



Frontender[1.0] JavaScript - Объекты, методы, деструктуризация, for in, strict mode

YouTube	https://youtu.be/32-UUtCpWf4
Telegram	https://t.me/Dmitry_Kolotilshikov
Github	https://github.com/DmitryKolotilshikov/
Boosty	https://boosty.to/dmitry_ko
# Номер урока	24

💡 Задачи к этому уроку тут https://boosty.to/dmitry_ko

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script src="script.js"></script>
  <title>Objects - methods, for in, destructuring</title>
</head>
<body>
</body>
</html>
```

```
// Объекты, методы, деструктуризация, for in, строгий режим (strict mode)

"use strict";

const log = console.log;

// -----
// for in (цикл по объекту) f
const user = {
  id: 4,
  name: "Alena",
  age: 25,
  position: "developer",

  address: {
```

```

        country: "Australia",
        city: "Sidney",
    }
};

for (const key in user) {
    log(`New user -> ${key}: ${user[key]}`);
}

log(Object.keys(user)); // создается массив ключей из объекта

for (const key of Object.keys(user)) {
    log(`New user -> ${key}: ${user[key]}`);
}
// ----- деструктуризация, деструктурирующее присваивание -----
const { name, age, ...restProps } = user;
log(name, age, restProps);

const newUser = {
    ...user
}

log(newUser);
log(newUser === user);
newUser.name = "Alex";

log(newUser, user);

newUser.address.city = "Melbourne";
log(newUser, user);

delete newUser.address.city;
delete newUser.address.country;
log(newUser, user);

const obj1 = { a: 10, b: 20};
const obj2 = { c: 30, d: 40};

const obj3 = {
    ...obj1,
    ...obj2,
    a: 100,
    e: 50
}

log(obj3);

const obj4 = Object.assign({}, obj1, obj2);

log(obj4);

// ----- Методы -----

let x = Object.keys(user);

x = Object.values(user);
x = Object.entries(user);

```

```
x = Object.hasOwnProperty(user, "address");
x = Object.hasOwnProperty(user, "skills");

x = "address" in user;

log(x);

Object.freeze(user);

// user.name = "Olga"; // переприсвоение значения поля freeze объекта - use strict
// delete user.name; // удаление свойства freeze объекта - use strict
log(user);

// function example(a, a) { // дублирование параметров - use strict
//     return a;
// }
```