**Проверяем свойство isAvailable**

Цикл работает и корректно перебирает коллекцию DOM-элементов. Займёмся реализацией программы.

Будем считать, что массив с данными о товарах и псевдомассив с DOM-элементами всегда имеют одинаковую длину, это позволит перебирать две структуры в одном цикле. Заведём переменную product, которая будет соответствовать текущему товару. Тогда текущий DOM-элемент (мы уже сохраняем его в переменную element) и текущий товар будут совпадать друг с другом.

Чтобы отобразить статус наличия товара, надо добавлять DOM-элементу разные классы. Сделаем это в несколько шагов:

* Создадим переменную для хранения класса, связанного с наличием. По умолчанию считаем, что товар в наличии, поэтому исходное значение у переменной будет product--available.
* Добавим проверку свойства isAvailable у товара. Если оно равно false (товар не в наличии), изменим значение переменной на product--unavailable.
* С помощью classList.add() добавим DOM-элементу класс, хранящийся в переменной.

Из [части «Условия»](https://htmlacademy.ru/courses/209) вы знаете, что булевы значения сами могут выступать в качестве условий. Поэтому наша проверка могла бы выглядеть примерно так:

var value = true;

if (value) {

// Код выполнится

}

Только в нашем случае проверка должна сработать, если значение свойства isAvailable равно false. Поэтому нужна проверка с отрицанием:

var value = false;

if (!value) {

// Код выполнится

}

JS

var catalogData = [

{

isAvailable: true,

isSpecial: false

},

{

isAvailable: false,

isSpecial: false

},

{

isAvailable: true,

isSpecial: true

},

{

isAvailable: true,

isSpecial: false

},

{

isAvailable: false,

isSpecial: false

}

];

var updateCards = function (products) {

var elements = document.querySelectorAll('.product');

for (var i = 0; i < elements.length; i++) {

var element = elements[i];

console.log(element);

element.classList.add('product--available');

}

};

updateCards(catalogData);

Задачивыполнено 0 из 5

1. После вывода в консоль element создайте переменную product, которая равна текущему элементу массива products, а затем выведите её в консоль.
2. После вывода product создайте переменную availabilityClass со значением 'product--available'.
3. После этой переменной добавьте проверку, что товара нет в наличии.
4. Если условие выполняется, переопределите значение availabilityClass на 'product--unavailable'.
5. В вызове метода classList.add() у текущего DOM-элемента замените строку 'product--available' на переменную availabilityClass.

Задачивыполнено 0 из 5

1. После вывода в консоль element создайте переменную product, которая равна текущему элементу массива products, а затем выведите её в консоль.

var product = products[i];

console.log(product);

1. После вывода product создайте переменную availabilityClass со значением 'product--available'.

var availabilityClass = 'product--available';

1. После этой переменной добавьте проверку, что товара нет в наличии.

if (!product.isAvailable) {

~~availabilityClass = 'product--unavailable';~~

}

1. Если условие выполняется, переопределите значение availabilityClass на 'product--unavailable'.

if (!product.isAvailable) {

availabilityClass = 'product--unavailable';

}

1. В вызове метода classList.add() у текущего DOM-элемента замените строку 'product--available' на переменную availabilityClass.

element.classList.add(availabilityClass);

var updateCards = function (products) {

var elements = document.querySelectorAll('.product');

for (var i = 0; i < elements.length; i++) {

var element = elements[i];

console.log(element);

var product = products[i];

console.log(product);

var availabilityClass = 'product--available';

if (!product.isAvailable) {

availabilityClass = 'product--unavailable';

}

element.classList.add(availabilityClass);

}

};

updateCards(catalogData);

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание