ORTL at a glance:
A new Rich Text Editor for Oracle APEX

APEX World 2025 - Thursday, 20 March

WHO AM I?

Contact me

in ⊙ X 🦋

Louis Moreaux

- ➢ Senior Oracle APEX Developer at Talan (formerly Insum)
- ➢ Work with Oracle DB since 2013 and APEX since 2017
- ➢ Flows for APEX project Core Member
- ➢ Oracle ACE Pro ♠
- ➢ Writing blogs at https://lmoreaux.hashnode.dev/

insum

A TALAN Company

TALAN

Positive innovation

OUR SERVICES

We are leaders in Oracle solutions. Our team of experts specialize in Oracle APEX and OCI, helping businesses achieve innovation and transform their operations.

ORACLE APEX DEVELOPMENT

- Business Process Automation
- Enterprise Applications & Extensions
- Oracle Forms Modernization
- Client Portals
- Spreadsheet Replacement
- Data Analysis and Reporting
- E-Commerce Solutions
- AI Integrations

ORACLE CONSULTING

- Oracle APEX Audits
- Best Practices
- Security Audits and Enhancements
- Performance Tuning
- System Integration
- AI Solutions
- Training and Support

ORACLE TECHNOLOGY AND MANAGED SOLUTIONS

- Lift & Shift to Oracle Cloud Infrastructure
- Oracle Upgrades
- Infrastructure Managed Services
- Oracle Cloud Infrastructure
- OCVS – Oracle Cloud VMWare Solution Integration
- Oracle License Assistance

Introduction

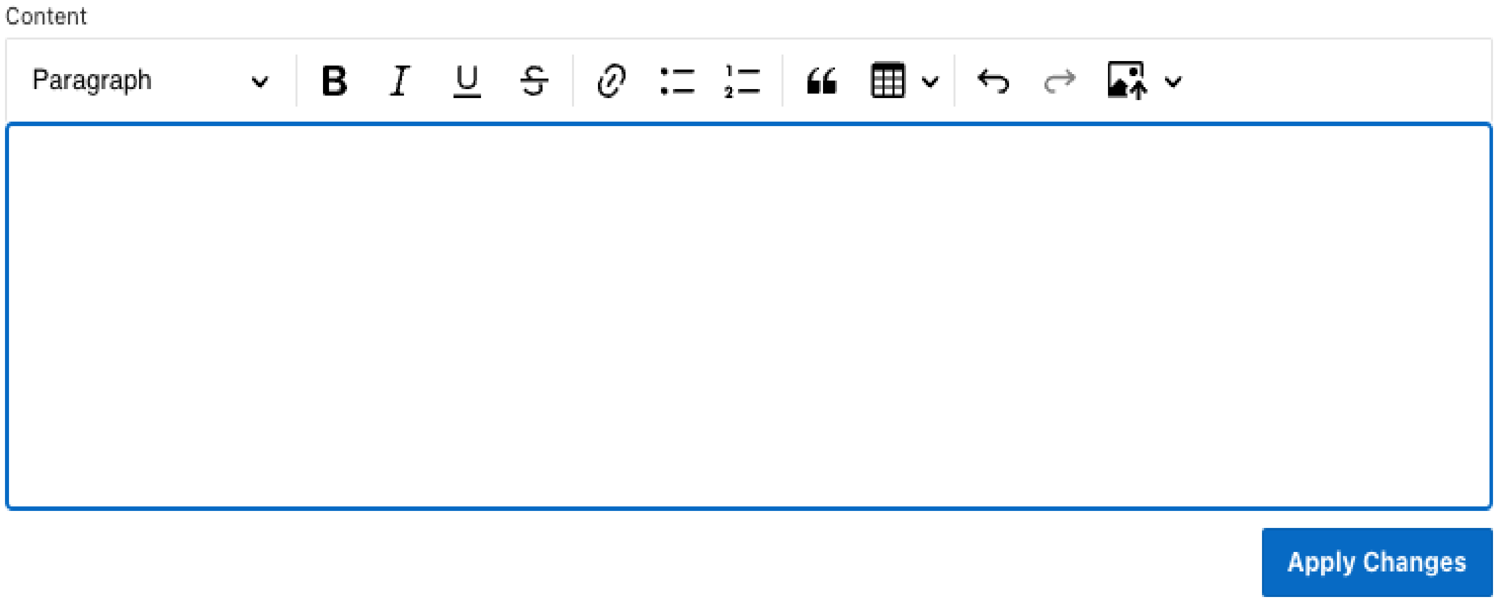
WHAT ARE WE TALKING ABOUT?

- RTE = Rich Text Editor
- Page item
- Support HTML and Markdown
- ORTL (Oracle Rich Text Library) = fork of CKEditor5

A.23 Oracle Rich Text Library (ORTL) 1.0.2 License

Copyright (c) 2003-2023, CKSource - Frederico Knabben. All rights reserved.

Parent topic: Open Source Software License Text

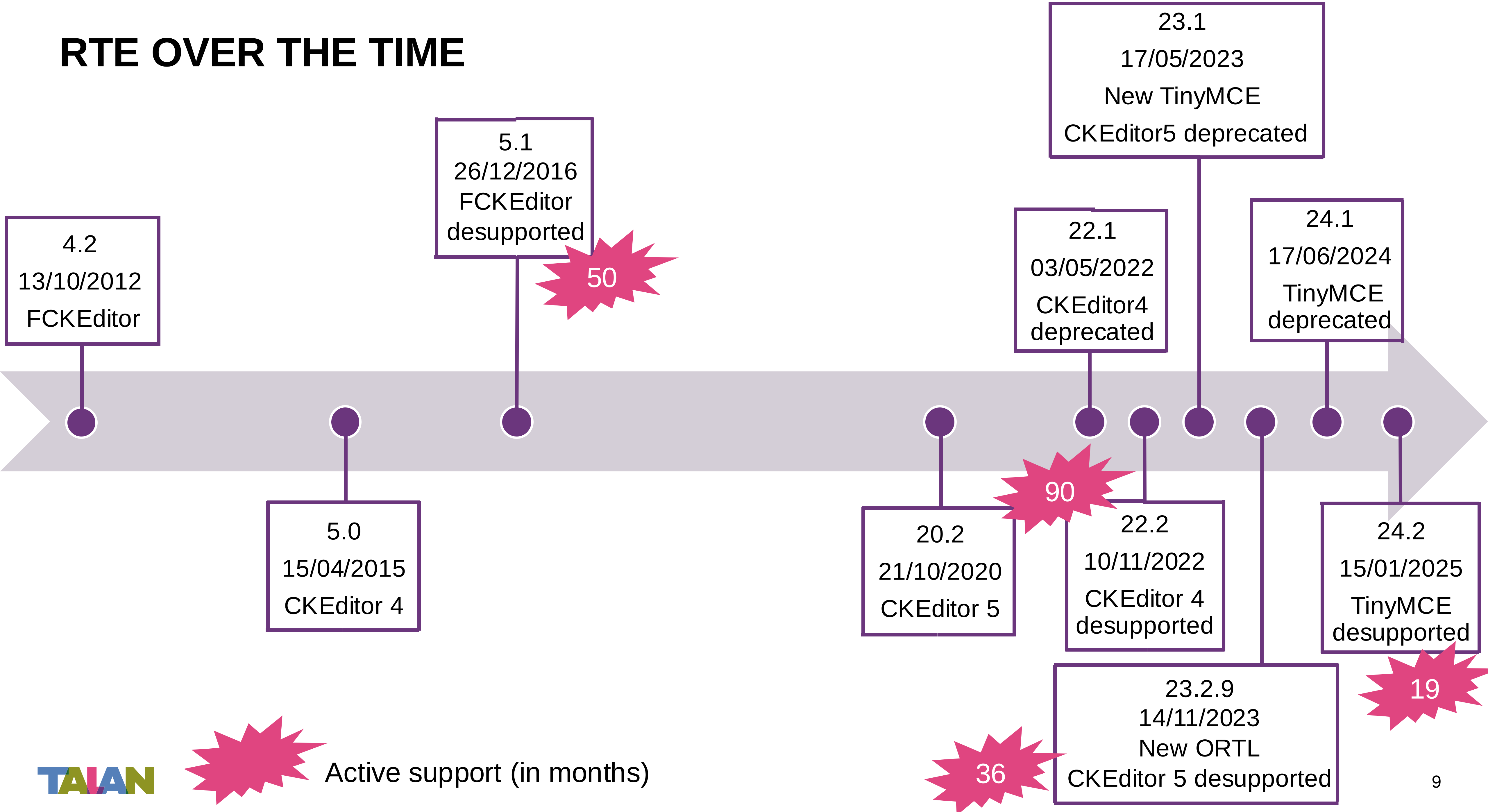


HIGHLY CUSTOMIZABLE

- Custom toolbar
- Image upload
- Autosave
- Mention aka @someone
- Word and characters count

RTE over the time

RTE OVER THE TIME



4.2
13/10/2012
FCKEditor

5.0
15/04/2015
CKEditor 4

5.1
26/12/2016
FCKEditor desupported

20.2
21/10/2020
CKEditor 5

22.1
03/05/2022
CKEditor4 deprecated

22.2
10/11/2022
CKEditor 4 desupported

23.1
17/05/2023
New TinyMCE
CKEditor5 deprecated

23.2.9
14/11/2023
New ORTL
CKEditor 5 desupported

24.1
17/06/2024
TinyMCE deprecated

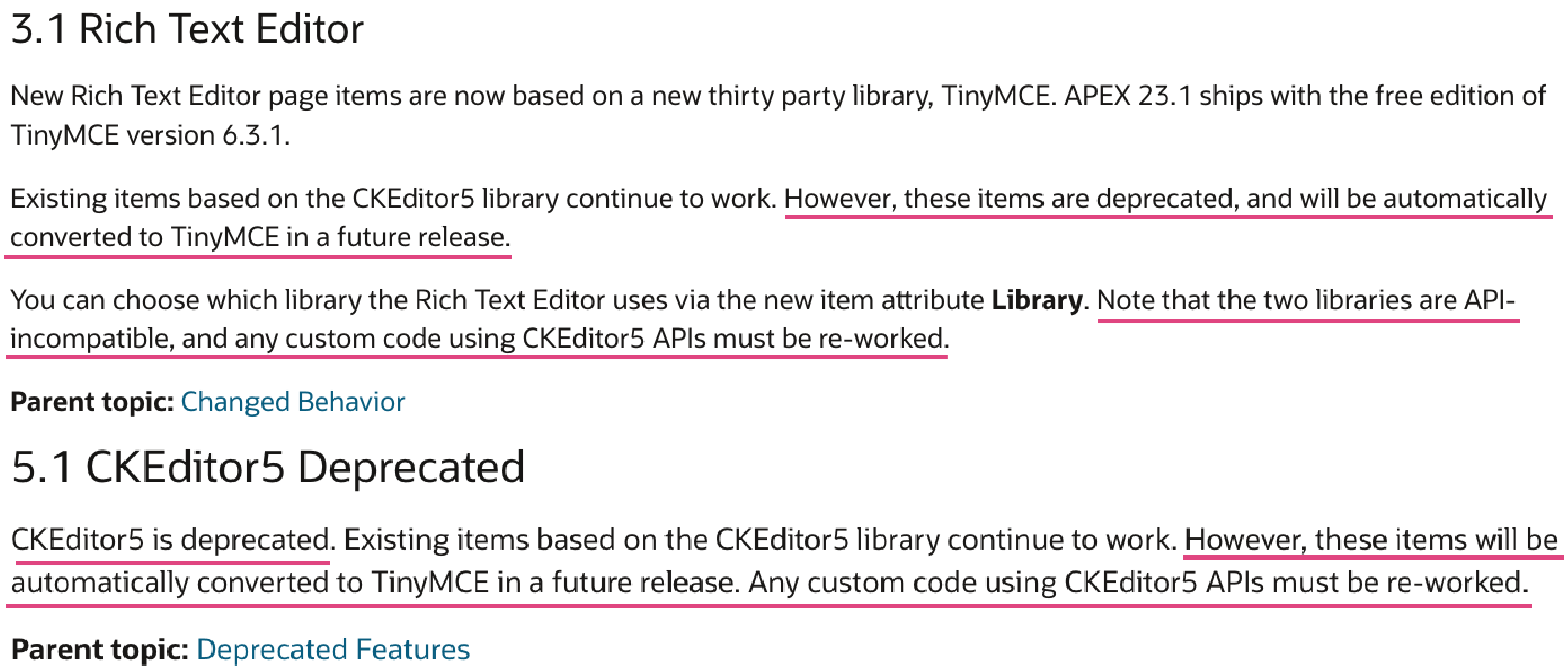
24.2
15/01/2025
TinyMCE desupported

Active support (in months)

What is the strategy for an easy migration (if you have customized your editor)?

MIGRATION FROM 22.2 TO 23.1 (17/05/2023)

1. Read the release note 👇



3.1 Rich Text Editor

New Rich Text Editor page items are now based on a new thirty party library, TinyMCE. APEX 23.1 ships with the free edition of TinyMCE version 6.3.1.

Existing items based on the CKEditor5 library continue to work. However, these items are deprecated, and will be automatically converted to TinyMCE in a future release.

You can choose which library the Rich Text Editor uses via the new item attribute **Library**. Note that the two libraries are API-incompatible, and any custom code using CKEditor5 APIs must be re-worked.

Parent topic: Changed Behavior

5.1 CKEditor5 Deprecated

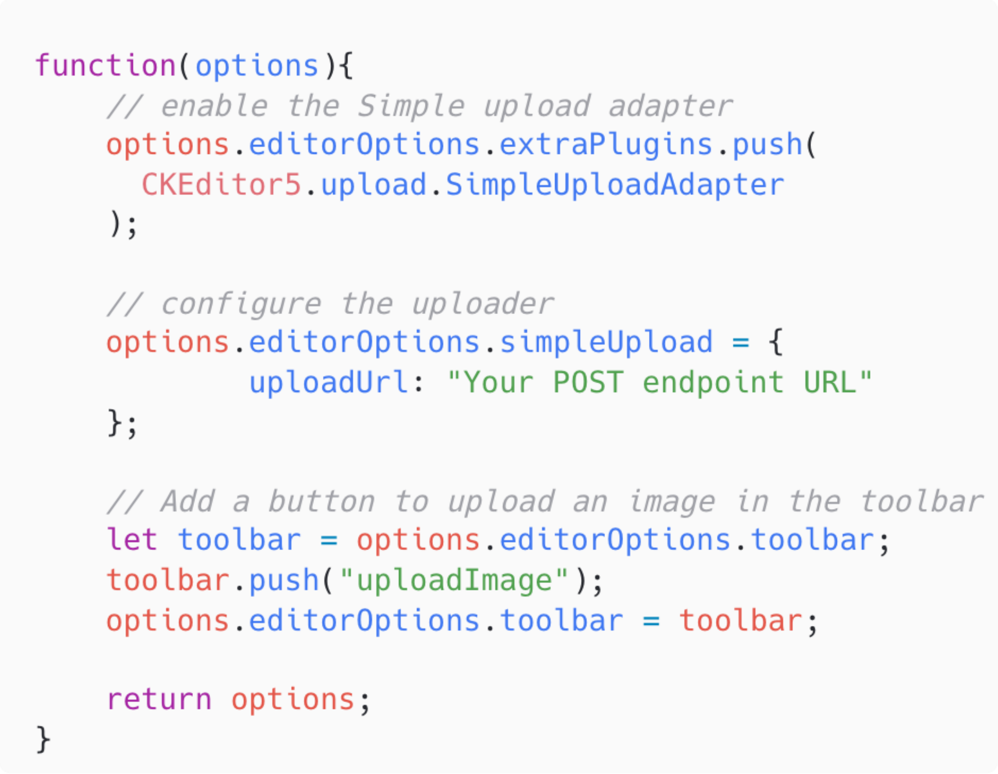
CKEditor5 is deprecated. Existing items based on the CKEditor5 library continue to work. However, these items will be automatically converted to TinyMCE in a future release. Any custom code using CKEditor5 APIs must be re-worked.

Parent topic: Deprecated Features

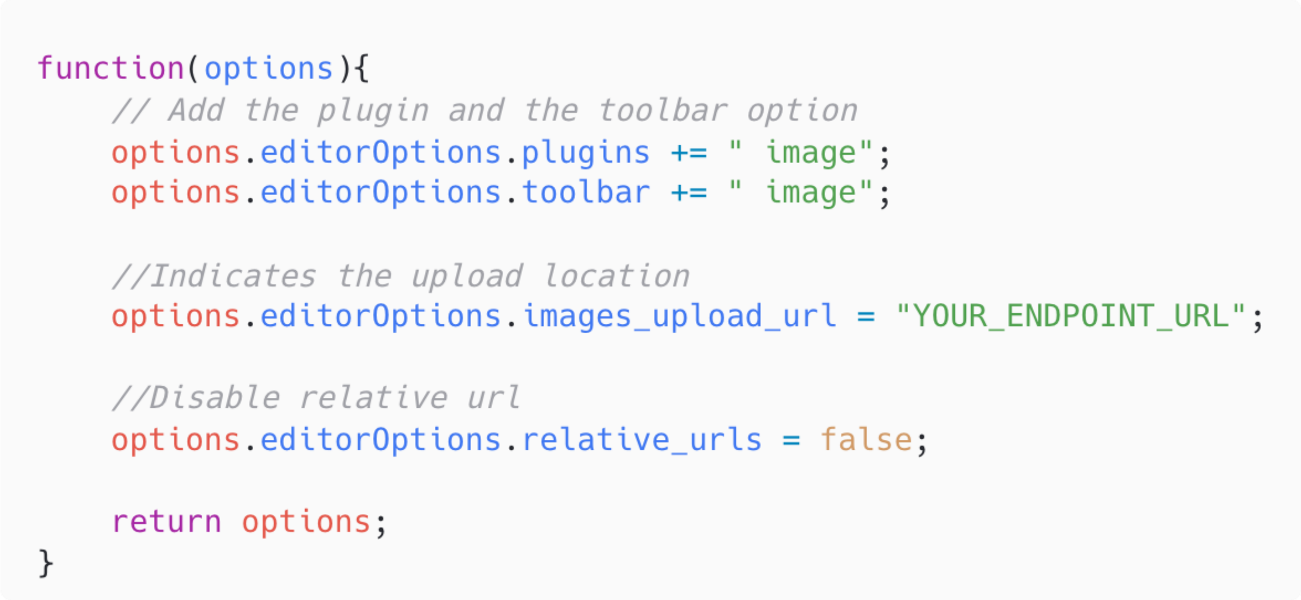
https://docs.oracle.com/en/database/oracle/apex/23.1/htmrn/about-release-notes.html

MIGRATION FROM 22.2 TO 23.1 (17/05/2023)

1. Read the release note
2. Migrate CKEditor5 to TinyMCE (learning curve, time, cost)



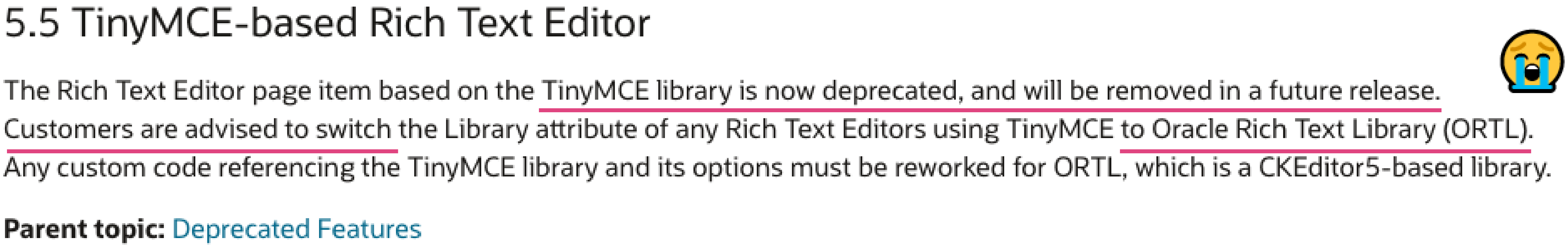
```
function(options){
    // enable the Simple upload adapter
    options.editorOptions.extraPlugins.push(
      CKEditor5.upload.SimpleUploadAdapter
    );

    // configure the uploader
    options.editorOptions.simpleUpload = {
            uploadUrl: "Your POST endpoint URL"
    };

    // Add a button to upload an image in the toolbar
    let toolbar = options.editorOptions.toolbar;
    toolbar.push("uploadImage");
    options.editorOptions.toolbar = toolbar;

    return options;
}
```



```
function(options){
    // Add the plugin and the toolbar option
    options.editorOptions.plugins += " image";
    options.editorOptions.toolbar += " image";

    //Indicates the upload location
    options.editorOptions.images_upload_url = "YOUR_ENDPOINT_URL";

    //Disable relative url
    options.editorOptions.relative_urls = false;

    return options;
}
```

MIGRATION FROM 23.1 TO 24.1 (17/06/2024)

1. Read the release note 👇



5.5 TinyMCE-based Rich Text Editor

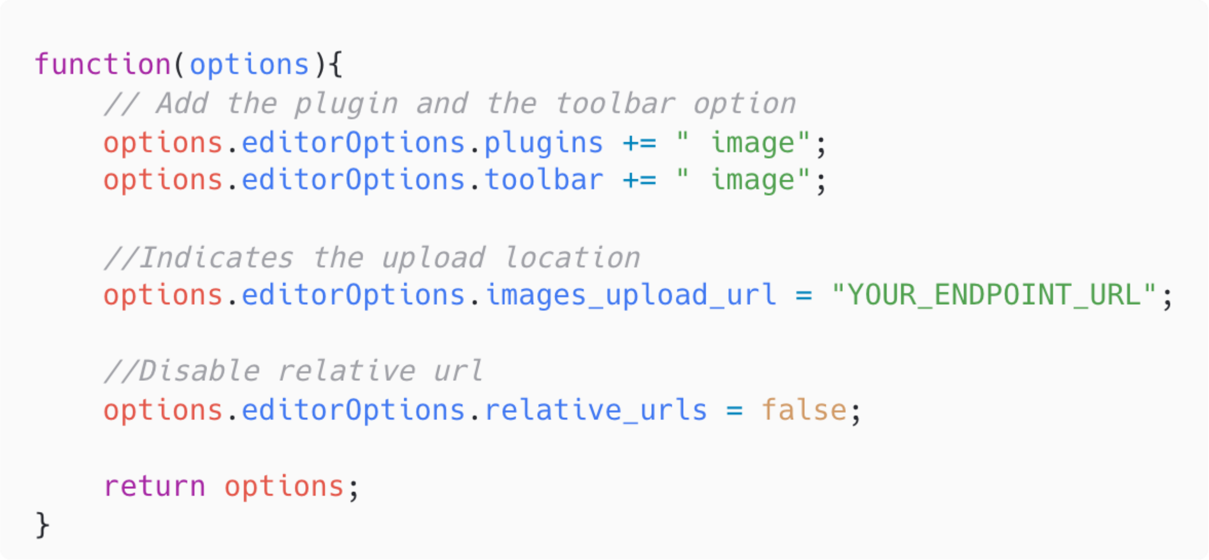
😭

The Rich Text Editor page item based on the TinyMCE library is now deprecated, and will be removed in a future release. Customers are advised to switch the Library attribute of any Rich Text Editors using TinyMCE to Oracle Rich Text Library (ORTL). Any custom code referencing the TinyMCE library and its options must be reworked for ORTL, which is a CKEditor5-based library.

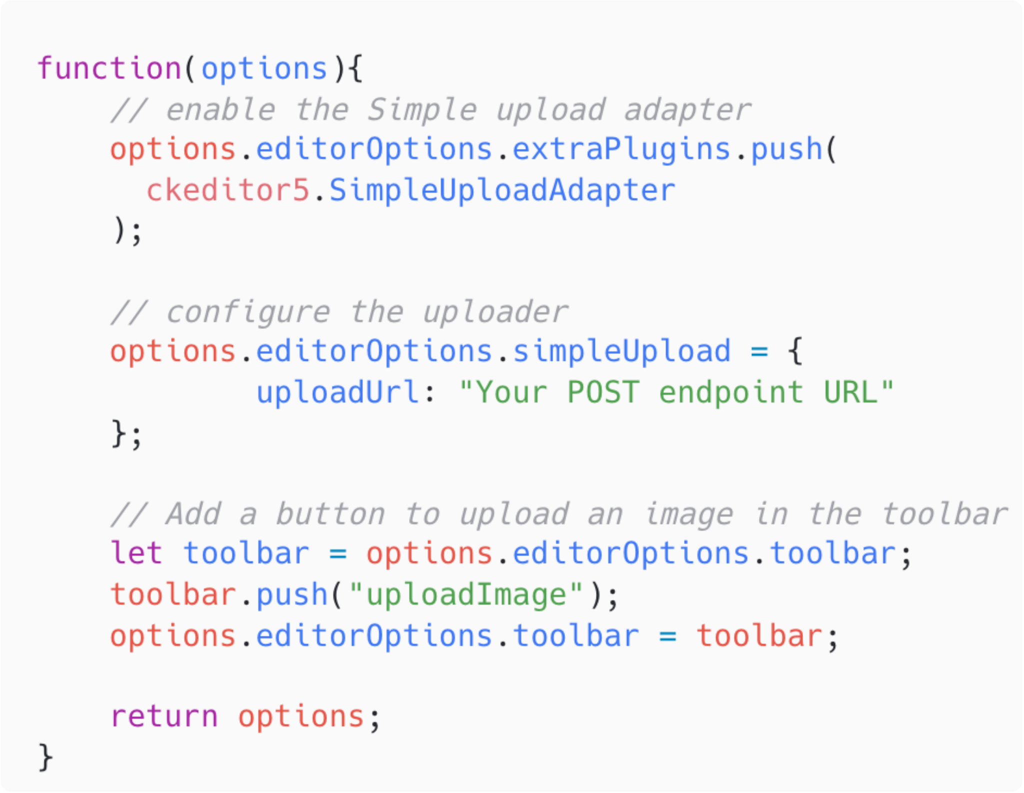
Parent topic: Deprecated Features

MIGRATION FROM 23.1 TO 24.1 (17/06/2024)

1. Read the release note
2. Migrate TinyMCE to ORTL (learning curve, time, cost)



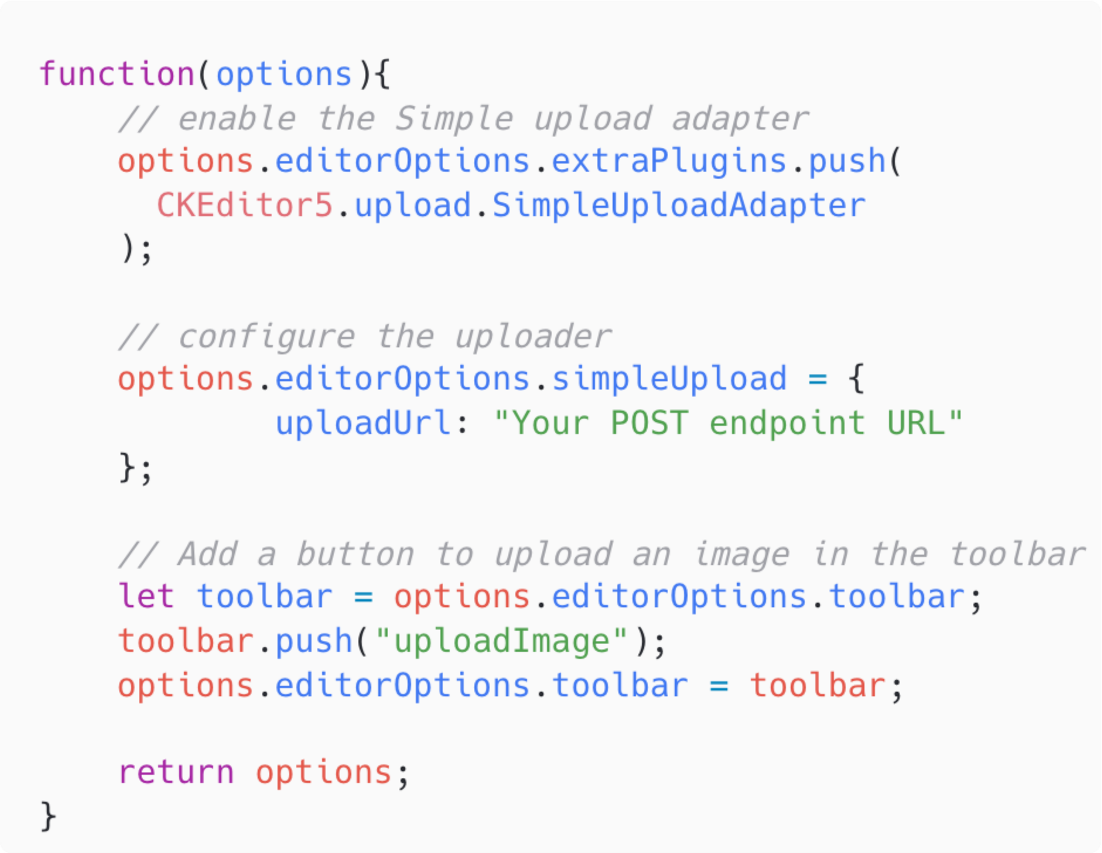
```
function(options){
    // Add the plugin and the toolbar option
    options.editorOptions.plugins += " image";
    options.editorOptions.toolbar += " image";

    //Indicates the upload location
    options.editorOptions.images_upload_url = "YOUR_ENDPOINT_URL";

    //Disable relative url
    options.editorOptions.relative_urls = false;

    return options;
}
```



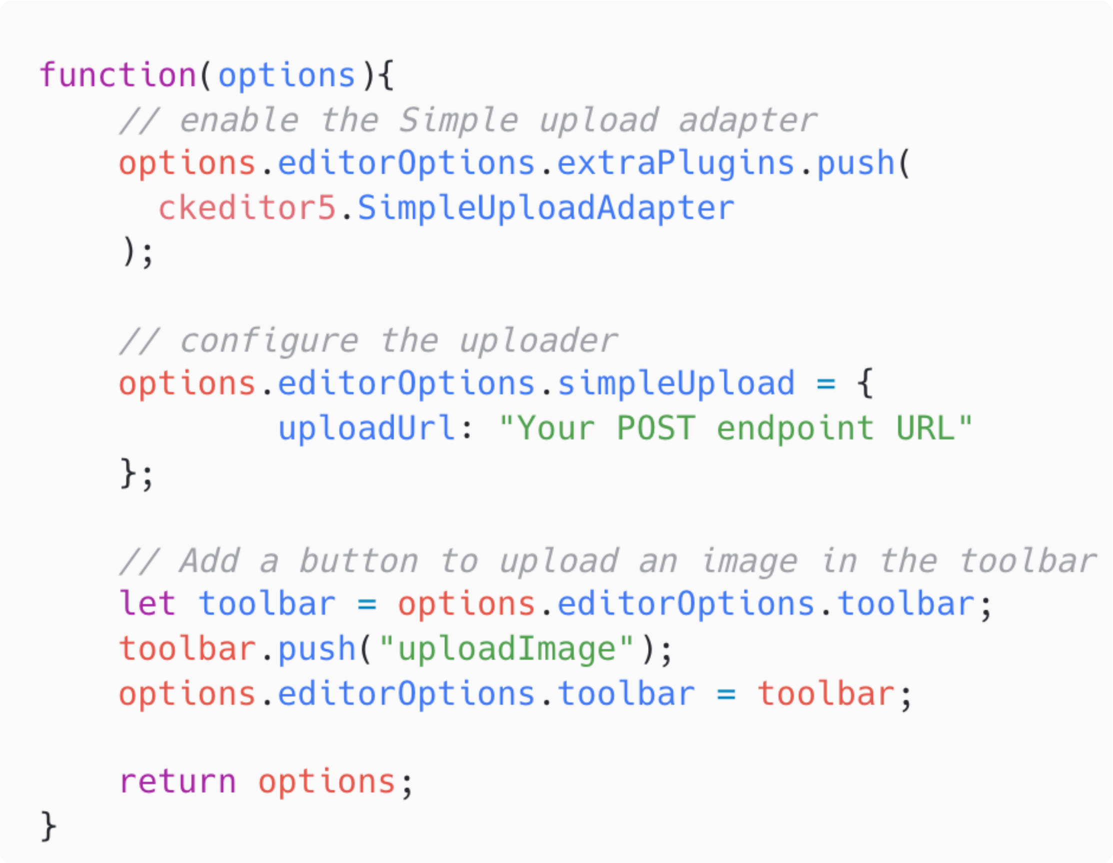
```
function(options){
    // enable the Simple upload adapter
    options.editorOptions.extraPlugins.push(
      ckeditor5.SimpleUploadAdapter
    );

    // configure the uploader
    options.editorOptions.simpleUpload = {
            uploadUrl: "Your POST endpoint URL"
    };

    // Add a button to upload an image in the toolbar
    let toolbar = options.editorOptions.toolbar;
    toolbar.push("uploadImage");
    options.editorOptions.toolbar = toolbar;

    return options;
}
```

Have you noticed anything?

LOOK CLOSELY



```
function(options){
    // enable the Simple upload adapter
    options.editorOptions.extraPlugins.push(
      CKEditor5.upload.SimpleUploadAdapter
    );

    // configure the uploader
    options.editorOptions.simpleUpload = {
            uploadUrl: "Your POST endpoint URL"
    };

    // Add a button to upload an image in the toolbar
    let toolbar = options.editorOptions.toolbar;
    toolbar.push("uploadImage");
    options.editorOptions.toolbar = toolbar;

    return options;
}
```



```
function(options){
    // enable the Simple upload adapter
    options.editorOptions.extraPlugins.push(
      ckeditor5.SimpleUploadAdapter
    );

    // configure the uploader
    options.editorOptions.simpleUpload = {
            uploadUrl: "Your POST endpoint URL"
    };

    // Add a button to upload an image in the toolbar
    let toolbar = options.editorOptions.toolbar;
    toolbar.push("uploadImage");
    options.editorOptions.toolbar = toolbar;

    return options;
}
```

CKEditor5: 22.2

ORTL: 24.1

THE RIGHT PATH (KNOWING ALL WE LEARNED)

Migration from <= 22.2 to 23.1

Keep CKEditor5 🤓

~~Migrate CKEditor5 to TinyMCE (learning curve, time, cost)~~

Migration from 23.1 to 24.1+

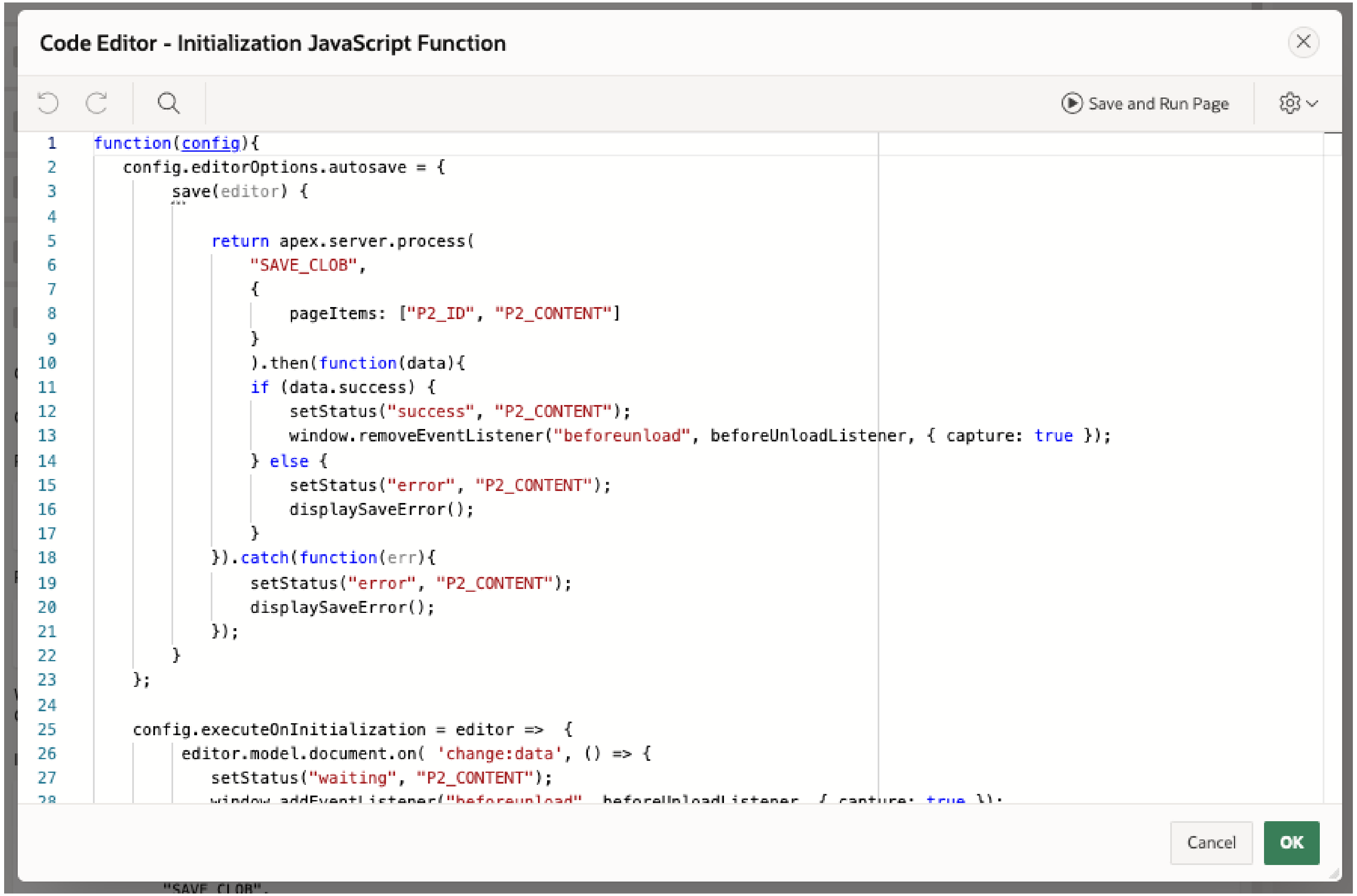
Migrate CKEditor5 to ORTL 🤓

~~Migrate TinyMCE to ORTL (learning curve, time, cost)~~

RTE customization

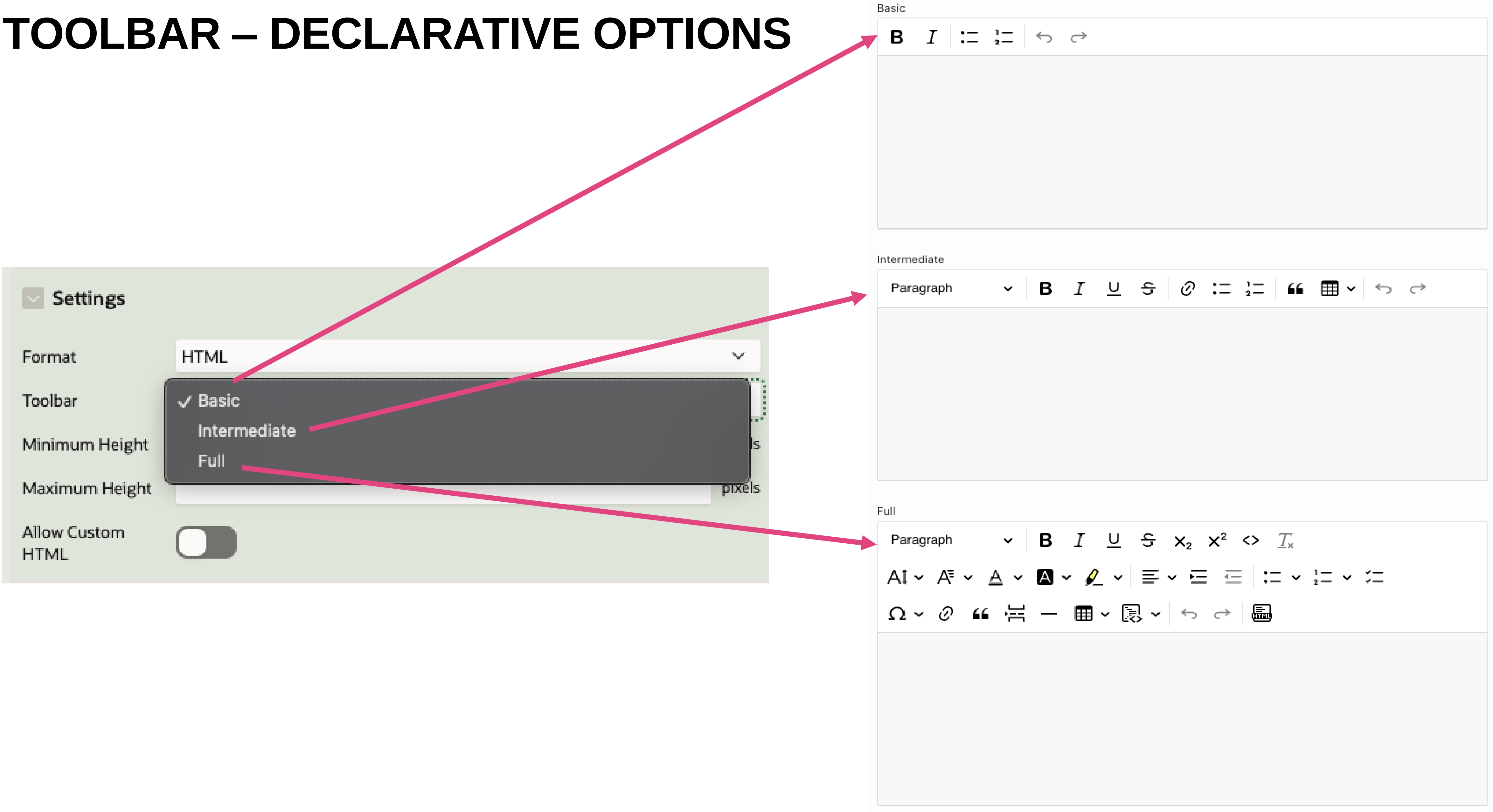
HOW TO CUSTOMIZE IT?

Add the code in the Initialization JavaScript Function attribute



```
function(config){
   config.editorOptions.autosave = {
        save(editor) {

            return apex.server.process(
                "SAVE_CLOB",
                {
                    pageItems: ["P2_ID", "P2_CONTENT"]
                }
                ).then(function(data){
                if (data.success) {
                    setStatus("success", "P2_CONTENT");
                    window.removeEventListener("beforeunload", beforeUnloadListener, { capture: true });
                } else {
                    setStatus("error", "P2_CONTENT");
                    displaySaveError();
                }
            }).catch(function(err){
                setStatus("error", "P2_CONTENT");
                displaySaveError();
            });
        }
    };

    config.executeOnInitialization = editor =>  {
         editor.model.document.on( 'change:data', () => {
            setStatus("waiting", "P2_CONTENT");
            window.addEventListener("beforeunload", beforeUnloadListener, { capture: true });
```

TOOLBAR – DECLARATIVE OPTIONS



Settings

Format: HTML

Toolbar: ✓ Basic / Intermediate / Full

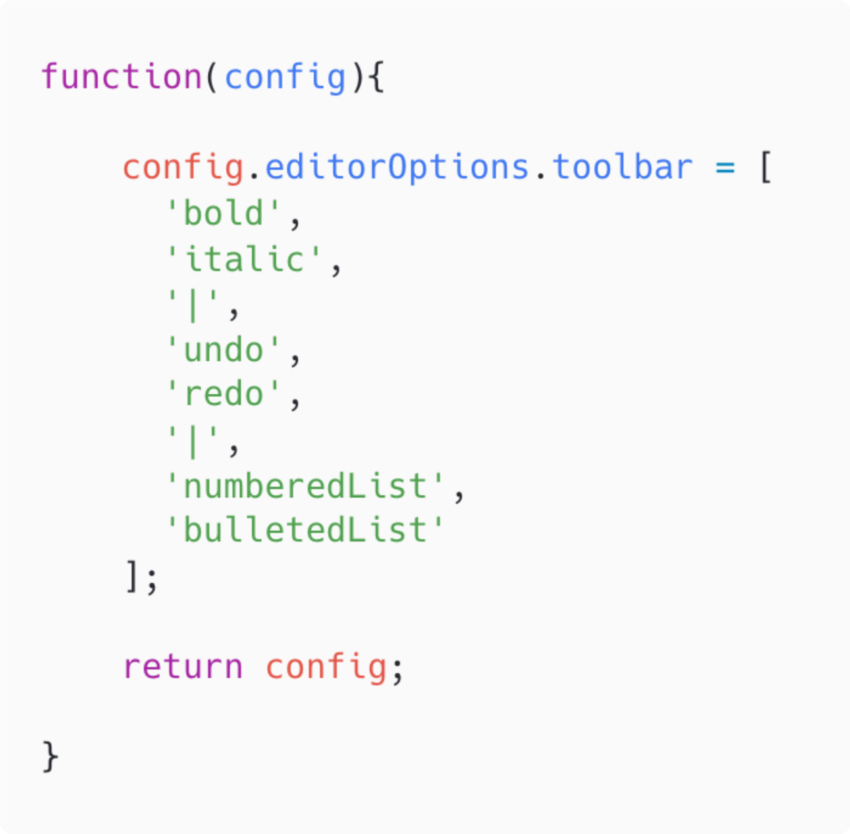
Minimum Height

Maximum Height — pixels

Allow Custom HTML: ○

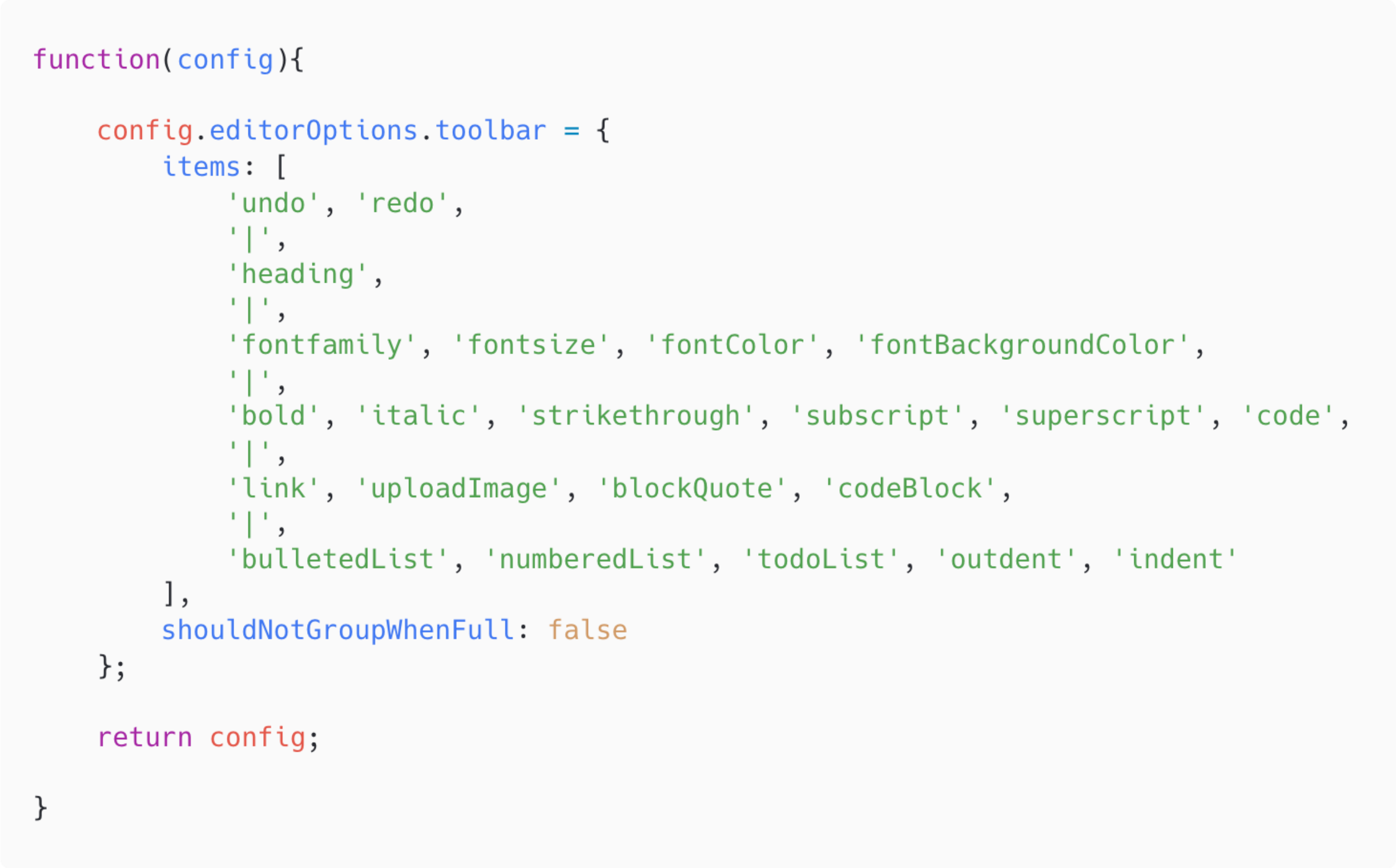
TOOLBAR – CONFIGURATION: TWO AVAILABLE FORMATS

Basic



```
function(config){

    config.editorOptions.toolbar = [
      'bold',
      'italic',
      '|',
      'undo',
      'redo',
      '|',
      'numberedList',
      'bulletedList'
    ];

    return config;

}
```

Extended



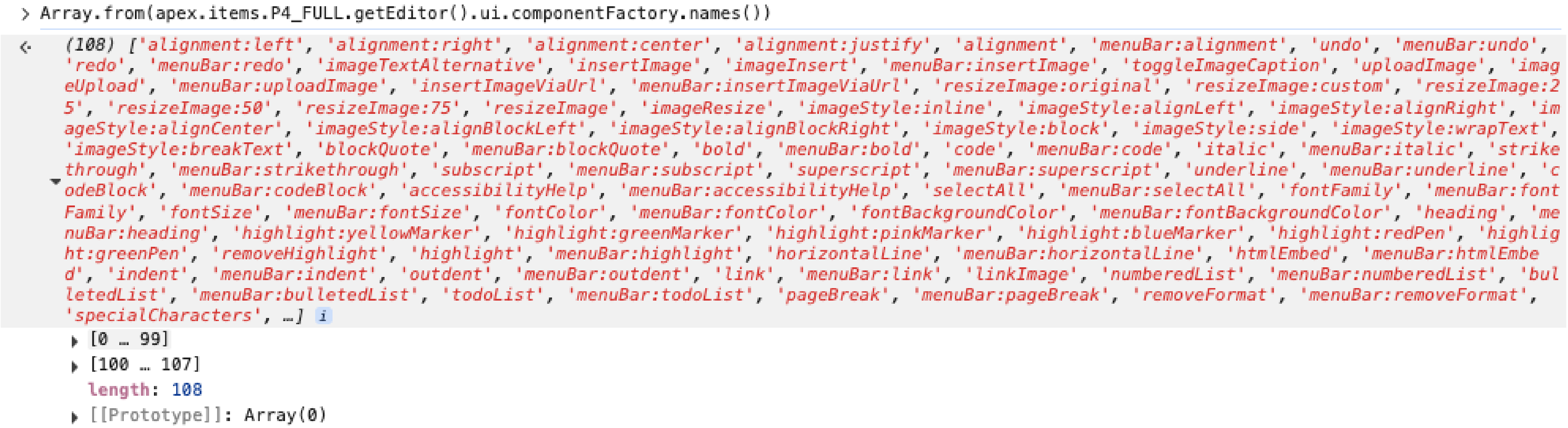
```
function(config){

    config.editorOptions.toolbar = {
        items: [
            'undo', 'redo',
            '|',
            'heading',
            '|',
            'fontfamily', 'fontsize', 'fontColor', 'fontBackgroundColor',
            '|',
            'bold', 'italic', 'strikethrough', 'subscript', 'superscript', 'code',
            '|',
            'link', 'uploadImage', 'blockQuote', 'codeBlock',
            '|',
            'bulletedList', 'numberedList', 'todoList', 'outdent', 'indent'
        ],
        shouldNotGroupWhenFull: false
    };

    return config;

}
```

https://ckeditor.com/docs/ckeditor5/latest/getting-started/setup/toolbar.html

TOOLBAR – WHAT ARE MY OPTIONS?

```
// List of available options for an editor
Array.from(apex.items.PAGE_ITEM.getEditor().ui.componentFactory.names())
```

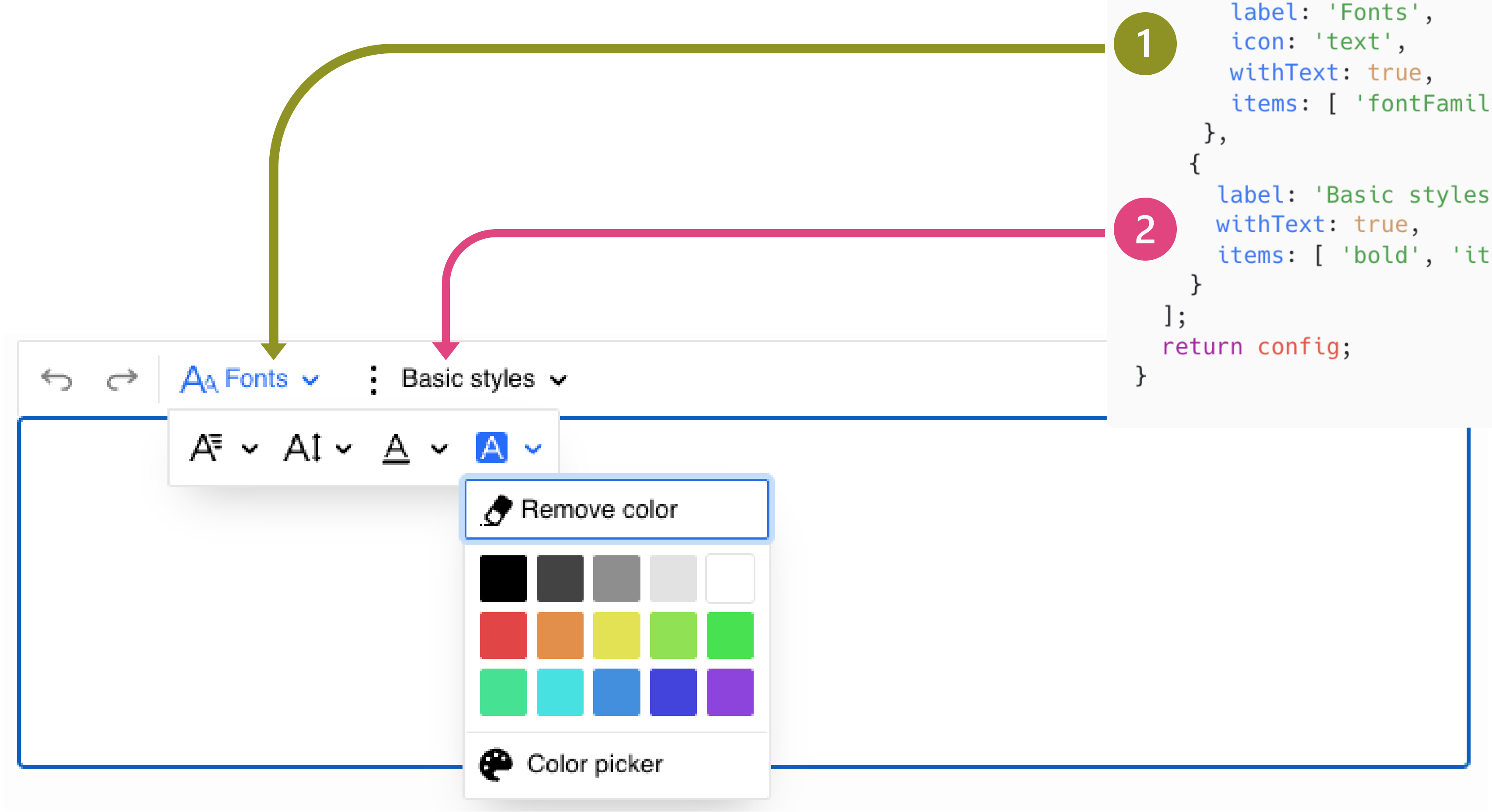
JavaScript console (browser development tools)



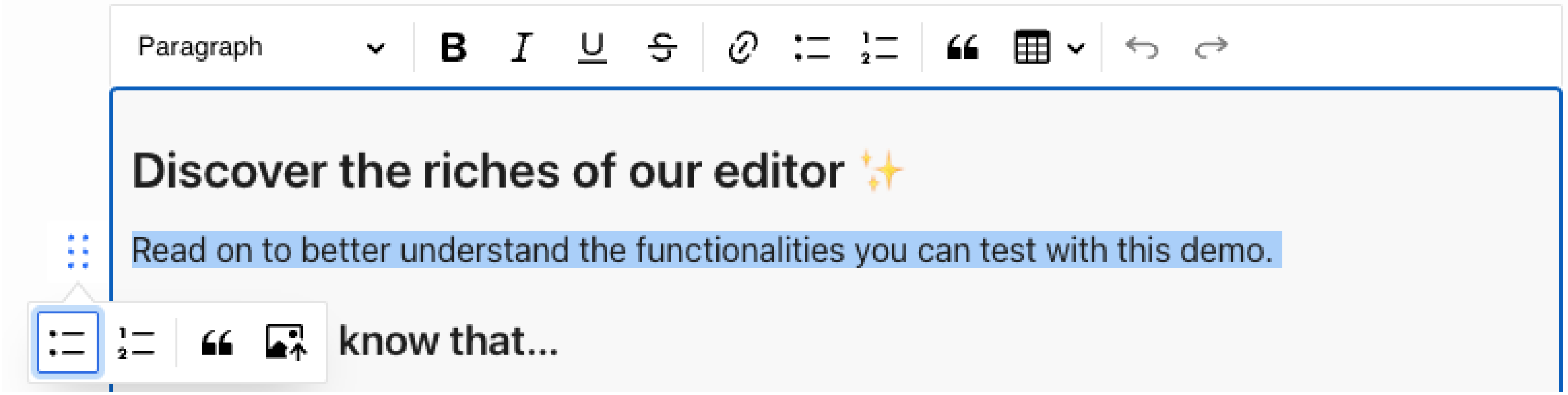
```
> Array.from(apex.items.P4_FULL.getEditor().ui.componentFactory.names())
<· (108) ['alignment:left', 'alignment:right', 'alignment:center', 'alignment:justify', 'alignment', 'menuBar:alignment', 'undo', 'menuBar:undo',
   'redo', 'menuBar:redo', 'imageTextAlternative', 'insertImage', 'imageInsert', 'menuBar:insertImage', 'toggleImageCaption', 'uploadImage', 'imag
   eUpload', 'menuBar:uploadImage', 'insertImageViaUrl', 'menuBar:insertImageViaUrl', 'resizeImage:original', 'resizeImage:custom', 'resizeImage:2
   5', 'resizeImage:50', 'resizeImage:75', 'resizeImage', 'imageResize', 'imageStyle:inline', 'imageStyle:alignLeft', 'imageStyle:alignRight', 'im
   ageStyle:alignCenter', 'imageStyle:alignBlockLeft', 'imageStyle:alignBlockRight', 'imageStyle:block', 'imageStyle:side', 'imageStyle:wrapText',
   'imageStyle:breakText', 'blockQuote', 'menuBar:blockQuote', 'bold', 'menuBar:bold', 'code', 'menuBar:code', 'italic', 'menuBar:italic', 'strike
   through', 'menuBar:strikethrough', 'subscript', 'menuBar:subscript', 'superscript', 'menuBar:superscript', 'underline', 'menuBar:underline', 'c
 ▾ odeBlock', 'menuBar:codeBlock', 'accessibilityHelp', 'menuBar:accessibilityHelp', 'selectAll', 'menuBar:selectAll', 'fontFamily', 'menuBar:font
   Family', 'fontSize', 'menuBar:fontSize', 'fontColor', 'menuBar:fontColor', 'fontBackgroundColor', 'menuBar:fontBackgroundColor', 'heading', 'me
   nuBar:heading', 'highlight:yellowMarker', 'highlight:greenMarker', 'highlight:pinkMarker', 'highlight:blueMarker', 'highlight:redPen', 'highlig
   ht:greenPen', 'removeHighlight', 'highlight', 'menuBar:highlight', 'horizontalLine', 'menuBar:horizontalLine', 'htmlEmbed', 'menuBar:htmlEmbe
   d', 'indent', 'menuBar:indent', 'outdent', 'menuBar:outdent', 'link', 'menuBar:link', 'linkImage', 'numberedList', 'menuBar:numberedList', 'bul
   letedList', 'menuBar:bulletedList', 'todoList', 'menuBar:todoList', 'pageBreak', 'menuBar:pageBreak', 'removeFormat', 'menuBar:removeFormat',
   'specialCharacters', …] i
    ▸ [0 … 99]
    ▸ [100 … 107]
      length: 108
    ▸ [[Prototype]]: Array(0)
```

TOOLBAR - GROUPING ITEMS (DROPDOWN)

```
function(config){
  config.editorOptions.toolbar = [
     'undo',
     'redo',
     '|',
     {
       label: 'Fonts',
       icon: 'text',
       withText: true,
       items: [ 'fontFamily', 'fontSize', 'fontColor', 'fontBackgroundColor']
     },
    {
      label: 'Basic styles',
      withText: true,
      items: [ 'bold', 'italic', 'strikethrough', 'superscript', 'subscript']
    }
  ];
  return config;
}
```



BLOCKTOOLBAR

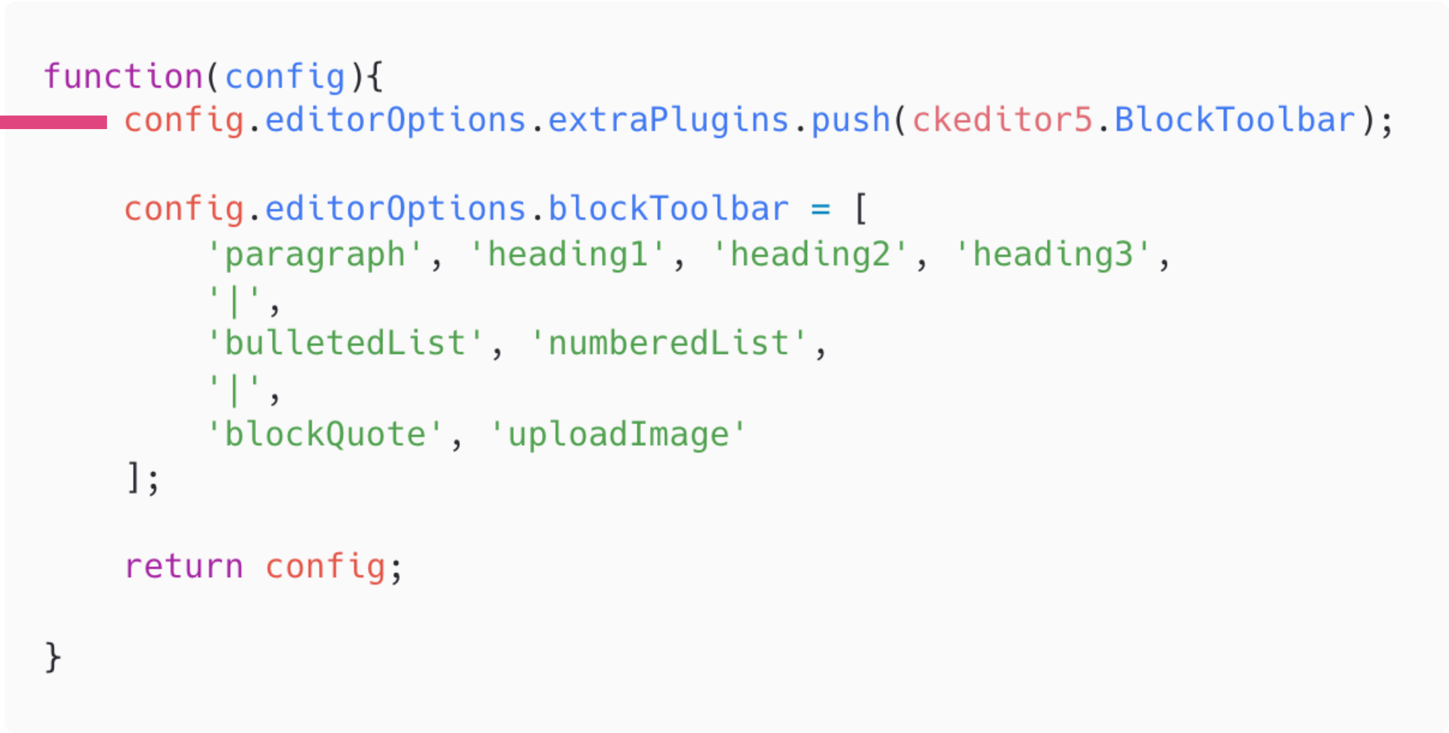


Discover the riches of our editor ✨

Read on to better understand the functionalities you can test with this demo.

know that...

Add external toolbar to the left



```
function(config){
    config.editorOptions.extraPlugins.push(ckeditor5.BlockToolbar);

    config.editorOptions.blockToolbar = [
        'paragraph', 'heading1', 'heading2', 'heading3',
        '|',
        'bulletedList', 'numberedList',
        '|',
        'blockQuote', 'uploadImage'
    ];

    return config;

}
```

IMAGE UPLOAD

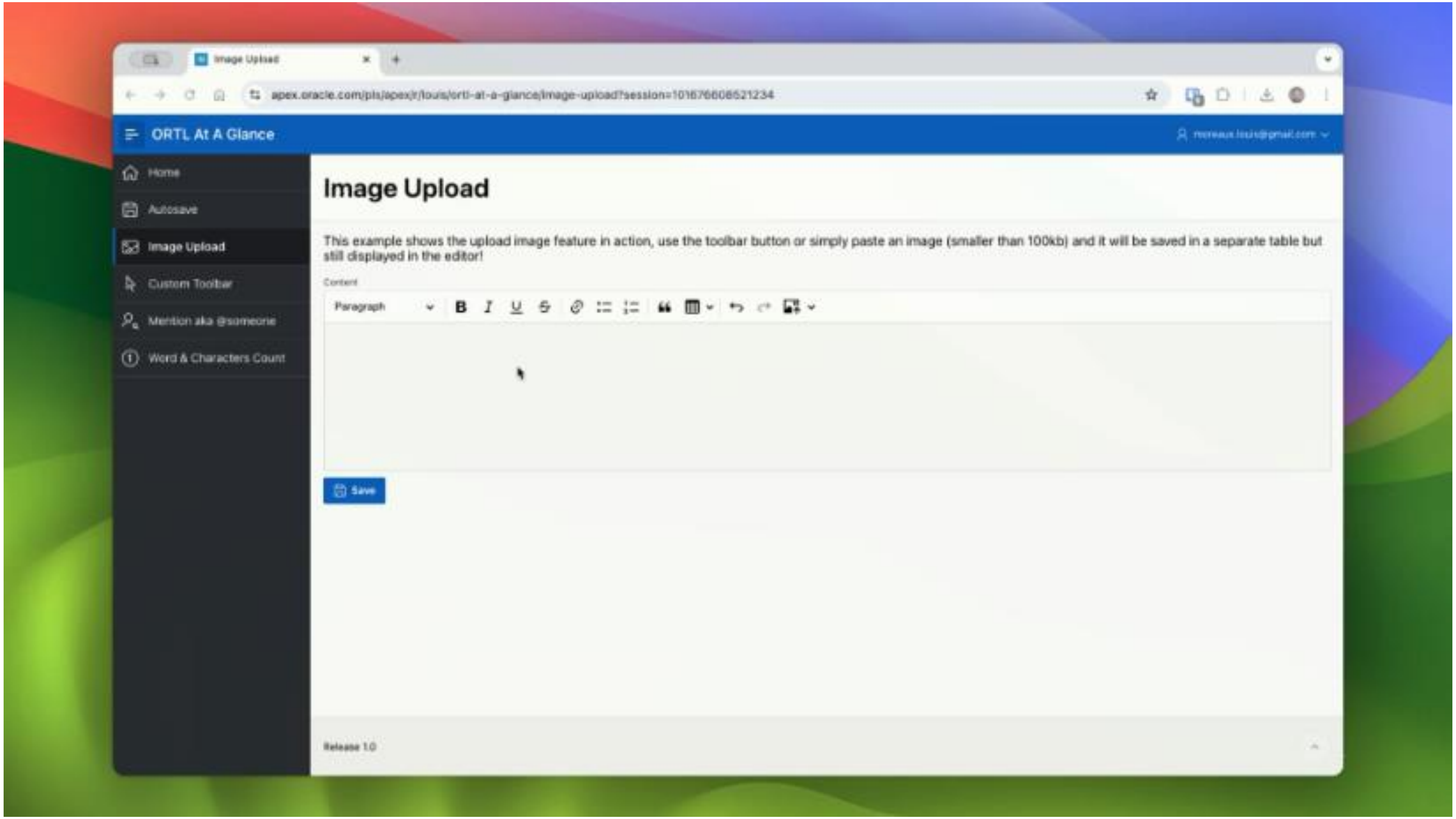


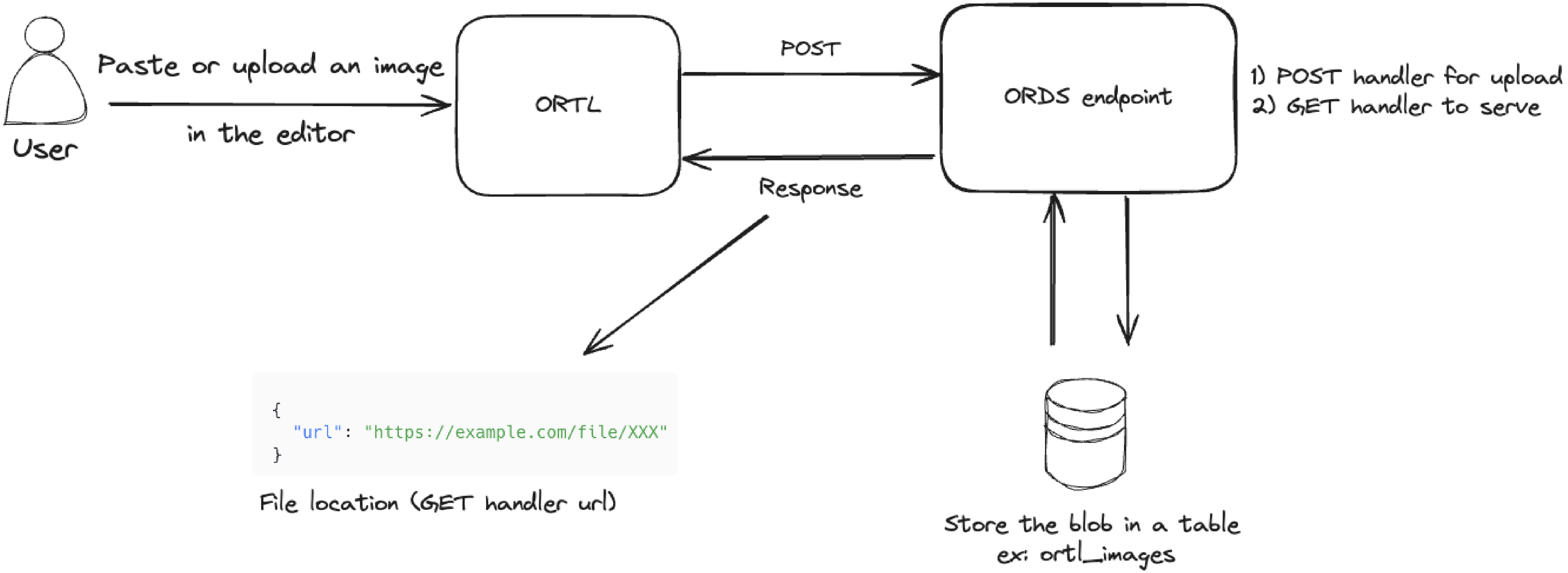
Image Upload

This example shows the upload image feature in action, use the toolbar button or simply paste an image (smaller than 100kb) and it will be saved in a separate table but still displayed in the editor!

Content

https://github.com/LouisMoreaux/talks/raw/b3234d2a8ed837753c50b95d562ef43c6a567755/APEX%20World%202025/videos/ORTL%20-%20image%20upload.mp4

IMAGE UPLOAD – HOW IT WORKS?



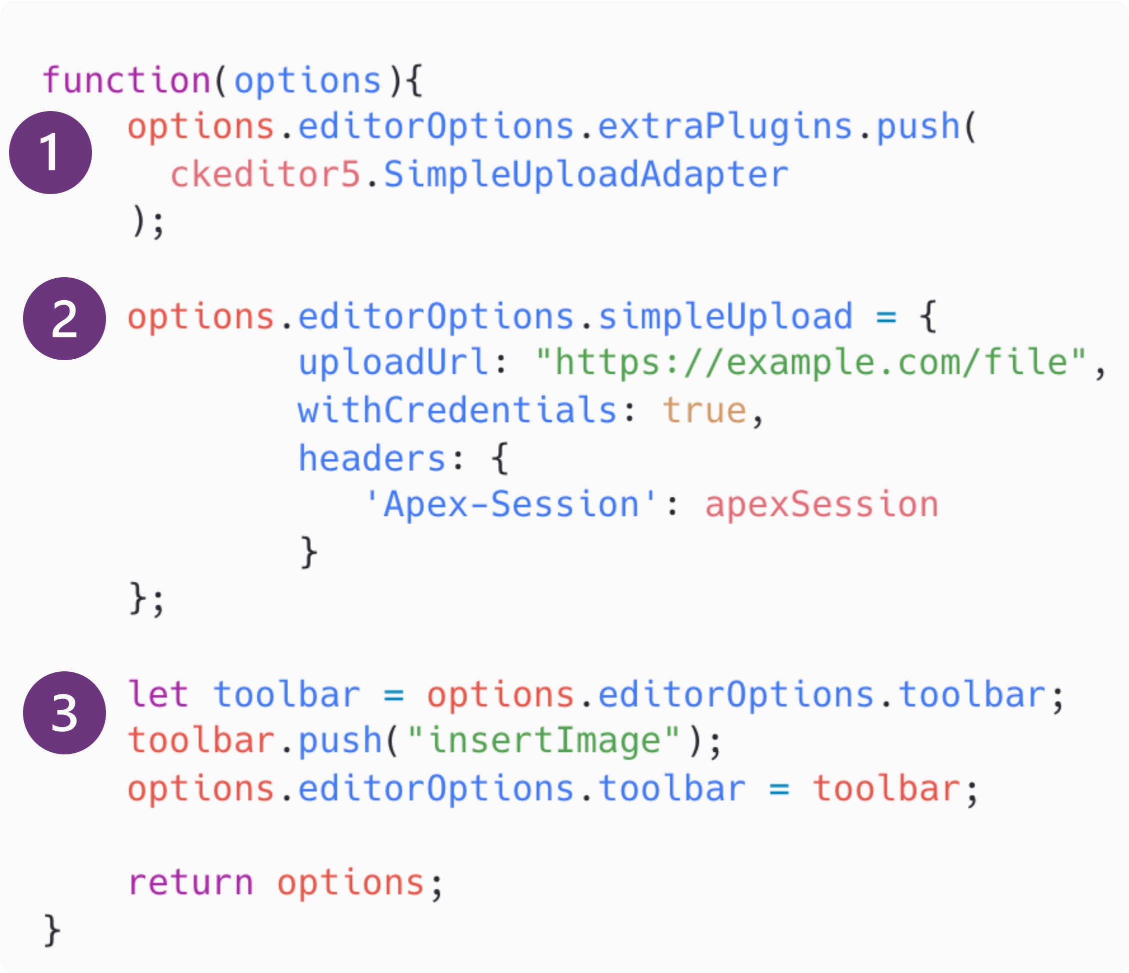
```
{
  "url": "https://example.com/file/XXX"
}
```

IMAGE UPLOAD – EDITOR CONFIGURATION

1 Enable the simple upload adapter

2 Configure it

3 Add a toolbar button



```
function(options){
    options.editorOptions.extraPlugins.push(
      ckeditor5.SimpleUploadAdapter
    );

    options.editorOptions.simpleUpload = {
            uploadUrl: "https://example.com/file",
            withCredentials: true,
            headers: {
                'Apex-Session': apexSession
            }
    };

    let toolbar = options.editorOptions.toolbar;
    toolbar.push("insertImage");
    options.editorOptions.toolbar = toolbar;

    return options;
}
```

https://ckeditor.com/docs/ckeditor5/latest/features/images/image-upload/simple-upload-adapter.html

IMAGE UPLOAD – ORDS (POST)

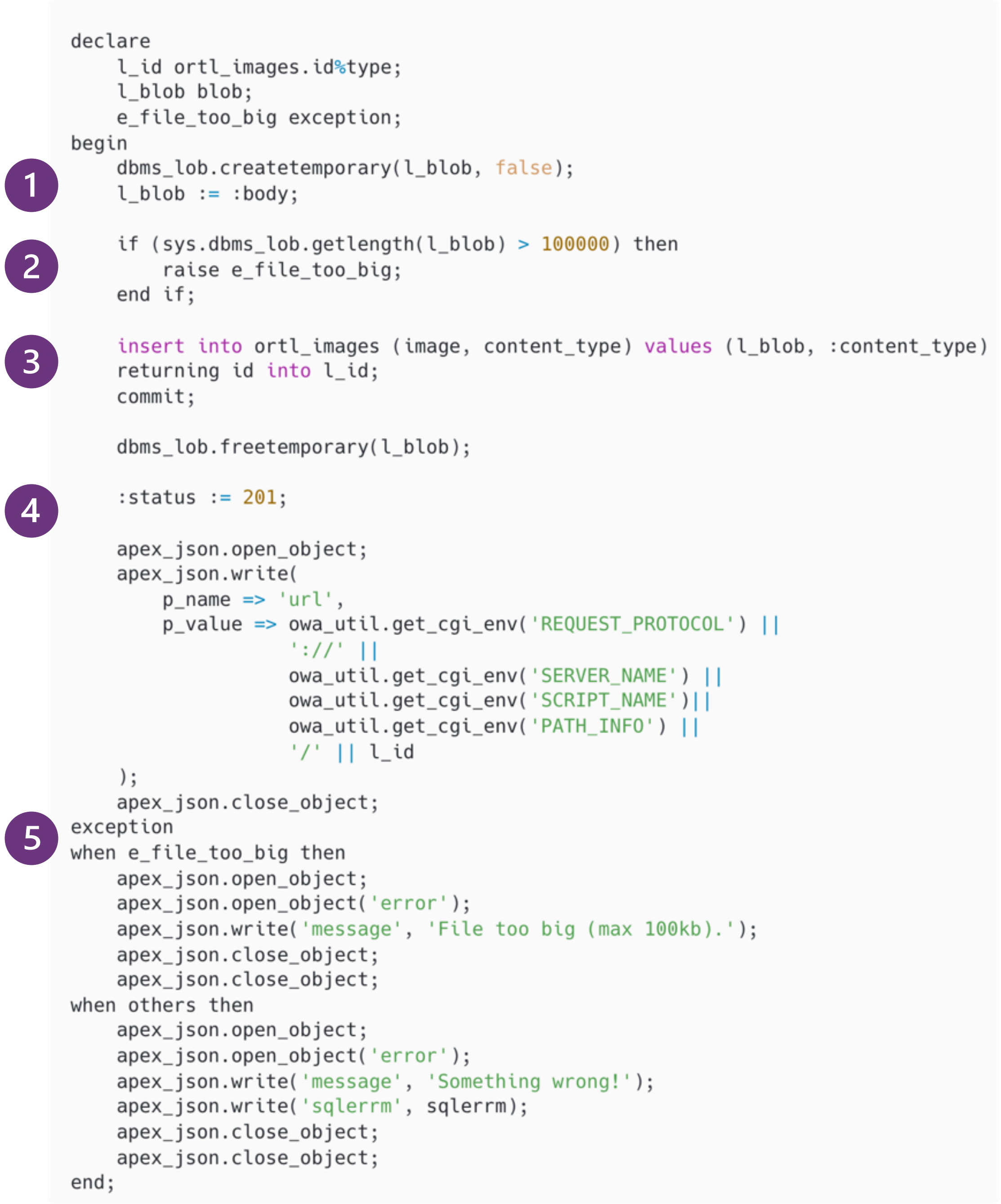
1 Get the blob from the request

2 Add some validation (ex: file size)

3 Store the file in the db and get the id

4 Create the success response

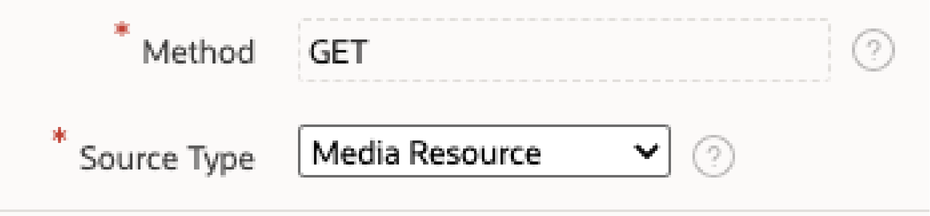
5 For errors, provide a user-friendly message



```
declare
    l_id ortl_images.id%type;
    l_blob blob;
    e_file_too_big exception;
begin
    dbms_lob.createtemporary(l_blob, false);
    l_blob := :body;

    if (sys.dbms_lob.getlength(l_blob) > 100000) then
        raise e_file_too_big;
    end if;

    insert into ortl_images (image, content_type) values (l_blob, :content_type)
    returning id into l_id;
    commit;

    dbms_lob.freetemporary(l_blob);

    :status := 201;

    apex_json.open_object;
    apex_json.write(
        p_name => 'url',
        p_value => owa_util.get_cgi_env('REQUEST_PROTOCOL') ||
                   '://' ||
                   owa_util.get_cgi_env('SERVER_NAME') ||
                   owa_util.get_cgi_env('SCRIPT_NAME')||
                   owa_util.get_cgi_env('PATH_INFO') ||
                   '/' || l_id
    );
    apex_json.close_object;
exception
when e_file_too_big then
    apex_json.open_object;
    apex_json.open_object('error');
    apex_json.write('message', 'File too big (max 100kb).');
    apex_json.close_object;
    apex_json.close_object;
when others then
    apex_json.open_object;
    apex_json.open_object('error');
    apex_json.write('message', 'Something wrong!');
    apex_json.write('sqlerrm', sqlerrm);
    apex_json.close_object;
    apex_json.close_object;
end;
```

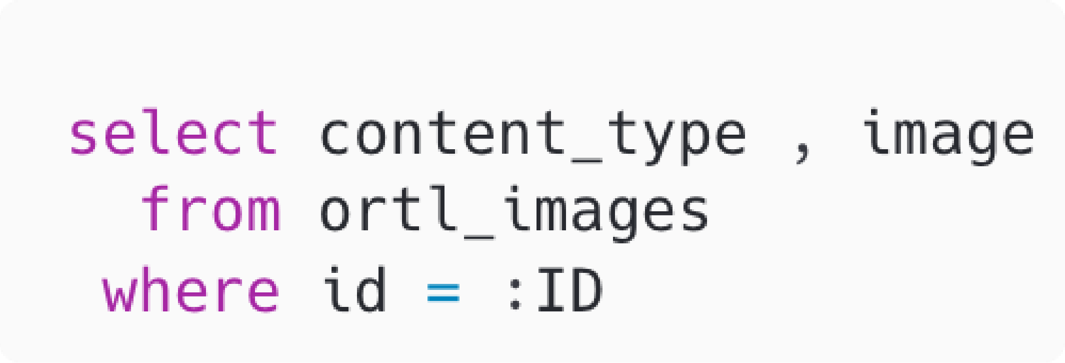
IMAGE UPLOAD – ORDS (GET)

Built-in source type



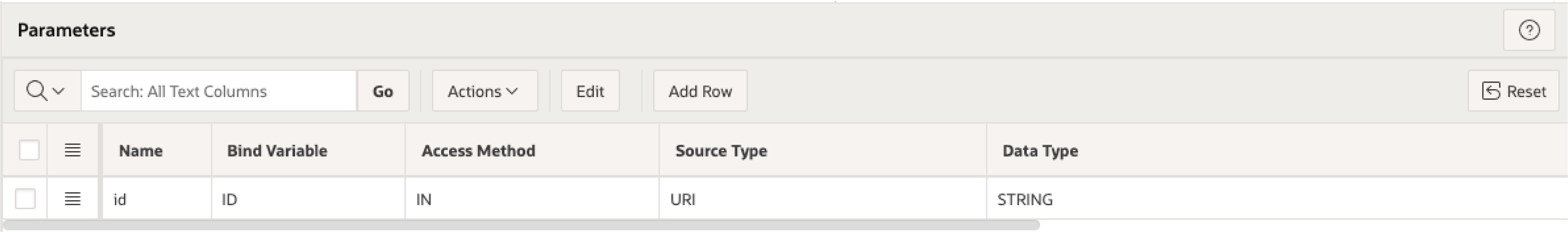
* Method	GET
* Source Type	Media Resource

SQL query



```
select content_type , image
  from ortl_images
 where id = :ID
```

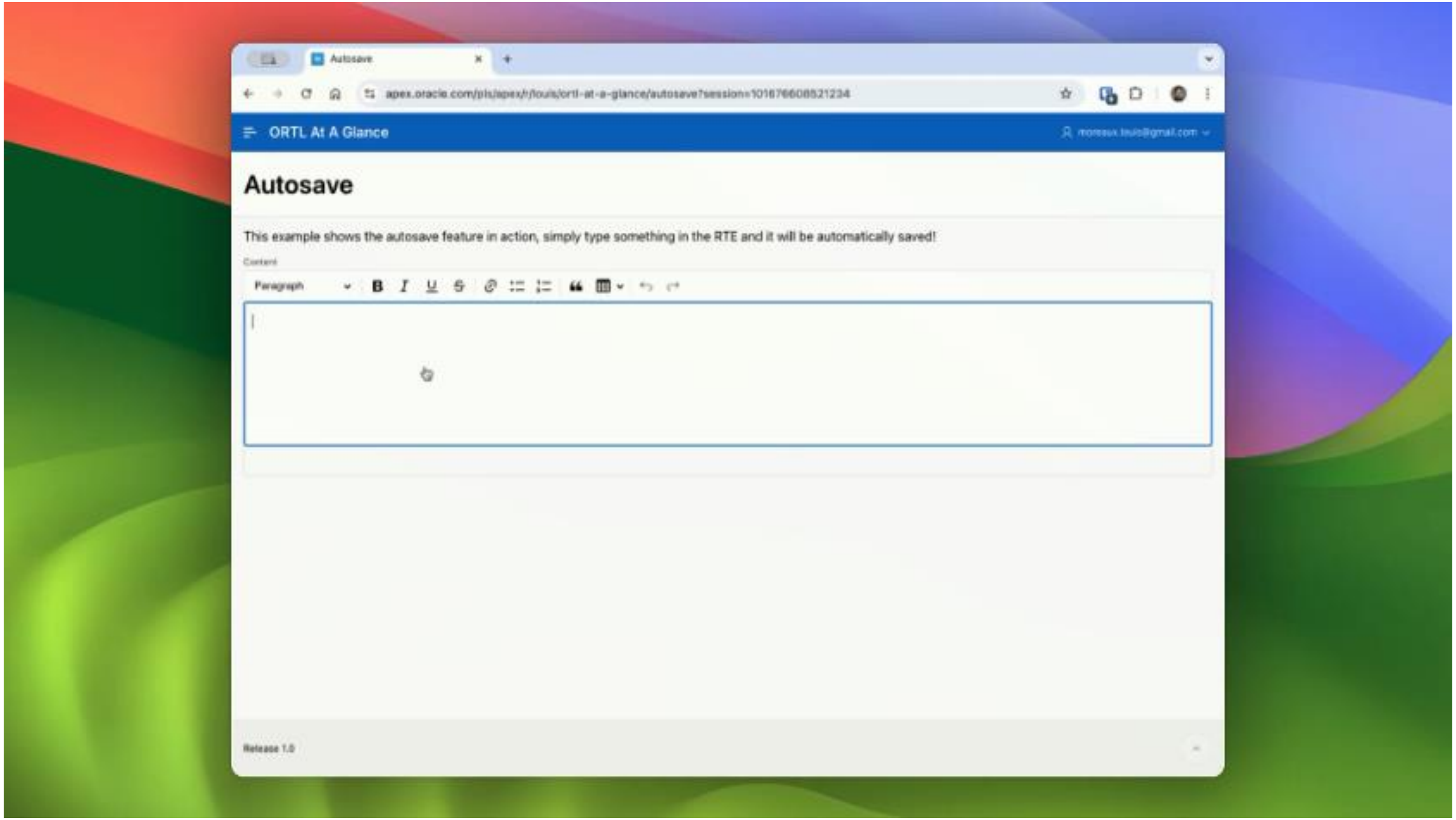
URI parameter



Parameters

Name	Bind Variable	Access Method	Source Type	Data Type
id	ID	IN	URI	STRING

AUTOSAVE


Autosave
apex.oracle.com/pls/apex/r/louis/ortl-at-a-glance/autosave?session=101676608521234
ORTL At A Glance
moreaux.louis@gmail.com
Paragraph
Release 1.0

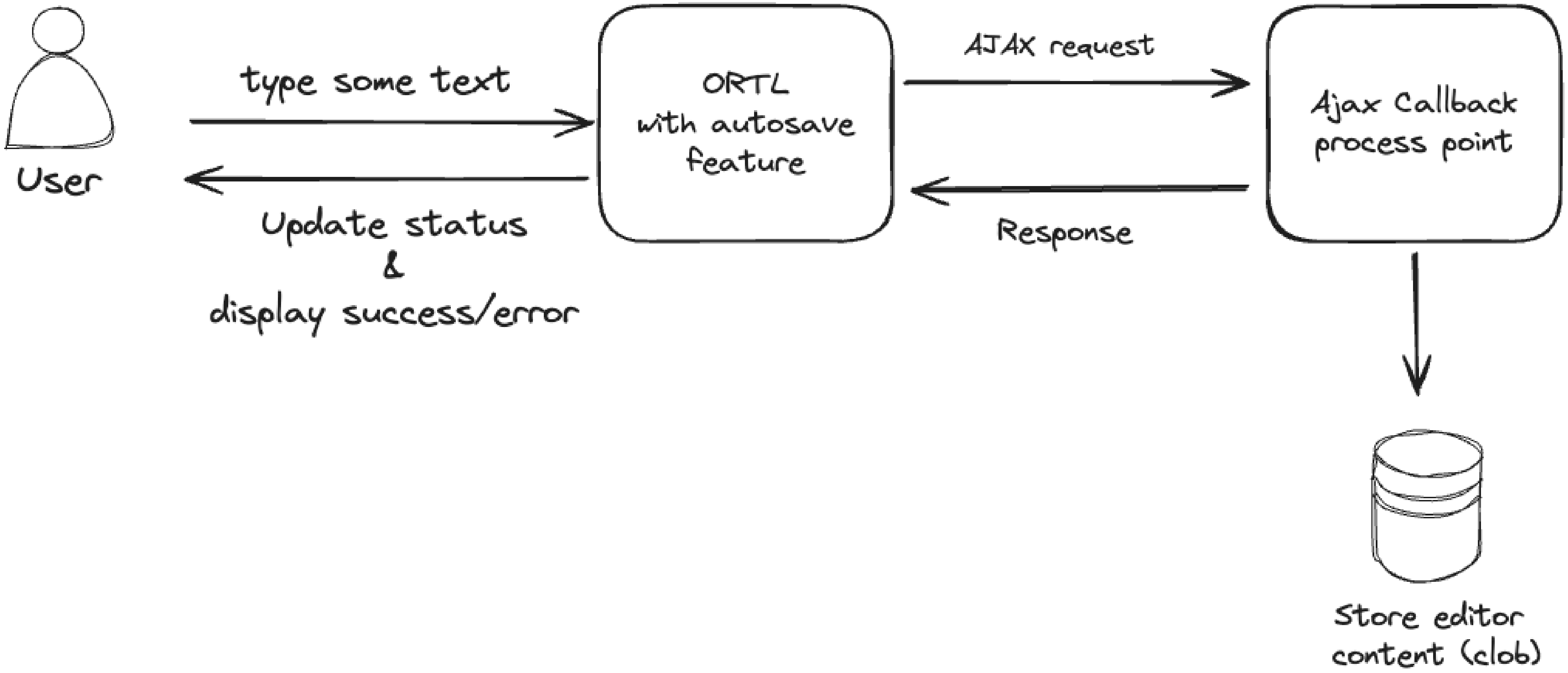
Autosave

This example shows the autosave feature in action, simply type something in the RTE and it will be automatically saved!

Content

https://github.com/LouisMoreaux/talks/raw/b3234d2a8ed837753c50b95d562ef43c6a567755/APEX%20World%202025/videos/ORTL%20-%20autosave.mp4

AUTOSAVE – HOW IT WORKS?



User

type some text

Update status
&
display success/error

ORTL
with autosave
feature

AJAX request

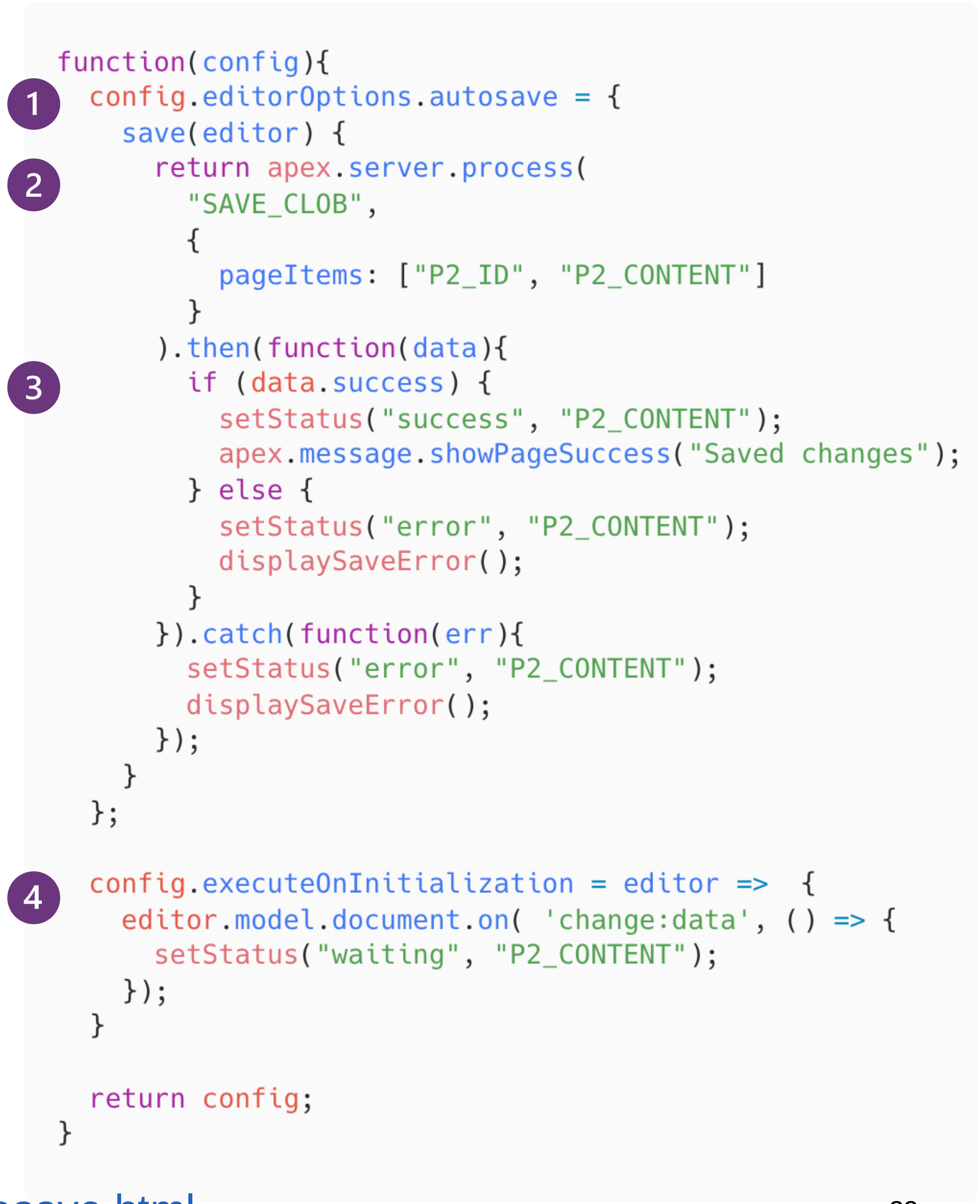
Response

Ajax Callback
process point

Store editor
content (clob)

AUTOSAVE – EDITOR CONFIGURATION

1. Configure autosave feature
2. Call Ajax Callback process
3. Update status and message
4. Update status on change event



```
function(config){
  config.editorOptions.autosave = {
    save(editor) {
      return apex.server.process(
        "SAVE_CLOB",
        {
          pageItems: ["P2_ID", "P2_CONTENT"]
        }
      ).then(function(data){
        if (data.success) {
          setStatus("success", "P2_CONTENT");
          apex.message.showPageSuccess("Saved changes");
        } else {
          setStatus("error", "P2_CONTENT");
          displaySaveError();
        }
      }).catch(function(err){
        setStatus("error", "P2_CONTENT");
        displaySaveError();
      });
    }
  };

  config.executeOnInitialization = editor =>  {
    editor.model.document.on( 'change:data', () => {
      setStatus("waiting", "P2_CONTENT");
    });
  }

  return config;
}
```

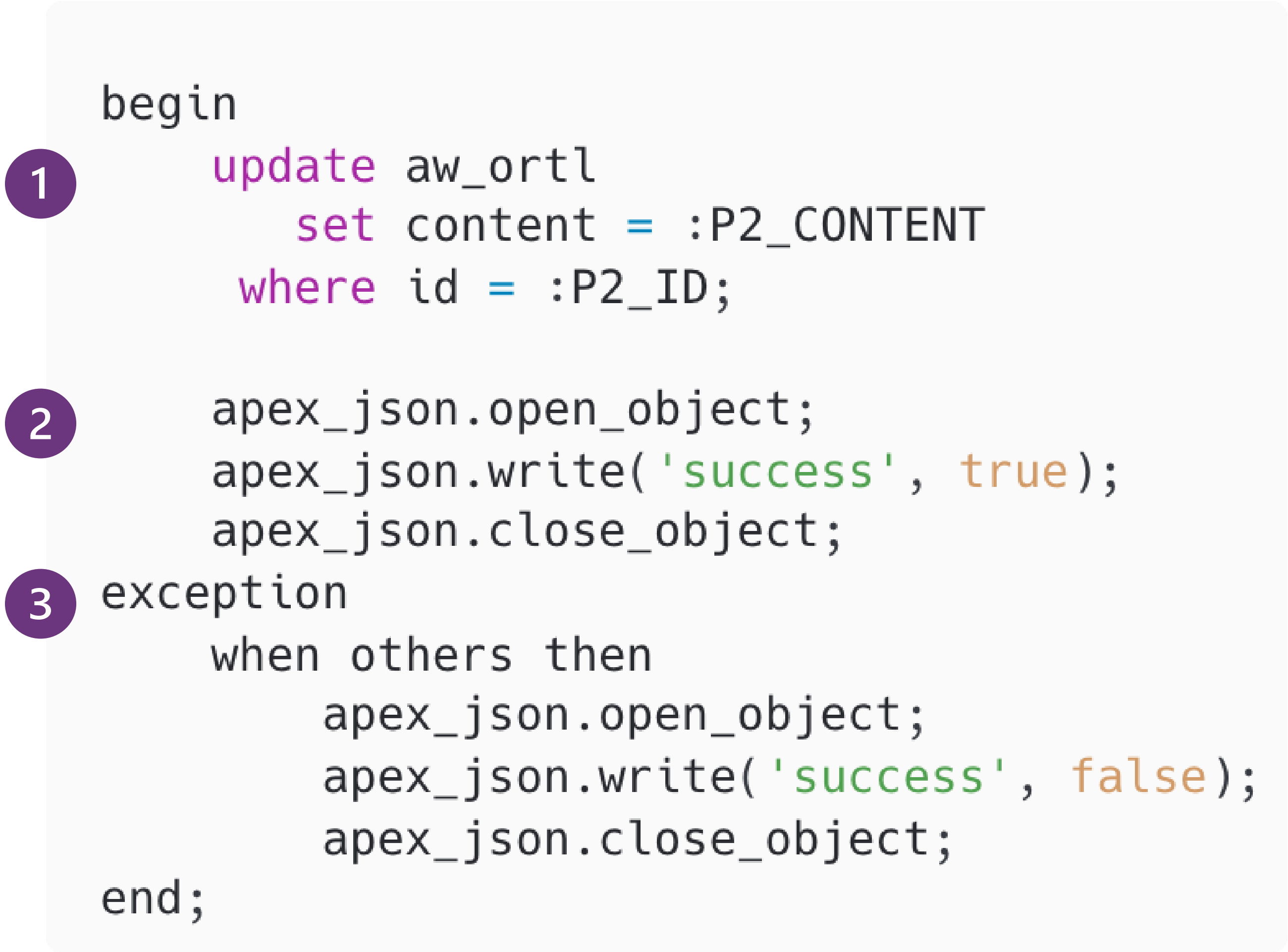
https://ckeditor.com/docs/ckeditor5/latest/features/autosave.html

AUTOSAVE – AJAX CALLBACK

1 Save editor content

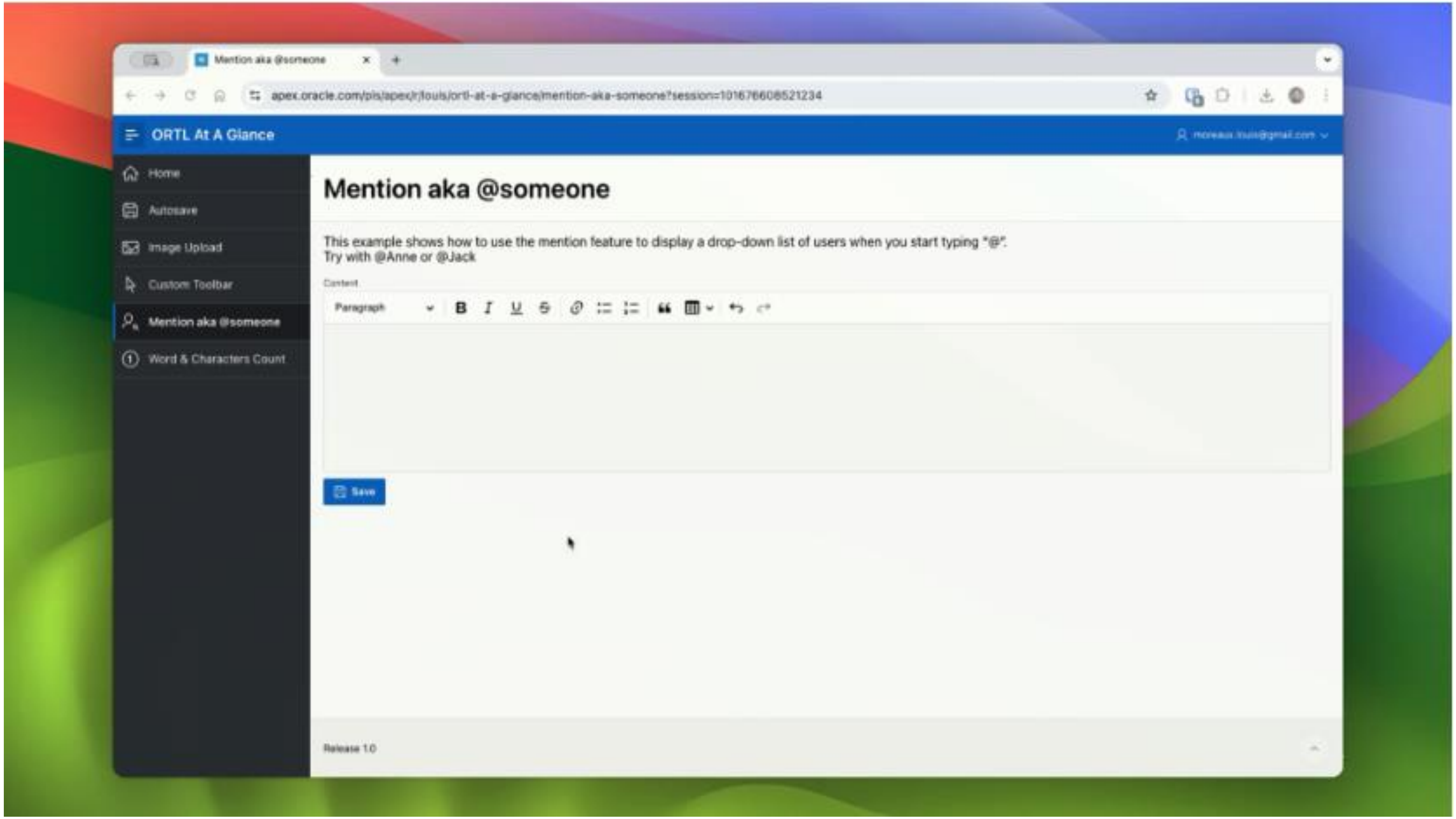
2 Successful response

3 Unsuccessful response



```
begin
    update aw_ortl
       set content = :P2_CONTENT
     where id = :P2_ID;

    apex_json.open_object;
    apex_json.write('success', true);
    apex_json.close_object;
exception
    when others then
        apex_json.open_object;
        apex_json.write('success', false);
        apex_json.close_object;
end;
```

MENTIONS AKA @SOMEONE



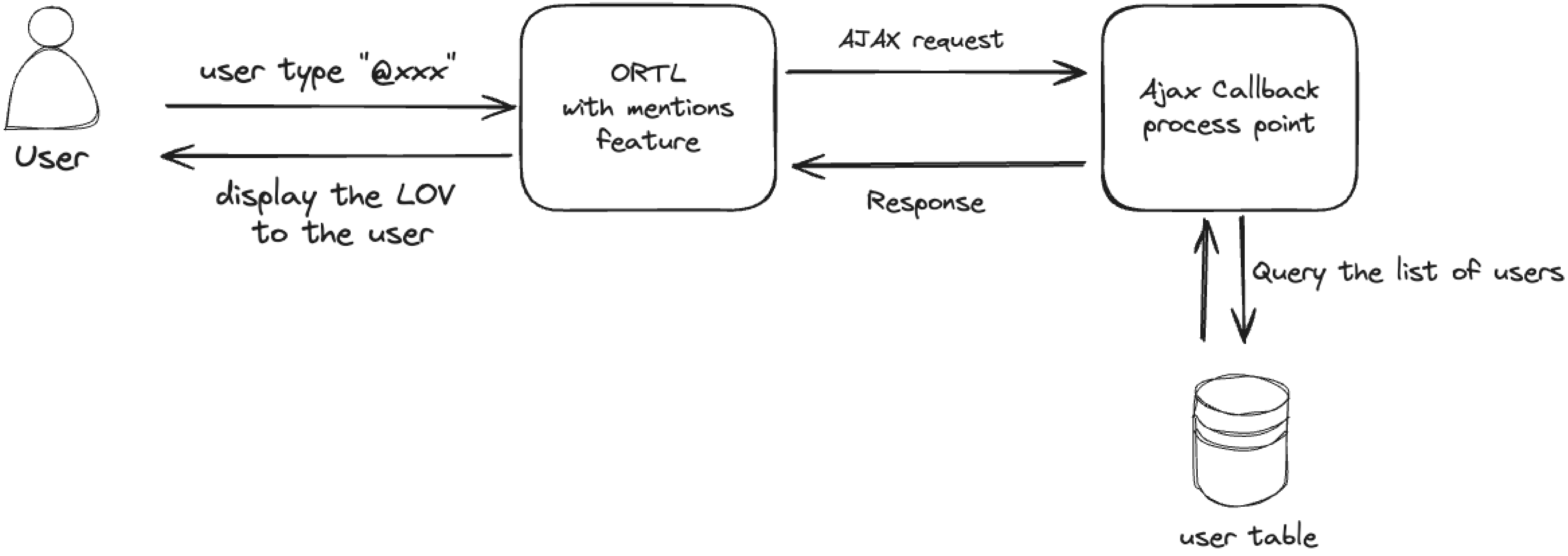
Mention aka @someone

This example shows how to use the mention feature to display a drop-down list of users when you start typing "@".
Try with @Anne or @Jack

Content

https://github.com/LouisMoreaux/talks/raw/b3234d2a8ed837753c50b95d562ef43c6a567755/APEX World 2025/videos/ORTL - mention.mp4

MENTIONS – HOW IT WORKS?



User

user type "@xxx"

display the LOV to the user

ORTL with mentions feature

AJAX request

Response

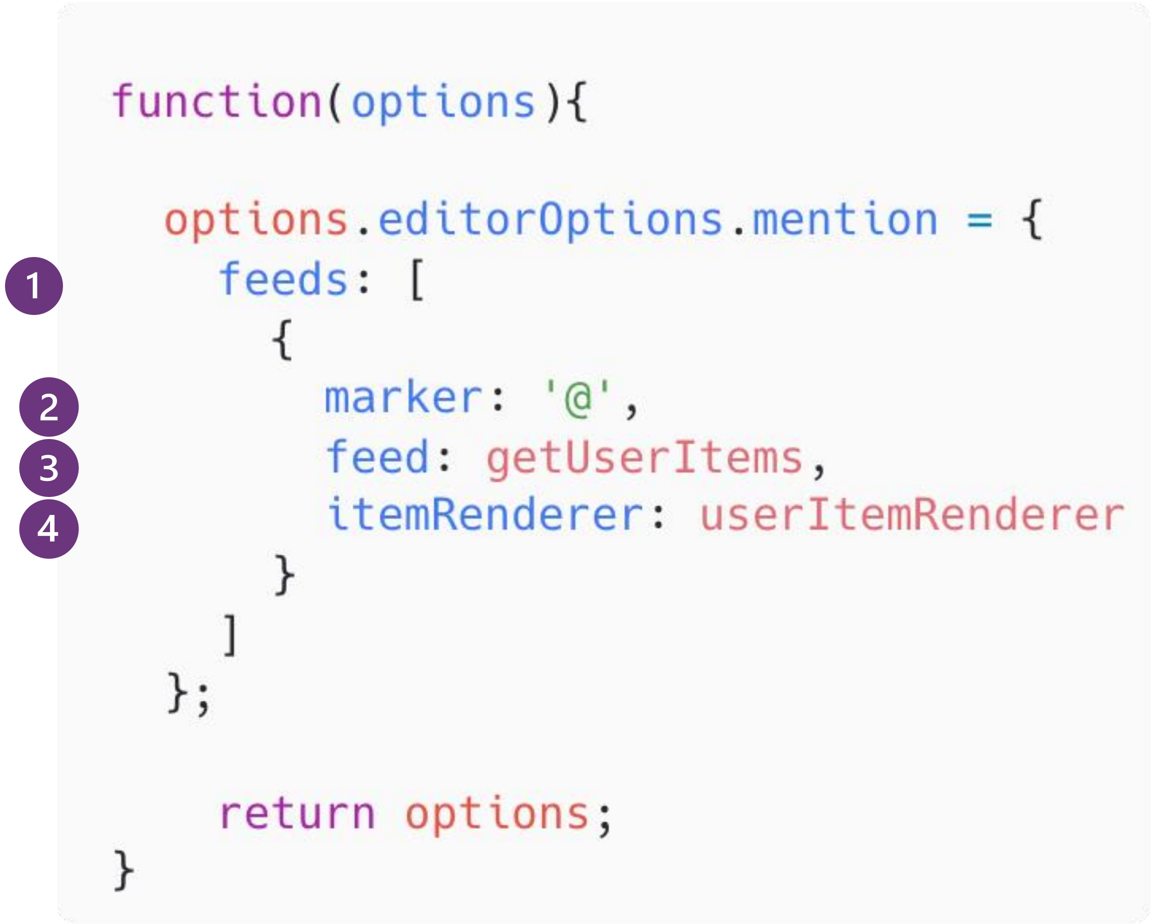
Ajax Callback process point

Query the list of users

user table

MENTIONS – EDITOR CONFIGURATION

1. Define one to many feeds
2. Provide a marker (@, #, etc)
3. JavaScript function to get the data
4. JavaScript function to format the list

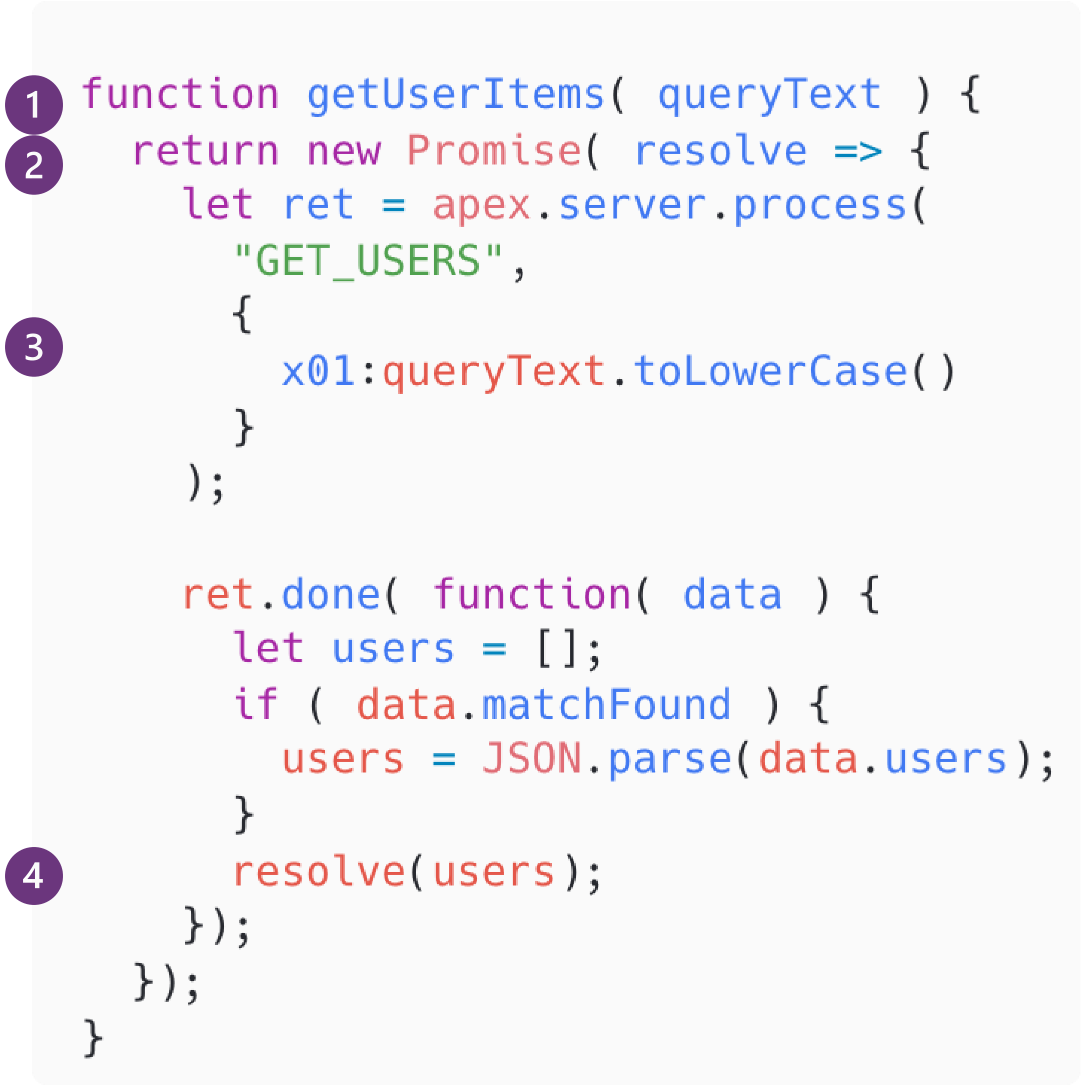


```
function(options){

  options.editorOptions.mention = {
    feeds: [
      {
        marker: '@',
        feed: getUserItems,
        itemRenderer: userItemRenderer
      }
    ]
  };

    return options;
}
```

https://ckeditor.com/docs/ckeditor5/latest/features/mentions.html

MENTIONS – FEED FUNCTION

1. Function with typed text as argument
2. Should return a Promise
3. Call Ajax Callback process
4. Returns aJSON array

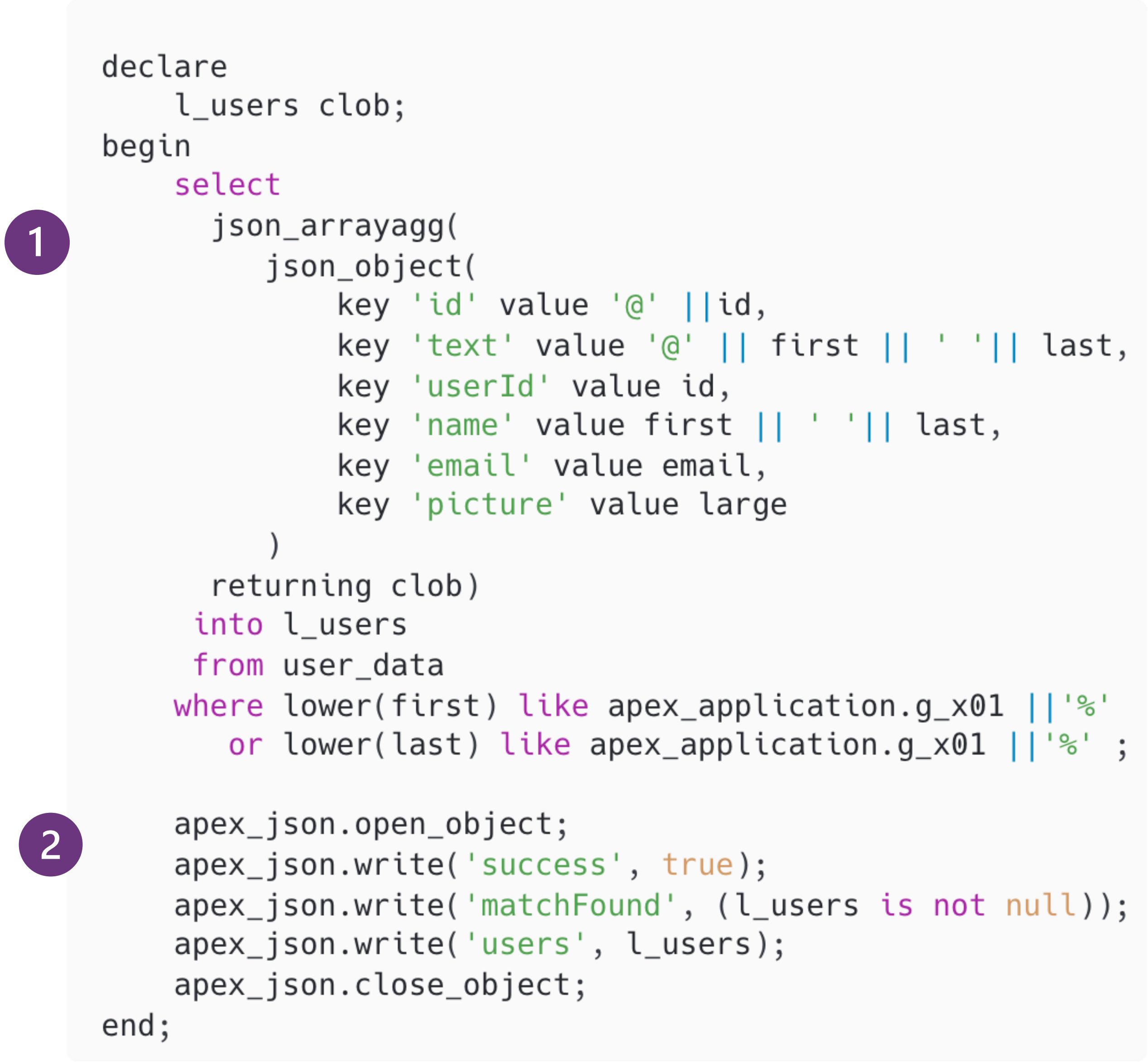


```
function getUserItems( queryText ) {
  return new Promise( resolve => {
    let ret = apex.server.process(
      "GET_USERS",
      {
        x01:queryText.toLowerCase()
      }
    );

    ret.done( function( data ) {
      let users = [];
      if ( data.matchFound ) {
        users = JSON.parse(data.users);
      }
      resolve(users);
    });
  });
}
```

MENTIONS – AJAX CALLBACK

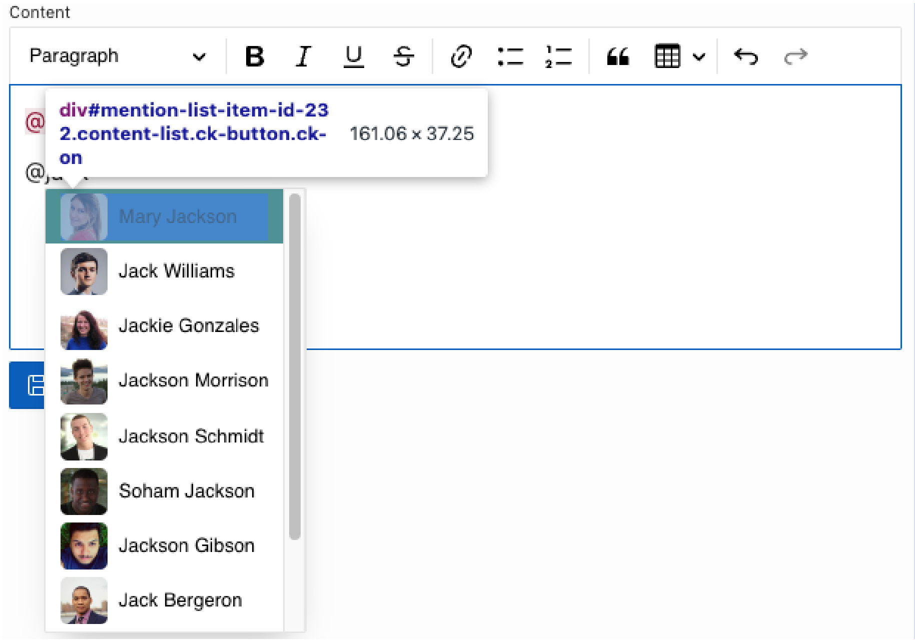
1 Query table (using user input)

2 Return JSON response with:
- matchFound (true/false)
- Users: data (JSON)



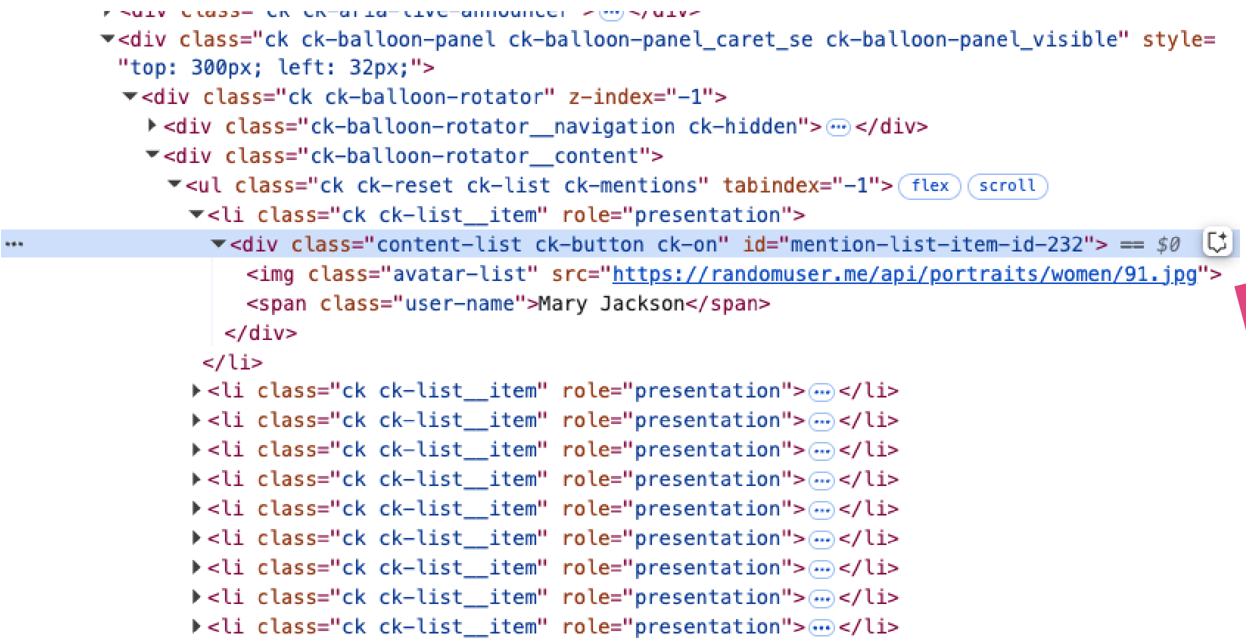
```
declare
    l_users clob;
begin
    select
      json_arrayagg(
         json_object(
             key 'id' value '@' ||id,
             key 'text' value '@' || first || ' '|| last,
             key 'userId' value id,
             key 'name' value first || ' '|| last,
             key 'email' value email,
             key 'picture' value large
         )
      returning clob)
     into l_users
     from user_data
    where lower(first) like apex_application.g_x01 ||'%'
       or lower(last) like apex_application.g_x01 ||'%' ;

    apex_json.open_object;
    apex_json.write('success', true);
    apex_json.write('matchFound', (l_users is not null));
    apex_json.write('users', l_users);
    apex_json.close_object;
end;
```

MENTIONS – RENDER FUNCTION

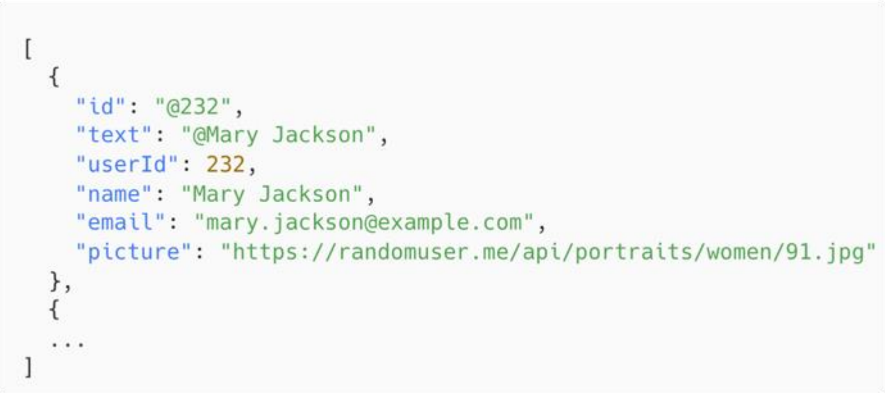


div#mention-list-item-id-232.content-list.ck-button.ck-on 161.06 × 37.25

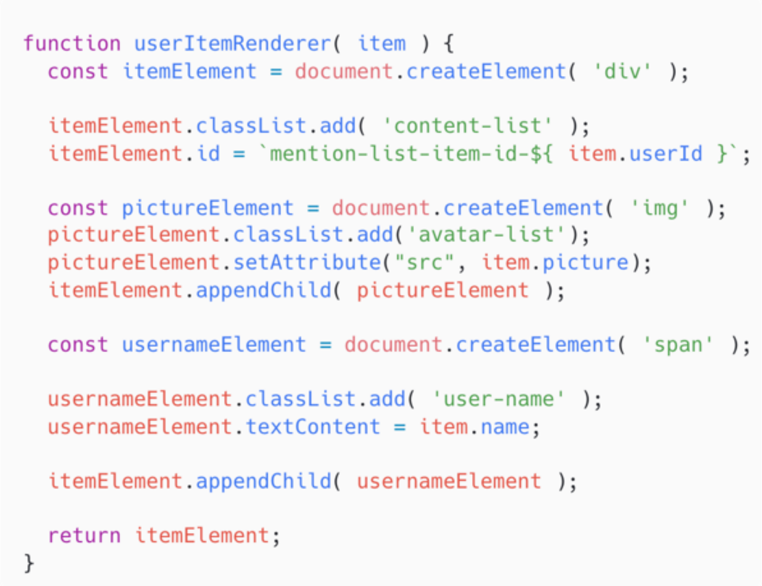
- Mary Jackson
- Jack Williams
- Jackie Gonzales
- Jackson Morrison
- Jackson Schmidt
- Soham Jackson
- Jackson Gibson
- Jack Bergeron



```
▸<div class="ck ck-aria-live-announcer">…</div>
▾<div class="ck ck-balloon-panel ck-balloon-panel_caret_se ck-balloon-panel_visible" style=
 "top: 300px; left: 32px;">
  ▾<div class="ck ck-balloon-rotator" z-index="-1">
    ▸<div class="ck-balloon-rotator__navigation ck-hidden">…</div>
    ▾<div class="ck-balloon-rotator__content">
      ▾<ul class="ck ck-reset ck-list ck-mentions" tabindex="-1"> flex  scroll
        ▾<li class="ck ck-list__item" role="presentation">
…         ▾<div class="content-list ck-button ck-on" id="mention-list-item-id-232"> == $0
             <img class="avatar-list" src="https://randomuser.me/api/portraits/women/91.jpg">
             <span class="user-name">Mary Jackson</span>
           </div>
         </li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
        ▸<li class="ck ck-list__item" role="presentation">…</li>
```

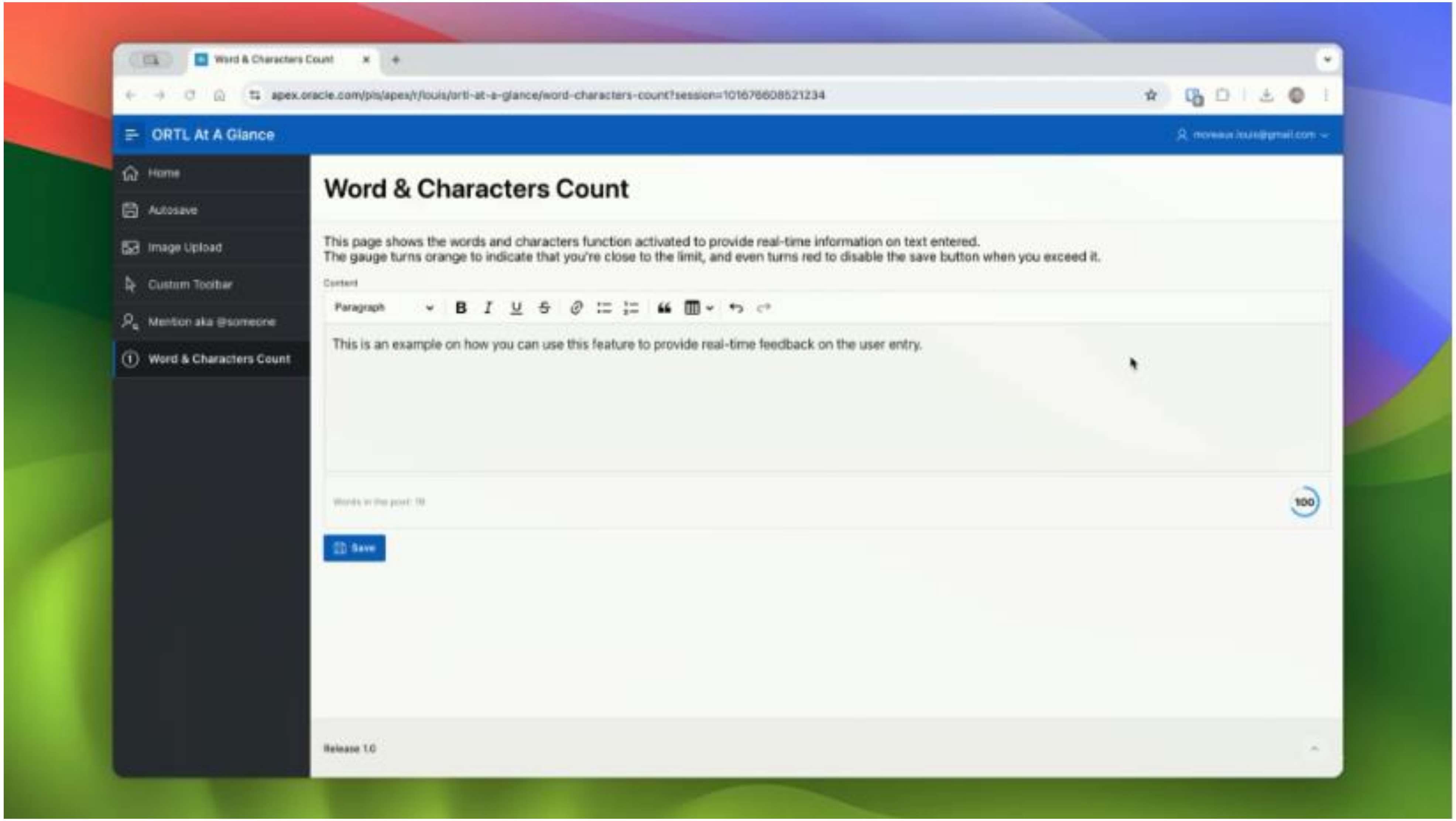


```json
[
  {
    "id": "@232",
    "text": "@Mary Jackson",
    "userId": 232,
    "name": "Mary Jackson",
    "email": "mary.jackson@example.com",
    "picture": "https://randomuser.me/api/portraits/women/91.jpg"
  },
  {
  ...
]
```



```javascript
function userItemRenderer( item ) {
  const itemElement = document.createElement( 'div' );

  itemElement.classList.add( 'content-list' );
  itemElement.id = `mention-list-item-id-${ item.userId }`;

  const pictureElement = document.createElement( 'img' );
  pictureElement.classList.add('avatar-list');
  pictureElement.setAttribute("src", item.picture);
  itemElement.appendChild( pictureElement );

  const usernameElement = document.createElement( 'span' );

  usernameElement.classList.add( 'user-name' );
  usernameElement.textContent = item.name;

  itemElement.appendChild( usernameElement );

  return itemElement;
}
```

WORD & CHARACTER COUNT



Word & Characters Count

This page shows the words and characters function activated to provide real-time information on text entered.
The gauge turns orange to indicate that you're close to the limit, and even turns red to disable the save button when you exceed it.

Content

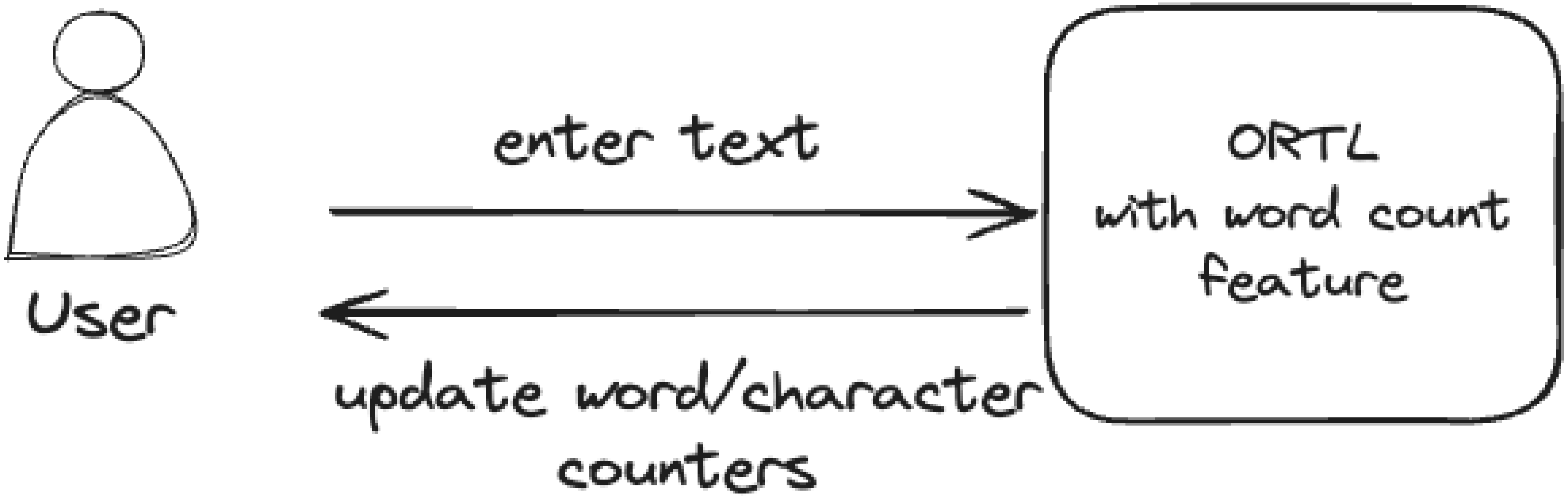
This is an example on how you can use this feature to provide real-time feedback on the user entry.

Words in the post: 19

100

https://github.com/LouisMoreaux/talks/raw/b3234d2a8ed837753c50b95d562ef43c6a567755/APEX World 2025/videos/ORTL - wordcount.mp4

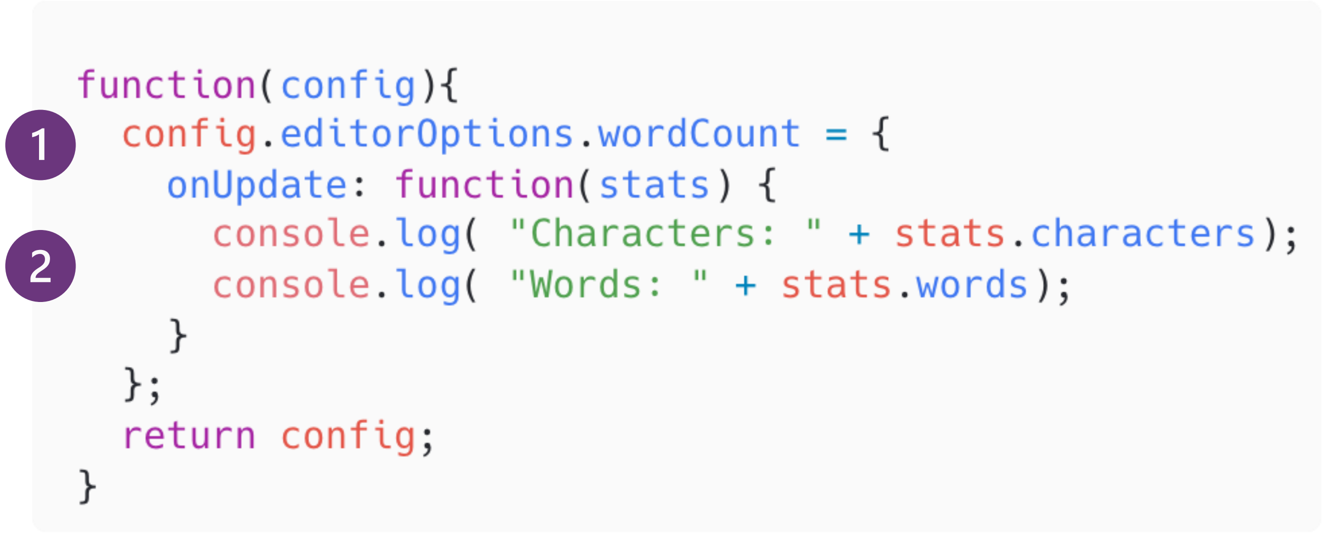
W&C COUNT – EDITOR CONFIGURATION



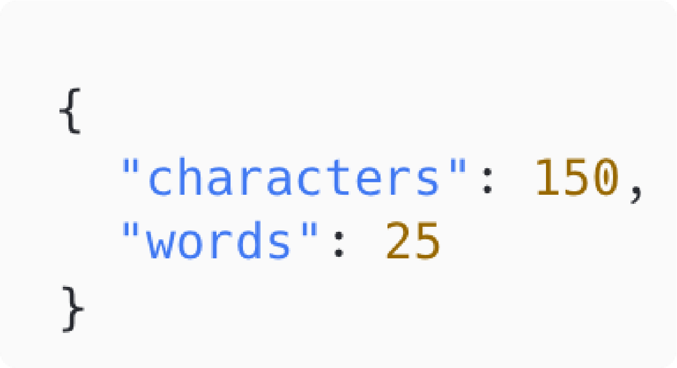
W&C COUNT – EDITOR CONFIGURATION

1 Configure WordCount feature

2 Provide onUpdate behavior



```
function(config){
  config.editorOptions.wordCount = {
    onUpdate: function(stats) {
      console.log( "Characters: " + stats.characters);
      console.log( "Words: " + stats.words);
    }
  };
  return config;
}
```



```
{
  "characters": 150,
  "words": 25
}
```

https://ckeditor.com/docs/ckeditor5/latest/features/word-count.html

Conclusion

Avoid using TinyMCE in 23.1+

Migrate CKEditor5 to ORTL is easier

Rich Text Editor is highly customizable and powerful

Thanks

TALAN

Contact me

Please fill in your evaluations