

Premium Rich Text Editor Component V1.0.0

Version 1.0.0

.....Column Break.....

Table of Contents

Table of Contents..... 2

Overview 3

Prerequisites 3

Premium Features..... 4

Component Parameter Configuration 6

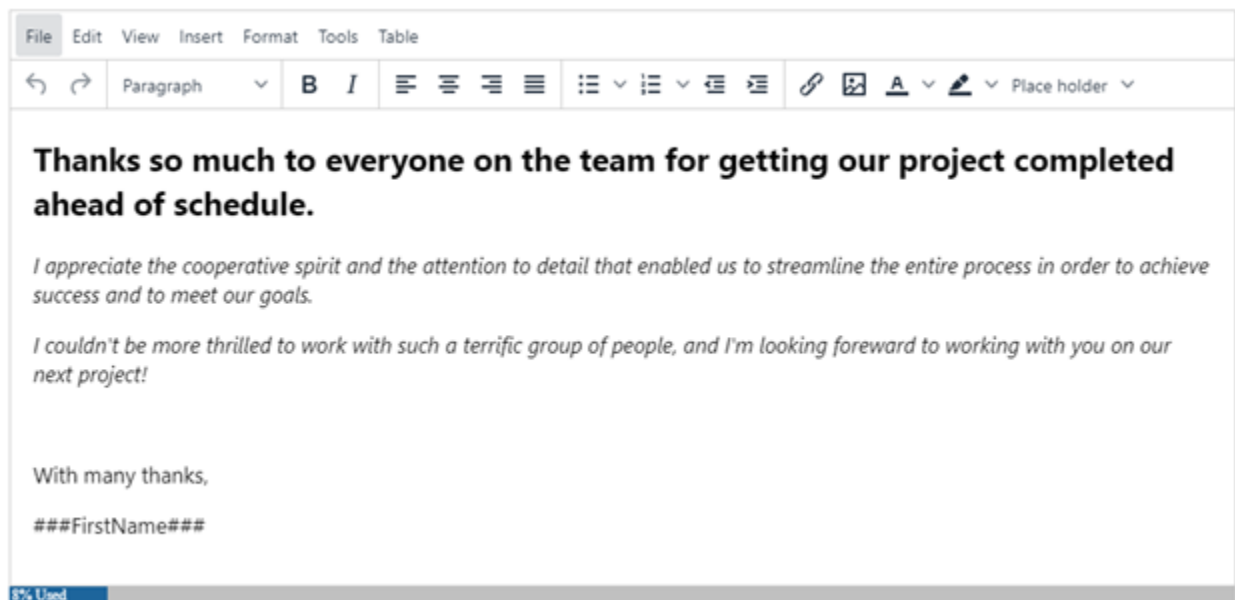
Component Usage Guide..... 11

Third Party Credential Setup..... 25

Advanced Rich Text Editor Connected System Version Compatibility 26

Overview

The Premium Rich Text Editor Component can be utilized on Appian Screens for applications where presenting rich text information to end-users in a readable format is essential. Additionally, it serves as an interface for composing email subjects. This component extends the capabilities of the existing [Advanced Rich Text Editor](#) Plugin by incorporating premium features. In comparison to the out-of-the-box rich text editor, this plugin enables users to add images, customize tables, and more. It also facilitates image uploads, which are stored in a specified Appian folder (requires separate installation of the Vuram Rich Text Editor Connected System Plugin from [Appian Appmarket](#)).



Prerequisites

- Download and configure the Advanced Rich Text Editor Connected

System found on the [Appian App Market](#). This allows uploading of images (above 1 KB) which get stored in the specified Appian folder.

- This plugin utilizes TinyMCE Cloud, so please ensure to purchase the [TinyMCECloud](#) subscription rather than opting for self-hosted deployment. Please check the pricing details [here](#).
- Note: Powerpaste, Spellchecker, spelling autocorrect, Accessibility Checker, and Advanced Typography are not available in the TinyMCE Essential plan. Therefore, please ensure to purchase the Professional or Enterprise plan to access these features.

Premium Features

1. **Accessibility Checker:** This feature allows users to examine HTML within the editor for various WCAG & Section 508 accessibility issues. An auto-repair feature enables users to rectify identified problems automatically.
2. **Advanced Code Editor:** This feature brings a more advanced code editor to TinyMCE. This code editor makes it easier to modify the HTML and is a useful add-on for power users. It comes with features often found in IDEs.
3. **Advanced Tables:** This feature extends the core table plugin by adding the following advanced functionality:
 - Sort options for rows and columns.
 - Row numbering column for tables.
4. **Advanced Typography:** This feature can replace common typewriter-style characters with their professional typesetter equivalents after the Advanced Typography button is pressed. This action will modify either the entire content for the TinyMCE document or the specific text selection highlighted by the user.

For example, if English-language typography rules are set, the Advanced Typography plugin can automatically replace an open-hyphen (-) with an em-dash (—). And it can automatically replace the teardrop apostrophe (') with opening and closing quote marks (‘ and ’). Click [here](#) for more details.
5. **Case Change:** This feature allows changing the case of text in block

selection to uppercase, lowercase, or title case. Perform the following steps to change the case of selected text in a document to lowercase, UPPERCASE, or Title Case.

6. **Checklist:** Stay organized with the Checklist feature. Easily create checkbox lists within your content, perfect for task management, to-do lists, and more.
7. **Enhanced Media Embed:** This feature that will take a URL, send it to a backend service that returns an embeddable snippet of code, and add that code to the editor. That typically takes the form of rich media such as audio, video and social media cards.
8. **Footnotes:** This feature allows the user to insert a footnote into the editor. Each time a footnote is added, a superscripted footnote number is placed at the current insertion point, or at the end of the current selection. The plugin then adds a new line to the footnotes section located at the bottom of the editor and moves the insertion point to this new line. This allows for the immediate creation of the contents of the footnote entry related to the newly created footnote.
9. **Link Checker:** This feature validates URLs, as you type. URLs considered invalid will be highlighted with red and will have a dedicated context menu with options to either edit the link, try and open it in a separate tab, remove the link, or ignore it.
10. **Page Embed:** This feature embeds a page in the content using an iframe (Inline frames). The size of these iframes can be configured to be responsive or another user-defined dimension. The ability to use iframes helps create visually engaging content which is very useful for:
 - Online courses and learning materials.
 - Online advertising
 - Creating multimedia content
11. **Permanent Pen:** This feature allows a user to apply a pre-defined format (for example, color, typeface and type style, font, etc.) to a new content, overriding the default values set for that content area. The Permanent Pen is useful when a distinct style text needs to be repeatedly added to the content. The user can pre-define the desired style in the Permanent Pen properties.
12. **PowerPaste:** It automatically cleans up content from Microsoft Word, Microsoft Excel, Google Docs, and HTML sources to ensure clean, compliant content that matches the

look and feel of the site.

Note: Due to limitations in Microsoft Excel online (part of Office Live) PowerPaste does not support pasting from Microsoft Excel online. If you paste content using Microsoft Excel in Office Live you will get a plain text representation of the content. Due to limitations in Microsoft Excel online (part of OfficeLive) PowerPaste does not support pasting from Microsoft Excel online. If you paste content using Microsoft Excel in Office Live you will get a plain text representation of the content.

13. **Table of Contents:** It generates a basic table of contents (ToC) and inserts it into the editor at the current cursor position. ToC entries are generated from header elements (h1 – h6 elements) in the content.
14. **Tiny Comments:** This feature provides the ability to start or join a conversation by adding comments to the content within the editor.

Component Parameter Configuration

Rule Name : premiumRichTextEditorField()

Name	Type	Description	Required
label	Text	Text to display as the field label.	No
labelPosition	Text	Determines where the label appears. Valid values: "ABOVE" (default), "ADJACENT", "JUSTIFIED", "COLLAPSED".	No
height	Text	Determines the layout height. Valid values: "SHORT", "MEDIUM", "TALL", "AUTO" (default).	No

apiKey	Text	Authentication key used for TinyMCE authorization. API Key can be obtained from TinyMCE Cloud . Please check the pricing details here .	Yes
isReadOnly	Boolean	Determines if the field should display as editable or not. Default: false.	No
hideBorderInRead Only	Boolean	Provide true to hide the border when the editor is in Read Only mode. Default : false	No
showMenubar	Boolean	If it is true then the menu bar displayed. Default true	No
showUsageBar	Boolean	Provide false to hide the usage bar at the bottom of editor. Default true.	No
includeHTMLlength	Boolean	Include the HTML format length, if set to "True" and vice versa. default : true.	No
enableFullpageOutput	Boolean	Enabling fullpage exposes head,body and various meta tags in richText output. Default true.	No
enableLatexConversion	Boolean	Provide false to disable 'Convert LaTeX to Mathematical Expression' button. Default: false.	No
maxSize	Number (Integer)	Maximum size of the rich text data. Default: 10000. MinSize 100.	No

allowImageUpload	Boolean	Determines if image uploads are allowed. Default: false.	No
imageStorageConnectedSystem	Connected System	The instance of the Vuram Rich Text Editor connected system used to store images from the rich text editor. This is required if allowImageUpload is set to TRUE. You can download the Connected System from the Appian App Market.	No
uploadedImagesDocIds	List of Save	It contains the document ids of all the uploaded images	No
enableExportPdf	Boolean	Provide false to disable 'Export as PDF' button. Default: True	No
exportedPdfId	List of Save	It has the document id of the pdf generated through Export as Pdf feature	No
user	Dictionary	Provide value for Author details. Example: {author:"userName",authorDisplayName:"User",isAdmin:true} author - Current Author username. Default: "User" isAdmin - To decide whether the author is admin or not. Only the admin user can delete the whole conversation. Default: false. authorDisplayName - Name of the Author which must be displayed in the editor. Default: "User"	No
spellCheck	Dictionary	This Parameter is used to configure spell check options. Example: {enableSpellCheck:true,spellCheckLang:"en",enableAutocorrect:true} enableSpellCheck- Provide true to	No

		<p>enable the spell checker. Default false.</p> <p>spellCheckLan - Spell Check Language. Check here for supported languages and their code.</p> <p>enableAutocorrect - This parameter is used to enable/disable spelling autocorrect feature. Default: false.</p>	
titleCaseMinors	List of Text String	<p>While using Title case option in case change, the values passed to this parameter will not be capitalized. Example: {"at", "by", "in"}.</p>	No
placeholder	List of Dictionary	<p>Place holder text to display in the component wherever you want. Example: { { name: "Name", values: { { text: "First Name", value: "###FirstName###" }, { text: "Last Name", value: "###LastName###" } } }, { name: "Address", values: { { text: "Permanent Address", value: "###PermanentAddress###" }, { text: "Temporary Address", value: "###TemporaryAddress###" } } } }</p>	No
richTextValue	Text	Rich text to display in the field.	No
richTextSaveInto	List of Save	<p>One or more variables that are updated with the richText value when the user changes it. Use a!save() to save a modified or alternative value to a variable.</p>	No
commentListValue	List of Dictionary	<p>List of comments that must be displayed in the Editor.</p> <p>Ex.</p> <pre>{ { uid: "96d067ad-d0bd-4748-a182-eecee7d32409", comments: {</pre>	No

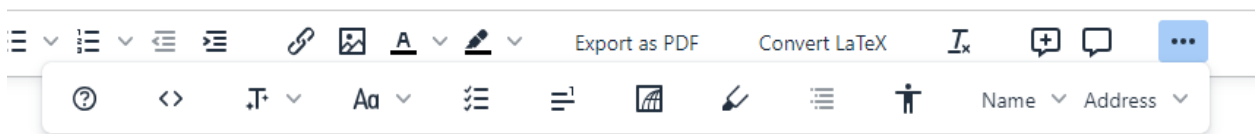
		<pre> { author: "ananth03", authorName: "ananth03", createdAt: "2024-03-25T11:13:50.906Z", content: "123qewrt", modifiedAt: "2024-03-25T11:13:50.906Z", uid: "96d067ad-d0bd-4748-a182-eecee7d32409" } }, { uid: "548d8aab-93b6-4226-952f-d1c10143d719", comments: { { author: "ananth03", authorName: "ananth03", createdAt: "2024-03-26T11:51:08.510Z", content: "test", modifiedAt: "2024-03-26T11:51:08.510Z", uid: "548d8aab-93b6-4226-952f-d1c10143d719" }, { author: "ananth03", authorName: "ananth03", createdAt: "2024-03-26T11:52:15.940Z", content: "ui,", modifiedAt: "2024-03-26T11:52:15.940Z", uid: "81a9e4cc-d226-4928-8f55-3caabb205528" } } } } </pre>	
--	--	--	--

commentListSaveIntro	List of Save	One or more variables that are updated with the commentList value when the user changes it. Use a!save() to save a modified or alternative value to a variable.	No
----------------------	--------------	---	----

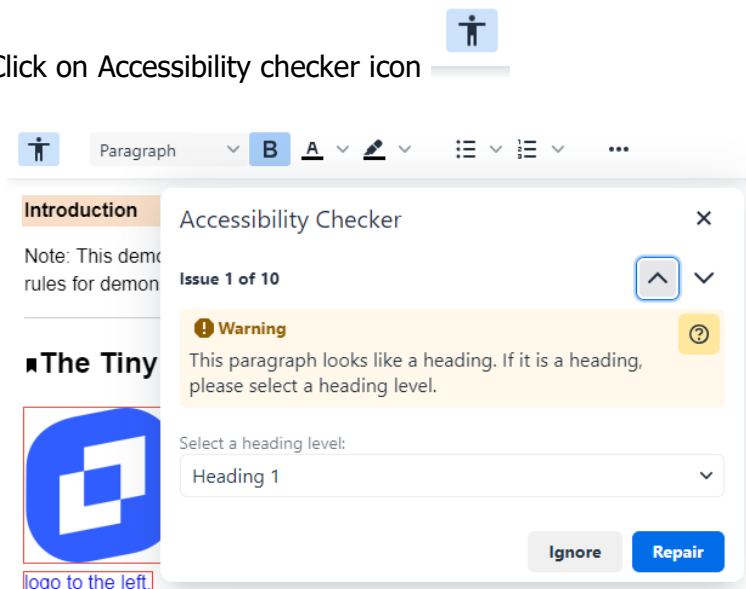
Component Usage Guide

Accessibility Checker:

Go to Additional tool bar items



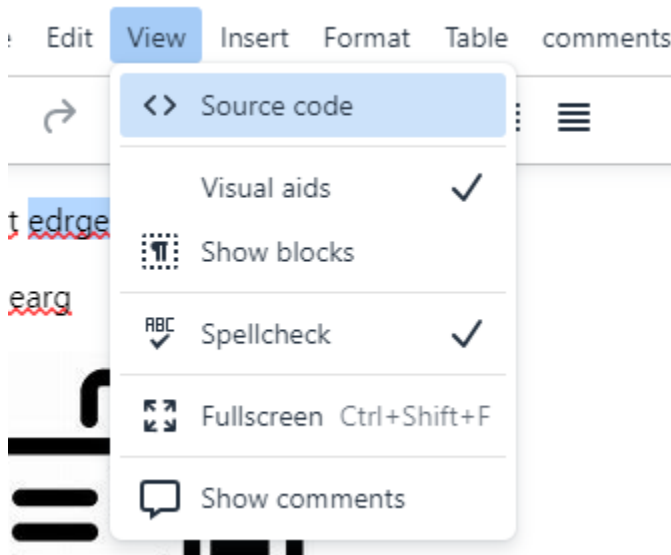
Click on Accessibility checker icon



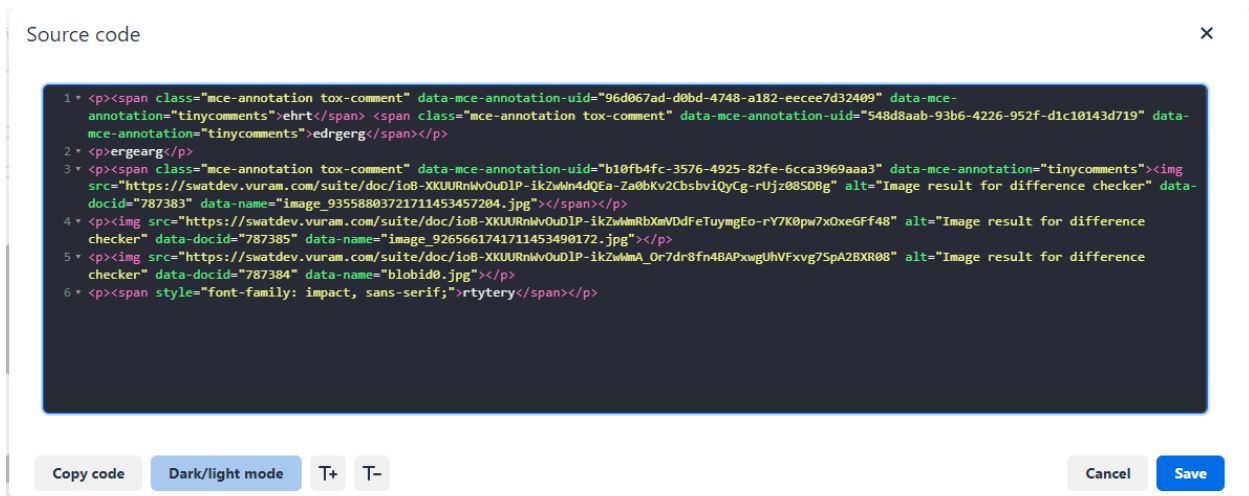
This dialog box will display the number of issues and suggestions detected by the accessibility checker. You will have the option to either ignore or repair these issues based on your preferences and requirements.

Advanced Code Editor:

Click on View ->Source Code

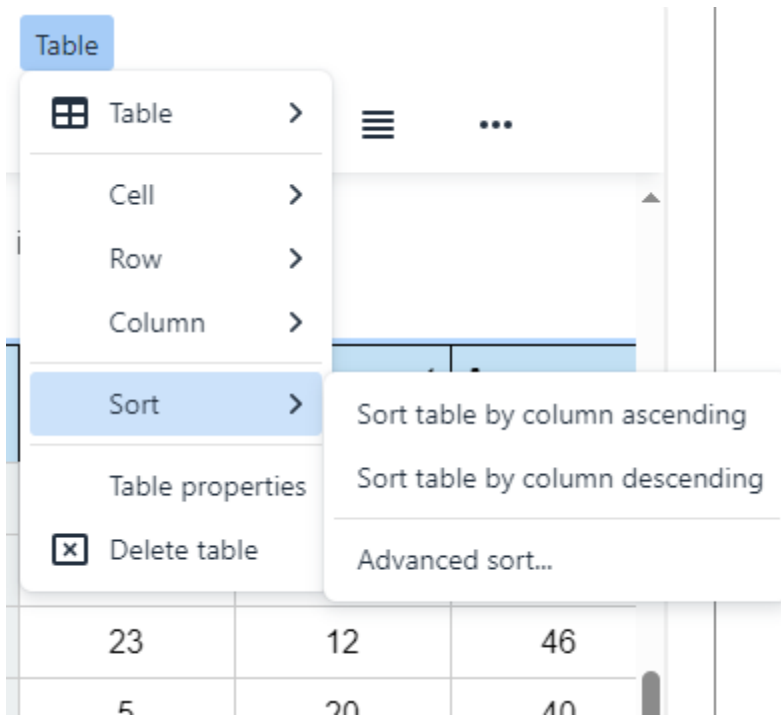


The source code is displayed here, you can make the changes and save those changes.

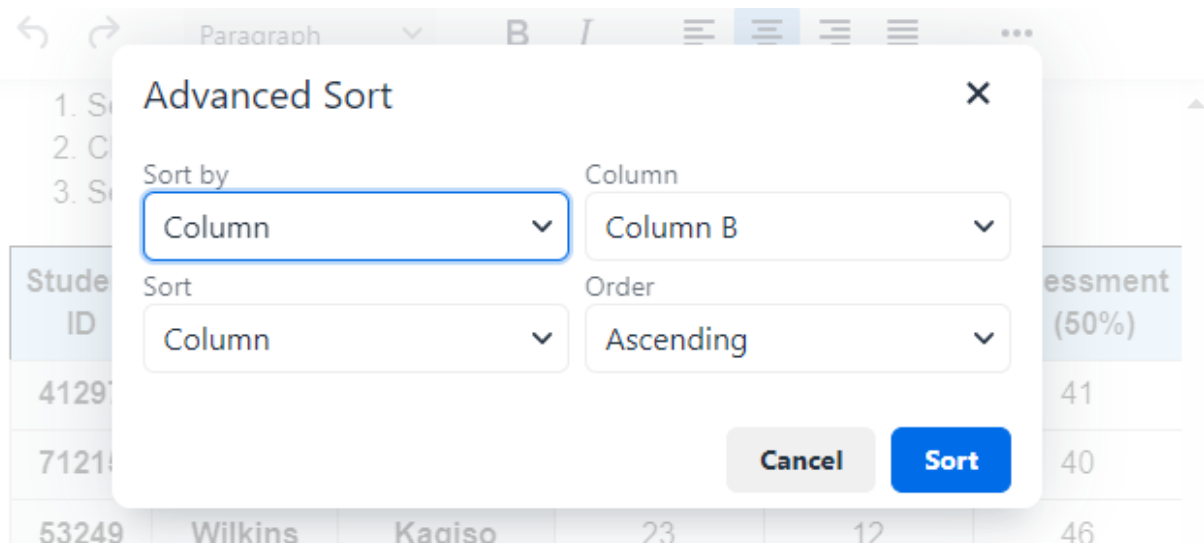


Advanced Tables:


Click View -> Table -> Sort

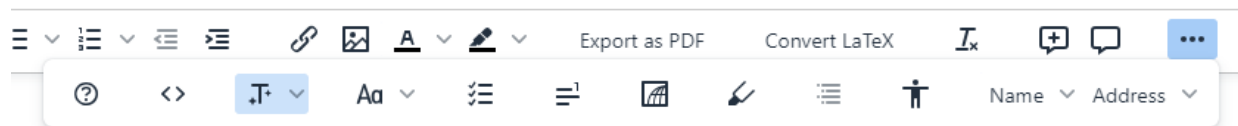


And there is an option for advanced sort

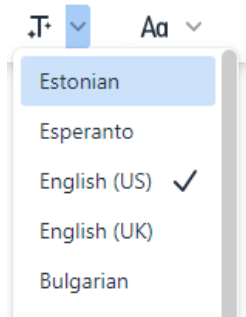


Advanced Typography:

Go to Additional tools option and click on .



Near the button, there is a dropdown menu that allows you to select the language.

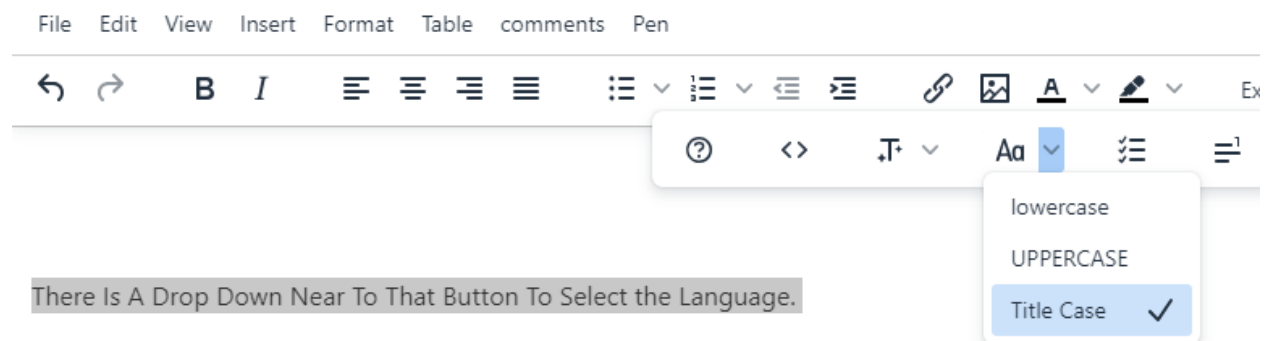


Case Change:

- Select the desired text to change the case.
- Click on the **Aa** icon in the toolbar.
- From the drop-down menu, choose the desired format by clicking on either lowercase, UPPERCASE or Title Case.

The Title Case option changes anything other than articles, coordinating conjunctions, or short prepositions to capital letters. The words that must not be capitalized can be passed to "titleCaseMinors" parameter.

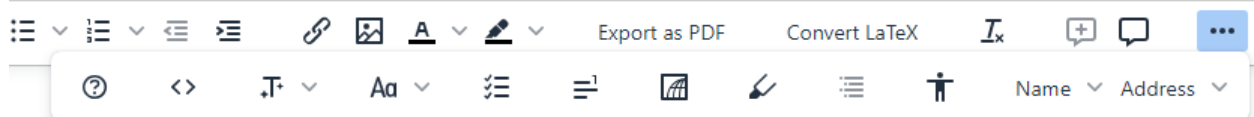
Example: {"by","from","the","at"}



In the above given text "the" is not capitalized because it is passed to "titleCaseMinors" parameter.

Checklist:

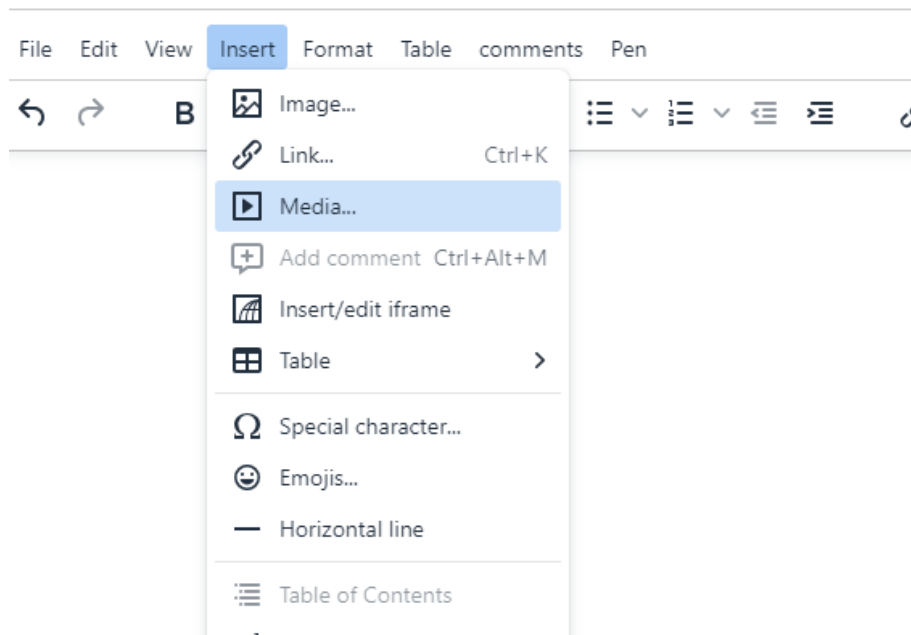
Go to additional tools option, click on checklist button



- List1
- List2
- List3
- List4
- List5

Enhanced Media Embed:

Insert -> Media



Insert/edit media

General

Embed

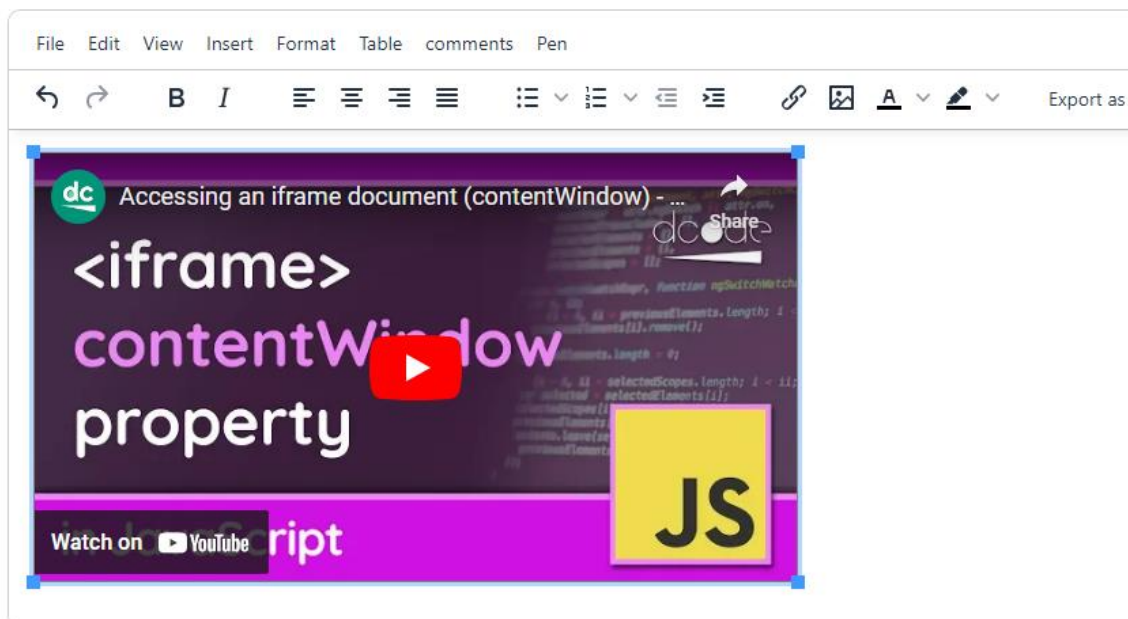
Advanced

Paste your embed code below:

```
<iframe style="border: 0px #ffffff  
none;"
```

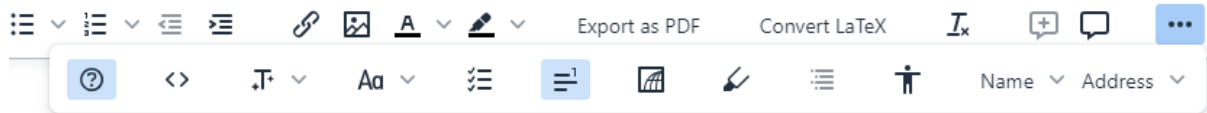
Cancel Save

sample iframe code - `<iframe width="560" height="315"
src="https://www.youtube.com/embed/-mNp3-UX9Qc"
frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-
media; gyroscope; picture-in-picture" allowfullscreen></iframe>`



Footnotes:

Go to Additional tools and select footnote option to insert the footnote for the particular text.

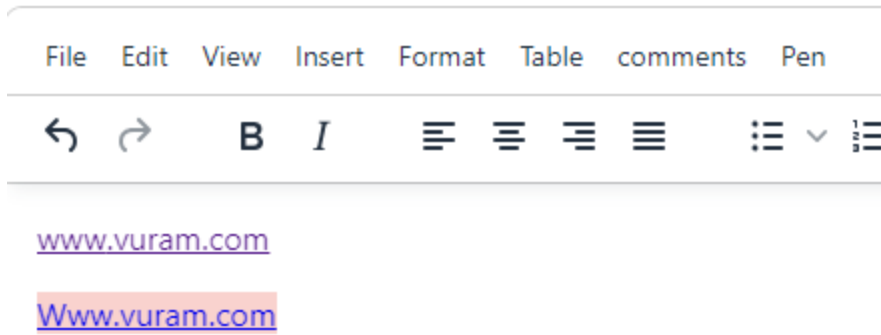


1. [^]Footnote1

Link Checker:

URLs considered invalid will be highlighted with red and will have a dedicated context menu with options to either edit the link, try and open it in a separate tab, remove the link, or ignore it.

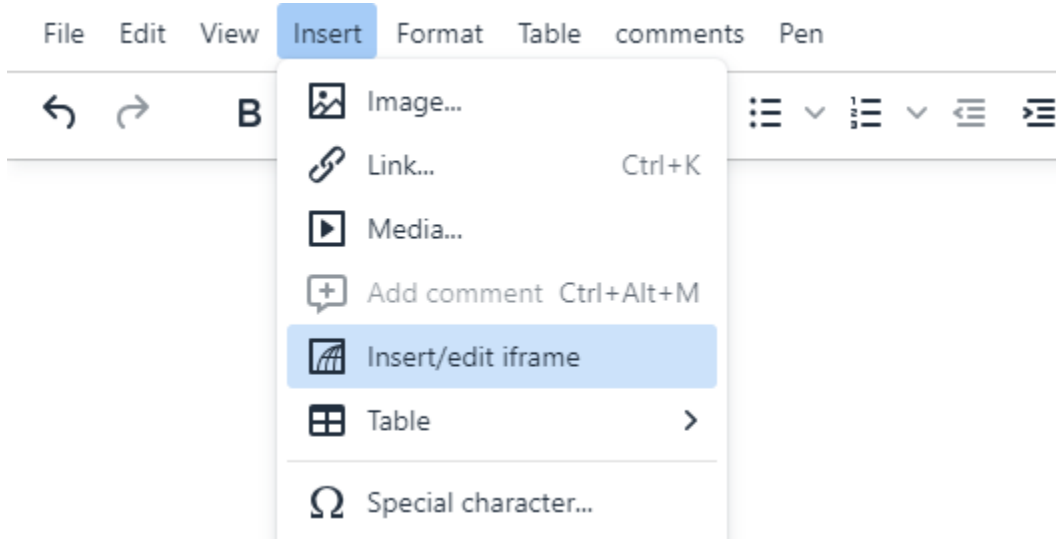
A screenshot of the 'Insert/edit link' dialog box. It has a title bar with a close button (X). The dialog contains several input fields: 'Url' with 'Www.vuram.com' and a red warning triangle icon; 'Text to display' with 'www.vuram.com'; 'Title' (empty); and 'Open link in...' with a dropdown menu set to 'Current window'. At the bottom, there are 'Cancel' and 'Save' buttons.

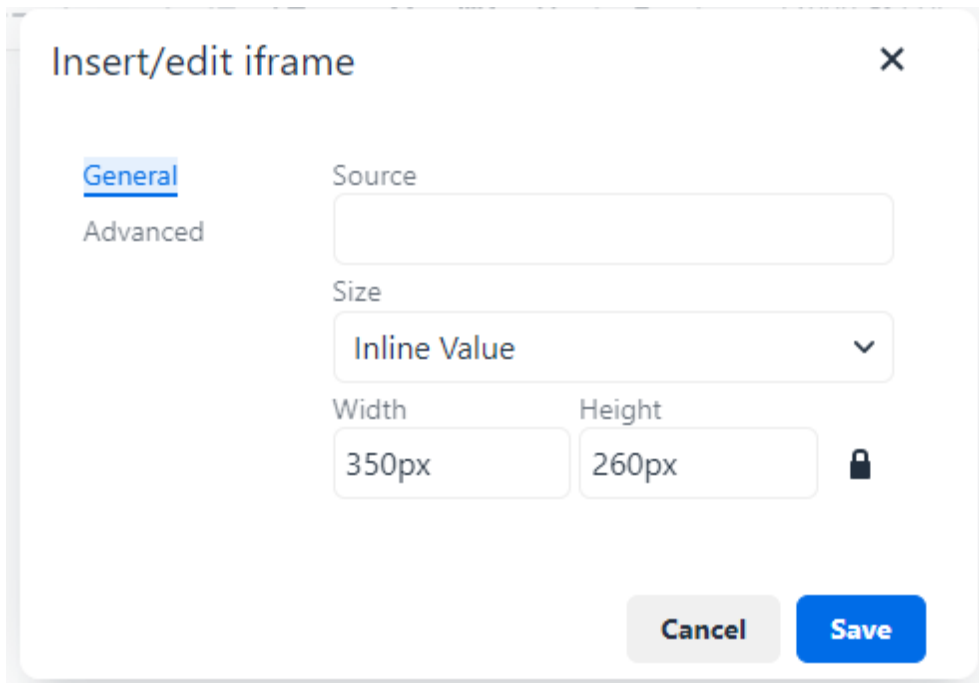


The Invalid URL is highlighted in red color.


Page Embed:

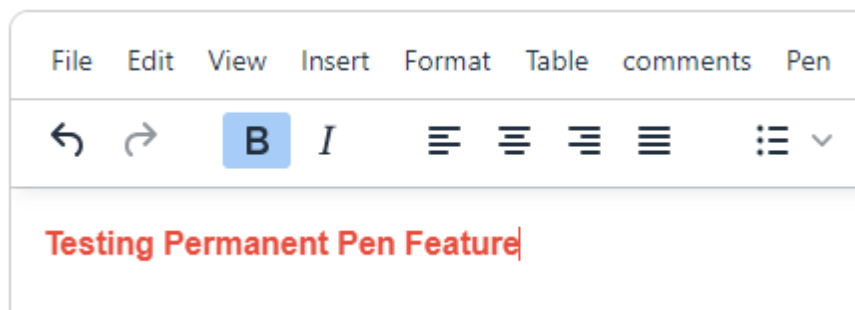
Insert-> Insert/edit iframe



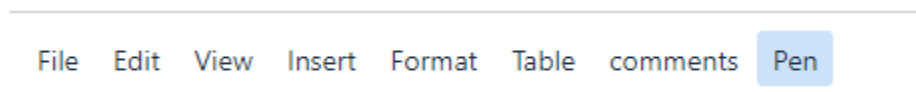


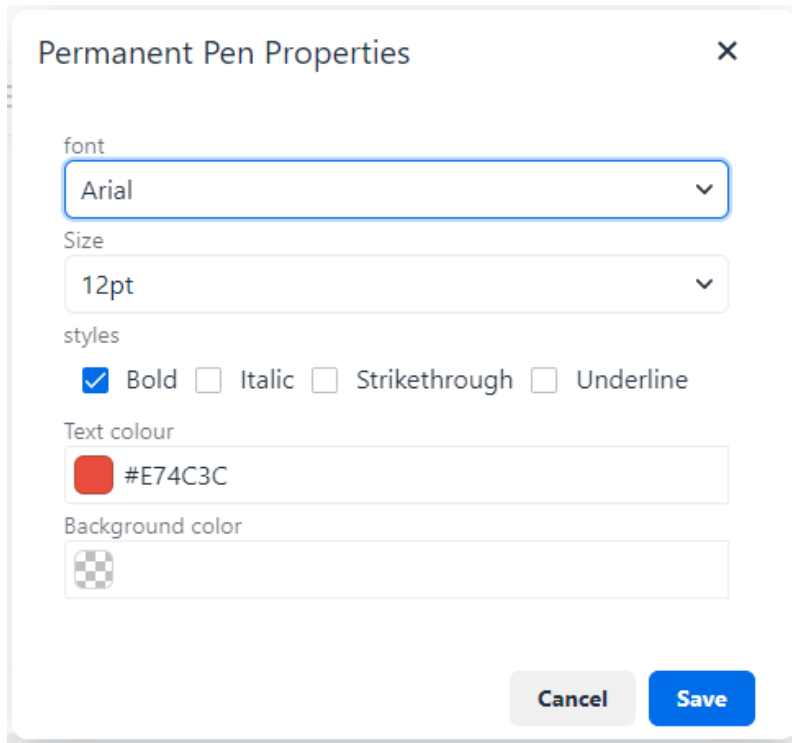
Permanent Pen:

Go to Additional Tools option and click the permanent pen icon .



Click on "Pen" button in the toolbar to customize the Permanent Pen.





PowerPaste:

When the user attempts to paste copied content, a prompt will be displayed, giving the option to paste the content with formatting or without formatting.

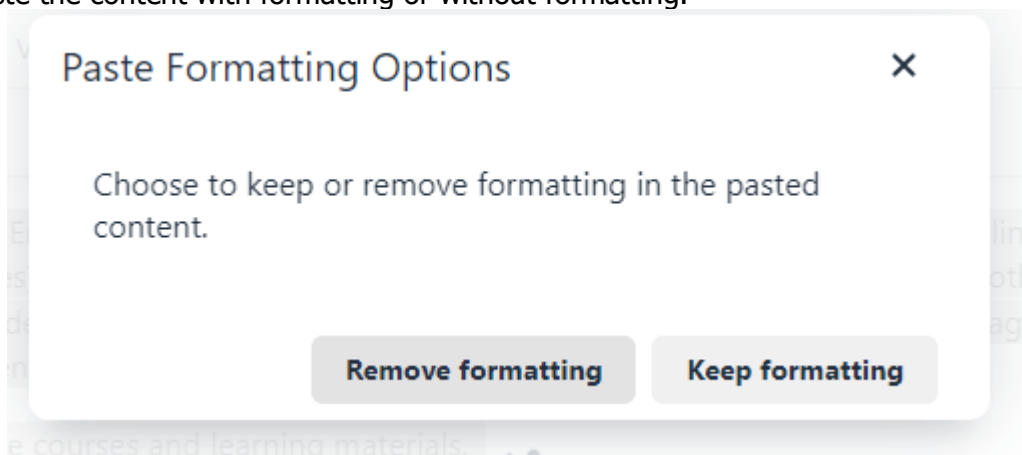
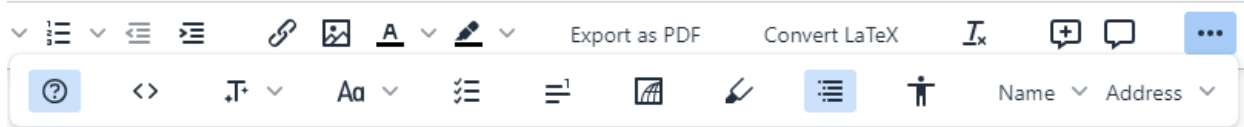
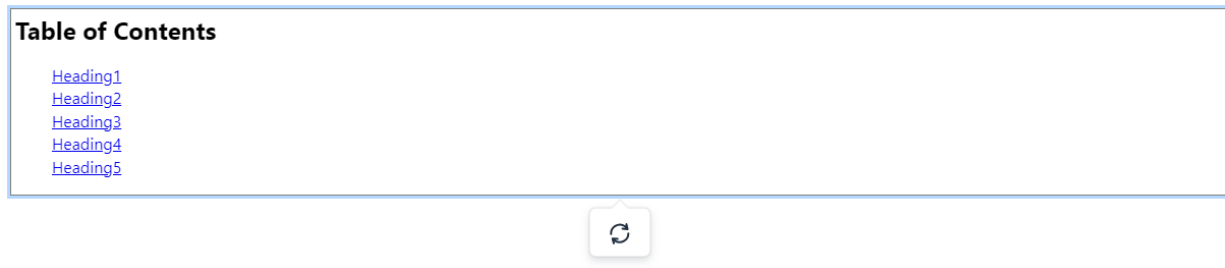


Table of Contents:

Go to Additional Tools and select Table of Contents icon .



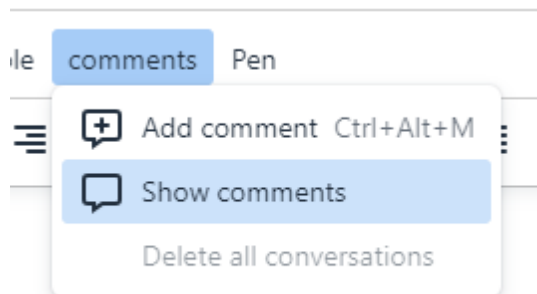
This Option will only be enabled if there are any headings (h1 – h6 elements) available in the editor.



Click on the update button to update the table of contents.

Tiny Comments:

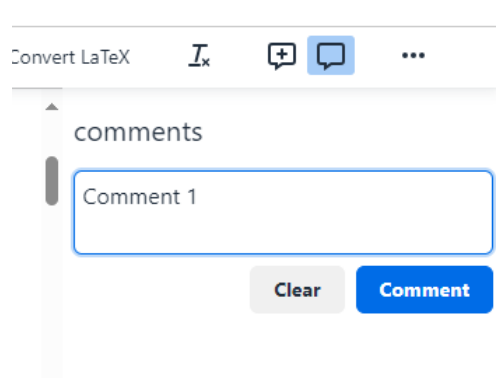
Primary Comments functions:



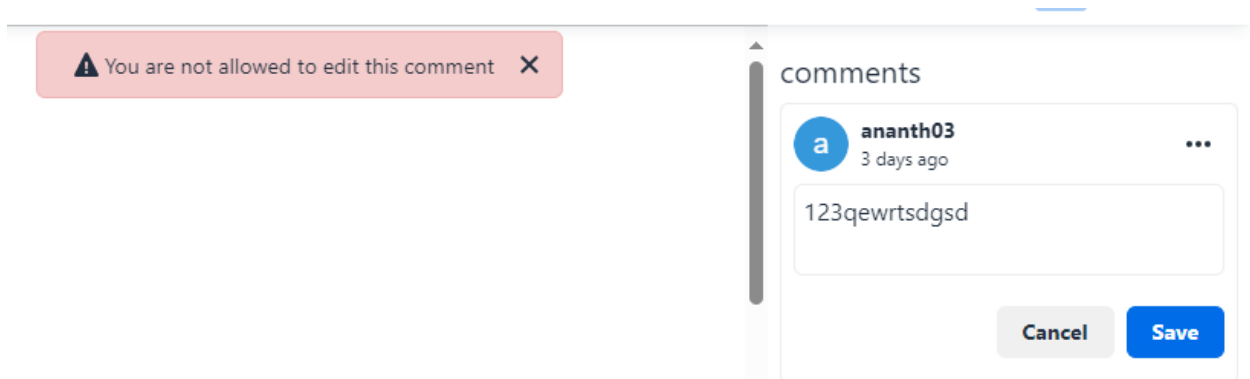
The Comments plugin allows the user to perform the following functions:

- Create a comment

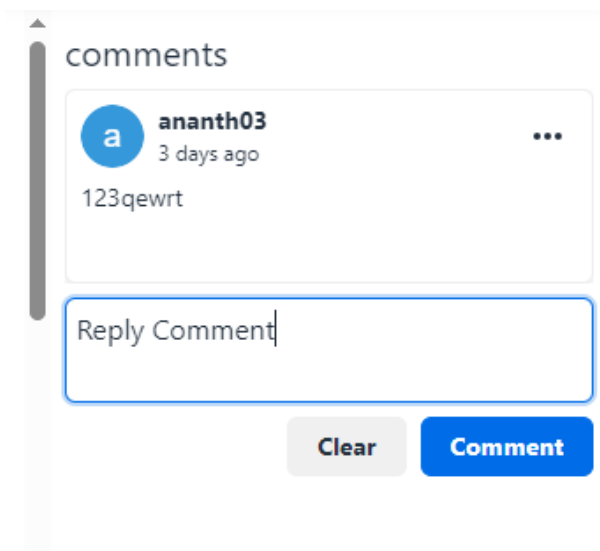
After selecting the text click on the add comment button in Comments button in toolbar or in additional tools option.

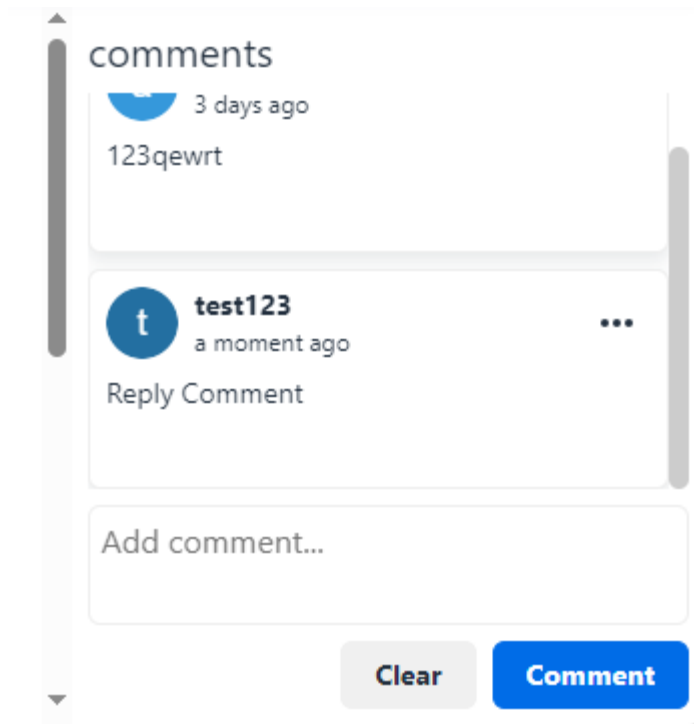


- Edit a comment: Edit the existing comment(Only the comment's author can edit the comment. If attempted by other users, a warning message will be displayed).

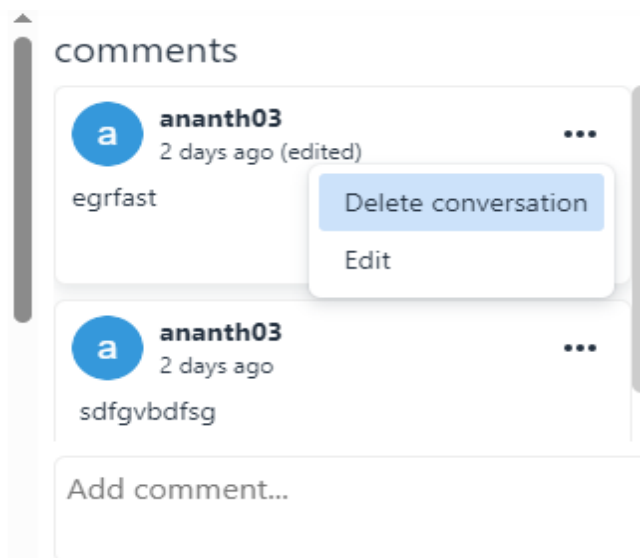


- Reply to a comment: Reply to a Conversation

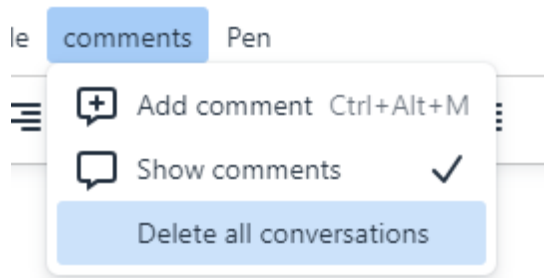




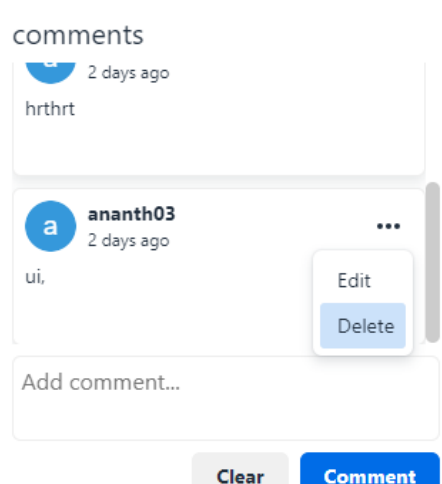
- Save the Conversations: The updated conversations can be saved with the help of “commentListSaveInto” parameter.
- Delete Comment Thread: Deleting the first comment in a comment thread (considered as the parent comment) will delete the entire thread. (Only the Admin user can delete the conversation).



- Delete All Conversations: Users have the option to delete all comment threads in the editor by selecting the "Delete All Conversations" button in the "Comments" option.



- Delete Individual Comments: Users can delete any comments they have posted within a comment thread.



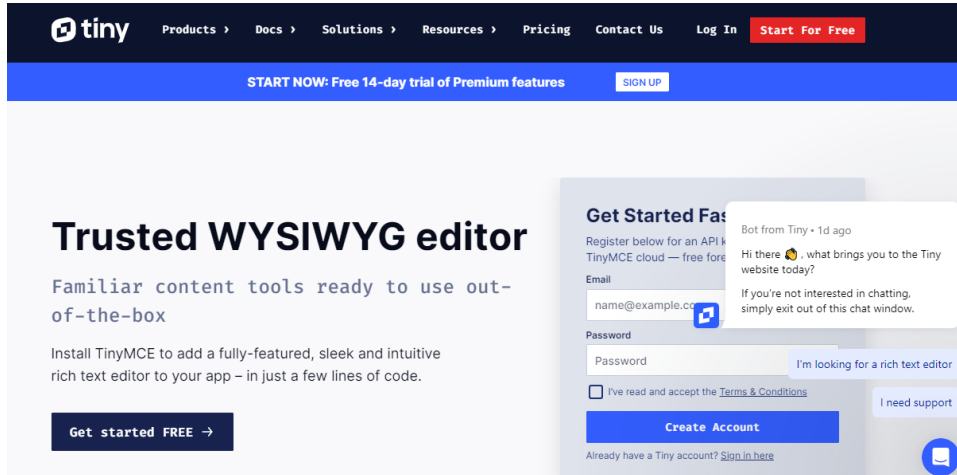
Note: When a new comment is created, a unique identifier (uid) attribute is appended to the HTML element. For example:

```
<p><span class="mce-annotation tox-comment" data-mce-annotation-uid="b10fb4fc-3576-4925-82fe-6cca3969aaa3" data-mce-annotation="tinycomments"></span></p>
```

This uid should be updated in the "commentListSaveInto" parameter. Therefore, when the user attempts to load the set of comments in the editor, ensure to pass the appropriate HTML code containing elements with the corresponding uid to the "richText" parameter.

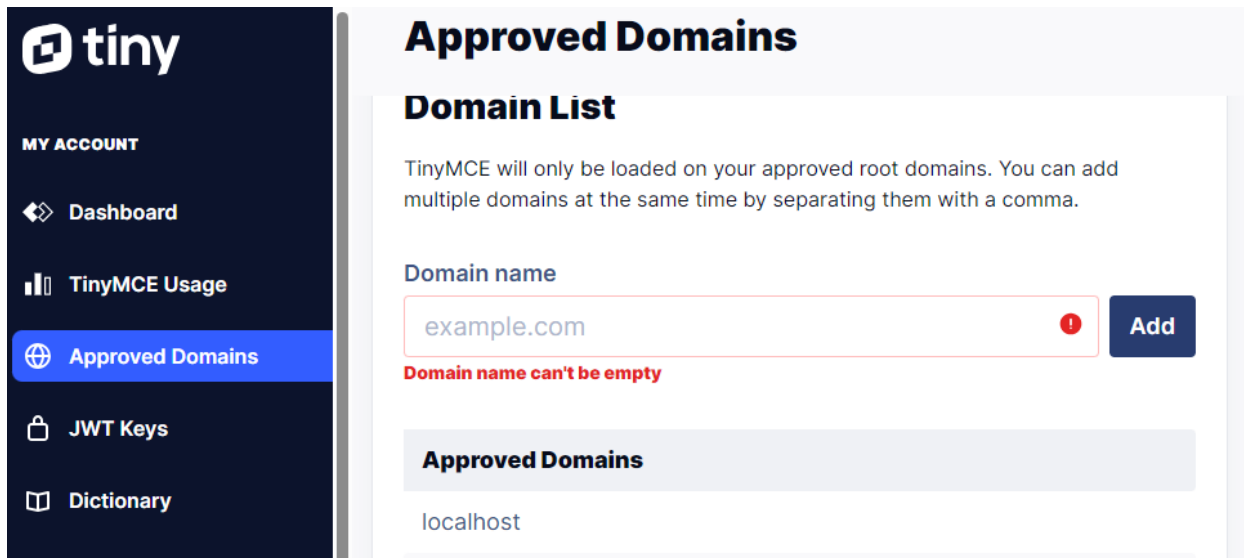
Third Party Credential Setup

Navigate to TinyMCE cloud and click on Get Started for free button to try trial version.

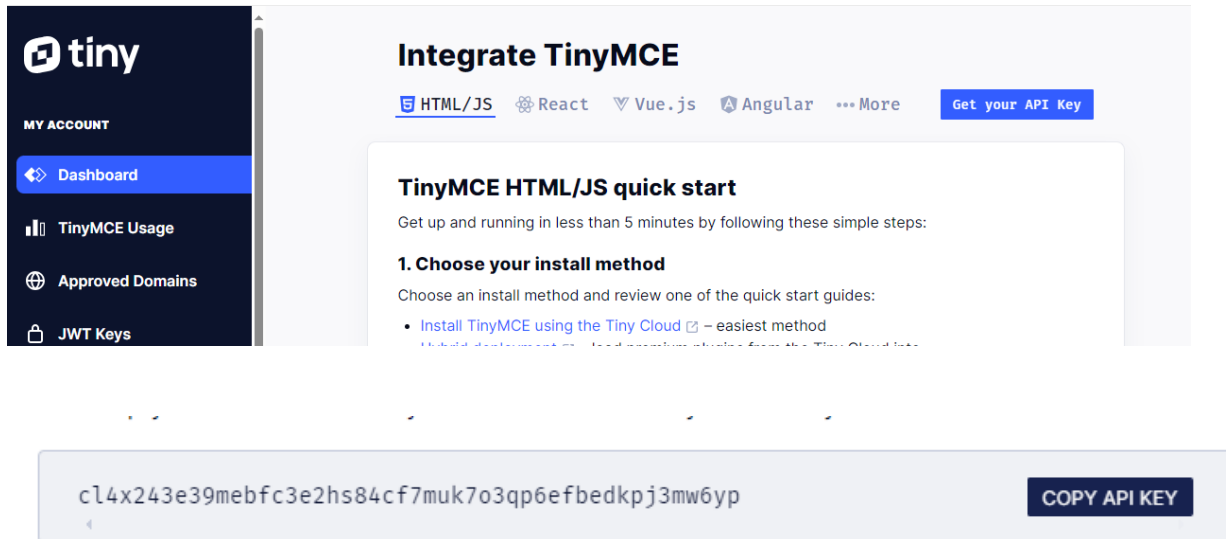


Then Sign In with your google account to get 14 days free trial.

After Signing in click on Approved Domains and add your Domain name and Click "Add".



To get the API key click navigate to Dashboard and click on "Get your API Key button"



Copy this API key and pass it to `apiKey` parameter in the expression mode.

Advanced Rich Text Editor Connected System Version Compatibility

The following table shows the compatible versions of Advanced Rich Text Editor Connected System for each Advanced Rich text Editor Premium Component version.

Sl. No	Component Version	Compatible Connected System Versions
1	1.0.0	1.1.0