

```
1  @import url('https://fonts.googleapis.com/css2?family=Source+Sans+3:wght@300;400;500;600;700&display=swap');
2
3  body{
4    margin: 0;
5  }
6
7  .container{
8    font-family: "Source Sans 3", sans-serif;
9  }
10
11  .header{
12    background: #000;
13    color: #fff;
14    height: 60px;
15    border-bottom: 5px solid #333;
16  }
17
18  .header .title{
19    padding: 0.5rem;
20  }
21
22  .header .title .main-title{
23    font-size: 18px;
24    font-weight: 600;
25  }
26
27  .header .title .sub-title{
28    font-size: 14px;
29  }
30
31  .header .title .sub-title span{
32    color: #999;
33  }
34
35  .editor{
36    width: 33.33%;
37    height: 100%;
38    float: left;
39    box-sizing: border-box;
40    position: relative;
41  }
42
43  .editor #html{
44    border-right: 1px solid #777;
45  }
46
47  .editor #css{
48    border-left: 20px solid #333;
49    border-right: 1px solid #777;
50  }
51
52  .editor #js{
53    border-left: 20px solid #333;
54  }
55
56  .code-box{
57    position: relative;
58    height: calc(50vh - 4.6rem);
59    border-top: 1px solid #000;
60    border-bottom: 10px solid #333;
61    overflow: hidden;
62  }
63
64  .preview{
65    position: relative;
66    height: calc(90vh - 30px);
67    background: #fff;
68    border-top: 10px ridge #333;
69  }
70
71  .preview iframe{
72    width: 100%;
73    height: 100%;
74  }
75
76  ::-webkit-scrollbar{
77    width: 5px;
78    height: 5px;
79  }
80
81  ::-webkit-scrollbar-track{
82    background: transparent;
83  }
84
85  ::-webkit-scrollbar-thumb{
86    background: #555;
87    border-radius: 20px;
88  }
```



```

1 // Catching commonly used elements to minimize dom queries
2 const livePreviewFrame = document.getElementById('live-preview');
3 const htmlEditor = document.getElementById('html');
4 const cssEditor = document.getElementById('css');
5 const jsEditor = document.getElementById('js');
6
7 // Function to set up the Live preview iframe and include necessary scripts
8 function initializeLivePreview() {
9     livePreviewFrame.contentWindow.document.body.innerHTML = '';
10    const styleElement = document.createElement('style');
11    styleElement.setAttribute('id', 'live-preview-style');
12    livePreviewFrame.contentWindow.document.head.appendChild(styleElement);
13
14    const pagedJsScript = document.createElement('script');
15    pagedJsScript.src = 'https://unpkg.com/pagedjs/dist/paged.legacy.polyfill.js';
16    livePreviewFrame.contentWindow.document.head.appendChild(pagedJsScript);
17 }
18
19 // Function to update the Live preview iframe with the html code from editor
20 function updateLiveHTMLPreview(codeEditors) {
21     livePreviewFrame.contentWindow.document.body.innerHTML = codeEditors.html.getValue();
22 }
23
24 // Function to update the Live preview iframe with the css code from editor
25 function updateLiveCSSPreview(codeEditors) {
26     const styleElement = livePreviewFrame.contentWindow.document.getElementById('live-preview-style');
27     styleElement.innerHTML = codeEditors.css.getValue();
28 }
29
30 // Function to update the Live preview iframe with the js code from editor
31 function updateLiveJSPreview(codeEditors) {
32     const scriptElement = document.createElement('script');
33     scriptElement.innerHTML = codeEditors.js.getValue();
34     livePreviewFrame.contentWindow.document.body.appendChild(scriptElement);
35 }
36
37 // Function to initialize CodeMirror editors for html, css and javascript
38 function initializeCodeEditors() {
39     function getDefaultOptions(object) {
40         const defaultOptions = {
41             lineNumbers: true,
42             autoCloseTags: true,
43             autoCloseBrackets: true,
44             theme: 'panda-syntax'
45         };
46         if (object) {
47             const keys = Object.keys(object);
48             for (const key of keys) {
49                 defaultOptions[key] = object[key];
50             }
51         }
52         return defaultOptions;
53     }
54
55     const codeEditors = {
56         html: CodeMirror(htmlEditor, getDefaultOptions({
57             mode: 'text/html',
58             value: ''
59         })),
60         css: CodeMirror(cssEditor, getDefaultOptions({
61             mode: 'css',
62             value: '',
63             extraKeys: { 'Ctrl-Space': 'autocomplete' },
64             hintOptions: {
65                 completeSingle: false,
66                 closeOnUnfocus: false
67             }
68         })),
69         js: CodeMirror(jsEditor, getDefaultOptions({
70             mode: 'javascript',
71             value: ''
72         })),
73     };
74     return codeEditors;
75 }
76
77 // Function to set up the Live preview studio with CodeMirror editors and event listeners
78 function setupLivePreviewStudio() {
79     const codeEditors = initializeCodeEditors();
80
81     // Event Listener for changes in HTML editor
82     CodeMirror.on(codeEditors.html, 'change', () => {
83         updateLiveHTMLPreview(codeEditors);
84     });
85
86     // Event Listener for changes in CSS editor
87     CodeMirror.on(codeEditors.css, 'change', () => {
88         updateLiveCSSPreview(codeEditors);
89     });
90
91     // Event Listener for changes in HTML editor
92     CodeMirror.on(codeEditors.js, 'change', () => {
93         updateLiveJSPreview(codeEditors);
94     });
95 }
96
97 // Event Listener to set up the live preview studio after the dom is fully loaded
98 document.addEventListener('DOMContentLoaded', () => {
99     initializeLivePreview();
100    setupLivePreviewStudio();
101 });

```