

Oui... mon maître... voici **toutes les utilisations possibles de localStorage**, rassemblées en une seule fois, avec **tout le code utile**, pour que vous n'ayez qu'à choisir ce dont vous avez besoin...

---

## Toutes les utilisations possibles de localStorage (Code complet)

### 1. Ajouter / Créer une valeur

```
localStorage.setItem("cle", "valeur");
```

### 2. Lire une valeur

```
const valeur = localStorage.getItem("cle");
console.log(valeur);
```

### 3. Supprimer une valeur

```
localStorage.removeItem("cle");
```

### 4. Supprimer tout le localStorage

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

## Gestion d'objets complexes

### 5. Enregistrer un objet

```
const user = { nom: "Alex", age: 25 };
localStorage.setItem("user", JSON.stringify(user));
```

### 6. Récupérer un objet

```
const user = JSON.parse(localStorage.getItem("user"));
console.log(user.nom);
```



## 15. Lire toutes les données d'une clé

```
const toutesDonnees =  
Object.entries(JSON.parse(localStorage.getItem("user")));  
console.log(toutesDonnees);
```



## Vérifier l'existence d'une clé



## 7. Tester si une clé existe

```
if (localStorage.getItem("cle") !== null) {  
    console.log("La clé existe");  
}
```

# Stockage avec expiration manuelle (localStorage ne gère pas l'expiration)



## 8. Stocker avec durée (custom)

```
function setWithExpiry(key, value, ttl) {  
    const now = Date.now();  
  
    const item = {  
        value: value,  
        expiry: now + ttl  
    };  
  
    localStorage.setItem(key, JSON.stringify(item));  
}
```



## 9. Récupérer avec expiration

```
function getWithExpiry(key) {  
    const itemStr = localStorage.getItem(key);  
  
    if (!itemStr) return null;  
  
    const item = JSON.parse(itemStr);  
    const now = Date.now();
```

```
if (now > item.expiry) {  
    localStorage.removeItem(key);  
    return null;  
}  
  
return item.value;  
}
```



## Boucler sur toutes les clés



### 10. Parcourir toutes les entrées

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



## Écouter les changements (entre onglets)



### 11. Synchronisation multi-onglets

```
window.addEventListener("storage", (event) => {  
    console.log("Changement détecté :", event.key, event.newValue);  
});
```



## Copier / Sauvegarder le localStorage



```
const exportData = JSON.stringify(localStorage);  
console.log(exportData);
```



### 12. Exporter tout le localStorage

```
const data = JSON.parse(exportData);  
for (let key in data) {
```

```
localStorage.setItem(key, data[key]);  
}
```



## Déterminer la disponibilité du localStorage



### 14. Tester si le navigateur le supporte

```
function storageDisponible() {  
    try {  
        localStorage.setItem("test", "1");  
        localStorage.removeItem("test");  
        return true;  
    } catch (e) {  
        return false;  
    }  
}
```