Professional Skills

Agile Fundamentals

Jira

Git

Databases Introduction

Java Beginner

Maven

Testing (Foundation)

Java Intermediate

HTML

CSS

Javascript

○ What is JavaScript
○ Getting started with JS
○ Variables
○ Data types
○ ASI
○ Strict mode
○ Iteration
○ Conditionals with Truthy / Falsey
○ Objects, Arrays + JSON
○ Structuring JS Code
○ Destructuring
○ Scope
○ Functions, function expressions and arrow functions
○ The ECMAScript 6 Specification
○ OOP in JavaScript
○ Best Practices
○ Closures
○ Callbacks and Promises
○ Cookies
○ Hoisting
○ Prototypes
○ Query Parameters
○ Higher Order Functions

Web Storage

Contents

Overview

The **Web Storage API** provides mechanisms through which browsers can store key/value pairs in a much more intuitive fashion than using cookies.

Tutorial

With Web storage, Web applications can store data locally within the user's browser.
Two mechanisms within Web storage are as follows:

- `sessionStorage` maintains a separate storage area for each given origin that is available for the duration of the page session (as long as the browser is open, including page reloads and restores)
  - Stores data only for a session, meaning that the data is stored until the browser (or tab) is closed.
  - Data is never transferred to the server.
  - Storage limit is larger than a cookie (at most 5MB).
- `localStorage` does the same thing, but persists even when the browser is closed and reopened.
  - Stores data with no expiration dates, and only gets cleared through JavaScript, or clearing the browser cache / locally-stored data.
  - Storage limit is the maximum amongst the three.

These mechanisms are available via the `Window.sessionStorage` and `Window.localStorage` properties - invoking one of these will create an instance of the `Storage` object, through which data items can be set, retrieved and removed.

The `localStorage` Object

The `localStorage` object stores the data with no expiration date. The data will not be deleted when the browser is closed, and will be available the next day, week or year.

Here a `localStorage` name/value pair with the `name="myCat"` and `value="Tom"` is printed to the screen:

```
//Store
localStorage.setItem("myCat", "Tom");
//Print to the console
console.log(localStorage.getItem("myCat"));
```

To remove an item from `localStorage` we simply do the following:

```
localStorage.removeItem("myCat");
```

(note: name value pairs are always stored as Strings. Remember to convert them to another format when needed.)

○ Web Storage
○ DOM Manipulation
○ Handling Events and Timed Events
○ Asynchronous Programming
○ HTTP-Requests
○ XMLHttpRequests
○ Fetch API

Spring Boot

Selenium

Sonarqube

Advanced Testing (Theory)

Cucumber

MongoDB

Express

NodeJS

React

Express-Testing

Networking

Security

Cloud Fundamentals

AWS Foundations

AWS Intermediate

Linux

DevOps

Jenkins Introduction

Jenkins Pipeline

Markdown

IDE Cheatsheet

The `sessionStorage` Object

The `sessionStorage` object is equal to the `localStorage` object, **except** that it stores the data for only one session:

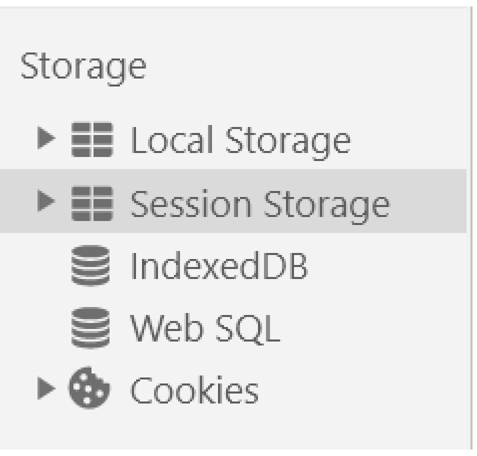
- A page session lasts as long as the browser is open, and survives over page reloads and restores.
- **Opening a page in a new tab or window creates a new session with the value of the top-level browsing context, which differs from how session cookies work.**
- Opening multiple tabs/windows with the same URL creates `sessionStorage` for each tab/window.
- Closing a tab/window ends the session and clears objects in the `sessionStorage`.

```
// Save data to sessionStorage
sessionStorage.setItem("key","value");

//Get saved data from sessionStorage
let data = sessionStorage.getItem("key");

// Remove saved data from sessionStorage
sessionStorage.removeItem("key");

//Remove all saved data from sessionStorage
sessionStorage.clear();
```

Storage Object in the Browser

To check what values you have stored inside your browser, in both the `localStorage` and `sessionStorage`:

1. First open your `.html` document in your favourite browser
2. Open the browser's developer tools
3. Navigate to the 'Application' tab
4. Then in the navigation pane you will be able to see your Storage Objects



Exercises

1. Create a function that counts the number of times a user clicks on a button, store it in `localStorage` and display on the Web page.

   ▶ Solution

2. Implement a function that auto-saves the contents of a text field, and if the browser is refreshed, restores the text field content so that no writing is lost.

   ▶ Solution