

GABRIELE ROMANATO

Gestione dei cookie in JavaScript

I cookie sono piccoli file di testo che i siti web memorizzano nel browser dell'utente per salvare informazioni come preferenze, sessioni di accesso e tracciamento delle attività. In JavaScript, è possibile leggere, scrivere e cancellare i cookie utilizzando la proprietà `document.cookie`.

Come Creare un Cookie

Per creare un cookie in JavaScript, basta assegnare una stringa alla proprietà `document.cookie`:

```
document.cookie = "username=JohnDoe; expires=Fri, 31 Dec 2025 23:59:59 GMT; path="/;
```

Questa riga di codice crea un cookie chiamato `username` con il valore `JohnDoe`, che scadrà alla data specificata e sarà accessibile in tutto il sito (`path=/`).

Come Leggere i Cookie

Per ottenere i cookie memorizzati nel browser, si utilizza nuovamente `document.cookie`:

```
console.log(document.cookie);
```

Il valore restituito è una stringa con tutti i cookie disponibili per il dominio corrente, separati da punto e virgola.

Come Modificare un Cookie

Per modificare un cookie, basta riscriverlo con lo stesso nome ma con un nuovo valore:

```
document.cookie = "username=JaneDoe; expires=Fri, 31 Dec 2025 23:59:59 GMT; path="/;
```

Il cookie `username` avrà ora il valore aggiornato.

Come Eliminare un Cookie

Per eliminare un cookie, si imposta la sua data di scadenza a una data passata:

```
document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path="/;
```

Questo metodo invalida il cookie e lo rimuove dal browser.

Limitazioni e Considerazioni

- I cookie hanno una dimensione massima di circa 4 KB.
- Possono essere letti solo dal dominio che li ha impostati (same-origin policy).
- Non sono il metodo più sicuro per memorizzare informazioni sensibili.

Utilizzo della API `document.cookie` con JavaScript

Un esempio di funzione per gestire i cookie in modo più comodo:

```
function setCookie(name, value, days) {
  let expires = "";
  if (days) {
    let date = new Date();
    date.setTime(date.getTime() + (days * 24 * 60 * 60 * 1000));
    expires = "; expires=" + date.toUTCString();
  }
  document.cookie = name + "=" + value + expires + "; path=/";
}

function getCookie(name) {
  let nameEQ = name + "=";
  let ca = document.cookie.split(';');
  for(let i = 0; i < ca.length; i++) {
    let c = ca[i];
    while (c.charAt(0) == ' ') c = c.substring(1, c.length);
    if (c.indexOf(nameEQ) == 0) return c.substring(nameEQ.length, c.length);
  }
  return null;
}

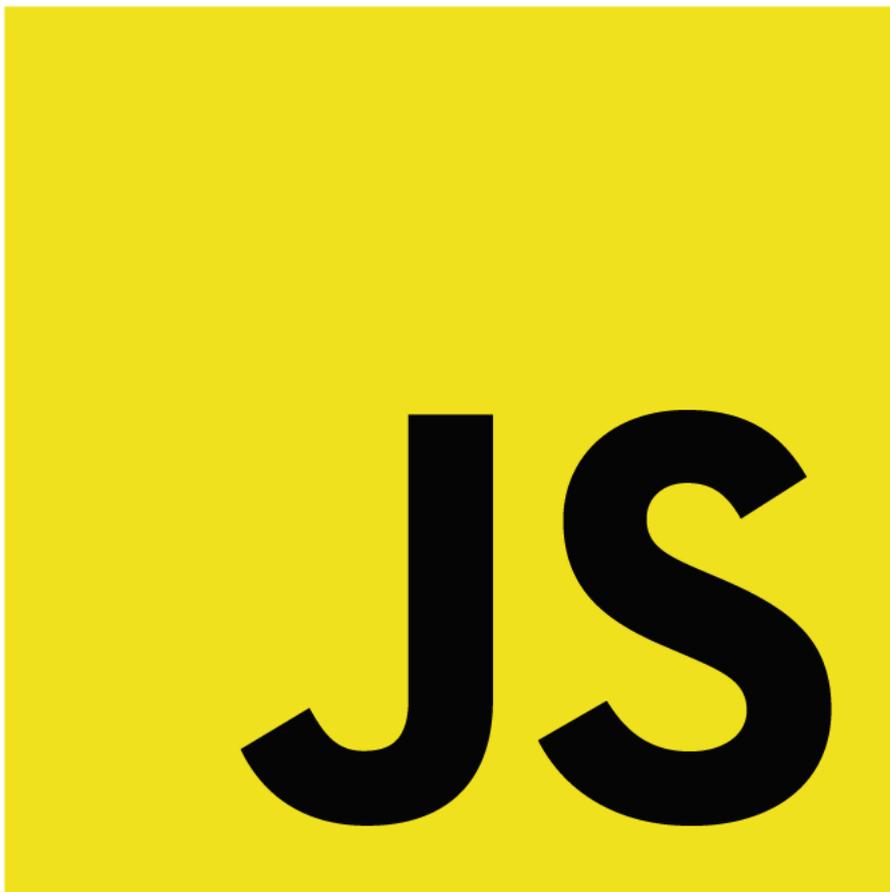
function deleteCookie(name) {
  document.cookie = name + "=" + "; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/";
}
```

Queste funzioni permettono di impostare, ottenere e cancellare un cookie in modo più strutturato.

Conclusione

I cookie sono un potente strumento per la gestione delle informazioni lato client, ma devono essere usati con attenzione per garantire la sicurezza e la privacy degli utenti. È consigliato l'uso di tecniche più sicure come LocalStorage o sessionStorage quando i dati non devono essere inviati al server.

Applicazioni Correlate

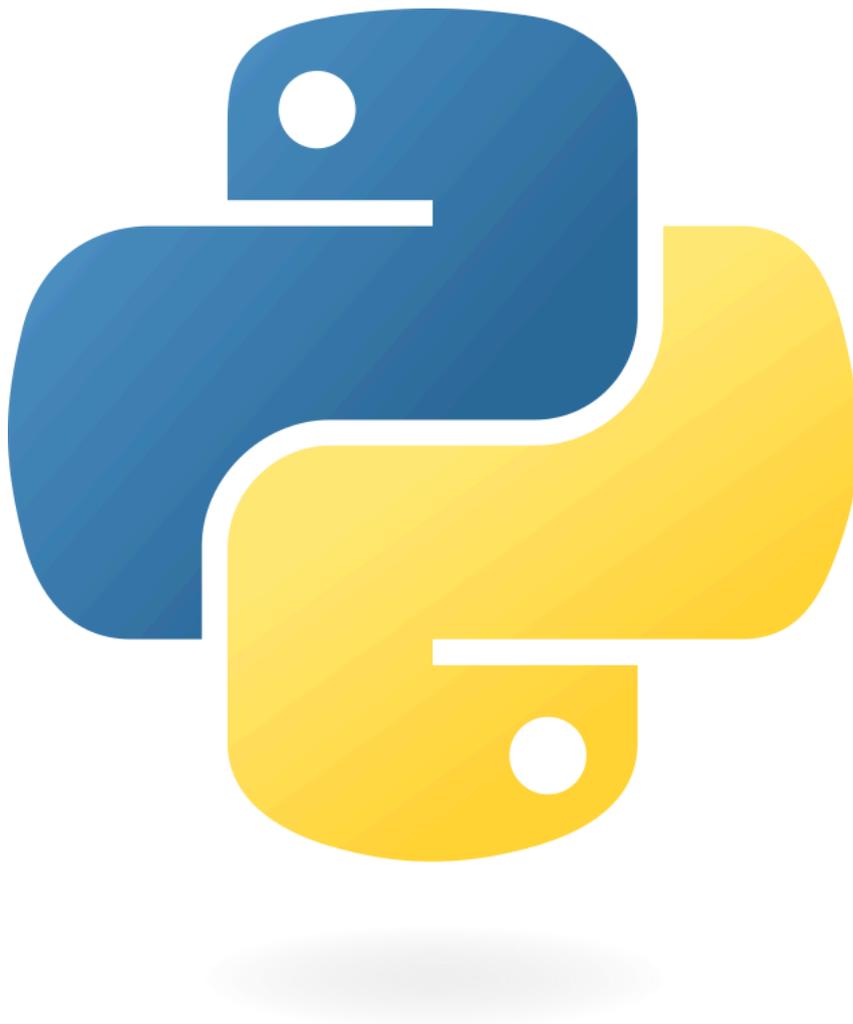


-

JavaScript Password Mask

Un esempio in JavaScript di mascheramento di una password con l'aggiunta della funzionalità di copia negli appunti.

Docker Docker Compose JavaScript



-

Python Placeholder Image

Applicazione sviluppata in Python con Flask per la creazione di immagini segnaposto.
Docker Docker Compose Python Flask JavaScript



-

Go Placeholder Image

Applicazione in Go per la creazione di immagini segnaposto.

Docker Docker Compose Go JavaScript