Member-only story

Session Storage vs. Local Storage in JavaScript with Examples

In modern web development, the ability to store data on the client side has become an essential feature.

Nil Lenon · Follow

3 min read · May 23

Listen Share More

local storage. Both options offer distinct advantages and use cases. In this article, we will explore the differences between session storage and local storage in JavaScript and provide examples to help you understand when to use each one.

Introduction to Session Storage and Local Storage

Session storage and local storage are part of the Web Storage API, which allows developers to store key-value pairs on the client side. These mechanisms provide a way to persist data across page refreshes, allowing web applications to store and retrieve information efficiently.

Session Storage: Session storage is designed to store data for a specific session. A session refers to the duration of a user's interaction with a website. When the user closes the browser or tab, the session storage is cleared. This means that session storage data is temporary and lasts only as long as the session is active.

Local Storage: Local storage, on the other hand, is intended for persistent storage. The data stored in local storage remains available even after closing and reopening the browser. It is not tied to a specific session, making it suitable for scenarios where you want to store data for a longer period, such as user preferences or application settings.

Differences between Session Storage and Local Storage

1. **Scope**: Session storage is scoped to a particular tab or window. Each tab or window maintains its own session storage, and the data stored in one tab is not accessible in another. Local storage, on the other hand, has a global scope. Data stored in local storage is accessible across different tabs and windows of the same origin.

2. **Persistence**: As mentioned earlier, session storage is temporary and cleared when the session ends. On the other hand, local storage is persistent and remains available even after closing and reopening the browser.

3. **Storage Limit:** The amount of data that can be stored in session storage and local storage varies. Session storage typically has a smaller storage limit, usually around 5MB, while local storage offers a larger storage capacity, often around 10MB or more. It's important to note that these limits may vary across different browsers.

Examples

Let's consider a scenario where we want to store a user's name using both session storage and local storage.

Using Session Storage:

```
// Store data in session storage
sessionStorage.setItem('name', 'John');

// Retrieve data from session storage
const storedName = sessionStorage.getItem('name');
console.log(storedName); // Output: John
```

```
// Clear session storage
sessionStorage.clear();
```

Using Local Storage:

```
// Store data in local storage
localStorage.setItem(‘name’, ‘John’);

// Retrieve data from local storage
const storedName = localStorage.getItem(‘name’);
console.log(storedName);
// Output: John

// Clear local storage
localStorage.clear();
```

In the above examples, we store the user’s name in both session storage and local storage using the **setItem()** method. We retrieve the stored name using the **getItem()** method, and finally, we clear the storage using the **clear()** method.

Session storage and local storage are powerful tools in JavaScript for client-side data storage. Session storage is suitable for storing temporary data within a specific session, while local storage is ideal for persistently storing data across sessions. By understanding their differences and use cases, you can leverage these mechanisms to enhance the functionality and user experience of your web applications.

P.S. *I’ve earned over $1,000 in the last 3months. If you’d also like a chance to earn money writing on Medium, we suggest **signing up as a m**ember. It only costs $5/month. That way, not only you’ll get a chance to write and earn, but you’ll also get unlimited access to all the stories like this one. The story above has been written with the help of AI tools.*

JavaScript Sessionstorage Javascript Tips Programming

Software Development

Follow

Written by Nil Lenon

2.5K Followers

A software specialist during the day and a side hustler during night. Writing about code, IT products, personal development and career tips.

More from Nil Lenon

Nil Lenon in Your Software Development Stories

The Future of DevOps: Trends and Predictions for 2023

The world of software development is ever-evolving, and DevOps has become a critical element of modern software development practices...

✦ · 4 min read · Feb 23

79

Nil Lenon in Level Up Coding

My Journey to Becoming a Chief Technology Officer (CTO)

Becoming a Chief Technology Officer (CTO) is a considerable achievement in one's profession, representing know-how, leadership, and...

✦ · 4 min read · May 24

96

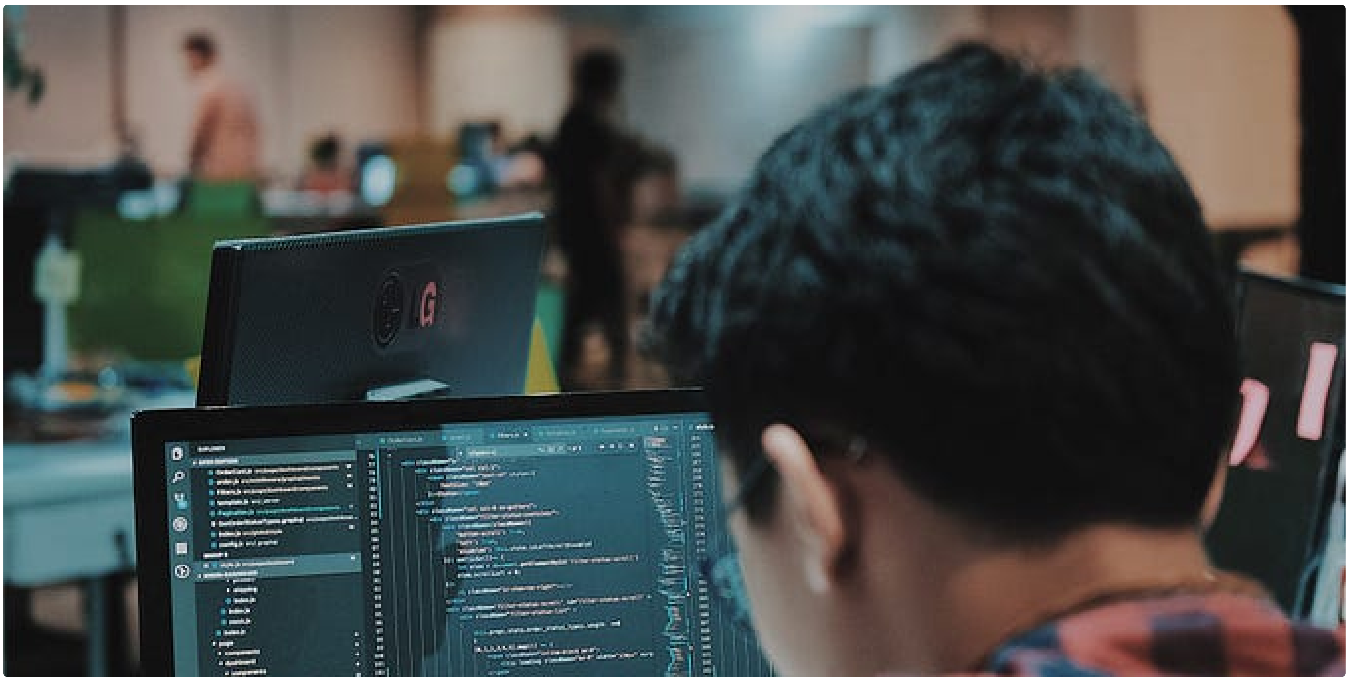
Nil Lenon in Your Software Development Stories

How To Make Money from StackOverflow

Stack Overflow is one of the most popular question-and-answer websites for software developers.

✦ · 4 min read · Feb 21

10



Nil Lenon in Level Up Coding

What I am Avoiding at Work as a Senior Software Developer

Being a senior software developer comes with its fair share of responsibilities and challenges.

✦ · 4 min read · Jun 5

91 2

See all from Nil Lenon