

## Copilot

Certainly! To persist your data even after refreshing the page, you can use **local storage** in JavaScript. Local storage allows you to save key-value pairs in the browser, and the data remains accessible across sessions. Here's how you can modify your code to utilize local storage:

```
let container = document.querySelector(".container");
let button = document.getElementById("add");

let card = document.createElement("div");
card.classList.add("childDiv");

let headers = ["Cheetah", "Buffalo", "Rhino"];
let animals = ["img/cheetah.jpeg", "img/buffalo.jpeg", "img/rhino.jpeg"];
let text = ["This is a cheetah", "This is a buffalo", "This is a rhino"];

let index = 0;

// Load data from local storage if available
if (localStorage.getItem("animalIndex")) {
  index = parseInt(localStorage.getItem("animalIndex"));
}

button.addEventListener("click", addAnimal);

function addAnimal() {
  let newCard = card.cloneNode(true);

  let h3 = document.createElement("h3");
  h3.textContent = headers[index];

  let img = document.createElement("img");
  img.src = animals[index];
  img.alt = "animal";

  let description = document.createElement("p");
  description.textContent = text[index];

  newCard.append(h3, img, description);
  container.append(newCard);

  // Save the current index to local storage
  localStorage.setItem("animalIndex", index);

  index = (index + 1) % headers.length;
}
```

In this modified code:

- We check if there's an existing index stored in local storage. If so, we load it.
- After displaying the animal, we save the current index back to local storage. Now your data will persist even after refreshing the page! 🌟