

## WEB STORAGE

A API Web Storage fornece mecanismos para armazenar informações na memória dos navegadores no formato de pares chave/valor.

## SESSION STORAGE

O primeiro mecanismo é o **sessionStorage** que mantém uma área de armazenamento separada para cada origem disponível durante a sessão da página, ou seja, enquanto o navegador estiver aberto, os dados são mantidos mesmo após refresh ou reabertura da aba.

- Armazena dados apenas para uma sessão, o que significa que os dados são armazenados até que o navegador (ou guia) seja fechado.
- Os dados nunca são transferidos para o servidor.
- \*O limite de armazenamento é de até 5 MB.

**Caminho** para acessar os dados gravados na session storage é

Abrir o console (F12), procurar por **APLICATIVO** e após, **armazenamento de sessão**

**OBS:** para utilizar é necessário usar **window.sessionStorage** e o método desejado (os métodos a serem utilizados estão no fim da página)

## LOCAL STORAGE

O **localStorage** faz a mesma coisa, mas persiste mesmo quando o navegador é fechado e reaberto.

- Armazena dados sem data de expiração e é limpo apenas por meio de JavaScript ou limpando o cache do navegador/dados armazenados localmente.
- \*O limite de armazenamento é de até 10MB.

Ambos utilizam uma interface simples de chave-valor para armazenar e recuperar dados.

**Caminho** para acessar os dados gravados na local storage é

Abrir o console (F12), procurar por **APLICATIVO** e após, **armazenamento Local**

**OBS:** para utilizar é necessário usar **window.localStorage** e o método desejado (os métodos a serem utilizados estão no fim da página)

## OS PRINCIPAIS MÉTODOS PARA UTILIZAR TANTO O LOCAL STORAGE QUANDO O SESSION STORAGE SÃO:

- `setItem()`: Este método permite armazenar um valor associado a uma chave especificada.
- `getItem()`: Este método permite recuperar o valor associado a uma chave especificada.
- `removeItem()`: Este método permite remover o valor associado a uma chave especificada.
- `clear()`: Este método permite remover todos os valores armazenados no `sessionStorage` ou `localStorage`.

```
localStorage.setItem('chave', 'valor')  
sessionStorage.getItem('chave')  
localStorage.removeItem('chave')  
sessionStorage.clear()
```