LOCAL STORAGE AND SESSION STORAGE

Interviewquestion-57

JS

Follow on in

@Duvvuru Kishore

what we will learn from here?

Local Storage

- what is local storage ?
- How local storage is useful?
- what kind of work needs local storage?

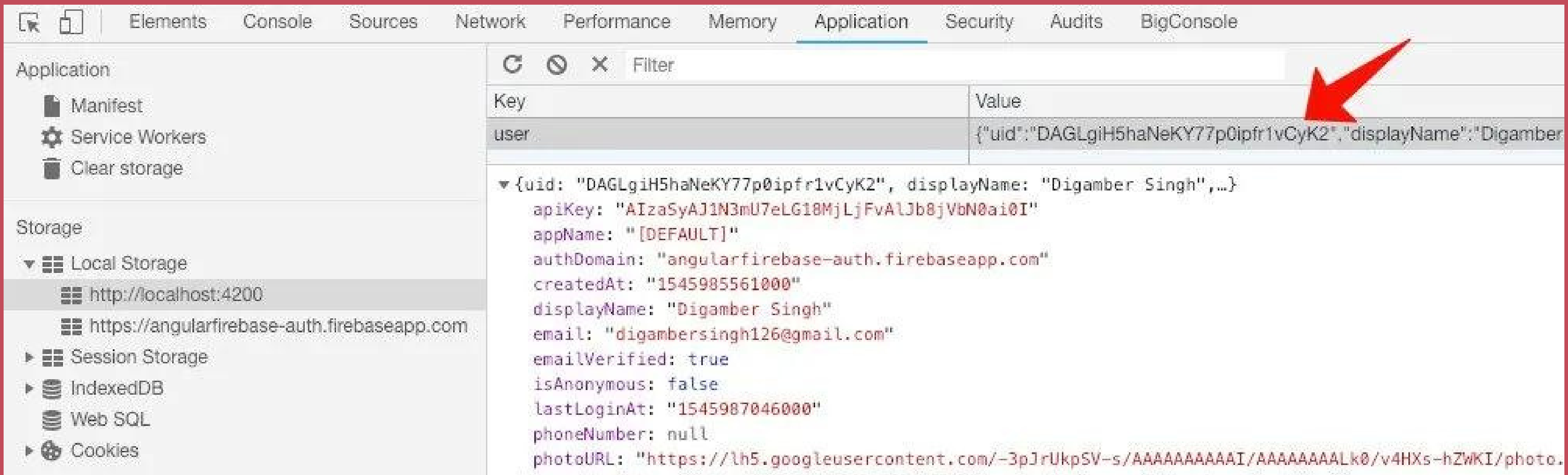
Session Storage

- what is session storage ?
- How session storage is useful?
- what kind of work needs session storage?

Local Storage

- LocalStorage is a web storage technology that allows you to store key-value pairs in a web browser with no expiration time.

- Data stored in LocalStorage remains available even after the browser is closed and can be accessed from the same origin



Key	Value
user	{"uid":"DAGLgiH5haNeKY77p0ipfr1vCyK2","displayName":"Digamber

```
▾{uid: "DAGLgiH5haNeKY77p0ipfr1vCyK2", displayName: "Digamber Singh",…}
   apiKey: "AIzaSyAJ1N3mU7eLG18MjLjFvAlJb8jVbN0ai0I"
   appName: "[DEFAULT]"
   authDomain: "angularfirebase-auth.firebaseapp.com"
   createdAt: "1545985561000"
   displayName: "Digamber Singh"
   email: "digambersingh126@gmail.com"
   emailVerified: true
   isAnonymous: false
   lastLoginAt: "1545987046000"
   phoneNumber: null
   photoURL: "https://lh5.googleusercontent.com/-3pJrUkpSV-s/AAAAAAAAAAI/AAAAAAAALk0/v4HXs-hZWKI/photo.
```

How local storage is useful?

- LocalStorage allows you to store data on the client-side (in the user's browser).

- LocalStorage data is accessible to all tabs or windows opened from the same domain. This allows you to share data between different parts of your web application

- LocalStorage is synchronous and relatively fast.

- Data stored in LocalStorage has no expiration date by default.

- Reduced Server Load:By storing data locally on the user's device,

what kind of work needs local storage?

Form Data:

Imagine you are filling the form of a bank and suddenly you opened other application for replying a important message. and you are back to form and still sees the data. How that’s possible , local storage make that possible.

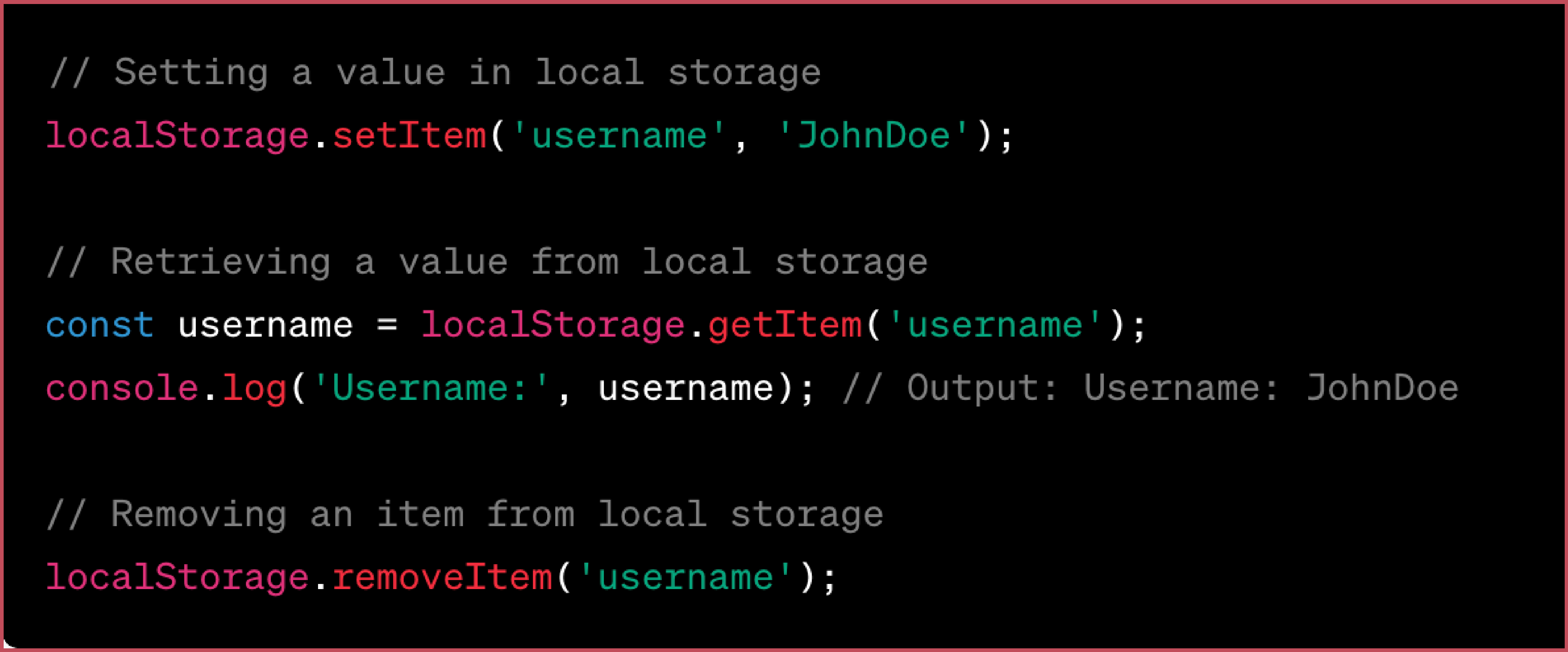
Shopping Carts:

Maintaining the contents of a shopping cart between visits, allowing users to continue shopping where they left off.

Persistent Tokens: **Save long-term tokens, such as access tokens or refresh tokens, for use in OAuth or similar authentication flows. This can help maintain user sessions and authentication across browser restarts.**

List goes according to requirment

Code

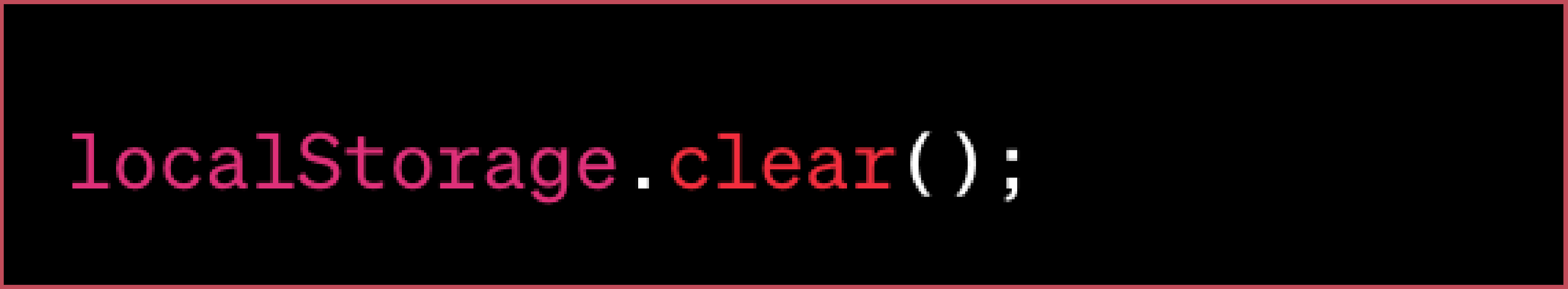


```javascript
// Setting a value in local storage
localStorage.setItem('username', 'JohnDoe');

// Retrieving a value from local storage
const username = localStorage.getItem('username');
console.log('Username:', username); // Output: Username: JohnDoe

// Removing an item from local storage
localStorage.removeItem('username');
```

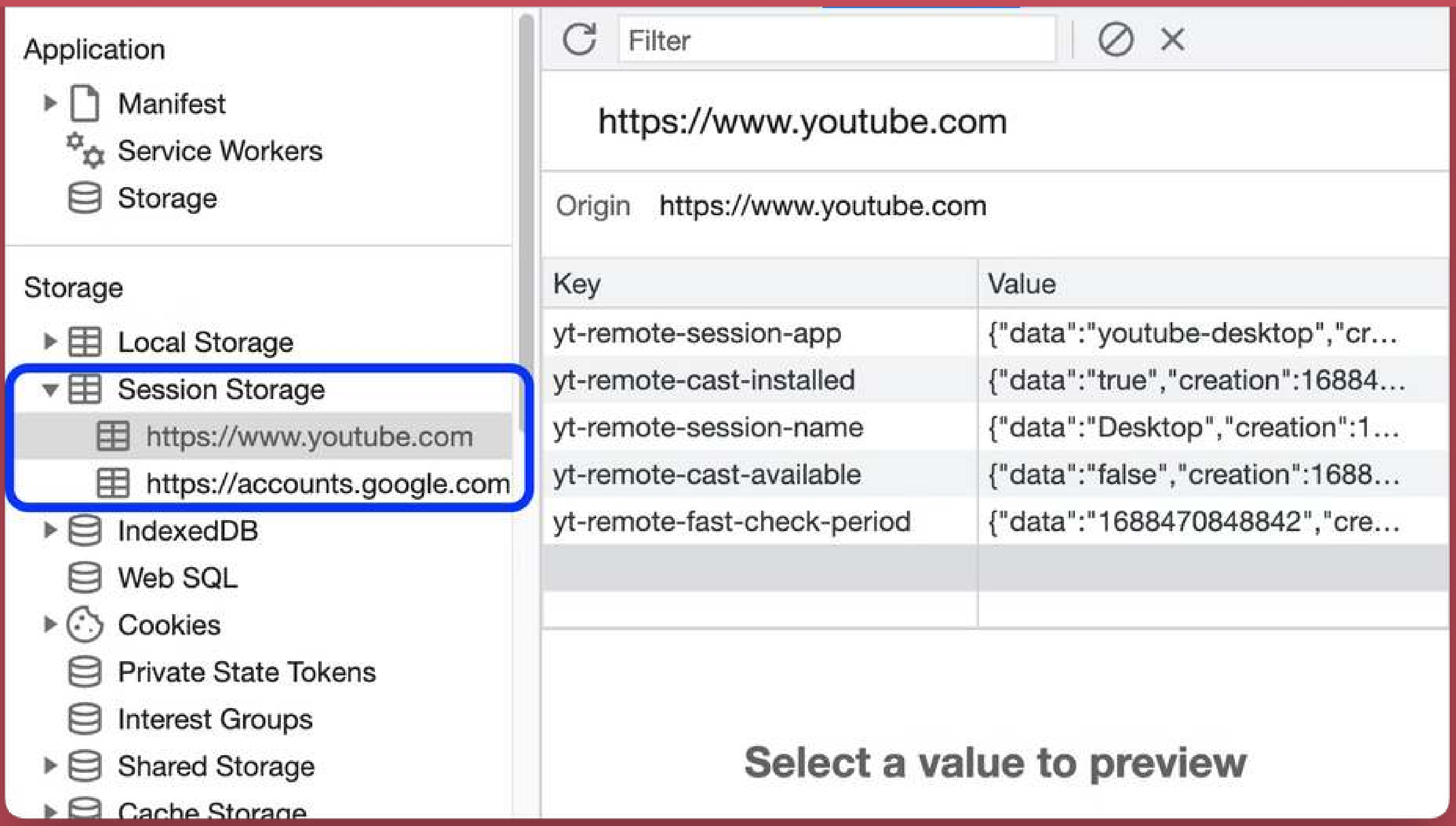
To clear localstorage



```javascript
localStorage.clear();
```

Session storage

- Session Storage is similar to Local Storage, but it has a shorter lifespan.
- Data stored in **Session Storage is only available for the duration of the page session**. Once the session ends (when you close the browser or tab), the data is deleted.



https://www.youtube.com

Origin https://www.youtube.com

Key	Value
yt-remote-session-app	{"data":"youtube-desktop","cr…
yt-remote-cast-installed	{"data":"true","creation":16884…
yt-remote-session-name	{"data":"Desktop","creation":1…
yt-remote-cast-available	{"data":"false","creation":1688…
yt-remote-fast-check-period	{"data":"1688470848842","cre…

Select a value to preview

How session storage is useful ?

- Session storage is designed for temporary data storage that should be available for the duration of a page session
- Session storage provides a way to persist data temporarily, even if the page is refreshed.This can be handy for saving user preferences, form data, or application state temporarily.
- Because session storage is client-side, data retrieval and storage are relatively fast compared to making network requests to a server

what kind of work needs session storage?

User Authentication:

You can store user authentication tokens, such as JSON Web Tokens (JWTs), in session storage after a user logs in. This allows the application to keep the user authenticated throughout their session without requiring repeated logins.

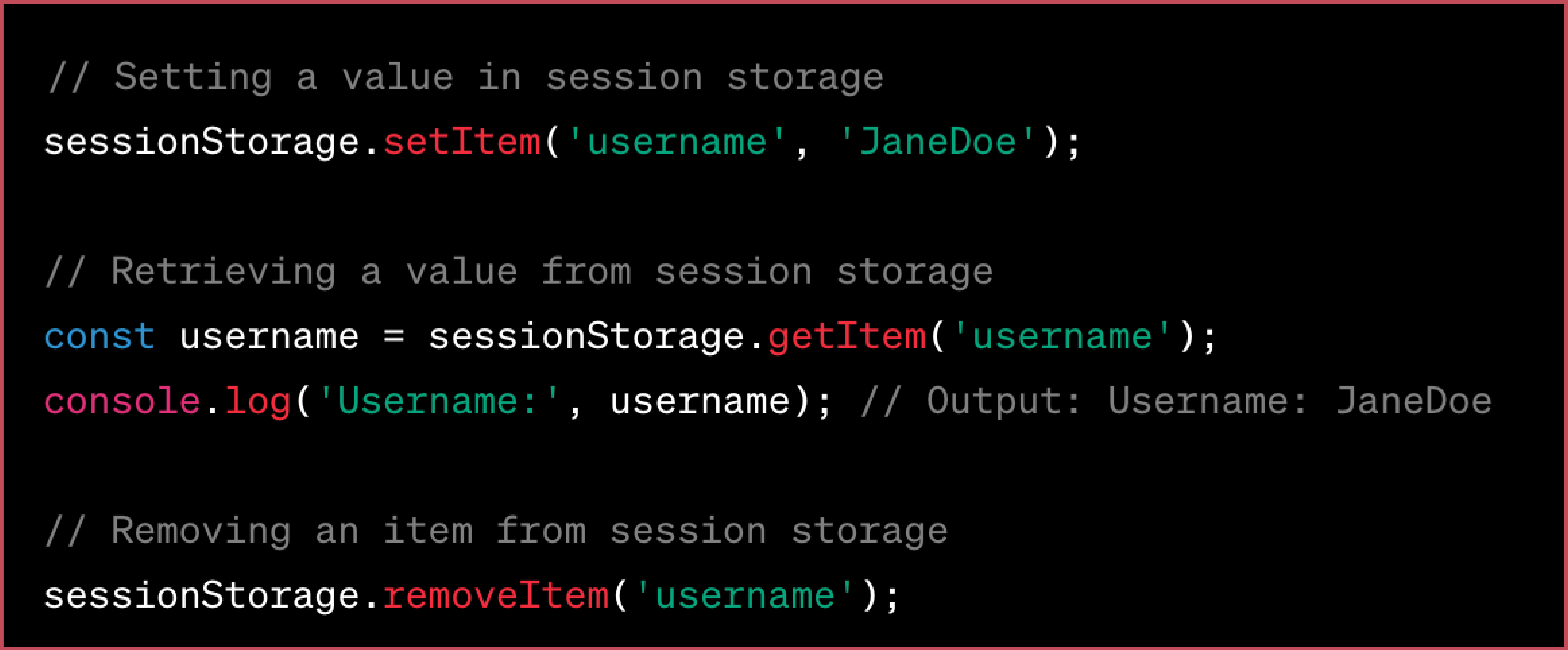
Form Data:

When users fill out a multi-step form, you can store their input data in session storage, ensuring that the data is not lost if they accidentally refresh the page or navigate to a different section of the form.

Shopping Carts: For e-commerce websites, session storage can be used to store the contents of a user's shopping cart. This ensures that items added to the cart remain available during the shopping session, even if the user navigates to other pages.

key difference is local storage will not expire but session storage expires on tab close

Code

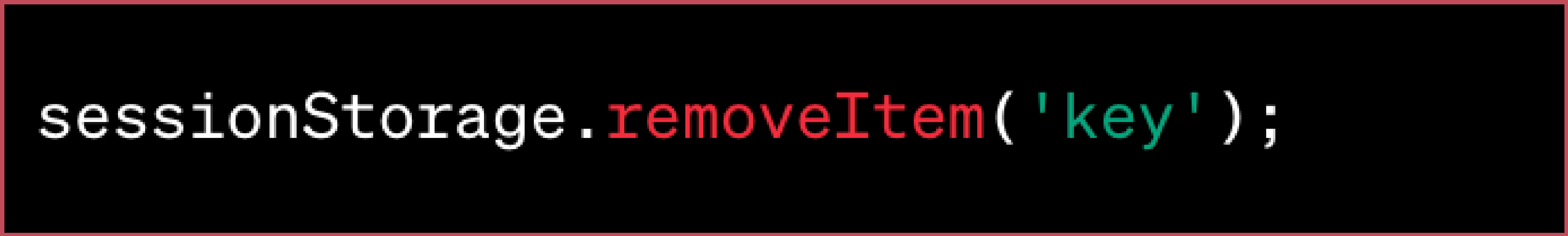


```
// Setting a value in session storage
sessionStorage.setItem('username', 'JaneDoe');

// Retrieving a value from session storage
const username = sessionStorage.getItem('username');
console.log('Username:', username); // Output: Username: JaneDoe

// Removing an item from session storage
sessionStorage.removeItem('username');
```

To clear session storage



```
sessionStorage.removeItem('key');
```

Follow on in

@Duvvuru Kishore