Remove all associated child objects to avoid memory leaks

To avoid memory leaks in JavaScript, you need to make sure that you remove all references to an object when you're done with it. This includes event listeners, timers, and any other objects that might hold references to the object you want to clean up.

Here's a basic example of a JavaScript function that can help you terminate or clean up an object and its associated web objects to avoid memory leaks:

```
function terminateObject(obj) {
  // Remove event listeners associated with the object
  obj.removeEventListener('click', clickHandler);
  // Clear any timeouts or intervals associated with the object
  clearInterval(obj.intervalID);
  // Remove any references to the object itself
  obj = null;
}
```