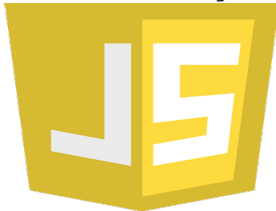


## Web Storage : `localStorage` et `sessionStorage`

**Achref El Mouelhi**

Docteur de l'université d'Aix-Marseille  
Chercheur en Programmation par contrainte (IA)  
Ingénieur en Génie logiciel

`elmouelhi.achref@gmail.com`



# localStorage et sessionStorage

## localStorage et sessionStorage

- deux propriétés permettant d'accéder respectivement aux deux objets `Storage` et `Session`
- `localStorage` : stocke les données sans date d'expiration
- `sessionStorage` : stocke les données pour une connexion (les données sont perdues lorsque l'onglet du navigateur est fermé)

# localStorage et sessionStorage

## localStorage et sessionStorage

- deux propriétés permettant d'accéder respectivement aux deux objets `Storage` et `Session`
- `localStorage` : stocke les données sans date d'expiration
- `sessionStorage` : stocke les données pour une connexion (les données sont perdues lorsque l'onglet du navigateur est fermé)

## Dans les exemples

- Que `localStorage` sera considéré
- Même syntaxe pour `sessionStorage`

# localStorage et sessionStorage

## Pour créer ou mettre à jour une variable session

```
localStorage.setItem('nom', 'wick');
```

# localStorage et sessionStorage

## Pour créer ou mettre à jour une variable session

```
localStorage.setItem('nom', 'wick');
```

## On peut aussi faire

```
localStorage['nom'] = 'wick';
```

# localStorage **et** sessionStorage

## Pour créer ou mettre à jour une variable session

```
localStorage.setItem('nom', 'wick');
```

## On peut aussi faire

```
localStorage['nom'] = 'wick';
```

## Ou encore

```
localStorage.nom = 'wick';
```

# localStorage et sessionStorage

**Pour récupérer une variable de** localStorage

```
let nom = localStorage.getItem('nom');
```

# localStorage et sessionStorage

**Pour récupérer une variable de** localStorage

```
let nom = localStorage.getItem('nom');
```

**Pour supprimer une variable de** localStorage

```
localStorage.removeItem('nom');
```



# localStorage et sessionStorage

**Pour récupérer une variable de** localStorage

```
let nom = localStorage.getItem('nom');
```

**Pour supprimer une variable de** localStorage

```
localStorage.removeItem('nom');
```

**Pour vider** localStorage

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

# localStorage et sessionStorage

**Pour connaître le nombre de variables session**

```
localStorage.length
```

# localStorage et sessionStorage

## Pour connaître le nombre de variables session

```
localStorage.length
```

## Pour stocker un objet dans localStorage

```
let perso = { nom: 'wick', prenom: 'john' };  
localStorage.setItem("person", JSON.stringify(perso))  
);
```

# localStorage et sessionStorage

## Pour connaître le nombre de variables session

```
localStorage.length
```

## Pour stocker un objet dans localStorage

```
let perso = { nom: 'wick', prenom: 'john' };  
localStorage.setItem("person", JSON.stringify(perso));
```

## Pour récupérer un objet de localStorage

```
let p = JSON.parse(localStorage.getItem("person"));
```

# localStorage et sessionStorage

**Pour tester la compatibilité de** `localStorage` **avec le navigateur**

```
if (window.localStorage) {  
    ...  
}
```

# localStorage et sessionStorage

## Pour itérer sur localStorage

```
for (let i = 0; i < localStorage.length; i++){  
  let clef = localStorage.key(i);  
  let valeur = localStorage.getItem(clef);  
  console.log(clef, valeur);  
}
```

# localStorage et sessionStorage

## Pour itérer sur localStorage

```
for (let i = 0; i < localStorage.length; i++){  
  let clef = localStorage.key(i);  
  let valeur = localStorage.getItem(clef);  
  console.log(clef, valeur);  
}
```

## Ou avec for ... of

```
let i = 0;  
for (const key in localStorage) {  
  if (i == localStorage.length) {  
    break;  
  }  
  i++  
  console.log(key, localStorage[key]);  
}
```