**Работа с локальным хранилищем**

Для работы с локальным хранилищем предназначен встроенный в браузер объект localStorage. У него есть 4 простых для понимания метода:

//Сохранение значения:

localStorage.setItem('Ключ', 'Значение');

//Получение значения:

Var value**=**localStorage.getItem('Ключ');

//Удалениезначения:

localStorage.removeItem('Ключ');

//Очистка всего хранилища:

localStorage.clear();

С **localStorage** можно также работать как с обычным массивом:

//Сохранение значения:

localStorage['Ключ'] **=**'Значение';

//Получение значения:

Var value**=**localStorage['Ключ'];

//Удаление значения:

Delete localStorage['Ключ'];

Кроме объекта **localStorage** есть также объект **sessionStorage**. Работа с ним осуществляется так же, единственное отличие в том, что все данные из него автоматически уничтожаются после закрытия браузера или вкладки с сайтом. Ну, а localStorage хранит данные длительное время, пока эти данные не будут удалены скриптом, или пользователь браузера очистит локальное хранилище с помощью настроек.

**Примеры**

В следующем примере мы запишем имя пользователя в локальное хранилище:

localStorage.setItem('name', 'Иван');

Через некоторое время получим это имя обратно:

alert(localStorage.getItem('name'));

Как вы видите – ничего сложно здесь нет, все гораздо проще, чем та же работа с куками.

**Сохранение объектов**

Локальное хранилище не способно хранить объекты и массивы JavaScript, хотя это зачастую бывает удобно. Но способ есть - нужно сериализовать эти данные в формат JSON – получится строка, которую уже можно будет сохранить в localStorage. Затем, когда нам понадобится достать этот объект обратно - преобразуем строку из JSON обратно в объект – и спокойно им пользуемся.

Давайте рассмотрим этот процесс на примере. Сериализуем объект и сохраним в локальное хранилище:

//Дан объект:

Var obj**=** {name: 'Иван', arr: [1, 2, 3]};

//Сериализуем его в '{"name": "Иван", "arr": [1, 2, 3]}':

var json **=**JSON.stringify(obj);

//Запишем в localStorage с ключом obj:

localStorage.setItem('obj', json);

Через некоторое время получим объект обратно:

//Получим данные обратно из localStorage в виде JSON:

var json **=**localStorage.getItem('obj');

//Преобразуем их обратно в объект JavaScript:

var obj **=**JSON.parse(json);

console.log(obj);

**Дополнительные возможности**

Определение количества записей в хранилище: alert(localStorage.length).

Определение названия ключа по его номеру: alert(localStorage.key(номер)).

При выполнении операций с хранилищем, срабатывает событие **onstorage**. Если к этому событию привязать функцию - то в ней будет доступен объект Event со следующими свойствами:

<body onstorage="func()">

Function func(event) {

var key**=**event.key; //ключ изменяемых данных

var oldValue**=**event.oldValue; //староезначение

var newValue**=**event.newValue; //новоезначение

var storageArea**=**event.storageArea; //storageArea

}

**Доп. материал**

Хранение массива в локальном хранилище: <https://youtu.be/sYUILPMnrIo>

1. На странице находится textarea. В него вводится текст. Сделайте так, чтобы при следующем заходе на страницу введенные им ранее данные стояли на своих местах.
2. На странице находится textarea. В него можно ввести данные, затем отредактировать их, затем еще отредактировать. Пусть textareaхранит историю изменений (даже после перезагрузки страницы). Над textareaдолжны находится кнопки, с помощью которых можно перемещаться по истории.
3. Дана форма с input иtextarea. Пользователь заполняет некоторые поля и закрывает страницу. Сделайте так, чтобы после закрытия страницы введенные пользователем данные находились на своих местах.
4. Реализуйте записную книгу, хранящую данные в локальном хранилище.
5. Реализуйте органайзер, хранящий данные в локальном хранилище.
6. Реализуйте экзаменационную ведомость (ФИО, дата, №билета, оценка), хранящую данные в локальном хранилище.
7. Реализуйте форму заказа блюд в ресторане (Название, страна кухни, масса, стоимость), хранящую данные в локальном хранилище.
8. Реализуйте форму регистрации жильцов многоквартирного дома (Номер квартиры, ФИО жильца, паспортные данные, дата регистрации), хранящую данные в локальном хранилище.
9. Дана форма, состоящая из нескольких инпутов. Сделайте так, чтобы после закрытия страницы введенные пользователем данные находились на своих местах.
10. На странице находятся несколько объектов input, расположенных в две колонки, в первый столбец вводятся ФИО, во второй –дни рождения. Сделайте так, чтобы при следующем заходе на страницу введенные им ранее данные стояли на своих местах.
11. Реализуйте форму учета кадров (ФИО, пол, код отдела, стаж, дата рождения), хранящую данные в локальном хранилище.
12. Реализуйте форму – справочник организаций (Наименование, отрасль, дата регистрации, ФИО генерального директора, уставной капитал), хранящую данные в локальном хранилище.