



Clicking on the Markdown box pastes text from the clipboard, automatically converts it to HTML, and copies the resulting HTML back to the clipboard.

Head Section

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Markdown to HTML Converter</title>
  <script
src="https://cdn.jsdelivr.net/npm/marked/marked.min.js"></script>
</head>
```

Body Section

```
<body>

  <h2>Markdown to HTML Converter</h2>

  <!-- Input area for Markdown text -->
  <textarea id="markdown-input" placeholder="Click here to paste from
clipboard..." rows="10" style="width: 100%;"
onclick="pasteAndConvert()"></textarea>

  <!-- Output area for HTML result -->
  <h3>HTML Output (automatically copied):</h3>
  <textarea id="html-output" readonly rows="10" style="width:
100%;"></textarea>

  <script>
    // Function to paste content from the clipboard, convert to HTML,
and copy HTML to clipboard
    async function pasteAndConvert() {
      try {
        // Paste clipboard text into the Markdown input
        const text = await navigator.clipboard.readText();
        const markdownInput = document.getElementById("markdown-
input");
      }
    }
  </script>
```



```
markdownInput.value = text;

// Convert the pasted Markdown to HTML
const htmlOutput = marked.parse(text);

// Display the HTML in the output box
document.getElementById("html-output").value = htmlOutput;

// Copy the HTML output to the clipboard
await copyToClipboard(htmlOutput);

// Notify the user
alert("HTML output has been copied to clipboard!");
} catch (err) {
  alert("Failed to read clipboard contents: " + err);
}
}

// Helper function to copy text to clipboard
async function copyToClipboard(text) {
  try {
    await navigator.clipboard.writeText(text);
  } catch (err) {
    console.error("Failed to copy text to clipboard: ", err);
  }
}
</script>

</body>
```

Explanation

1. **pasteAndConvert()**: This function runs when you click on the Markdown input box.
 - It reads the clipboard content and pastes it into the Markdown box.
 - It converts the pasted Markdown to HTML using `marked.parse(text)`.
 - It copies the converted HTML back to the clipboard and alerts the user.



2. **copyToClipboard()**: A helper function that copies text to the clipboard, used to ensure that the HTML output is copied back smoothly.

Now, you can click on the Markdown box, and it will:

- Paste clipboard content into the Markdown box.
- Convert it to HTML.
- Automatically copy the HTML output back to the clipboard and notify you that it's done.