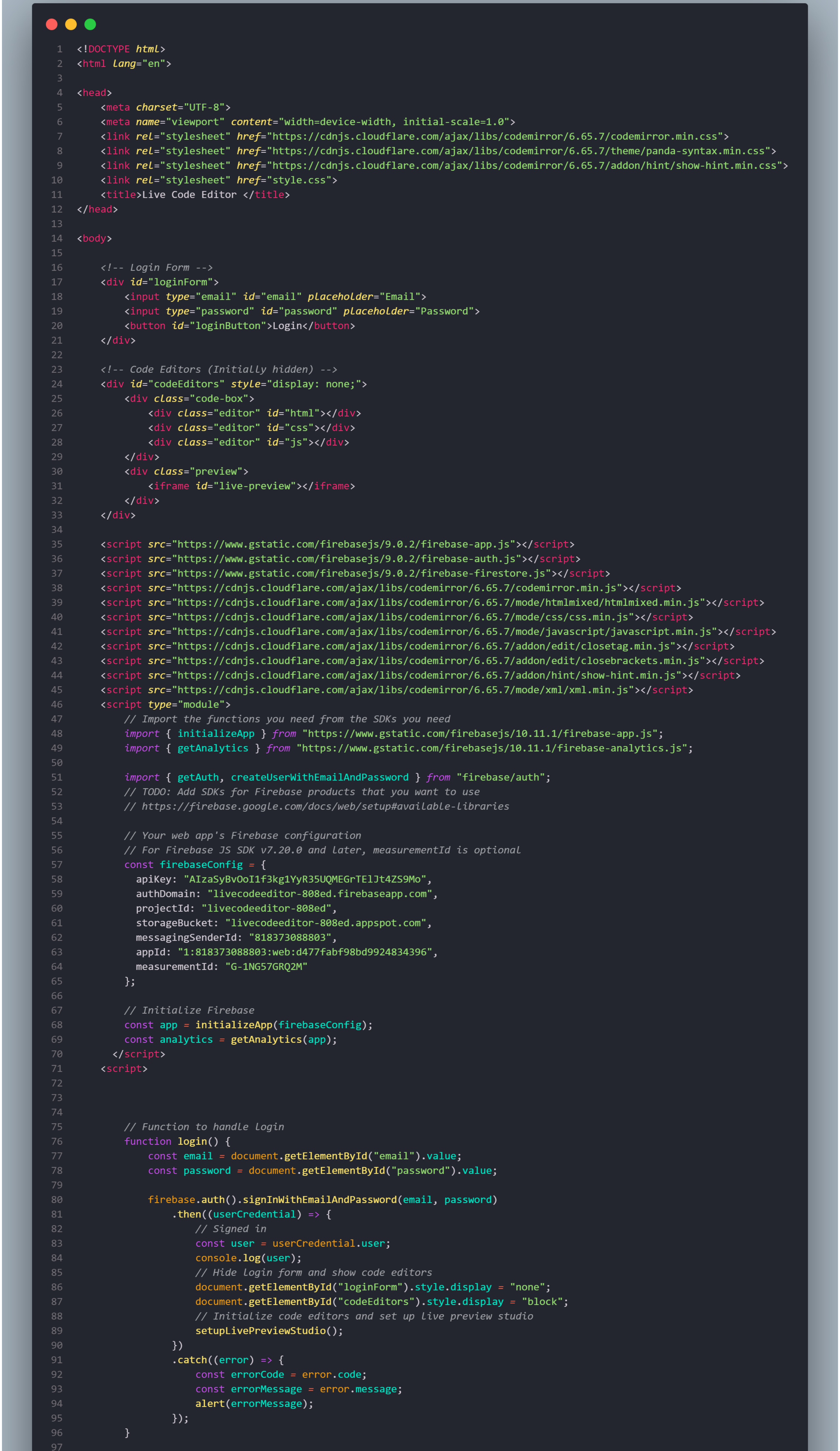
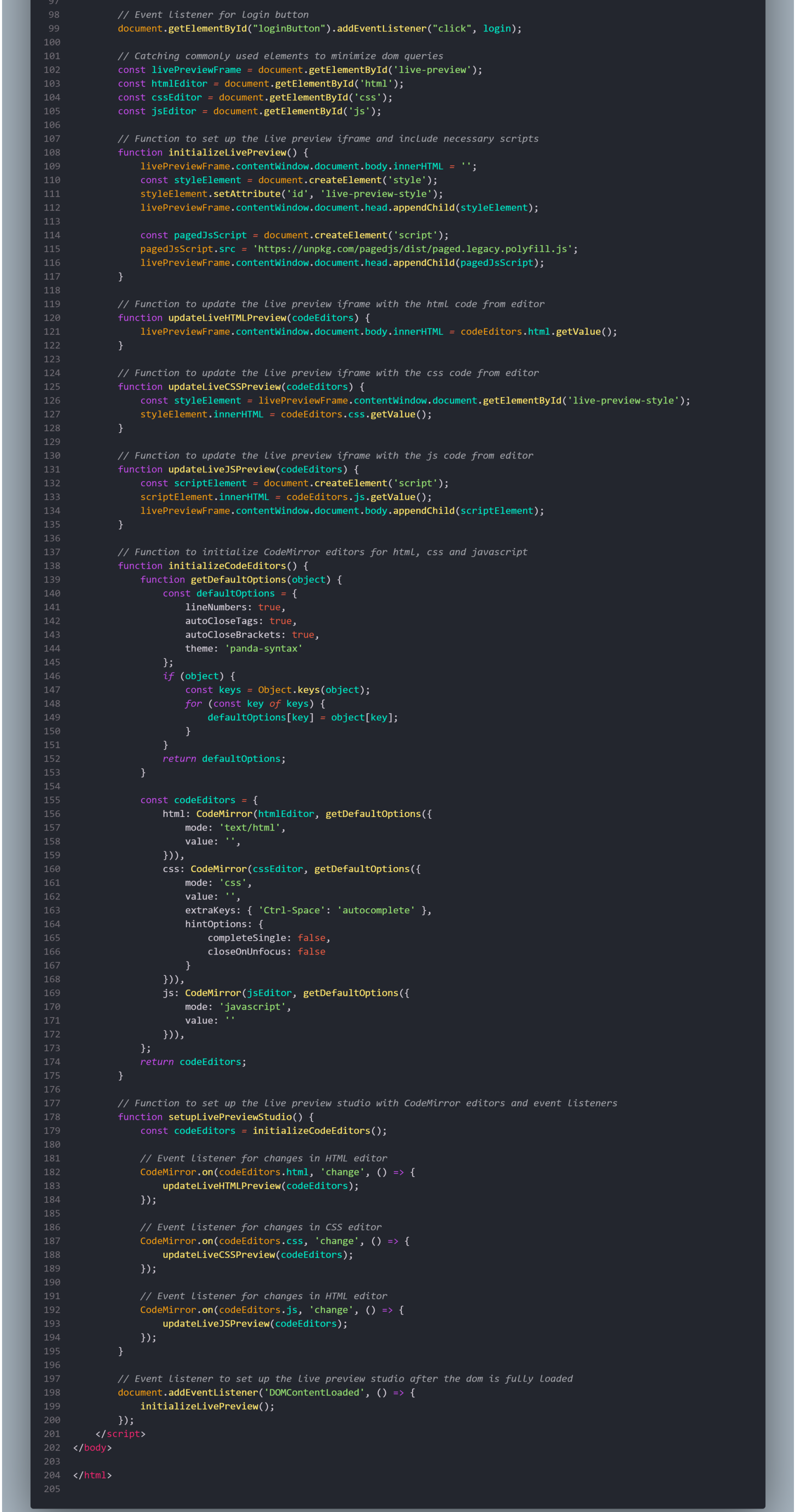
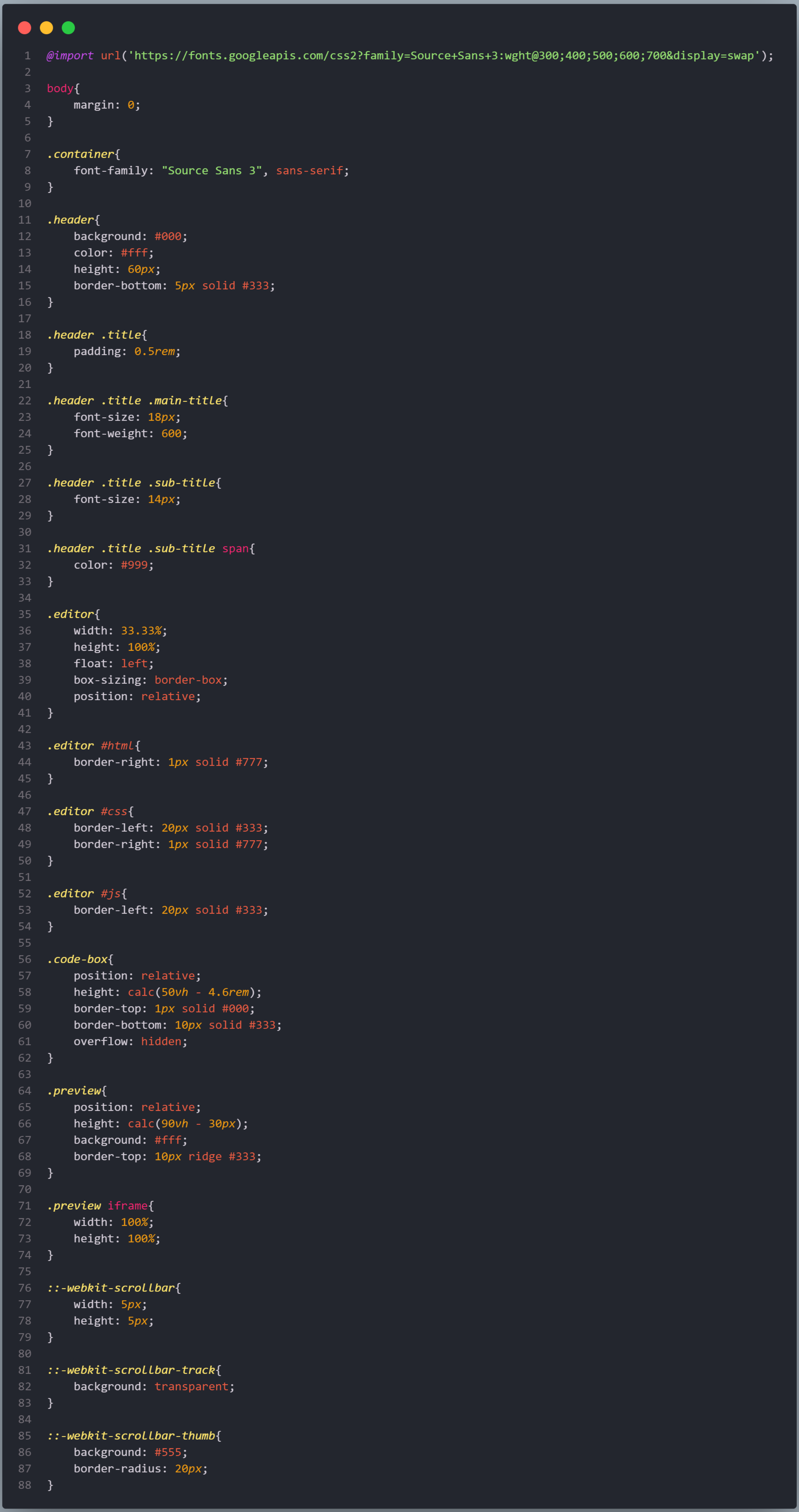


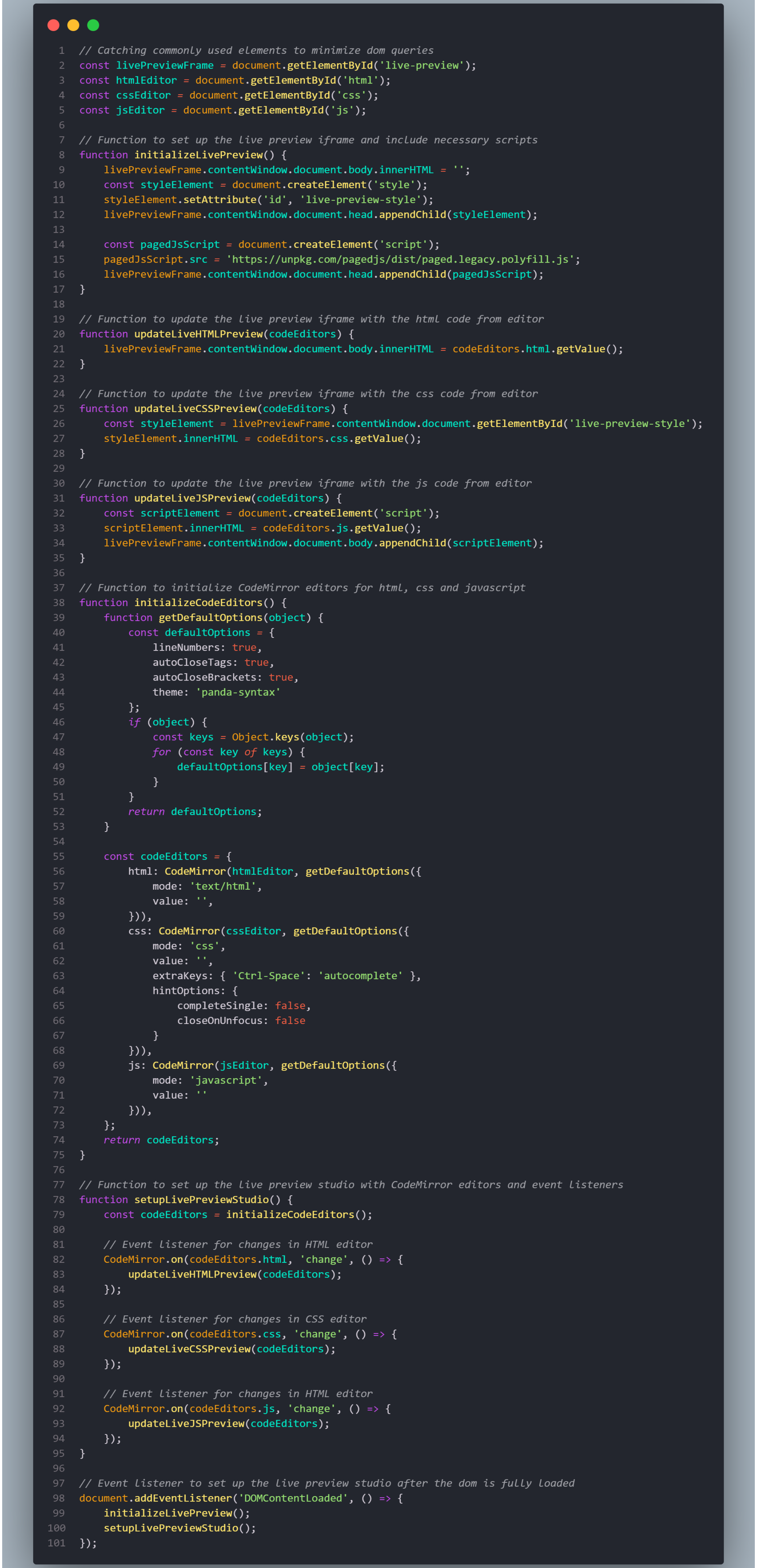
```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/codemirror.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/theme/panda-syntax.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/hint/show-hint.min.css">
    <link rel="stylesheet" href="style.css">
    <title>Live Code Editor </title>
</head>

<body>

    <!-- Login Form -->
    <div id="loginForm">
        <input type="email" id="email" placeholder="Email">
        <input type="password" id="password" placeholder="Password">
        <button id="loginButton">Login</button>
    </div>

    <!-- Code Editors (Initially hidden) -->
    <div id="codeEditors" style="display: none;">
        <div class="code-box">
            <div class="editor" id="html"></div>
            <div class="editor" id="css"></div>
            <div class="editor" id="js"></div>
        </div>
        <div class="preview">
            <iframe id="live-preview"></iframe>
        </div>
    </div>

    <script src="https://www.gstatic.com/firebasejs/9.0.2/firebase-app.js"></script>
    <script src="https://www.gstatic.com/firebasejs/9.0.2/firebase-auth.js"></script>
    <script src="https://www.gstatic.com/firebasejs/9.0.2/firebase-firestore.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/codemirror.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/htmlmixed/htmlmixed.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/css/css.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/javascript/javascript.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/edit/closetag.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/edit/closebrackets.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/hint/show-hint.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/xml/xml.min.js"></script>
    <script type="module">
        // Import the functions you need from the SDKs you need
        import { initializeApp } from "https://www.gstatic.com/firebasejs/10.11.1/firebase-app.js";
        import { getAnalytics } from "https://www.gstatic.com/firebasejs/10.11.1/firebase-analytics.js";

        import { getAuth, createUserWithEmailAndPassword } from "firebase/auth";
        // TODO: Add SDKs for Firebase products that you want to use
        // https://firebase.google.com/docs/web/setup#available-libraries

        // Your web app's Firebase configuration
        // For Firebase JS SDK v7.20.0 and later, measurementId is optional
        const firebaseConfig = {
          apiKey: "AIzaSyBvOoI1f3kg1YyR35UQMEGrTElJt4ZS9Mo",
          authDomain: "livecodeeditor-808ed.firebaseapp.com",
          projectId: "livecodeeditor-808ed",
          storageBucket: "livecodeeditor-808ed.appspot.com",
          messagingSenderId: "818373088803",
          appId: "1:818373088803:web:d477fabf98bd9924834396",
          measurementId: "G-1NG57GRQ2M"
        };

        // Initialize Firebase
        const app = initializeApp(firebaseConfig);
        const analytics = getAnalytics(app);
      </script>
    <script>



        // Function to handle login
        function login() {
            const email = document.getElementById("email").value;
            const password = document.getElementById("password").value;

            firebase.auth().signInWithEmailAndPassword(email, password)
                .then((userCredential) => {
                    // Signed in
                    const user = userCredential.user;
                    console.log(user);
                    // Hide login form and show code editors
                    document.getElementById("loginForm").style.display = "none";
                    document.getElementById("codeEditors").style.display = "block";
                    // Initialize code editors and set up live preview studio
                    setupLivePreviewStudio();
                })
                .catch((error) => {
                    const errorCode = error.code;
                    const errorMessage = error.message;
                    alert(errorMessage);
                });
        }

```



```
        // Event listener for login button
        document.getElementById("loginButton").addEventListener("click", login);

        // Catching commonly used elements to minimize dom queries
        const livePreviewFrame = document.getElementById('live-preview');
        const htmlEditor = document.getElementById('html');
        const cssEditor = document.getElementById('css');
        const jsEditor = document.getElementById('js');

        // Function to set up the live preview iframe and include necessary scripts
        function initializeLivePreview() {
            livePreviewFrame.contentWindow.document.body.innerHTML = '';
            const styleElement = document.createElement('style');
            styleElement.setAttribute('id', 'live-preview-style');
            livePreviewFrame.contentWindow.document.head.appendChild(styleElement);

            const pagedJsScript = document.createElement('script');
            pagedJsScript.src = 'https://unpkg.com/pagedjs/dist/paged.legacy.polyfill.js';
            livePreviewFrame.contentWindow.document.head.appendChild(pagedJsScript);
        }

        // Function to update the live preview iframe with the html code from editor
        function updateLiveHTMLPreview(codeEditors) {
            livePreviewFrame.contentWindow.document.body.innerHTML = codeEditors.html.getValue();
        }

        // Function to update the live preview iframe with the css code from editor
        function updateLiveCSSPreview(codeEditors) {
            const styleElement = livePreviewFrame.contentWindow.document.getElementById('live-preview-style');
            styleElement.innerHTML = codeEditors.css.getValue();
        }

        // Function to update the live preview iframe with the js code from editor
        function updateLiveJSPreview(codeEditors) {
            const scriptElement = document.createElement('script');
            scriptElement.innerHTML = codeEditors.js.getValue();
            livePreviewFrame.contentWindow.document.body.appendChild(scriptElement);
        }

        // Function to initialize CodeMirror editors for html, css and javascript
        function initializeCodeEditors() {
            function getDefaultOptions(object) {
                const defaultOptions = {
                    lineNumbers: true,
                    autoCloseTags: true,
                    autoCloseBrackets: true,
                    theme: 'panda-syntax'
                };
                if (object) {
                    const keys = Object.keys(object);
                    for (const key of keys) {
                        defaultOptions[key] = object[key];
                    }
                }
                return defaultOptions;
            }

            const codeEditors = {
                html: CodeMirror(htmlEditor, getDefaultOptions({
                    mode: 'text/html',
                    value: '',
                })),
                css: CodeMirror(cssEditor, getDefaultOptions({
                    mode: 'css',
                    value: '',
                    extraKeys: { 'Ctrl-Space': 'autocomplete' },
                    hintOptions: {
                        completeSingle: false,
                        closeOnUnfocus: false
                    }
                })),
                js: CodeMirror(jsEditor, getDefaultOptions({
                    mode: 'javascript',
                    value: ''
                })),
            };
            return codeEditors;
        }

        // Function to set up the live preview studio with CodeMirror editors and event listeners
        function setupLivePreviewStudio() {
            const codeEditors = initializeCodeEditors();

            // Event listener for changes in HTML editor
            CodeMirror.on(codeEditors.html, 'change', () => {
                updateLiveHTMLPreview(codeEditors);
            });

            // Event listener for changes in CSS editor
            CodeMirror.on(codeEditors.css, 'change', () => {
                updateLiveCSSPreview(codeEditors);
            });

            // Event listener for changes in HTML editor
            CodeMirror.on(codeEditors.js, 'change', () => {
                updateLiveJSPreview(codeEditors);
            });
        }

        // Event listener to set up the live preview studio after the dom is fully loaded
        document.addEventListener('DOMContentLoaded', () => {
            initializeLivePreview();
        });
    </script>
</body>

</html>

```



```css
@import url('https://fonts.googleapis.com/css2?family=Source+Sans+3:wght@300;400;500;600;700&display=swap');

body{
    margin: 0;
}

.container{
    font-family: "Source Sans 3", sans-serif;
}

.header{
    background: #000;
    color: #fff;
    height: 60px;
    border-bottom: 5px solid #333;
}

.header .title{
    padding: 0.5rem;
}

.header .title .main-title{
    font-size: 18px;
    font-weight: 600;
}

.header .title .sub-title{
    font-size: 14px;
}

.header .title .sub-title span{
    color: #999;
}

.editor{
    width: 33.33%;
    height: 100%;
    float: left;
    box-sizing: border-box;
    position: relative;
}

.editor #html{
    border-right: 1px solid #777;
}

.editor #css{
    border-left: 20px solid #333;
    border-right: 1px solid #777;
}

.editor #js{
    border-left: 20px solid #333;
}

.code-box{
    position: relative;
    height: calc(50vh - 4.6rem);
    border-top: 1px solid #000;
    border-bottom: 10px solid #333;
    overflow: hidden;
}

.preview{
    position: relative;
    height: calc(90vh - 30px);
    background: #fff;
    border-top: 10px ridge #333;
}

.preview iframe{
    width: 100%;
    height: 100%;
}

::-webkit-scrollbar{
    width: 5px;
    height: 5px;
}

::-webkit-scrollbar-track{
    background: transparent;
}

::-webkit-scrollbar-thumb{
    background: #555;
    border-radius: 20px;
}
```



```javascript
// Catching commonly used elements to minimize dom queries
const livePreviewFrame = document.getElementById('live-preview');
const htmlEditor = document.getElementById('html');
const cssEditor = document.getElementById('css');
const jsEditor = document.getElementById('js');

// Function to set up the live preview iframe and include necessary scripts
function initializeLivePreview() {
    livePreviewFrame.contentWindow.document.body.innerHTML = '';
    const styleElement = document.createElement('style');
    styleElement.setAttribute('id', 'live-preview-style');
    livePreviewFrame.contentWindow.document.head.appendChild(styleElement);

    const pagedJsScript = document.createElement('script');
    pagedJsScript.src = 'https://unpkg.com/pagedjs/dist/paged.legacy.polyfill.js';
    livePreviewFrame.contentWindow.document.head.appendChild(pagedJsScript);
}

// Function to update the live preview iframe with the html code from editor
function updateLiveHTMLPreview(codeEditors) {
    livePreviewFrame.contentWindow.document.body.innerHTML = codeEditors.html.getValue();
}

// Function to update the live preview iframe with the css code from editor
function updateLiveCSSPreview(codeEditors) {
    const styleElement = livePreviewFrame.contentWindow.document.getElementById('live-preview-style');
    styleElement.innerHTML = codeEditors.css.getValue();
}

// Function to update the live preview iframe with the js code from editor
function updateLiveJSPreview(codeEditors) {
    const scriptElement = document.createElement('script');
    scriptElement.innerHTML = codeEditors.js.getValue();
    livePreviewFrame.contentWindow.document.body.appendChild(scriptElement);
}

// Function to initialize CodeMirror editors for html, css and javascript
function initializeCodeEditors() {
    function getDefaultOptions(object) {
        const defaultOptions = {
            lineNumbers: true,
            autoCloseTags: true,
            autoCloseBrackets: true,
            theme: 'panda-syntax'
        };
        if (object) {
            const keys = Object.keys(object);
            for (const key of keys) {
                defaultOptions[key] = object[key];
            }
        }
        return defaultOptions;
    }

    const codeEditors = {
        html: CodeMirror(htmlEditor, getDefaultOptions({
            mode: 'text/html',
            value: '',
        })),
        css: CodeMirror(cssEditor, getDefaultOptions({
            mode: 'css',
            value: '',
            extraKeys: { 'Ctrl-Space': 'autocomplete' },
            hintOptions: {
                completeSingle: false,
                closeOnUnfocus: false
            }
        })),
        js: CodeMirror(jsEditor, getDefaultOptions({
            mode: 'javascript',
            value: ''
        })),
    };
    return codeEditors;
}

// Function to set up the live preview studio with CodeMirror editors and event listeners
function setupLivePreviewStudio() {
    const codeEditors = initializeCodeEditors();

    // Event listener for changes in HTML editor
    CodeMirror.on(codeEditors.html, 'change', () => {
        updateLiveHTMLPreview(codeEditors);
    });

    // Event listener for changes in CSS editor
    CodeMirror.on(codeEditors.css, 'change', () => {
        updateLiveCSSPreview(codeEditors);
    });

    // Event listener for changes in HTML editor
    CodeMirror.on(codeEditors.js, 'change', () => {
        updateLiveJSPreview(codeEditors);
    });
}

// Event listener to set up the live preview studio after the dom is fully loaded
document.addEventListener('DOMContentLoaded', () => {
    initializeLivePreview();
    setupLivePreviewStudio();
});
```