Copy

Download

// Function Declaration - работает

console.log(sum(2, 3)); // 5

function sum(a, b) { return a + b; }

// Function Expression - ошибка

console.log(multiply(2, 3)); // ReferenceError

const multiply = function(a, b) { return a \* b; };

Рекурсия с Named Function Expression

javascript

Copy

Download

const power = function pow(x, n) {

return n == 1 ? x : x \* pow(x, n - 1);

};

console.log(power(2, 3)); // 8

Присвоение функции другой переменной

javascript

Copy

Download

function sayHi() { console.log('Привет'); }

const greet = sayHi;

greet(); // 'Привет'

sayHi = null;

greet(); // Все еще работает

Современные альтернативы (ES6+)

Стрелочные функции

javascript

Copy

Download

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

**Отличия:**

* Нет собственного this
* Нет arguments
* Не могут быть использованы как конструкторы

Выводы

1. Для обычных случаев используйте Function Declaration
2. Для callback'ов и анонимных функций - Function Expression
3. Для сложных рекурсивных функций - Named Function Expression
4. В современном коде предпочтительны стрелочные функции для лаконичности