

Provide Tax Code in Billing Document Output

SAP S/4HANA Cloud



THE BEST RUN



TABLE OF CONTENTS

1	<i>Provide Tax Code in Billing Document Output</i>	3
1.1	Aim	3
1.2	Steps	3
1.2.1	Creation of a custom extension field on billing document item level	3
1.2.2	Extend the Form Template with the Custom Field	4
1.2.3	Creation of custom logic	4

1 Provide Tax Code in Billing Document Output

1.1 Aim

Add item level tax code on form template for billing document output.

1.2 Steps

1.2.1 Creation of a custom extension field on billing document item level

Use the app *Custom Fields and Logic*, tab *Custom Fields*, which is part of the Extensibility business catalog. Create a new custom field in *Business Context Billing Document Item* with an appropriate type.

The screenshot shows a 'New Field' configuration window. The 'Field Properties' section includes the following fields:

- *Business Context:** Sales: Billing Document Item
- *Label:** TaxCode
- *Identifier:** YY1_ TaxCode
- *Tooltip:** TaxCode
- *Type:** Text
- *Length:** 2

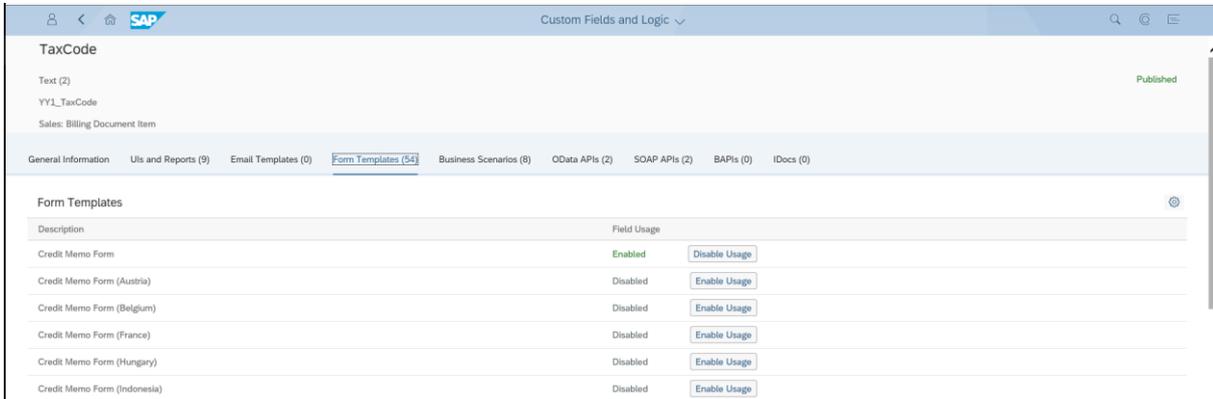
At the bottom, there is a 'Business Context Capacity' indicator showing 16% and three buttons: 'Create', 'Create and Edit', and 'Cancel'.

This custom field is intended to store the tax code on billing document item level.

Note: The custom field is not filled until the output request is triggered.

1.2.2 Extend the Form Template with the Custom Field

Within the same app *Custom Fields and Logic*, and on the tab *Custom Field*, enable the form templates for the billing document with the custom field.



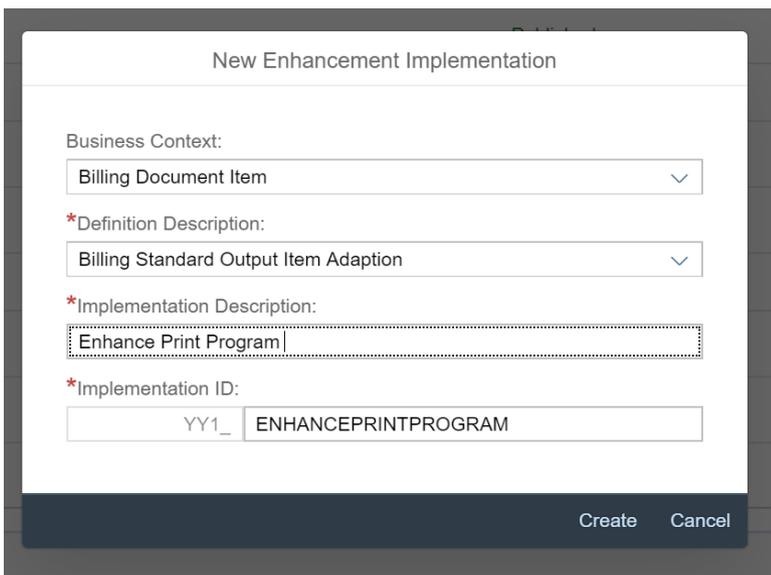
Save and publish the custom field.

1.2.3 Creation of custom logic

Use the app *Custom Fields and Logic*, and go to the *Custom Logic* tab.

Create a new Enhancement Implementation in *Business Context Sales: Billing Document Item* with *Definition Description Billing Standard Output Item Adaption* and add an appropriate Implementation Description.

In case the Enhancement Implementation already exists, you can skip this step.



Add the following custom logic to the Draft:

```
* First of all assign existing custom field values to output parameter:
billingdocitem_ext_out = billingdocitem_ext_in.
billingdocitemdescr_ext_out = billingdocitemdescr_ext_in.
*
* Read tax code for specific billing document item:
SELECT SINGLE taxcode
  FROM i_billingdocitemprcgelmntbasic
 WHERE billingdocument = @billingdocitem-billingdocument
 AND   billingdocumentitem = @billingdocitem-billingdocumentitem
 AND   pricingprocedurestep = '850'
 INTO @billingdocitem_ext_out-yy1_taxcode_bdi.
```

Alternative:

```
* First of all assign existing custom field values to output parameter:
billingdocitem_ext_out = billingdocitem_ext_in.
billingdocitemdescr_ext_out = billingdocitemdescr_ext_in.
*
* Read tax code for specific billing document item:
SELECT SINGLE taxcode
  FROM i_billingdocitemprcgelmntbasic
 WHERE billingdocument = @billingdocitem-billingdocument
 AND   billingdocumentitem = @billingdocitem-billingdocumentitem
 AND ( ConditionCategory = 'D' or ConditionCategory = '+' )
 AND ConditionInactiveReason IS INITIAL
 INTO @billingdocitem_ext_out-yy1_itmtaxcode_bdi.
```

Save and publish the custom logic.

The screenshot shows the SAP Custom Fields and Logic editor interface. The title bar indicates 'Custom Fields and Logic'. The main content area is titled 'EnhancePrintProgramm' and shows 'Last Changed By billing_clerk John, 20.06.2018 14:20:53' and a 'Published' status. Below this, there are input fields for 'billingdoc:', 'billingdocitem:', 'billingdocitem_ext_in:', and 'billingdocitemdescr_ext_in:', each with a 'Click to add value' button. At the bottom, the 'Published Logic' section contains the following code:

```
1 This method can be used to set values for output to billing document item extension fields.
2 * Example:
3 -
4 - First of all assign existing extension field values to output parameter:
5 billingdocitem_ext_out = billingdocitem_ext_in.
6 billingdocitemdescr_ext_out = billingdocitemdescr_ext_in.
7 -
8 - Read tax code for specific billing document item:
9 -
10 SELECT SINGLE taxcode
11   FROM i_billingdocitemprcgelmntbasic
12  WHERE billingdocument = @billingdocitem-billingdocument
13        AND   billingdocumentitem = @billingdocitem-billingdocumentitem
14        AND   pricingprocedurestep = '850'
15        INTO @billingdocitem_ext_out-yy1_taxcode_bdi.
```

Note:

- *The pricing procedure step depends on the pricing procedure and, in case of custom pricing procedures, the value could differ.*
- *To see the newly created custom field on the billing document printout, it's necessary to create a new form template.*