

Você sabe a diferença entre **Session Storage**, **Local Storage** e **Cookies**?



Helton Quintães
@helton_quintans

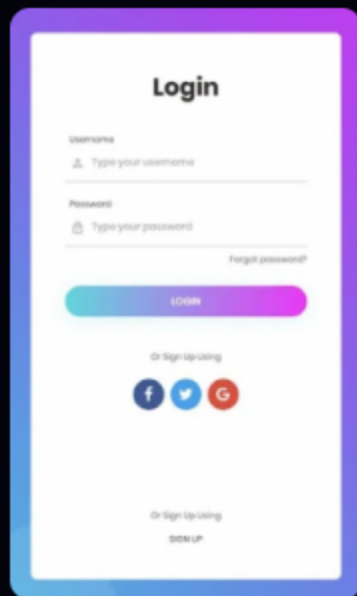


Salve este post

Contextualizando

Imagina que você está fazendo uma aplicação que precisa de **autenticação dos usuários**.

Quando o usuário faz o login o back end vai retornar **dados valiosos** desse usuário.



Helton Quintães

@helton_quintans



Salve este post

Contextualizando

A sua aplicação pode ter **várias páginas** e **vários tipos de usuários**. Os usuários podem ter **permissions** (ex: Admin / End user) e **roles** (ex: criar/deletar usuários)

```
async function signIn({ email, password }: SignInCredentials) {  
  try {  
    const response = await api.post('sessions', {  
      email,  
      password,  
    })  
  
    const { token, refreshToken, permissions, roles } = response.data;
```



Helton Quintães

@helton_quintans



Salve este post

Contextualizando



Um usuário pode **não ter acesso a todas as páginas** do sistema, ou ele pode listar usuários mas não pode deletá-los, por exemplo.

Por isso é muito importante que alguns **dados** do usuário **persistam** e nós podemos usar os browsers pra isso.



Helton Quintães

@helton_quintans



Salve este post

E agora, José?

Agora com esses dados na palma da mão, **onde** você pode persistir esses dados?

```
const { token, refreshToken, permissions, roles } = response.data;
```

SessionStorage

LocalStorage ou nos

Cookies?



Helton Quintães

@helton_quintans



Salve este post

A Resposta é: Depende

*pronto, virei Sr. kkk



Helton Quintães

@helton_quintans



Salve este post

Session Storage

Os dados armazenados aqui ficam **disponíveis durante uma sessão**, ou seja, se o usuário **fechar o navegador** perde-se os dados armazenados.

*perdeu, playboy!

Storage

- ▶  Local Storage
- ▶  Session Storage
- ▶  IndexedDB
-  Web SQL
- ▶  Cookies
-  Trust Tokens
-  Interest Groups



Helton Quintães

@helton_quintans



Salve este post

Local Storage

Os dados armazenados no localStorage **persistem até serem excluídos explicitamente**, Ou seja, ficam disponíveis para todas as visitas atuais e futuras ao site.

O tamanho disponível é de **5 MB**, o que representa consideravelmente **mais espaço** para trabalhar do que um cookie típico de 4KB.

Storage

- ▶ Local Storage
- ▶ Session Storage
 - IndexedDB
 - Web SQL
- ▶ Cookies
 - Trust Tokens
 - Interest Groups



Helton Quintães

@helton_quintans



Salve este post

Local Storage

É importante saber que
pelo lado do servidor não
se tem acesso ao
localStorage

É nesta situação que os
cookies vem salvar a
gente...



Helton Quintães

@helton_quintans



Salve este post

Cookies

Estrutura mais antiga que se tem dentro dos browsers. Podemos definir o tempo de expiração para cada cookie.

Storage

- ▶ Local Storage
- ▶ Session Storage
- IndexedDB
- Web SQL
- ▶ Cookies
- Trust Tokens
- Interest Groups

Para suportar a maioria dos navegadores, mantenha o nome abaixo de 4000 bytes e o tamanho geral do cookie abaixo de 4093 bytes.



Helton Quintães

@helton_quintans



Salve este post

e Pasmem...

Os **cookies** podem ser acessados tanto pelo **browser** quanto pelo lado do **servidor**

E isso faz todo sentido quando, por exemplo, usamos o framework **Next.js**, que nos dá a possibilidade de usar o **SSR** (Server Side Redenring)



Helton Quintães

@helton_quintans





@helton_quintans

**Gostou?
segue para ver mais!**