**[Добавьте toString в словарь](https://learn.javascript.ru/prototype-methods" \l "dobavte-tostring-v-slovar)**

важность: 5

Имеется объект dictionary, созданный с помощью Object.create(null) для хранения любых пар ключ/значение.

Добавьте ему метод dictionary.toString(), который должен возвращать список ключей, разделённых запятой. Ваш toString не должен выводиться при итерации объекта с помощью цикла for..in.

Вот так это должно работать:

let dictionary = Object.create(null);

// ваш код, который добавляет метод dictionary.toString

// добавляем немного данных

dictionary.apple = "Apple";

dictionary.\_\_proto\_\_ = "test"; // здесь \_\_proto\_\_ -- это обычный ключ

// только apple и \_\_proto\_\_ выведены в цикле

for(let key in dictionary) {

alert(key); // "apple", затем "\_\_proto\_\_"

}

// ваш метод toString в действии

alert(dictionary); // "apple,\_\_proto\_\_"

Ответ:

let dictionary = Object.create(null, {

toString: {

value() {

return Object.keys(this).join();

}

}

});

**[Разница между вызовами](https://learn.javascript.ru/prototype-methods" \l "raznitsa-mezhdu-vyzovami)**

важность: 5

Давайте создадим новый объект rabbit:

function Rabbit(name) {

this.name = name;

}

Rabbit.prototype.sayHi = function() {

alert(this.name);

};

let rabbit = new Rabbit("Rabbit");

Все эти вызовы делают одно и тоже или нет?

rabbit.sayHi();

Rabbit.prototype.sayHi();

Object.getPrototypeOf(rabbit).sayHi();

rabbit.\_\_proto\_\_.sayHi();

Ответ: В первом вызове this == rabbit, во всех остальных this равен Rabbit.prototype, так как это объект перед точкой.

Так что только первый вызов выведет Rabbit, а остальные – undefined