**Лабораторная работа №7. Методы объектов и контекст вызова**

*Методы объектов, this. Создание объектов через "new". Флаги и дескрипторы свойств. Геттеры и сеттеры.*

**Задание:** изучите теорию и решите задачи.

**Задачи:**

1. Создайте объект products={}, в котором хранится весь имеющийся товар. Весь товар разделен на категории, одна из них «Обувь», которая в свою очередь делится на виды «Ботинки», «Кроссовки», «Сандалии». О каждой паре обуви хранится следующая информация: уникальный номер товара, размер обуви, цвет, **цена**.
2. Создайте функцию-фильтр обуви по цене в заданном диапазоне, по заданному размеру, по заданному цвету. Выведите номера товаров, соответствующих заданному условию (фильтру).
3. Создайте объект newProducts аналогичный первому. Добавьте свойства «скидка» и «стоимость товара». Преобразуйте свойство «**цена**» в свойство-аксессор таким образом, чтобы цена товара рассчитывалась с учетом стоимости и скидки на товар.
4. Переопределите дескриптор свойств:

- «**цена**»: свойство можно изменять, можно перечислять, нельзя удалять;

- “уникальный номер товара”: нельзя изменять, можно перечислять, нельзя удалять.

1. Для каждого товара (пара обуви) создайте функцию-конструктор.
2. Создайте объект allProducts аналогичный newProducts. Каждый товар (пару обуви) создайте с помощью конструктора.

*Методы объектов, this*

let user = {

name: "John",

age: 30,

sayHi() {

// "this" - это "текущий объект".

alert(this.name);

}

};

user.sayHi(); // John

Здесь во время выполнения кода user.sayHi() значением this будет являться user (ссылка на объект user).

В JavaScript ключевое слово «this» ведёт себя иначе, чем в большинстве других языков программирования. Его можно использовать в любой функции, даже если это не метод объекта.

*Создание объектов через "new".*

function Rabbit(name) {

this.name = name;

alert( name );

}

var rabbit = new Rabbit("Кроль");

var rabbit2 = new rabbit.constructor("Крольчиха");

Эта возможность бывает полезна, когда, получив объект, мы не знаем в точности, какой у него был конструктор (например, сделан вне нашего кода), а нужно создать такой же.

*Флаги и дескрипторы свойств.*

## [Флаги свойств](https://learn.javascript.ru/property-descriptors" \l "flagi-svoystv)

Помимо значения **value**, свойства объекта имеют три специальных атрибута (так называемые «флаги»).

* **writable** – если true, свойство можно изменить, иначе оно только для чтения.
* **enumerable** – если true, свойство перечисляется в циклах, в противном случае циклы его игнорируют.
* **configurable** – если true, свойство можно удалить, а эти атрибуты можно изменять, иначе этого делать нельзя.

Возвращаемое значение – это объект, так называемый «дескриптор свойства»: он содержит значение свойства и все его флаги.

*Геттеры и сеттеры.*

* Для большего контроля над присвоением и чтением значения вместо свойства делают «функцию-геттер» и «функцию-сеттер», геттер возвращает значение, сеттер – устанавливает.
* Если свойство предназначено только для чтения, то может быть только геттер, только для записи – только сеттер.
* В качестве альтернативы к паре геттер/сеттер применяют единую функцию, которая без аргументов ведёт себя как геттер, а с аргументом – как сеттер.