

Домашнее задание к лекции 2.2 «Прототип и конструктор объекта»

Перед началом работы

- 1.Активируйте строгий режим соответствия.
- 2.Скопируйте код ниже и вставьте его в начало своей работы:

```
var positions = [  
  {  
    title: 'Телепорт бытовой VZННН-101',  
    price: 10000,  
    discount: 7,  
    available: 3  
  },  
  {  
    title: 'Ховерборд Mattel 2016',  
    price: 9200,  
    discount: 4,  
    available: 14  
  },  
  {  
    title: 'Меч световой FORCE (синий луч)',  
    price: 57000,  
    discount: 0,  
    available: 1  
  }  
];
```

Задача № 1. Резервирование.

Компания наняла еще одного разработчика, который реализовал нам функционал резервирования товаров. Нужно разобраться, как он работает, и добавить в него функцию отмены резерва.

```
const itemPrototype = {  
  hold(amount = 1) {  
    if (this.available < amount) {  
      return false;  
    }  
    this.available -= amount;  
  }  
};
```

```

        this.holded += amount;
        return true;
    },
    toString() {
        return `${this.title} (остаток ${this.available}, в резерве $
{this.holded})`;
    }
};

function createItem(title, amount) {
    const item = Object.create(itemPrototype);
    item.title = title;
    item.available = amount;
    item.holded = 0;
    return item;
}

const items = [];
for (let item of positions) {
    items.push(createItem(item.title, item.available));
}

items[0].hold(2);
items[1].hold(8);
items[1].hold(12);
items[2].hold(1);

for (let item of items) {
    console.log(`Товар ${item}`);
}

```

Реализовать функцию unhold, которая будет отменять резервирование товара. Функция должна принимать количество товара, которое нужно снова сделать доступным. Функция должна вернуть истину, если отменить резервирование удалось, и ложь, если нет. Она должна проверить, есть ли достаточное количество товара в резерве (свойство helded), и если есть, сделать его доступным (уменьшить hold и увеличить available на указанное количество). Если вызвать функцию без аргументов, то должен быть отменен весь резерв.

Пример использования функции

```

console.log(`Товар ${items[0]}`);
// Товар Телепорт бытовой VZНИН-101 (остаток 1, в резерве 2)
items[0].unhold(1);

```

```
console.log(`Товар ${items[0]}`);  
// Товар Телепорт бытовой VZHIH-101 (остаток 2, в резерве 1)  
  
console.log(`Товар ${items[1]}`);  
// Товар Ховерборд Mattel 2016 (остаток 6, в резерве 8)  
items[1].unhold();  
console.log(`Товар ${items[1]}`);  
// Товар Ховерборд Mattel 2016 (остаток 14, в резерве 0)
```

Процесс выполнения

- 1.Скопируйте код, написанный вашим коллегой, в конец вашей домашней работы.
- 2.Убедитесь, что он работает, разберитесь, как он устроен.
- 3.Реализуйте у всех товаров функцию `unhold`, принимающую нужное число аргументов.
- 4.Проверьте, можно ли снять из резерва указанное количество штук. Если нет, верните ложь.
- 5.Пересчитайте остаток и резерв.
- 6.Верните истину.
- 7.Вызовите метод `unhold` у некоторых товаров из массива `items`.
- 8.Выведите все товары из массива `items` в формате:

```
Товар Телепорт бытовой VZHIH-101 (остаток 2, в резерве 1)  
Товар Ховерборд Mattel 2016 (остаток 14, в резерве 0)  
Товар Меч световой FORCE (синий луч) (остаток 0, в резерве 1)
```

Задача № 2. Цена со скидкой

Без скидок торговля совсем не идет. Но наши менеджеры настолько плохо понимают проценты, что иногда выставляют счета, в которых цена со скидкой больше базовой цены. Чтобы нам защититься от них, добавьте в объекты товаров информацию о цене со скидкой.

Описание задачи

Добавить всем объектам из массива `positions` свойство с конечной ценой `finalPrice`, в котором будет базовая цена товара из свойства `price` с учетом скидки из свойства `discount`. При этом при изменении размера скидки или базовой цены конечная цена должна меняться. Если задать новую конечную цену товара, то должна обновиться скидка. Если задать конечную цену, большую, чем базовая цена, должно бросаться исключение с пояснением причины.

Пример использования свойства

```
console.log(positions[0].finalPrice); // 9300
```

```
positions[2].finalPrice = 28500;  
console.log(positions[2].discount); // 50
```

Процесс выполнения

- 1.Перебрать все элементы массива `positions`.
- 2.Для каждого элемента добавить свойство `finalPrice` так, чтобы оно отвечало требованиям. Убедитесь, что пример использования после добавления свойств работает так, как описано в примере. А также проверьте свои варианты использования нового свойства.

Задача № 3. Проверка полей

Наш контент-менеджер создает в день до сотни новых товаров. Но как мы выяснили, он просто не заполняет половину полей в форме. Нужно запретить ему это делать.

Описание функции

Реализовать функцию `isValidPosition`, которая будет принимать объект товара и массив обязательных свойств. Если все свойства в объекте присутствуют, функция должна вернуть истину, иначе — ложь. Аргументы:

- 1.Товар из формы, заполненной пользователем, объект;
- 2.Список обязательных свойств, массив. Функция должна вернуть `true`, если все обязательные поля есть в товаре, иначе вернуть `false`.

Пример использования

```
const requiredFields = [ 'title', 'price', 'discount' ];  
let form1 = {  
  title: 'Товар Телепорт бытовой VZHIH-101',  
  price: 7800,  
  discount: 0  
};  
let form2 = {  
  title: 'Товар Телепорт бытовой VZHIH-101',  
  discount: 10  
}  
  
if ( isValidPosition(form1, requiredFields) ) {  
  console.log('Форма № 1 заполнена верно');  
} else {  
  console.log('В форме № 1 не заполнены необходимые поля');
```

```
}  
  
if ( isValidPosition(form2, requiredFields) ) {  
    console.log('Форма № 2 заполнена верно');  
} else {  
    console.log('В форме № 2 не заполнены необходимые поля');  
}
```

Если функция реализована верно, то получите такой вывод в консоль:

Форма № 1 заполнена верно

В форме № 2 не заполнены необходимые поля

Процесс выполнения

- 1.Создайте функцию `isValidPosition`.
- 2.Убедитесь, что все поля, имена которых переданы в массиве из второго аргумента, есть в объекте из первого аргумента.
- 3.Если это так, верните `true`, иначе верните `false`. После создания функции убедитесь, что код из примера использования даёт такой же результат в вашем случае. Попробуйте проверить свои формы, используя свои обязательные поля.