

SCM-Procurement. Defaulting with our logic in descriptive flexfields in Responsive Self Service Procurement (RSSP) pages (26).

Version 2. With Redwood customization+Visual Builder.

Introduction/Goal

Requirements would be:

“We want to default information in Descriptive flexfields with some logic that depends for example on the business unit”.

DISCLAIMER: this demo is for investigation, use at your discretion, no guarantee provided.

204160
Requisition

Charge account

☒ Select favorite charge account ☐ Enter charge account number

Charge Account Nickname
Main

101.10.60540.120.000.000

Budgetary control

Budget Date
5/21/25

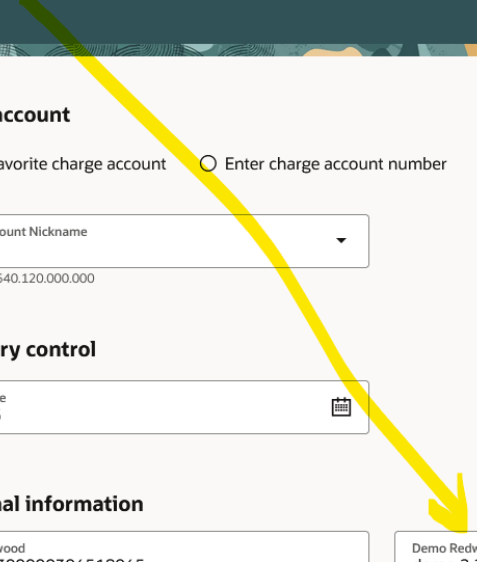
Additional information

Demo Redwood
demo 1.300000306518065

Demo Redwood 2
demo 2.204160

Notes

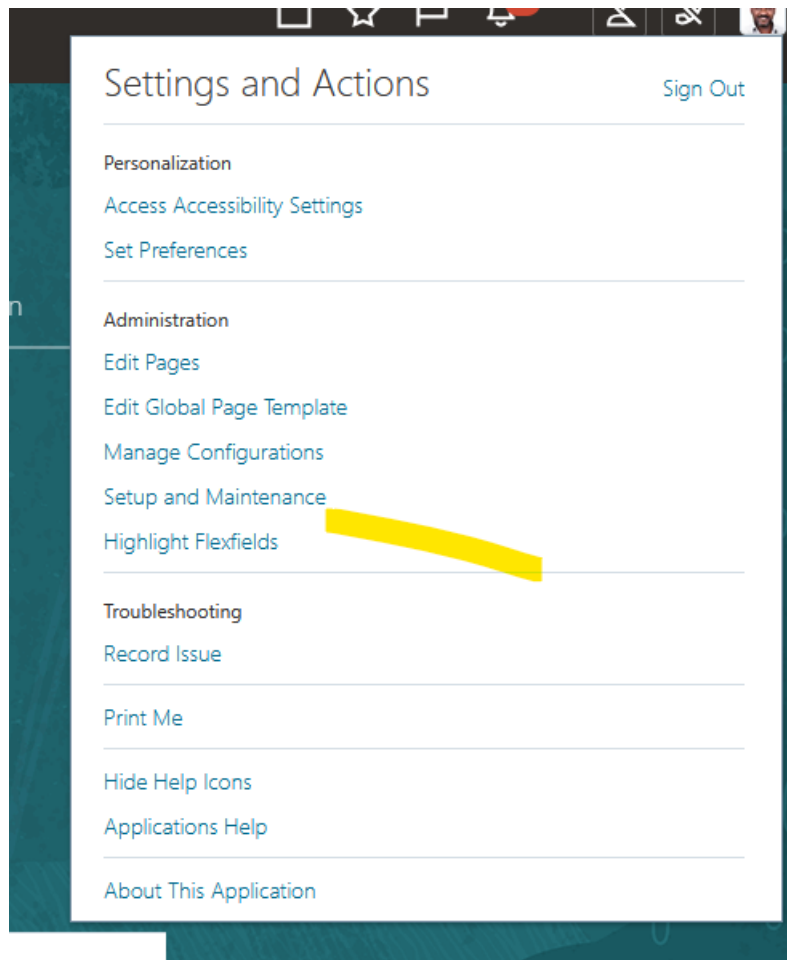
Note to Supplier



Preparatory steps

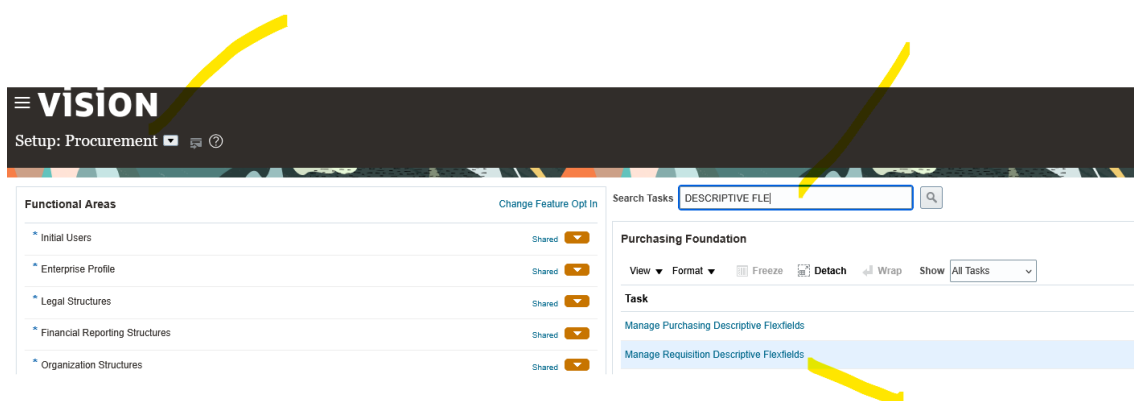
Setup Descriptive flexfields in Oracle ERP.

From main menu, access this option:

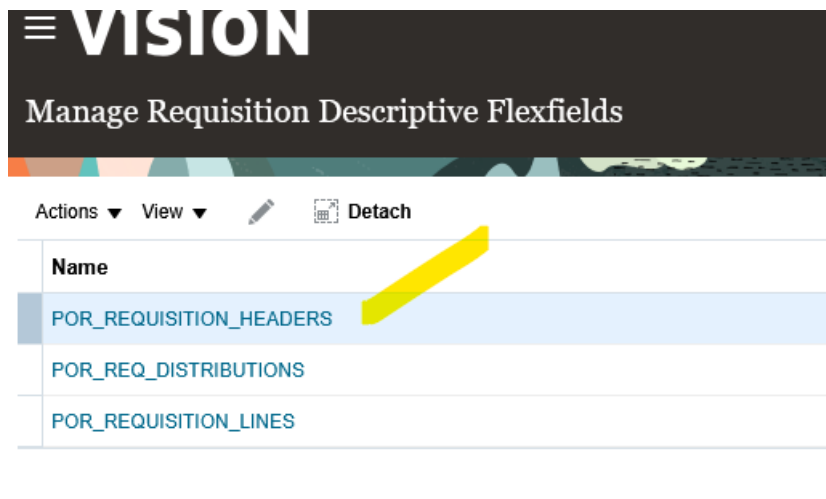


Then select Procurement module and search for “descriptive flex..” as shown.

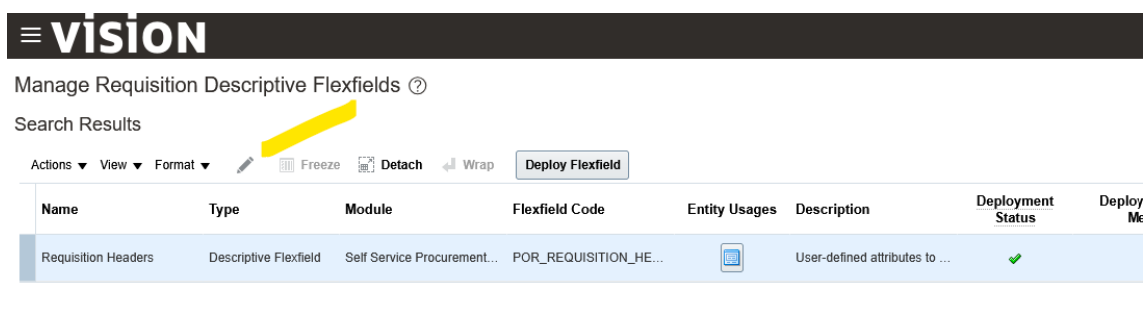
Select option “Manage Requisition Descriptive Flexfields”.



Then select headers option.

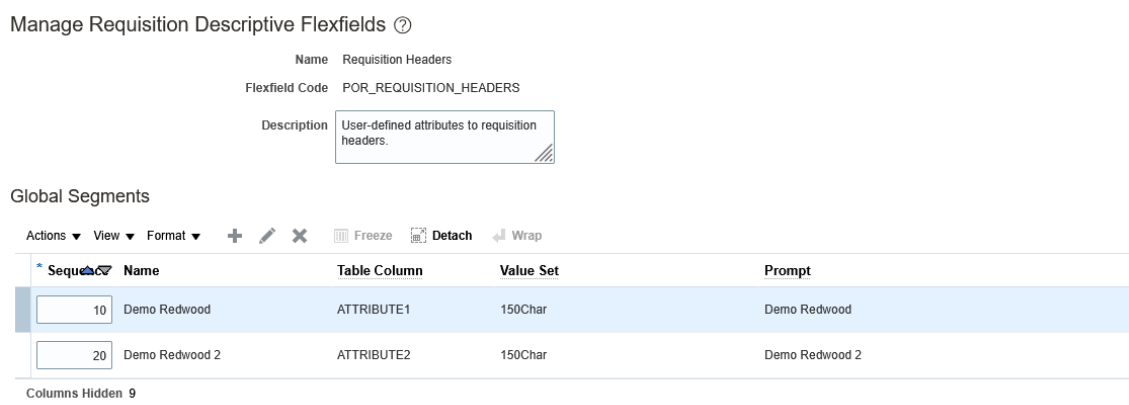


Then “Edit”.



Here add your Descriptive flexfields, according to your needs.

We will add these 2 for demo purposes.



Edit Segment ②

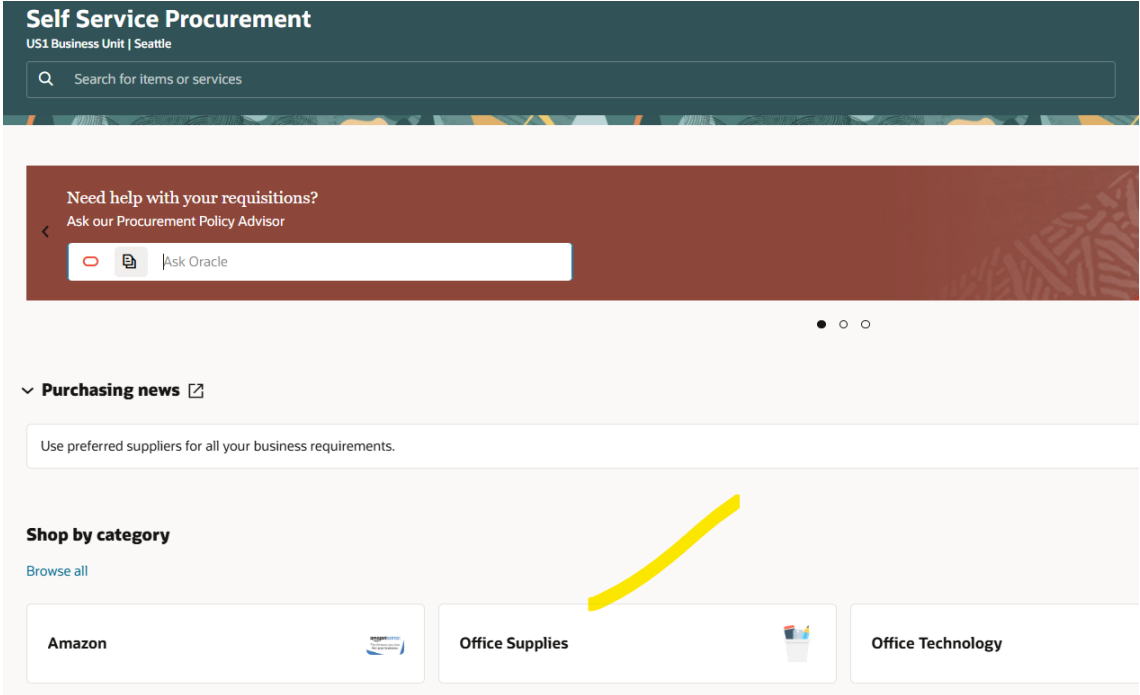
Flexfield Name Requisition Headers		Flexfield Code POR_REQUISITION_HEADERS
* Name Demo Redwood		Description
Code Demo Redwood		<input checked="" type="checkbox"/> Enabled
* API Name demoRedwood		
Column Assignment		Table Column ATTRIBUTE1
Data Type Character		
Validation		Range Type
* Value Set 150Char		<input type="checkbox"/> Required
Value Set Description 150Char		
Initial Default		
Default Type		
Display Properties		Definition Help Text
* Prompt Demo Redwood		
* Display Type Text Box		Instruction Help Text
Display Size 20		
Display Height		
<input type="checkbox"/> Read-only		

Steps in Visual Builder

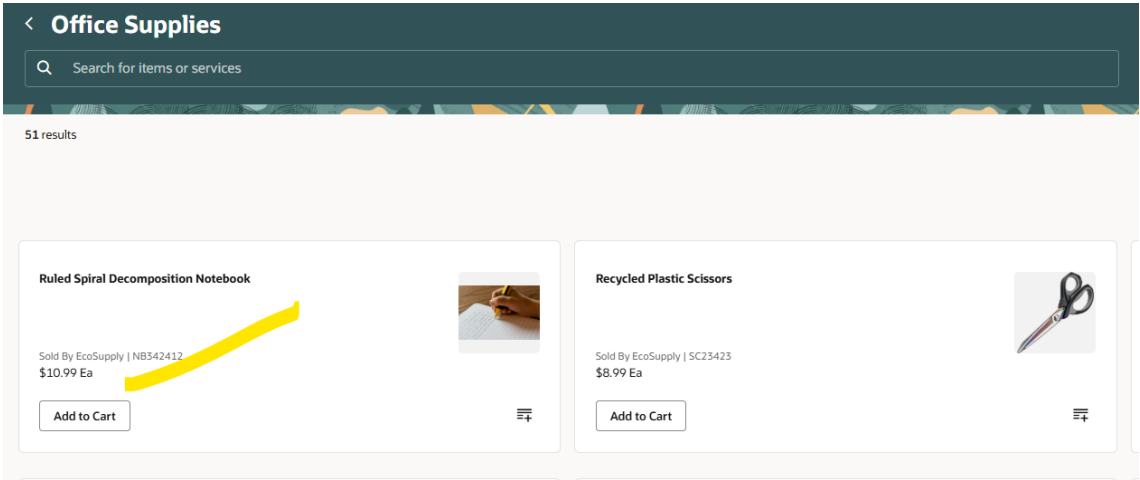
Navigate from main menu.



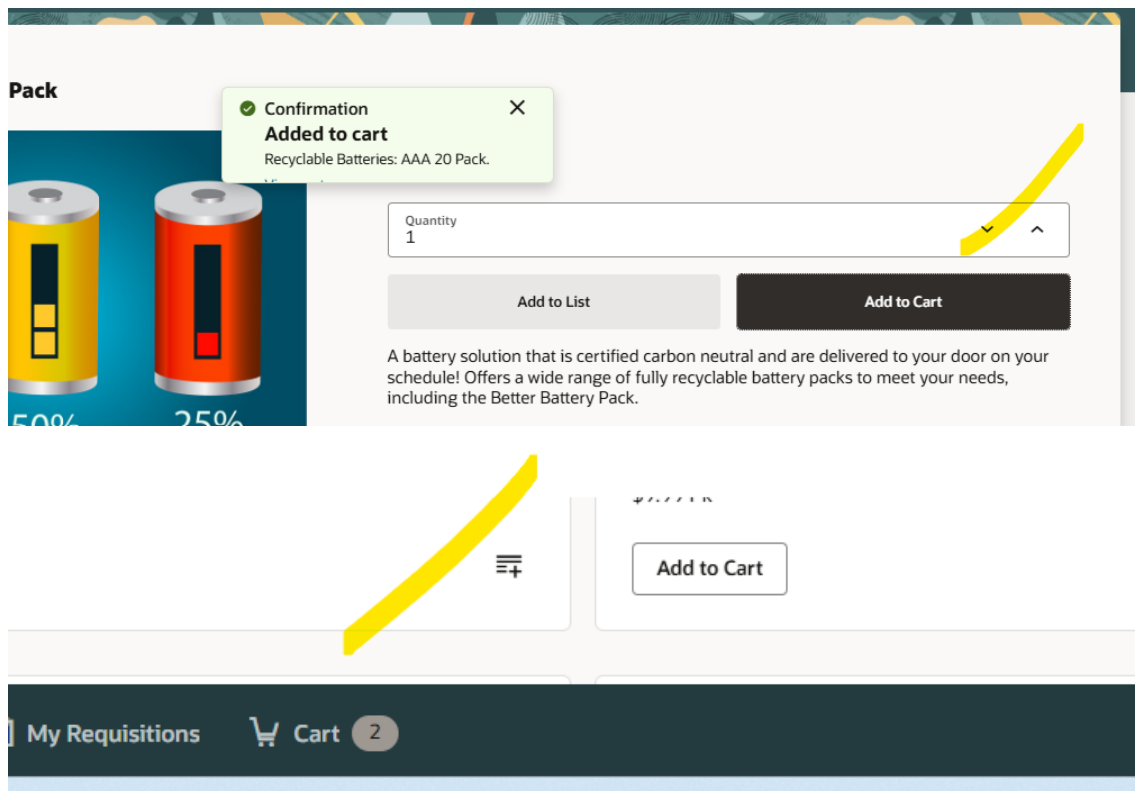
Select for example “Office supplies”.



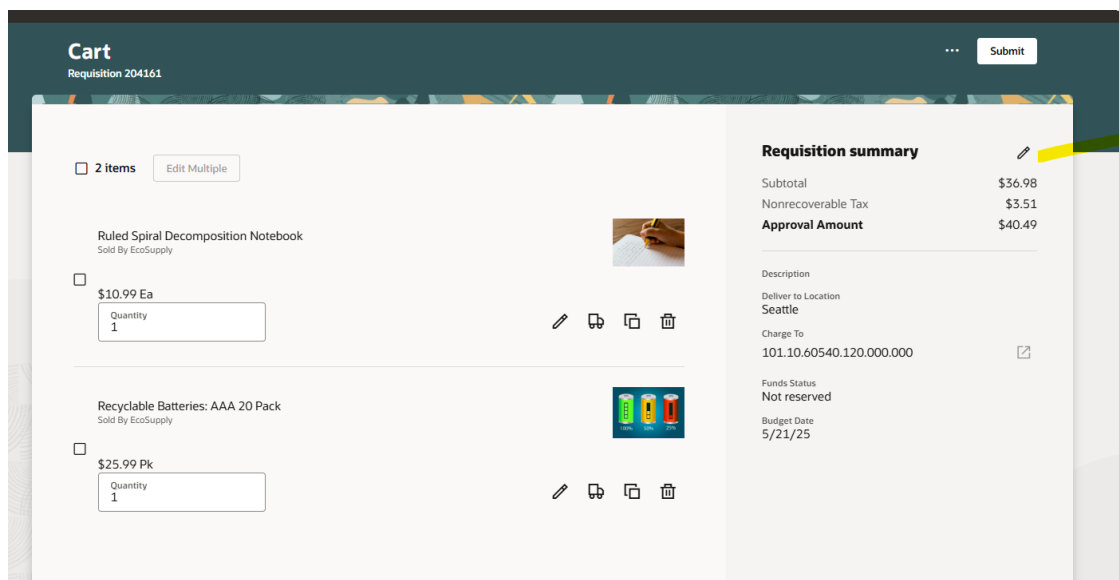
Add to Cart.



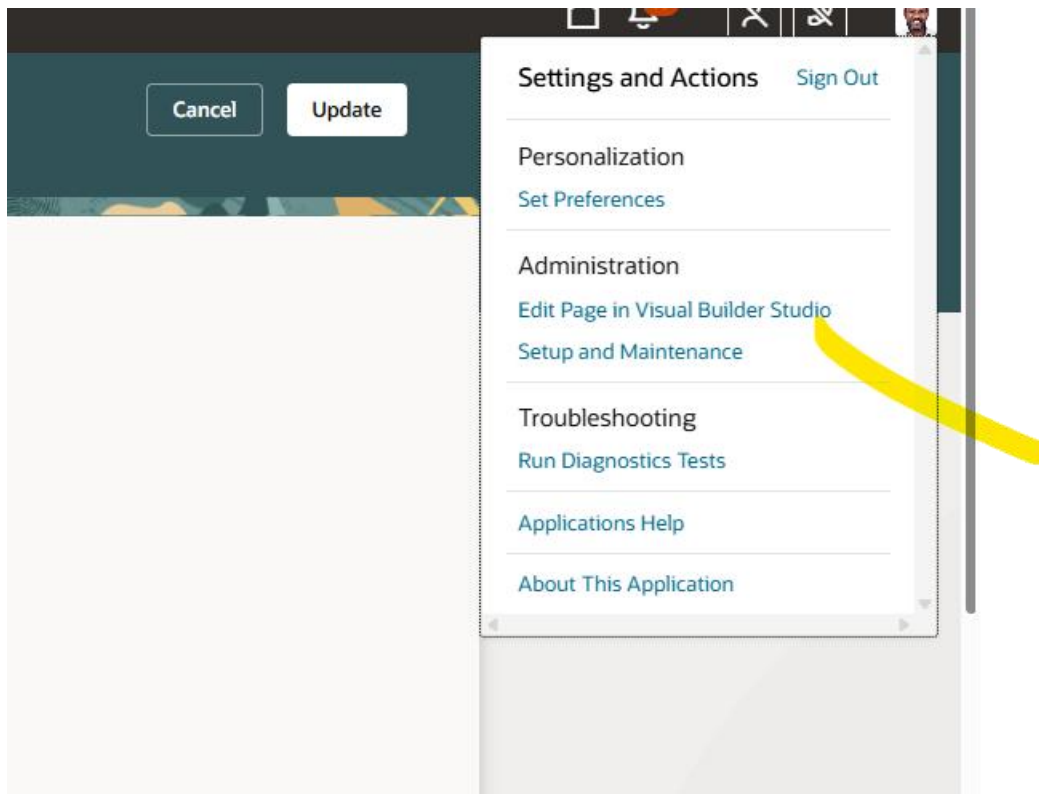
Press button.



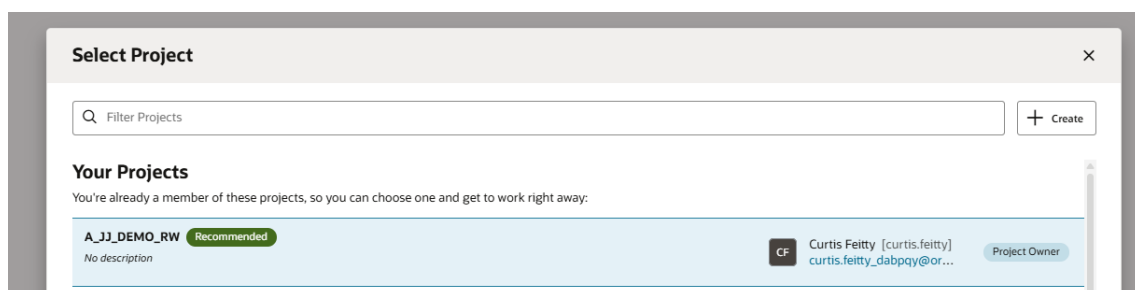
Here click on “Edit”.



Now, open Visual Builder tool as shown below.

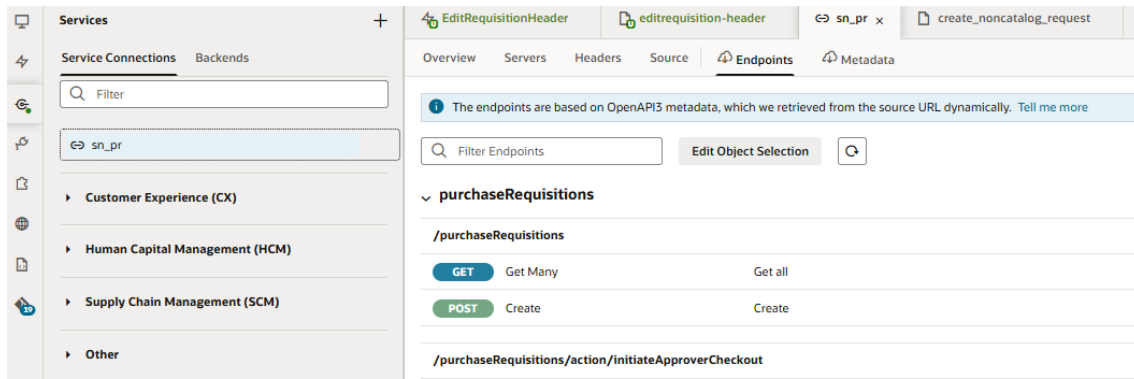


Use project or create one.

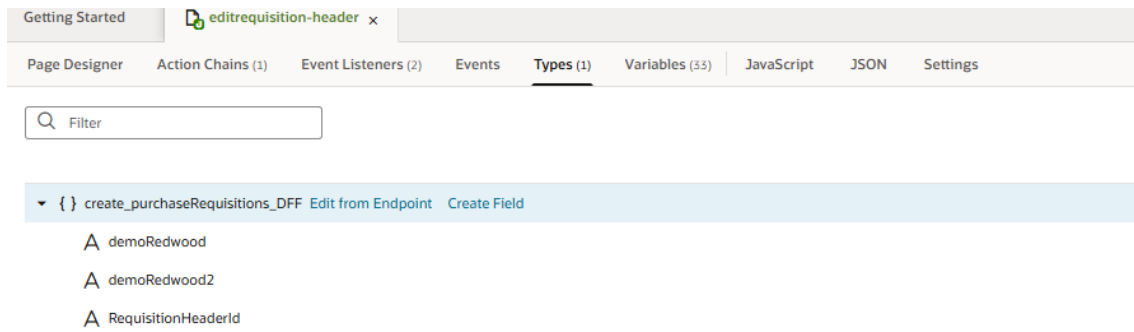


Now in Visual Builder tool.

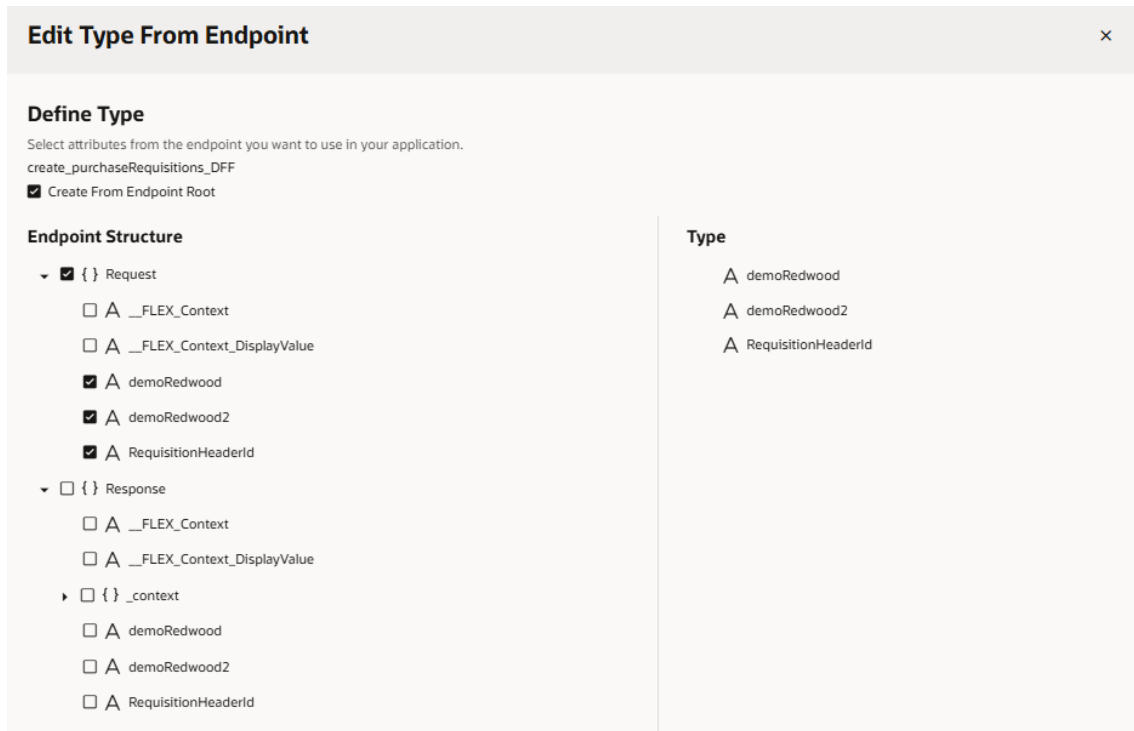
Create a connection for Purchase Requisitions REST API.



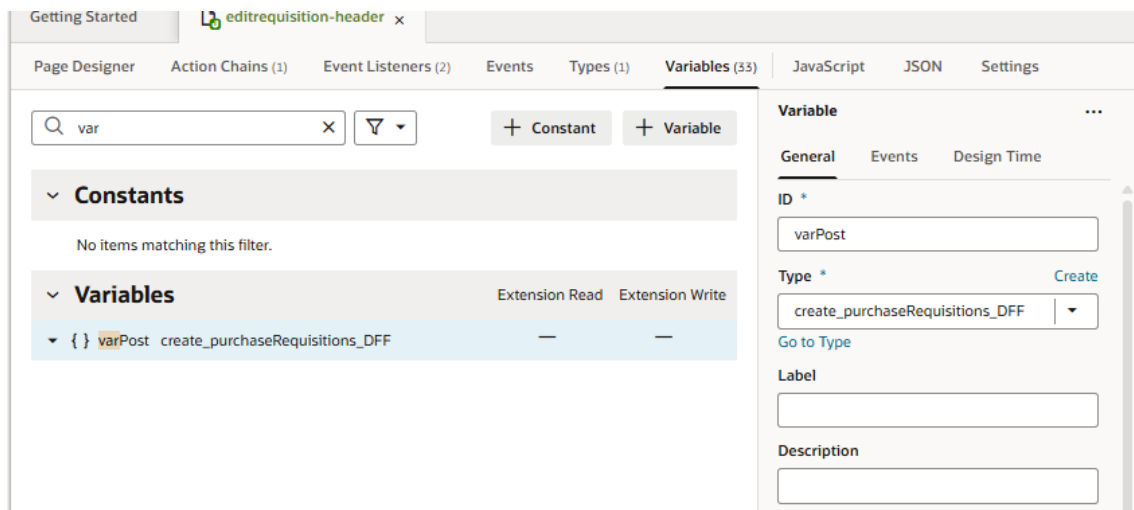
Create this type using connection created above:



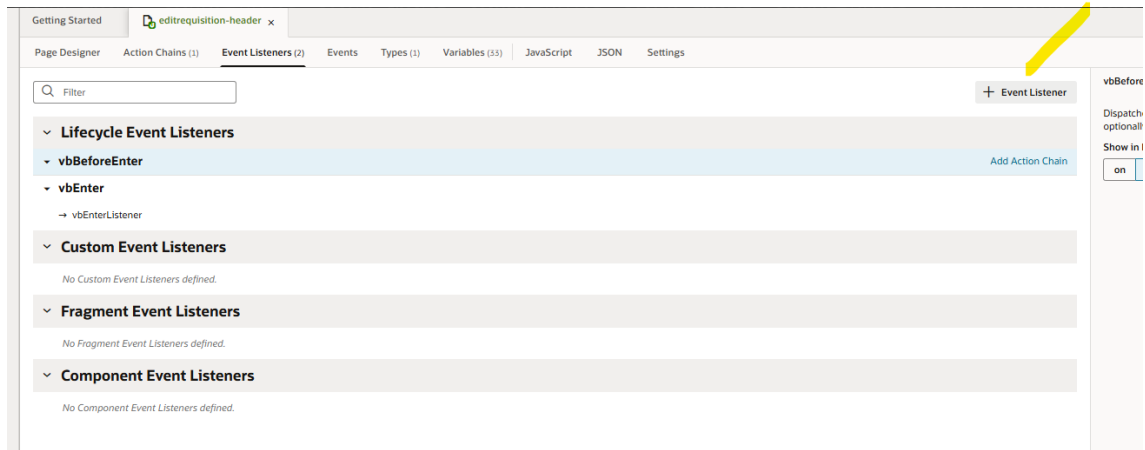
Select fields as shown.



Create variable using type above:

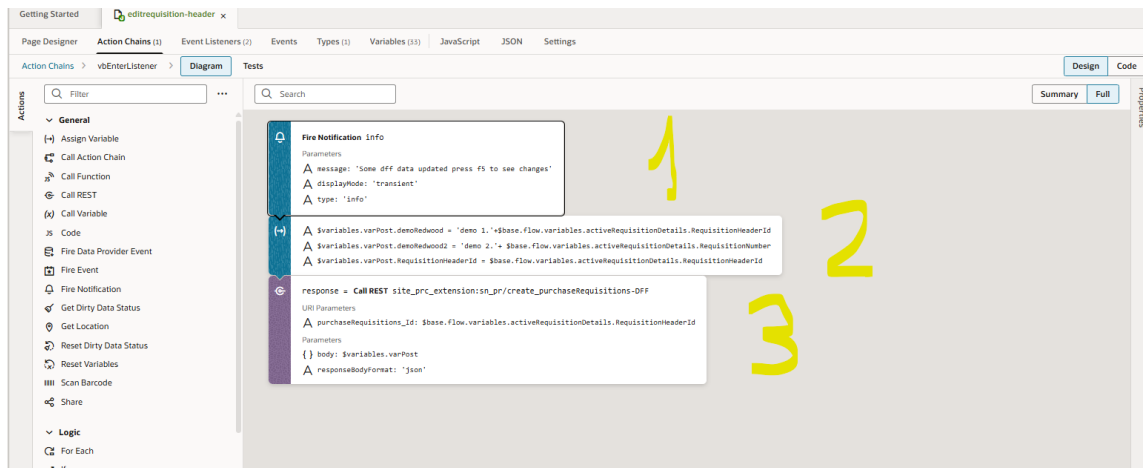


Create Event listener “vbEnter”.



In the Action chain we will have this code.

Note: code provided, see below.



1. Message to the user for debugging purposes.
2. Fill the payload for the REST API
3. Post the data from the REST API

Code in Action Chain.

```
define([
  'vb/action/actionChain',
  'vb/action/actions',
```

```

        'vb/action/actionUtils',
    ], (
        ActionChain,
        Actions,
        ActionUtils
    ) => {
        'use strict';

        class vbEnterListener extends ActionChain {

            /**
             * @param {Object} context
             */
            async run(context) {

                const { $page, $flow, $application, $base, $extension, $constants, $variables } =
                    context;

                await Actions.fireNotificationEvent(context, {
                    summary: 'info',
                    message: 'Some Descriptive flexfields data updated press f5 to see changes',
                    displayMode: 'transient',
                    type: 'info',
                });

                $variables.varPost.demoRedwood = 'demo
1.'+$base.flow.variables.activeRequisitionDetails.RequisitionHeaderId;

                $variables.varPost.demoRedwood2 = 'demo 2.'+
$base.flow.variables.activeRequisitionDetails.RequisitionNumber;

                $variables.varPost.RequisitionHeaderId =
$base.flow.variables.activeRequisitionDetails.RequisitionHeaderId;

                const response = await Actions.callRest(context, {
                    endpoint: 'site_prc_extension:sn_pr/create_purchaseRequisitions-Descriptive
flexfields',
                    uriParams: {
                        'purchaseRequisitions_Id':
$base.flow.variables.activeRequisitionDetails.RequisitionHeaderId,
                    },
                    body: $variables.varPost,
                    responseBodyFormat: 'json',
                });
            }
        }
    }
}

```

```

        });
    }
}

return vbEnterListener;
});

```

Technical

Code example	Comments
scm_po_requisitions_default_dff.zip	To be used in following chapters to save time. Use it as reference and make refinements at will.

Bonus track

In this and next described step with the information obtained from the rest you could add your logic.

Example:

```

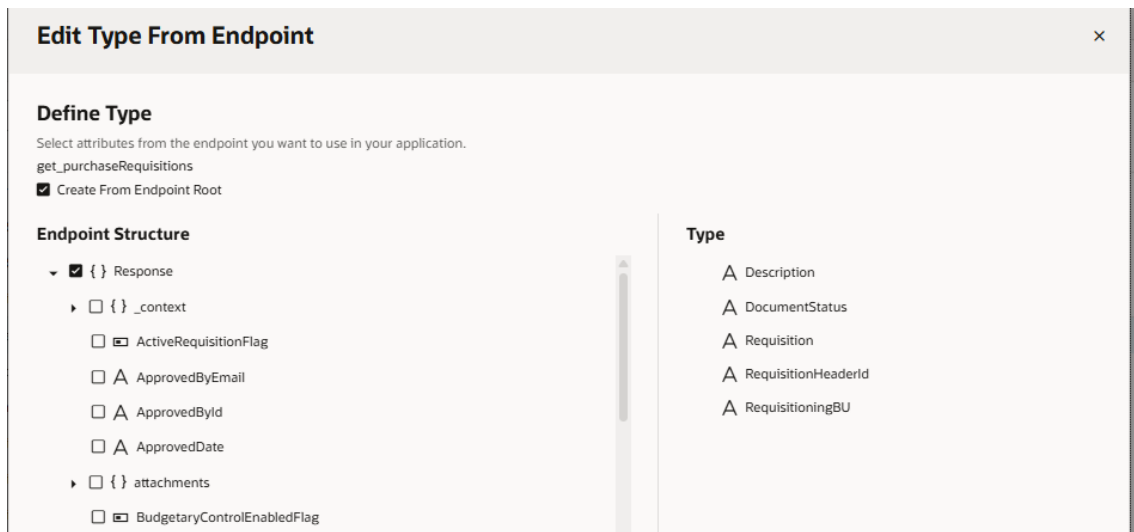
IF BUSINESS UNIT=A
    Descriptive flexfields1=WHATEVER
IF BUSINESS UNIT=B
    Descriptive flexfields1=WHATEVER
ELSE
    Descriptive flexfields1=OTHERS

```

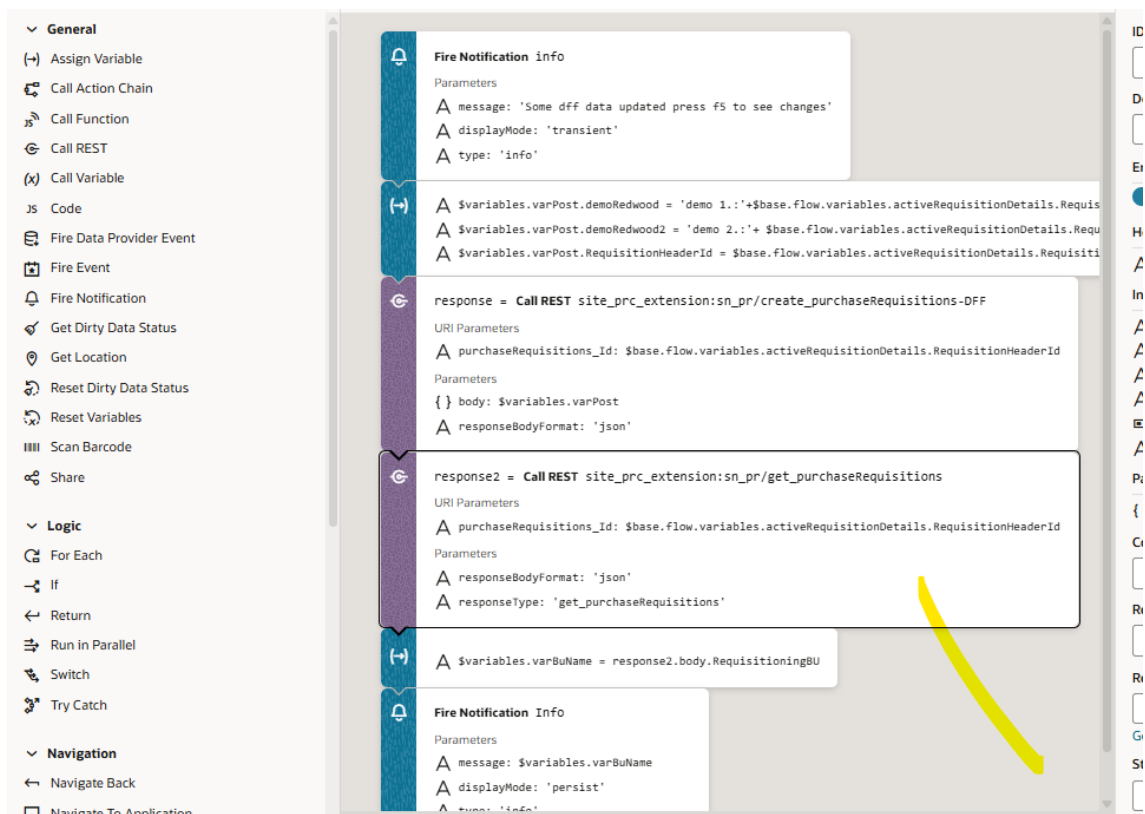
To achieve this, we should get the data from purchase requisitions using REST API and then build with the if above our logic.

To run the REST API, we have included the following steps.

- New type



- Run REST call.



- Show message for learning purposes.

When testing:

The screenshot shows a web interface for a requisition form. At the top left, the requisition number '204161' is displayed. A modal window titled 'Information' is open, showing 'Info' and 'US1 Business Unit'. The form has several sections: 'Charge account' with radio buttons for 'Select favorite charge account' (selected) and 'Enter charge account number'; a dropdown for 'Charge Account Nickname' set to 'Main'; a 'Budgetary control' section with a 'Budget Date' of '5/21/25'; an 'Additional information' section with two text boxes containing 'Demo Redwood demo 1.:300000306453471' and 'Demo Redwood 2 demo 2.:204161'; and a 'Notes' section with a text box labeled 'Note to Supplier'. Two yellow arrows are drawn on the form: one points from the 'Charge account' section to the 'Charge Account Nickname' dropdown, and the other points from the 'Additional information' section to the 'Demo Redwood 2' text box.

IMPORTANT:

Feel free to improve with a control to avoid overwriting if the user has changed manually the Descriptive Flexfield data.

Conclusion

We are able to default with specific logic in Descriptive Flexfield data in RSSP pages.

Technical

Code example	Comments
scm_po_requisitions_default_dff_2.zip	To be used in following chapters to save time. Use it as reference and make refinements at will.

