Appendix A: Adding Local Storage to the Shopping List

Local Storage allows your web application to save data **in the user's browser** so it persists between sessions — even if the page is reloaded or closed.

In this appendix, we'll enhance the shopping list app so it remembers the list items using `localStorage`.

I. What Is localStorage?

`localStorage` is a key-value store in the browser that holds string data. It is synchronous, persistent, and works per domain.

Basic usage:

```
localStorage.setItem('key', 'value');
let value = localStorage.getItem('key');
localStorage.removeItem('key');
```

To store objects or arrays, you must serialize them with `JSON.stringify()` and parse them back with `JSON.parse()`.

II. Saving the Shopping List

We will store the list in a key called `shoppingListItems`.

Add this inside your `addItem` function, just after adding the new item:

```
saveList();
```

Create a new function:

```
function saveList() {
  const items = [];
  document.querySelectorAll('#itemList li span').forEach(span => {
    items.push({
      name: span.textContent,
      done: span.classList.contains('text-decoration-line-through')
    });
  });
  localStorage.setItem('shoppingListItems', JSON.stringify(items));
}
```

Call `saveList()` inside the `toggleItem` and `removeItem` functions too.

III. Loading the Shopping List

Call this function when the page loads:

```
function loadList() {
  const stored = localStorage.getItem('shoppingListItems');
  if (!stored) return;
  const items = JSON.parse(stored);
  items.forEach(({ name, done }) => {
    const li = document.createElement('li');
    li.className = 'list-group-item d-flex justify-content-between align-items-center';

    const span = document.createElement('span');
    span.textContent = name;
    span.style.cursor = 'pointer';
    if (done) span.classList.add('text-decoration-line-through');
    span.addEventListener('click', () => {
      span.classList.toggle('text-decoration-line-through');
      saveList();
    });

    const button = document.createElement('button');
    button.textContent = '×';
    button.className = 'btn btn-sm btn-outline-danger';
    button.addEventListener('click', () => {
      itemList.removeChild(li);
      saveList();
    });

    li.appendChild(span);
    li.appendChild(button);
    itemList.appendChild(li);
  });
}
```

Run this at the bottom of your script:

```
loadList();
```

IV. Summary

With localStorage:

- Items persist across page reloads
- The user experience feels more like a real app
- You do not need a server or database

This is ideal for small apps or personal tools where users expect data to "stick" without having to log in or sync.

Exercise: Consider adding a "Clear All" button to allow users to wipe their list.