

Домашнее задание к лекции 2.5 «Прототип и конструктор массива»

Перед началом работы

- 1.Активируйте строгий режим соответствия.
- 2.Добавьте в редактор следующий блок кода:

```
const clients = [{
  name: 'Филип Фрай',
  email: 'fray@mail.un',
  isSubscribed: false,
  orders: [ 11700, 1980, 450, 5500 ]
}, {
  name: 'Бендер Сгибатель Родригес',
  email: 'bender.rodrigues@rambler.un',
  isSubscribed: true,
  orders: [ 440, 226, 7650, 2990, 70 ]
}, {
  name: 'Доктор Джон Зоидберг',
  email: 'zoidberg-md@list.un',
  isSubscribed: true,
  orders: [ 720 ]
}];
```

Задача № 1. Поиск клиентов.

У нас огромная база клиентов. В массиве `clients` представлен её фрагмент для экспериментов. Нам нужно реализовать возможность получить всю информацию о клиенте по его имени.

Описание функции

Реализовать метод `findByName` в массиве `clients`, который будет принимать имя клиента и возвращать объект клиента с таким именем (свойство `name`, строгое соответствие). Метод принимает следующие аргументы:

- 1.`name` — имя клиента, строка.

Функция должна вернуть объект клиента с таким именем, либо `undefined`, если такого клиента нет в нашем списке.

Пример использования функции

```
const clientOne = clients.findByName('Доктор Джон Зоидберг');
```

```
console.log(clientOne.email); // zoidberg-md@list.un

const clientTwo = clients.findByName('Люпп');
console.log(typeof clientTwo); // undefined
```

Процесс реализации

- 1.Создайте свойство в массиве `clients` с именем `findByName`, присвоив туда функцию.
- 2.В теле функции найдите подходящий элемент самым оптимальным и очевидным способом.
- 3.Верните найденный объект или `undefined`.

Убедитесь, что пример использования функции работает так, как описано выше. И попробуйте свои варианты использования созданного метода.

Задача № 2. Сортируем по сумме покупок.

Бывает, что нам нужно отсортировать список клиентов по убыванию сумм их покупок. Все суммы покупок доступны в свойстве `orders`.

Описание функции

Реализовать функцию `compareByTotalSumm`, которая принимает два объекта пользователя и возвращает 1, если второй объект имеет сумму покупок больше первого, 0, если суммы покупок равны, и -1, если второй объект имеет меньшую сумму покупок.

- 1.`left` — клиент из массива `clients`, объект;
- 2.`right` — клиент из массива `clients`, объект.

Возвращает 1, 0 или -1. Для того, чтобы получить общую сумму объекта `left` или `right`, нужно просуммировать все элементы массива в свойстве `orders`.
Функция будет использоваться для передачи в метод `sort` массива.

Пример использования функции

```
clients
  .sort(compareByTotalSumm)
  .forEach(client => console.log(client.name));
```

Если функция `compareByTotalSumm` реализована правильно, то вы увидите такой результат:

```
Филип Фрай
Бендер Сгибатель Родригес
Доктор Джон Зоидберг
```

Процесс реализации

- 1.Создайте функцию `compareByTotalSumm`.
- 2.Посчитайте сумму продаж для клиента из аргумента `left`.

- 3.Посчитайте сумму продаж для клиента из аргумента `right`.
- 4.Сравните их и верните результат.

Задача № 3. Рассылка писем.

Перед началом работы добавьте в редактор следующий код:

```
function sendMail(email) {  
  console.log(`Письмо отправлено на адрес ${email}`);  
}
```

Наш отдел маркетинга рассылает письма с информацией об акциях всем клиентам, которые подписаны на рассылку. У таких клиентов свойство `isSubscribed` равно `true`.

Описание функции

Создать функцию `getSubscribedEmails`, которая принимает список клиентов, получает электронные адреса подписанных пользователей и возвращает их массив. Принимает аргументы:

- 1.`list` — список клиентов, аналогичный массиву `clients`, массив.

Функция должна вернуть массив электронных адресов подписанных клиентов либо пустой массив.

Пример использования функции

```
getSubscribedEmails(clients).forEach(sendMail);
```

Если функция `getSubscribedEmails` реализована верно, то получим такой вывод:

Письмо отправлено на адрес `bender.rodriges@rambler.un`

Письмо отправлено на адрес `zoidberg-md@list.un`

Процесс реализации

- 1.Создайте функцию `getSubscribedEmails`.
- 2.Получите оптимальным и логичным способом все электронные адреса клиентов, у которых `isSubscribed` равно `true`.
- 3.Верните полученный массив электронных адресов.