

Diego Fernandes Martinez - 2350947

## Web Storage

Permite o armazenamento local de informações definidas por aplicações web no navegador.

### - LocalStorage ou SessionStorage

O LocalStorage armazena os dados mesmo ao fechar o browser, e não tem data de expiração.

O SessionStorage armazena os dados na aba, e persistem se a página é recarregada, mas ao fechar a aba do navegador eles são perdidos.

No projeto foram utilizadas em quatro funções, três sendo LocalStorage e uma SessionStorage.

### - Leitura e escrita de dados simples

O WebStorage armazena os dados com uma chave e um valor, ambos do tipo string. Dados simples sendo valores únicos e não complexos como JSON.

No index foram armazenados a quantidade de acessos na página no LocalStorage, e depois foi inserido no HTML.

```
27 (function () {  
28     if (localStorage.accessCounter) {  
29         localStorage.accessCounter = parseInt(localStorage.accessCounter) + 1;  
30         $('#accessCounter').insertAdjacentText('beforeend', `${localStorage.accessCounter}`);  
31     } else {  
32         localStorage.accessCounter = 1;  
33     }  
34 })();
```

Também foram armazenados no SessionStorage a quantidade de acessos da sessão em particular.

```
36 (function () {  
37     if (sessionStorage.reloadCounter) {  
38         sessionStorage.reloadCounter = parseInt(sessionStorage.reloadCounter) + 1;  
39         $('#reloadCounter').insertAdjacentText('beforeend', `${sessionStorage.reloadCounter}`);  
40     } else {  
41         sessionStorage.reloadCounter = 0;  
42     }  
43 })();
```

### - Leitura e escrita de JSON

JavaScript Object Notation é um formato usado para o armazenamento de dados, independente do JavaScript. Aceita outros valores além da string, numéricos, boolean, array, null e outros JSON.

Na página de login foi utilizado o JSON para armazenar os usuários e as tentativas de senha. Conforme o script, os valores vão sendo inseridos em um array de um data type, e só são armazenados no LocalStorage quando a senha correta é enviada.

```
4  const users = [];
```

```
11  if (x === 13) {
12      // user declarado como var na 1ª avaliação
13      const username = document.getElementById('name').value;
14
15      // pss declarado como var na 1ª avaliação
16      const pss = document.getElementById('password').value;
17
18      let user = {
19          name: username,
20          triedPassword: pss
21      };
22      users.push(user);
23
24      if (pss === '12345') {
25          console.log(users);
26          alert(`Bem vindo(a) de volta ${username}`);
27          localStorage.setItem('users', JSON.stringify(users));
28      }
29  }
```

O nome que foi enviado junto com a senha correta (12345) é impresso na página index.

```
18  (function () {
19      if (localStorage.users) {
20          let arrayUsers = JSON.parse(localStorage.users);
21          $$('loggedUser').insertAdjacentText('beforeend',
22              `${arrayUsers[arrayUsers.length - 1].name}`);
23      } else {
24          $$('loggedUser').insertAdjacentText('beforeend', 'Nenhum usuário logado');
25      }
26  })();
```

```
177  <ul class="collection with-header">
178      <li class="collection-header"><h5>Estatísticas</h5></li>
179      <li id="loggedUser" class="collection-item">Usuário logado: </li>
180      <li id="accessCounter" class="collection-item">Acessos à esta página: </li>
181      <li id="reloadCounter" class="collection-item">Recarregamentos da página (nesta sessão): </li>
182  </ul>
```