

innerHTML vs. createElement

createElement is an alternative method of innerHTML, so here we'll look at the two.

createElement is faster, as browsers are not required to parse the HTML string and then build a node tree out of it; it also doesn't have to attach event listeners as innerHTML does. Using innerHTML will cause browsers to reparse and recreate all DOM nodes inside the element whose innerHTML is modified.

However, if you're writing a dynamic solution like a Markdown to HTML converter with a real-time preview, then innerHTML is the way to go as it is a "one-size-fits-all" approach. This is precisely what's required for this particular conversion and real-time preview. Building the same logic with createElement would be a pain and render highly coupled logic with unextendable code.

This will work for this particular problem, but it's not a scalable approach.

You'd be adding a lot of work for yourself and your team if, for example, you wanted to build a Rich Text Editor where you have to write separate logic for each of the new functionalities you might add.

Using innerHTML with a simple HTML sanitizer is a much more efficient and practical approach.

Conclusion

In HTML, assigning a string to the innerHTML property is okay in some situations. However, if you can't be sure of what the string contains—for example, if a user provides it and it is possibly malicious—it's best to use createElement or sanitize the input before storing it in a database.

The open-source "sanitization" libraries strip off specific tags and attributes to make the desired HTML string XSS proof. Sanitization should be done on both the frontend and backend as a best practice to help reduce the risk of XSS attacks. When implementing sanitization on the frontend, it should be done at the time of render when the user provides input. And, on the backend, it should be done before storage in the database.

This post was written by Keshav Malik. Keshav is a full-time developer who loves to build and break stuff. He is constantly looking for new and exciting technologies and enjoys working with diverse technologies in his spare time. He loves music and plays badminton whenever the opportunity presents itself.

