

This is a Practice Based Working Document. Everyone can change by their own requirements.

Scenario:

The scenario is when a Purchase Order Document creates a workflow will send an approval request email to the approver. That email will be received in Gmail Inbox or any other email server.

The approver will open that email in any device by using Gmail or mail server app or website. He will have two options Approve or Reject the purchase order.

He will just click on approve/reject button in email and the effect will happen in SAP Inbox.

The Purchase Order will be approved or rejected directly in SAP.

Transaction Codes Used:

1. SWO1 – To create business object.
2. SE37 – To create Function Module for Email Body
3. PFTC – To Create Workflow Task and Template
4. SEGW – Create ODATA
5. SOST – To Check Emails in Inbox

Go to SW01:

Create a business object by follow these the steps.

Step 1:

Business Object Builder: Initial Screen

Business Object Repository

Object/Interface Type **ZEMAPBOB_T**

Category

☒ Object type ☐ Interface type

Test

Display Change Create

2

Must Assign Package and TR. Otherwise It May create issues.

Business Object Builder: Initial Screen

Business Object Repository

Object/Interface Type

Category

☒ Object type ☐ Interface type

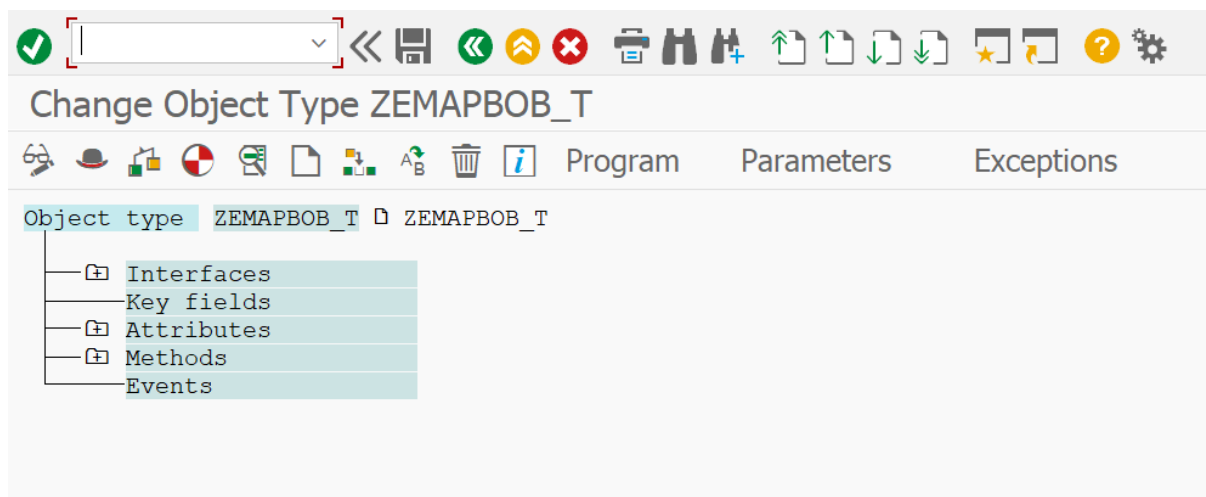
Display

Create Object Type

Supertype	
Object Type	ZEMAPBOB_T
Object name	ZEMAPBOB_T
Name	ZEMAPBOB_T
Description	ZEMAPBOB_T
Program	ZEMAPBOB_T
Application	*

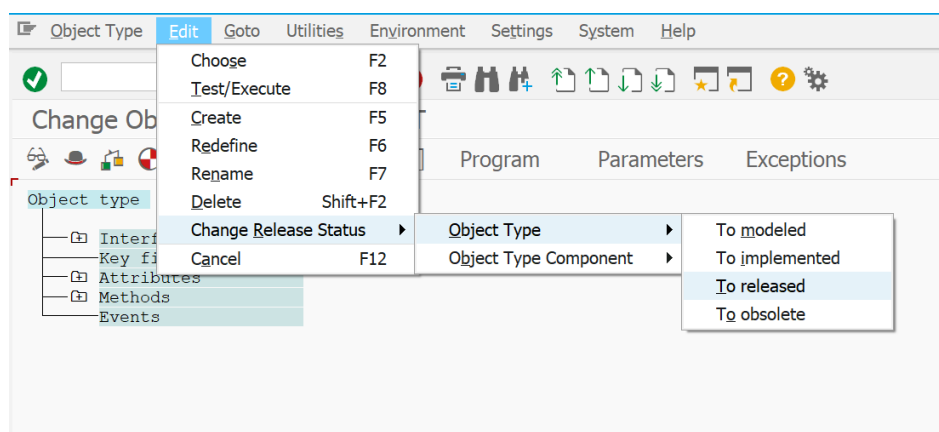
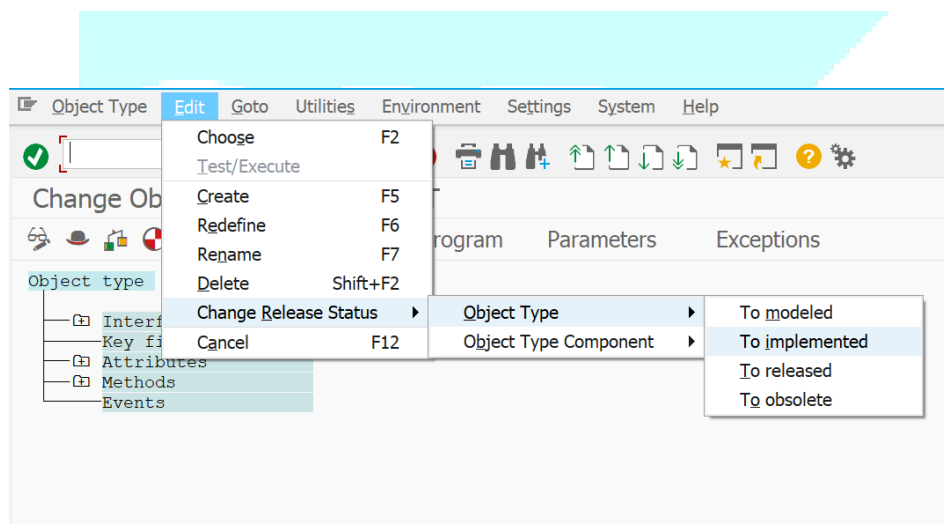
✓ ✗

The Business Object is created. Now we have to arrange it according to the need.

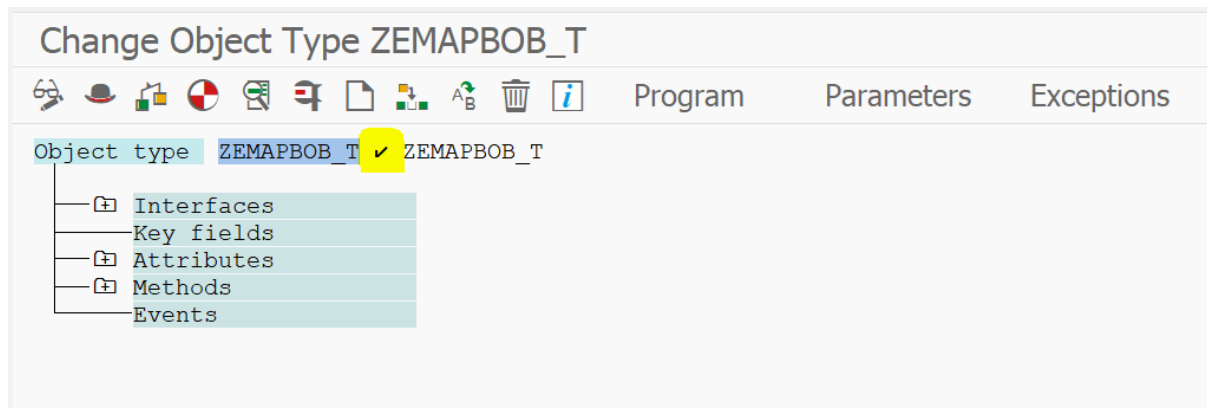


Step 2:

Must Implement and Release the Business object to make it available for use.

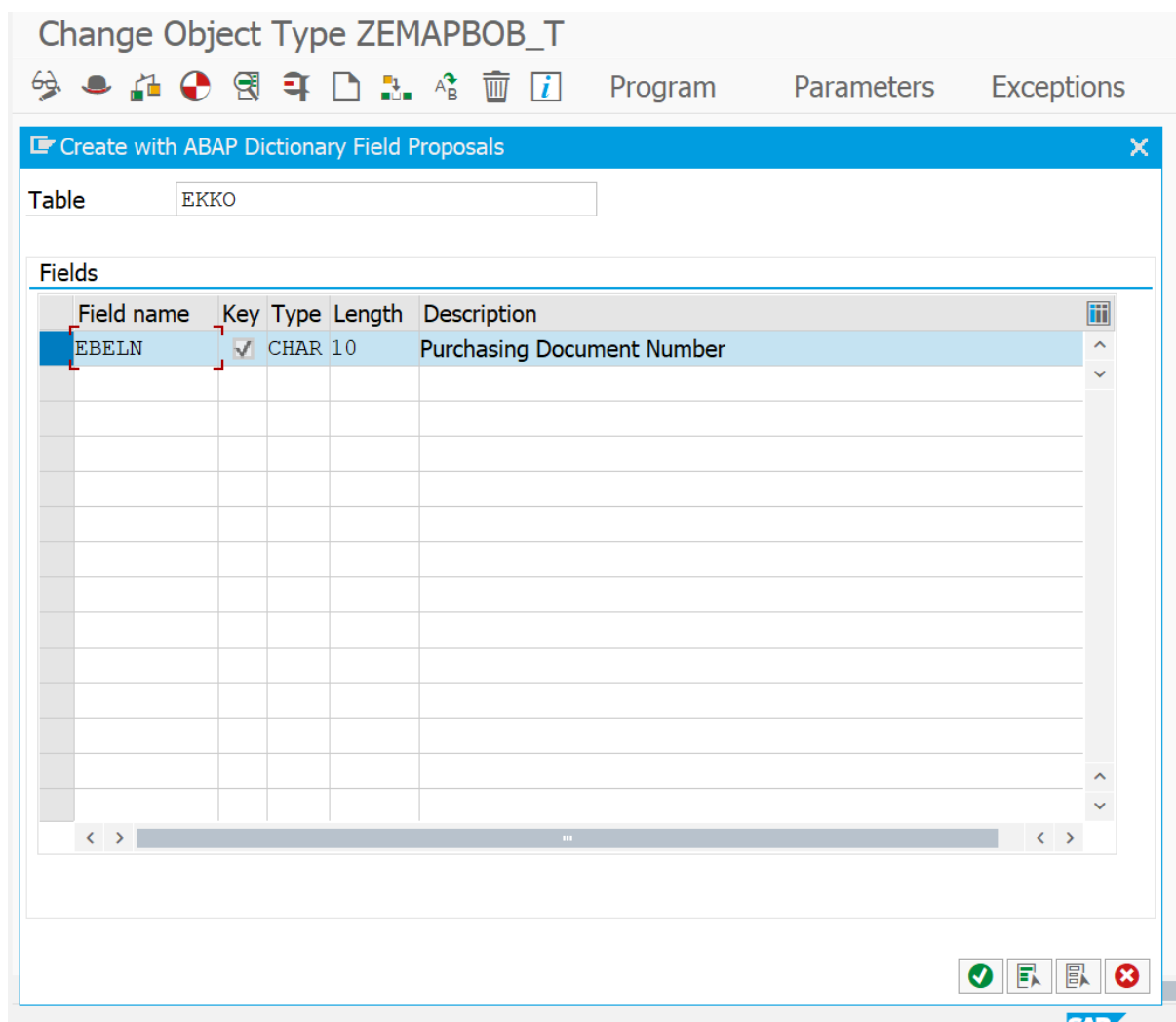
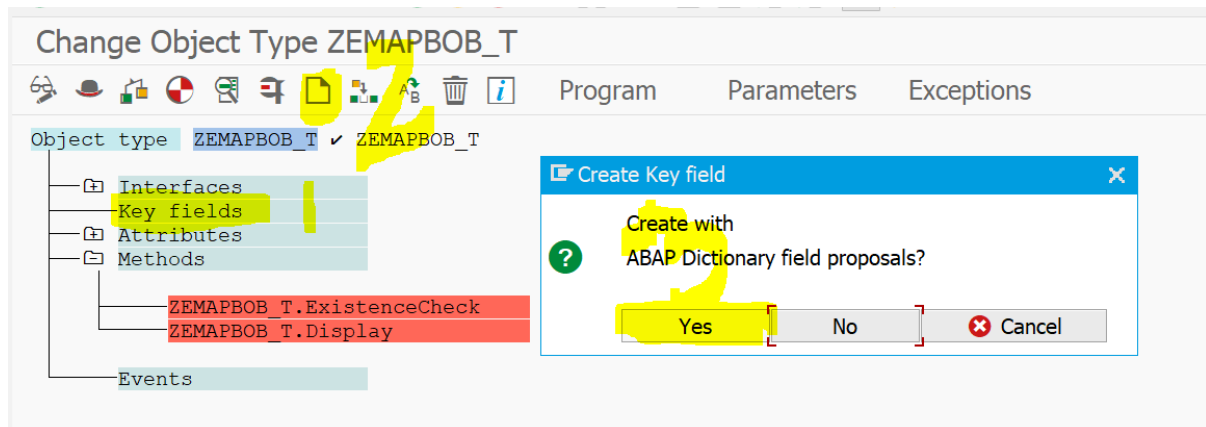


Now, it is available to use for Workflow.



Step 3:

We must create Key Field that is Purchase Order Number in Our case.



Create



Key field: PurchasingDocument

Texts

Name	Purchasing Document
Description	Purchasing Document Number

DDIC table field

Reference table	EKKO
Reference field	EBELN
	Purchasing Document Number

The Key Field is created successfully.

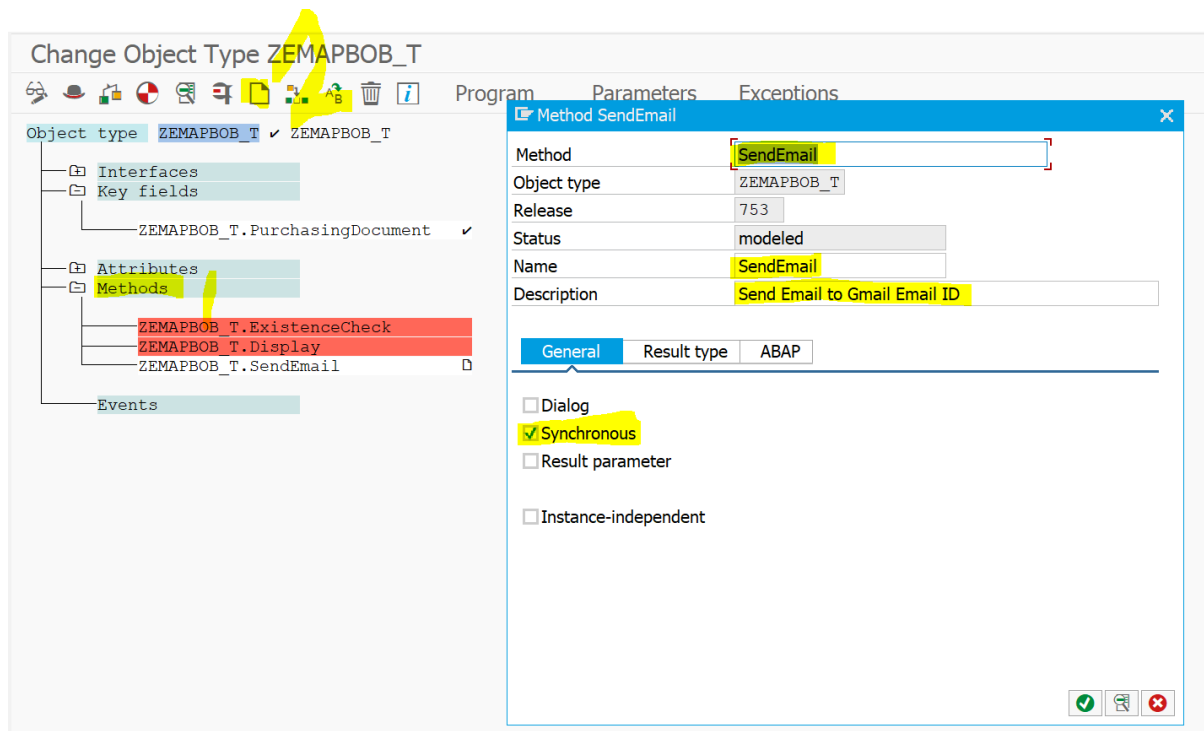
Change Object Type ZEMAPBOB_T

Object type: ZEMAPBOB_T ✓ ZEMAPBOB_T

- Interfaces
- Key fields
 - ZEMAPBOB_T.PurchasingDocument ✓ Purchasing Document Number
- Attributes
- Methods
 - ZEMAPBOB_T.ExistenceCheck Check existence of object
 - ZEMAPBOB_T.Display Display object
- Events

Step 4:

We have to create a method in this Business Object to send Email to the recipient.



Method SendEmail

Method	SendEmail
Object type	ZEMAPBOB_T
Release	753
Status	modeled
Name	SendEmail
Description	Send Email to Gmail Email ID




General Result type ABAP

☐ ABAP Dictionary

Reference table	
Reference field	
Search Help	

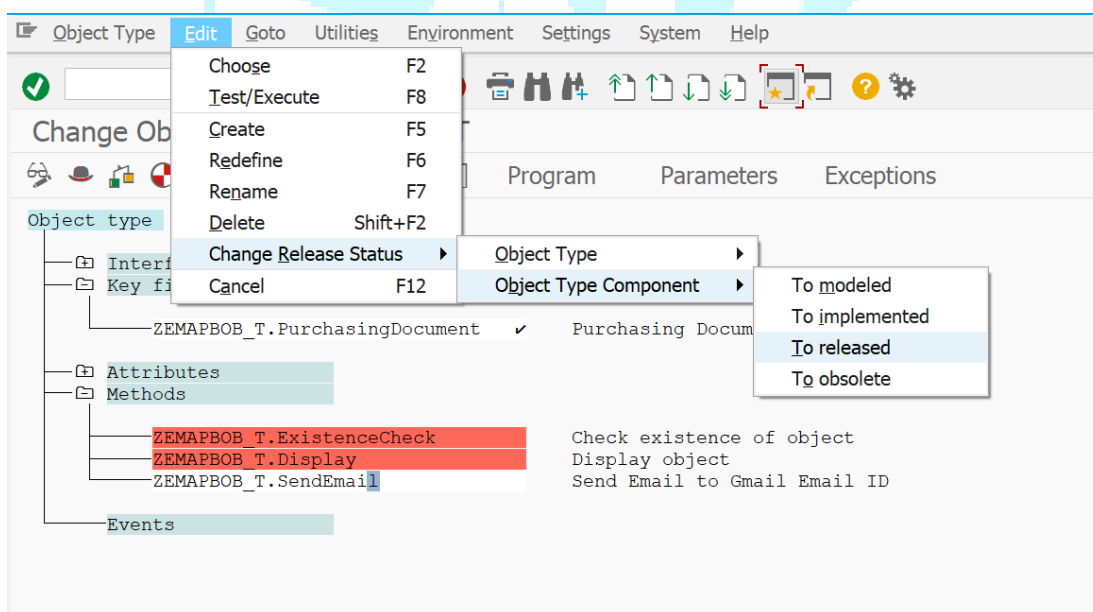
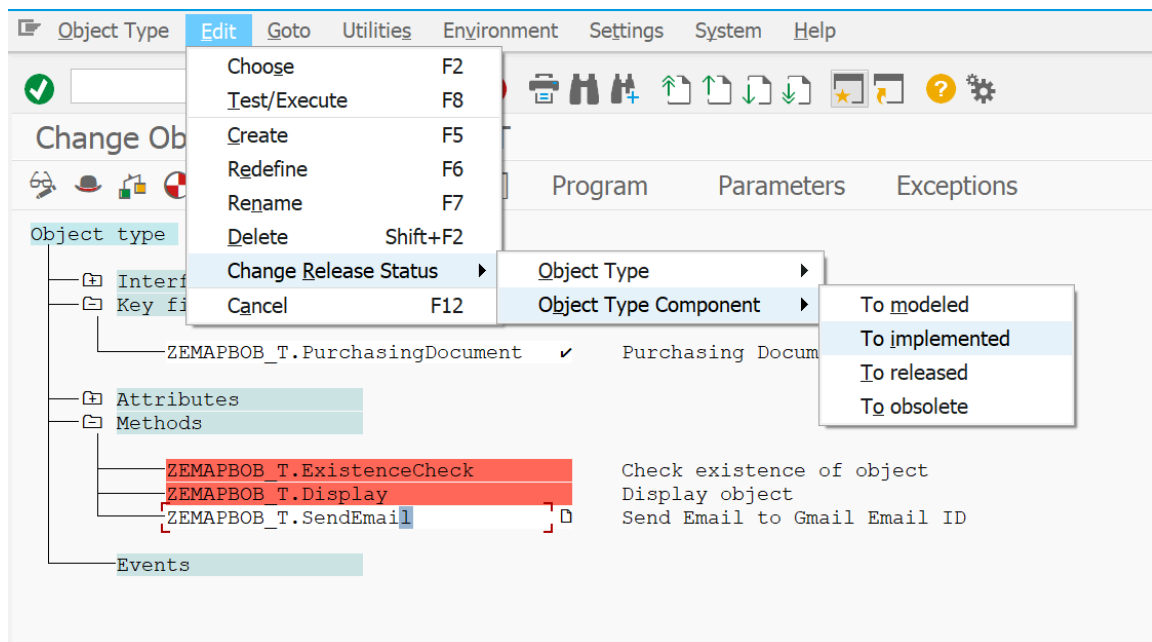
☒ Object type ZEMAPBOB_T

☐ Multiline

Note: Must step after creating any object in BO.

We must need to Implement and Release the Method to make it available for use in Workflow Tasks.



Method is now available to use.

Change Object Type ZEMAPBOB_T

Object type **ZEMAPBOB_T** ✓ ZEMAPBOB_T

- Interfaces
- Key fields
 - ZEMAPBOB_T.PurchasingDocument ✓ Purchasing Document Number
- Attributes
- Methods
 - ZEMAPBOB_T.ExistenceCheck Check existence of object
 - ZEMAPBOB_T.Display Display object
 - ZEMAPBOB_T.SendEmail1 ✓ Send Email to Gmail Email ID
- Events



Step 5:

Create the parameters for the method that should be passed to that method during workflow running.

Change Object Type ZEMAPBOB_T

Object type **ZEMAPBOB_T** ✓ ZEMAPBOB_T

- Interfaces
- Key fields
 - ZEMAPBOB_T.PurchasingDocument ✓ Purchasing Document Number
- Attributes
- Methods
 - ZEMAPBOB_T.ExistenceCheck Check existence of object
 - ZEMAPBOB_T.Display Display object
 - ZEMAPBOB_T.SendEmail ✓ Send Email to Gmail Email ID
- Events
 - ZEMAPBOB_T.Create ✓ Create Event

Object Type ZEMAPBOB_T: Edit Parameters for Method SENDEMAIL

Overview

Parameter	Obj. Type	First Release	Imp.	Man.	Exp.	Key
-----------	-----------	---------------	------	------	------	-----

Create Parameter

Create with
ABAP Dictionary field proposals?

Yes No Cancel

Object Type ZEMAPBOB_T: Edit Parameters for Method SENDEMAIL

Other View Program Exceptions

Create with ABAP Dictionary Field Proposals

Table:

Fields

Field name	Key	Type	Length	Description
MANDT	<input checked="" type="checkbox"/>	CLNT	3	Client
EBELN	<input checked="" type="checkbox"/>	CHAR	10	Purchasing Document Number
BUKRS	<input type="checkbox"/>	CHAR	4	Company Code
BSTYP	<input type="checkbox"/>	CHAR	1	Purchasing Document Category
BSART	<input type="checkbox"/>	CHAR	4	Purchasing Document Type
BSAKZ	<input type="checkbox"/>	CHAR	1	Control indicator for purchasing document type
LOEKZ	<input type="checkbox"/>	CHAR	1	Deletion indicator in purchasing document
STATU	<input type="checkbox"/>	CHAR	1	Status of Purchasing Document
AEDAT	<input type="checkbox"/>	DATS	8	Date on which the record was created
ERNAM	<input type="checkbox"/>	CHAR	12	Name of Person who Created the Object
LASTCHANG...	<input type="checkbox"/>	DEC	21	Change Time Stamp
PINCR	<input type="checkbox"/>	NUMC	5	Item Number Interval
LPONR	<input type="checkbox"/>	NUMC	5	Last Item Number
LIFNR	<input type="checkbox"/>	CHAR	10	Vendor's account number

✓ ✗

Object Type ZEMAPBOB_T: Edit Parameters for Method SENDEMAIL

Other View Program Exceptions

Overview **Parameter**

Create

Parameter:

Texts

Name:

Description:

Parameter attributes

☒ Import ☐ Mandatory

☒ Export ☐ Supplied from Key

☐ Multiline

DDIC table field

Reference table:

Reference field:

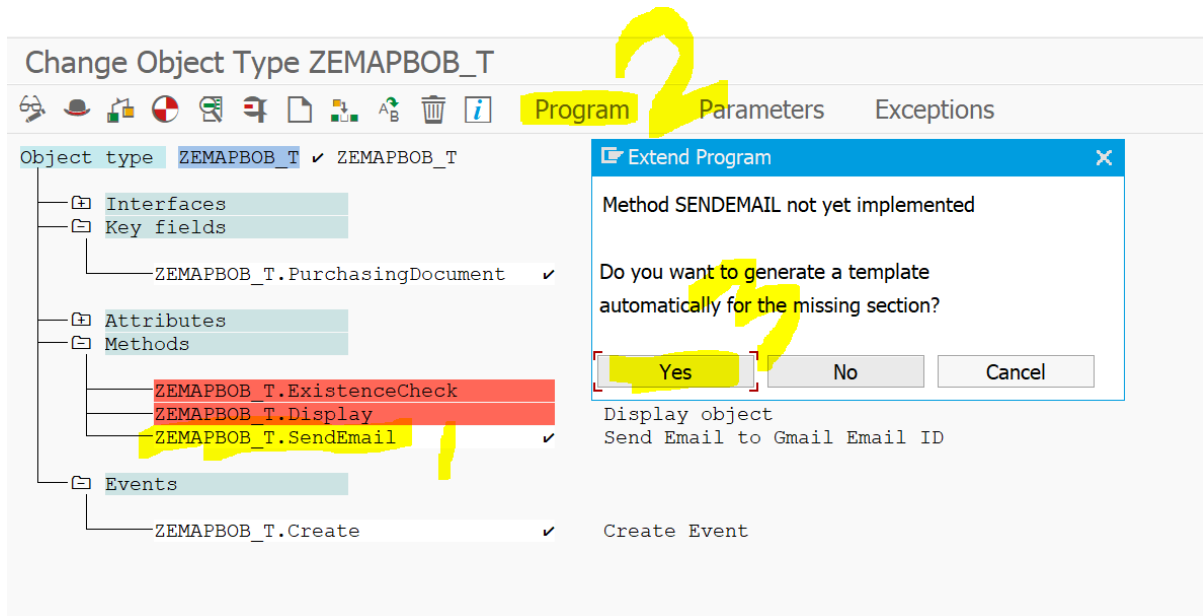
Purchasing Document Number

 ✗

Step 6:

Now we have to write code in the created method to send Email.

Note: The code is written for practice only. You can write by your own way with the new ABAP syntax as well.



Code for the Email Send:

```
BEGIN_METHOD SENDEMAIL CHANGING CONTAINER.
```

```
DATA: PURCHASINGDOCUMENT TYPE EKKO-EBELN.
```

```
SWC_GET_ELEMENT CONTAINER 'PurchasingDocument' PURCHASINGDOCUMENT.
```

CONSTANTS:

*-- Constants used in the body of the Email (HTML)

```
C_HTM      TYPE CHAR3  VALUE 'HTM',
```

```
C_STYLE_START TYPE CHAR255 VALUE '<FONT face=Arial size=2>',
```

```
C_NEW_LINE  TYPE CHAR255 VALUE '<br>',
```

```
C_LINK_START TYPE CHAR128 VALUE '<A href="www.w3schools.com">',
```

```
C_LINK_TEXT  TYPE CHAR32  VALUE 'Link for learning HTML',
```

```
C_LINK_END   TYPE CHAR4   VALUE '</A>',
```

```
C_SPACE(6)  TYPE C       VALUE '&nbsp;';
```

```
DATA: GT_CONT TYPE STANDARD TABLE OF BAPIASCONT255,
```

GW_CONT LIKE LINE OF GT_CONT.

DATA: SEND_REQUEST TYPE REF TO CL_BCS,
MAILSUBJECT TYPE SO_OBJ_DES,
MAILTEXT TYPE BCSY_TEXT,
DOCUMENT TYPE REF TO CL_DOCUMENT_BCS,
SENDER TYPE REF TO CL_CAM_ADDRESS_BCS,
RECIPIENT_TO TYPE REF TO CL_CAM_ADDRESS_BCS,
RECIPIENT_CC TYPE REF TO CL_CAM_ADDRESS_BCS,
RECIPIENT_BCC TYPE REF TO CL_CAM_ADDRESS_BCS,
BCS_EXCEPTION TYPE REF TO CX_BCS.

TRY.

SEND_REQUEST = CL_BCS=>CREATE_PERSISTENT().

MAILSUBJECT = 'Release PO Notification'.

CALL FUNCTION 'ZSENDEMAIL_THROUGH_WORKFLOW'

EXPORTING

EBELN = PURCHASINGDOCUMENT

TABLES

GT_CONT = GT_CONT.

DOCUMENT = CL_DOCUMENT_BCS=>CREATE_DOCUMENT(

I_TYPE = C_HTM

I_TEXT = GT_CONT

I_LANGUAGE = SY-LANGU

I_SUBJECT = MAILSUBJECT).

SEND_REQUEST->SET_DOCUMENT(DOCUMENT).

SENDER = CL_CAM_ADDRESS_BCS=>CREATE_INTERNET_ADDRESS('practice@gmail.com').

SEND_REQUEST->SET_SENDER(SENDER).

RECIPIENT_TO = CL_CAM_ADDRESS_BCS=>CREATE_INTERNET_ADDRESS('practice@gmail.com').

SEND_REQUEST->ADD_RECIPIENT(I_RECIPIENT = RECIPIENT_TO).

RECIPIENT_CC = CL_CAM_ADDRESS_BCS=>CREATE_INTERNET_ADDRESS('practice@gmail.com').

```

SEND_REQUEST->ADD_RECIPIENT( I_RECIPIENT = RECIPIENT_CC
I_COPY = 'X' ).
RECIPIENT_BCC = CL_CAM_ADDRESS_BCS=>CREATE_INTERNET_ADDRESS( 'practice@gmail.com' ).
SEND_REQUEST->ADD_RECIPIENT( I_RECIPIENT = RECIPIENT_BCC
I_BLIND_COPY = 'X' ).
DATA(LV_SENT_TO_ALL) = SEND_REQUEST->SEND( ).
IF LV_SENT_TO_ALL = 'X'.
*   WRITE 'Email sent to all recipients'.
ELSE.
*   WRITE 'Email could not be sent to all recipients!'.
ENDIF.
COMMIT WORK.
CATCH CX_BCS INTO BCS_EXCEPTION.
*   WRITE: 'Error occurred while sending email: Error Type', BCS_EXCEPTION->ERROR_TYPE.
ENDTRY.
SWC_SET_ELEMENT CONTAINER 'PurchasingDocument' PURCHASINGDOCUMENT.
END_METHOD.

```

Function Module Code: ZSENDEMAIL_THROUGH_WORKFLOW

```

DATA: G_URL_FIOR1 TYPE TEXT200,
      G_URL_FIOR2 TYPE TEXT200,
      LV_WIID  TYPE C LENGTH 12,
      LV_PO    TYPE C LENGTH 25.

" READ SELECTED WORK ITEM ID
SELECT WI_ID FROM SWWWIHEAD INTO TABLE @DATA(IT_WI_ID) WHERE TOP_TASK EQ 'WS90000135' AND
D WI_STAT EQ 'STARTED' AND WI_RH_TASK EQ 'TS90000200'.
SORT IT_WI_ID DESCENDING BY WI_ID.
READ TABLE IT_WI_ID INTO DATA(WA_WI_ID) INDEX 1.
WA_WI_ID-WI_ID = WA_WI_ID-WI_ID + 1.
WA_WI_ID-WI_ID = |{ WA_WI_ID-WI_ID ALPHA = IN }|.
MOVE WA_WI_ID-WI_ID TO LV_WIID.
CLEAR: LV_PO,WA_WI_ID,IT_WI_ID.

" CREATE LINK TO APPROVE WORK ITEM
MOVE EBELN TO LV_PO.
CONCATENATE LV_PO 'A' LV_WIID INTO LV_PO SEPARATED BY SPACE.
CONCATENATE 'https://-----/sap/opu/odata/SAP/ZAPPROVEPO_SRV/poSet(' LV_PO ') ' INTO G_URL_FIOR1.

" CREATE LINK TO REJECT WORK ITEM
CLEAR: LV_PO.
MOVE EBELN TO LV_PO.

```

```
CONCATENATE LV_PO 'R' LV_WIID INTO LV_PO SEPARATED BY SPACE.
CONCATENATE 'https://-----/sap/opu/odata/SAP/ZAPPROVEPO_SRV/poSet(' LV_PO ' ' )' INTO G_URL_FIOR2.
```

CONSTANTS:

```
*-- Constants used in the body of the Email (HTML)
C_HTM      TYPE CHAR3  VALUE 'HTM',
C_STYLE_START TYPE CHAR255 VALUE '<FONT face=Arial size=2>',
C_NEW_LINE  TYPE CHAR255 VALUE '<br>',
C_LINK_START TYPE CHAR128 VALUE '<A href="www.w3schools.com">',
C_LINK_TEXT  TYPE CHAR32  VALUE 'Link for learning HTML',
C_LINK_END   TYPE CHAR4   VALUE '</A>',
C_SPACE(6)  TYPE C       VALUE '&nbsp;';
```

DATA: GW_CONT LIKE LINE OF GT_CONT.

**start-> === Filling Email header =====*

```
GW_CONT-LINE = '<html>'.
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
MOVE '<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">' TO GW_CONT-LINE.
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
MOVE '<head>' TO GW_CONT-LINE.
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
GW_CONT-LINE = '<em><font ' .
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
GW_CONT-LINE = 'color="#0000FF" size="+7" face="Arial,'.
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
GW_CONT-LINE = 'Helvetica, sans-serif"> </font></em>'.
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
MOVE '</head>' TO GW_CONT-LINE.
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
GW_CONT-LINE = '<br>'.
APPEND GW_CONT TO GT_CONT.
```

CLEAR GW_CONT.

```
CONCATENATE 'Dear' 'Sir, ' INTO GW_CONT-LINE SEPARATED BY SPACE.
```

APPEND GW_CONT TO GT_CONT.

```
CLEAR GW_CONT.
GW_CONT-LINE = '<br>'.
APPEND GW_CONT TO GT_CONT.
```

```
CONCATENATE 'Please Release PO' " INTO GW_CONT-LINE SEPARATED BY SPACE.
```



```
GW_CONT-LINE = '<body>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
CONCATENATE '<p>Purchase number' EBELN 'is waiting for your decision,'  
            'kindly login the following applications to approve the document:</p>' INTO GW_CONT-  
LINE SEPARATED BY SPACE.
```

```
APPEND GW_CONT TO GT_CONT.
```

"Insert URLs in bullets

```
GW_CONT-LINE = '<ul>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

"Option 1

```
CONCATENATE '<li>'
```

```
            '<a href="'
```

```
            G_URL_FIOR1
```

```
            '">'
```

```
            'Approve</a>'
```

```
            '</li>'
```

```
            INTO GW_CONT-LINE.
```

```
APPEND GW_CONT TO GT_CONT.
```

"Option 2

```
CONCATENATE '<li>'
```

```
            '<a href="'
```

```
            G_URL_FIOR2
```

```
            '">'
```

```
            'Reject</a>'
```

```
            '</li>'
```

```
            INTO GW_CONT-LINE.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
GW_CONT-LINE = '</ul>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
GW_CONT-LINE = '</body>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
GW_CONT-LINE = '<br><p>Regards</P>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
GW_CONT-LINE = '<p><i>This e-mail was generated automatically from'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
GW_CONT-LINE = 'SAP; please do not reply.</i></P>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
GW_CONT-LINE = '</body></html>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

```
CLEAR GW_CONT.
```

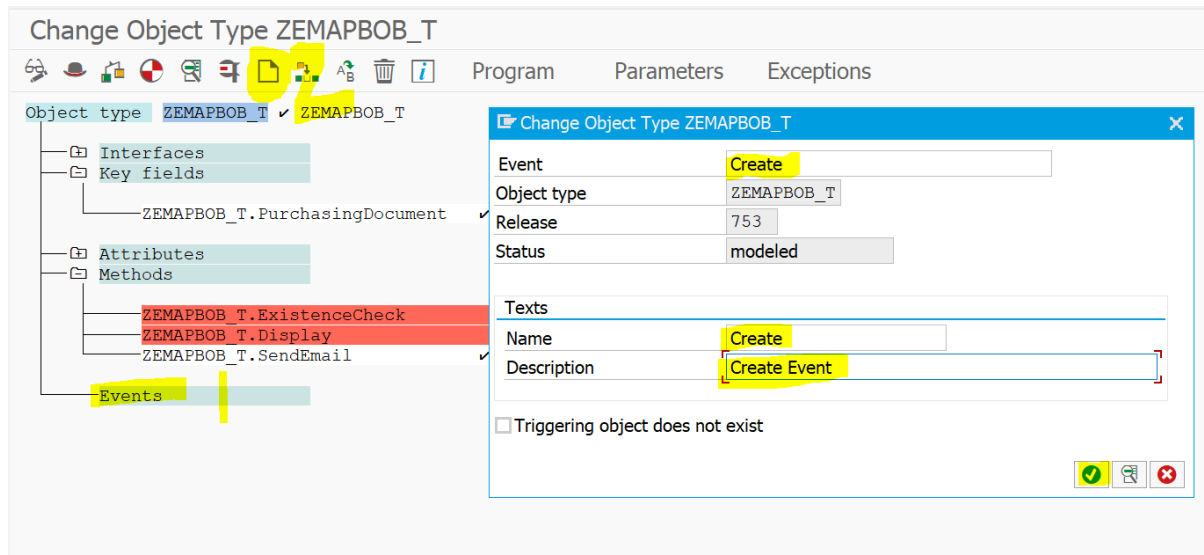
```
GW_CONT-LINE = '<br>'.
```

```
APPEND GW_CONT TO GT_CONT.
```

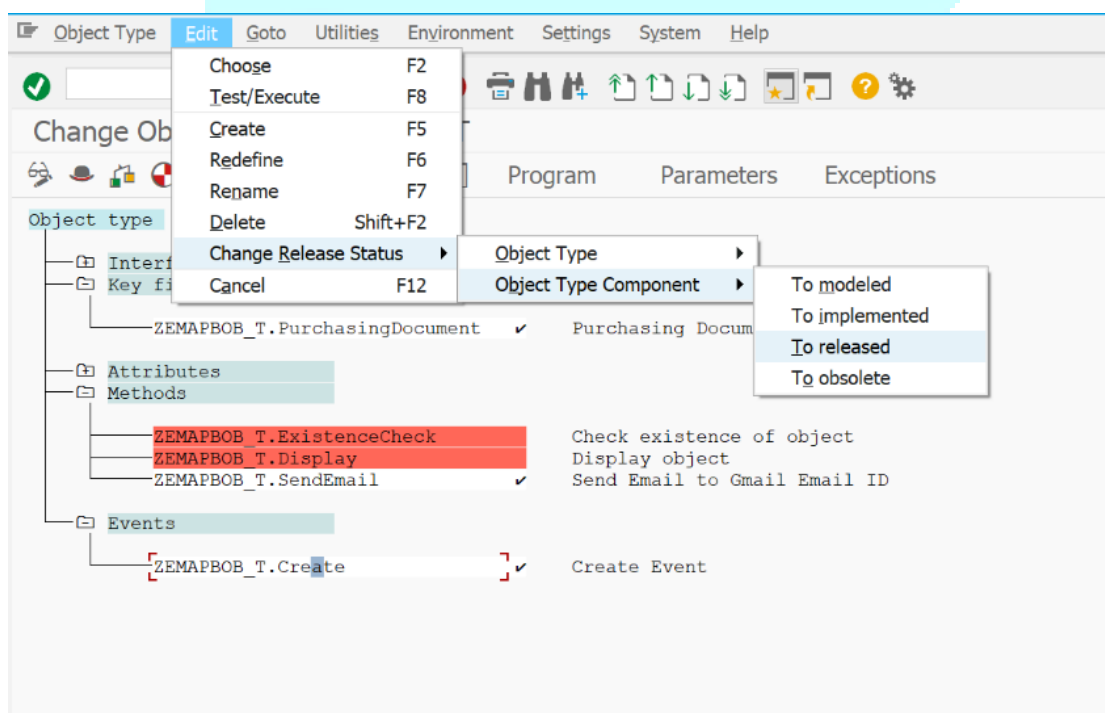
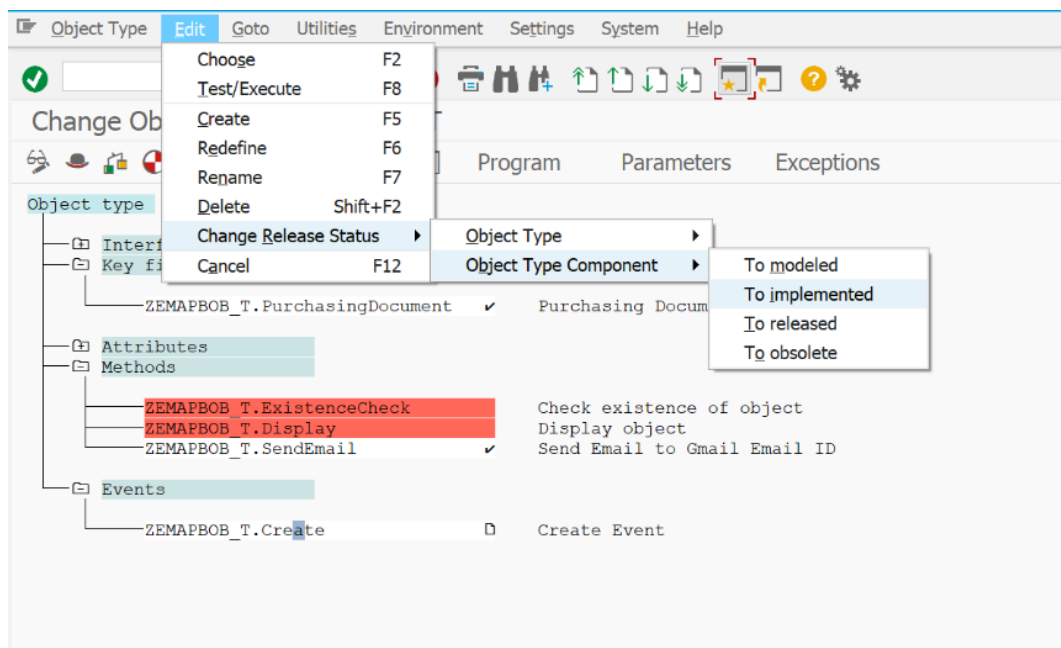
Note: After writing code in the BO Method must need to click on generate button to make it available.

Step 7:

Now we have to create event to use in Workflow.



Same as before, must need to implement and release it to make it available for use.



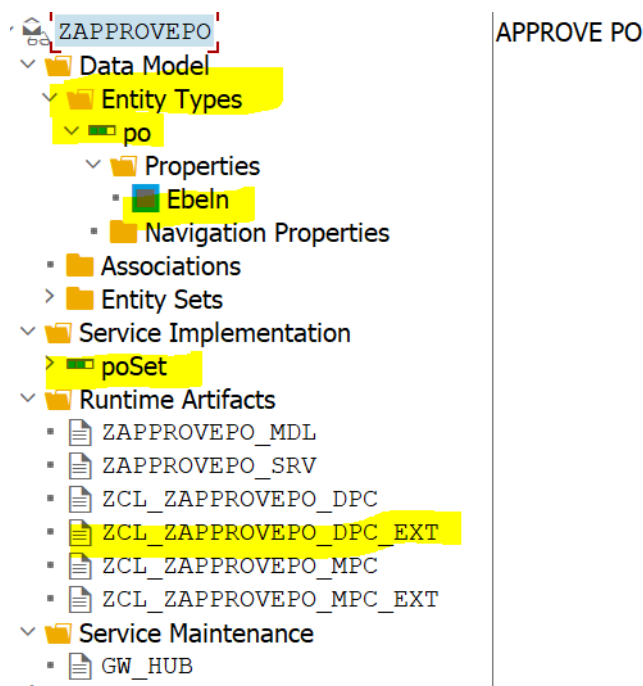
Step 8:

We need an API that will send a request of approve or reject case selected by approver from email.

In Our case, We Create ODATA and Implement Get Entity Method:

Go To Transaction Code SEGW and create a project with the following objects:

Note: I am not telling ODATA Creation in Detail here. Also change length of Ebeln to 25.



After Creating and Generating the ODATA Project we have to **redefine** the Get Entity Method.

Why we use get entity method?

As we have to send 2 parameters from Email to Back SAP.

1. Purchase Order Number
2. Approve or Reject Selection

Code for Get Entity Method:






```
DATA: WA_KEY_TAB LIKE LINE OF IT_KEY_TAB,  
      LV_EBELN  TYPE EBELN,  
      LV_VAR1   TYPE C LENGTH 10,  
      LV_VAR2   TYPE C LENGTH 1,  
      LV_VAR3   TYPE C LENGTH 12.  
  
DATA: WORKITEM_ID      TYPE SWR_STRUCT-WORKITEMID.  
  
READ TABLE IT_KEY_TAB INTO WA_KEY_TAB WITH KEY NAME = 'Ebeln'.  
  
SPLIT WA_KEY_TAB-VALUE AT SPACE INTO LV_VAR1 LV_VAR2 LV_VAR3.  
  
IF SY-SUBRC EQ 0.  
  
  IF LV_VAR2 EQ 'A'.  
  
    MOVE LV_VAR3 TO WORKITEM_ID.  
    CALL FUNCTION 'SAP_WAPI_FORWARD_WORKITEM'  
      EXPORTING  
        WORKITEM_ID = WORKITEM_ID.  
  
    COMMIT WORK AND WAIT.  
  
  ELSEIF LV_VAR2 EQ 'R'.  
  
    MOVE LV_VAR3 TO WORKITEM_ID.  
    CALL FUNCTION 'SAP_WAPI_WORKITEM_DELETE'  
      EXPORTING  
        WORKITEM_ID = WORKITEM_ID.  
  
    COMMIT WORK AND WAIT.  
  
  ENDIF.  
  
ENDIF.  
  
MOVE LV_VAR1 TO LV_EBELN.  
SELECT SINGLE * FROM EKKO INTO ER_ENTITY WHERE EBELN EQ LV_EBELN.
```

Step 9:

Create Workflow and Workflow Task:

Go to PFTC:

Task: Maintain



Task type

Workflow Template

Task

Name

Workflow Pattern

Name

Package

Applic. Component

Basic data

Description

Container

Triggering events

SAPphone

Name

Abbr.

Name

Release status

Work Item Text

Work item text

Workflow Definition

Workflow definition

Workflow Builder

ZPOAPPROVE

Approve PO from Email

Not defined

Approve PO

Link the event here that was created in Business Object.

Workflow Pattern: Display

Workflow Pattern

90000135

ZPOAPPROVE

Name

Approve PO from Email

Package

Z_ASAD

Applic. Component

Basic data

Description

Container

Triggering events

SAPphone

Standard events

	Binding...	Object Category	Object Type	Event	Name
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	BOR Object Type	ZEMAPBOB_T	CREATE	MAPBOB_T Create
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				



Step 10:

Create workflow.

Workflow Pattern: Display

Workflow Pattern 90000135 ZPOAPPROVE

Name Approve PO from Email

Package Z_ASAD Applic. Component

Basic data Description Container Triggering events SAPphone

Name

Abbr. ZPOAPPROVE

Name Approve PO from Email

Release status Not defined

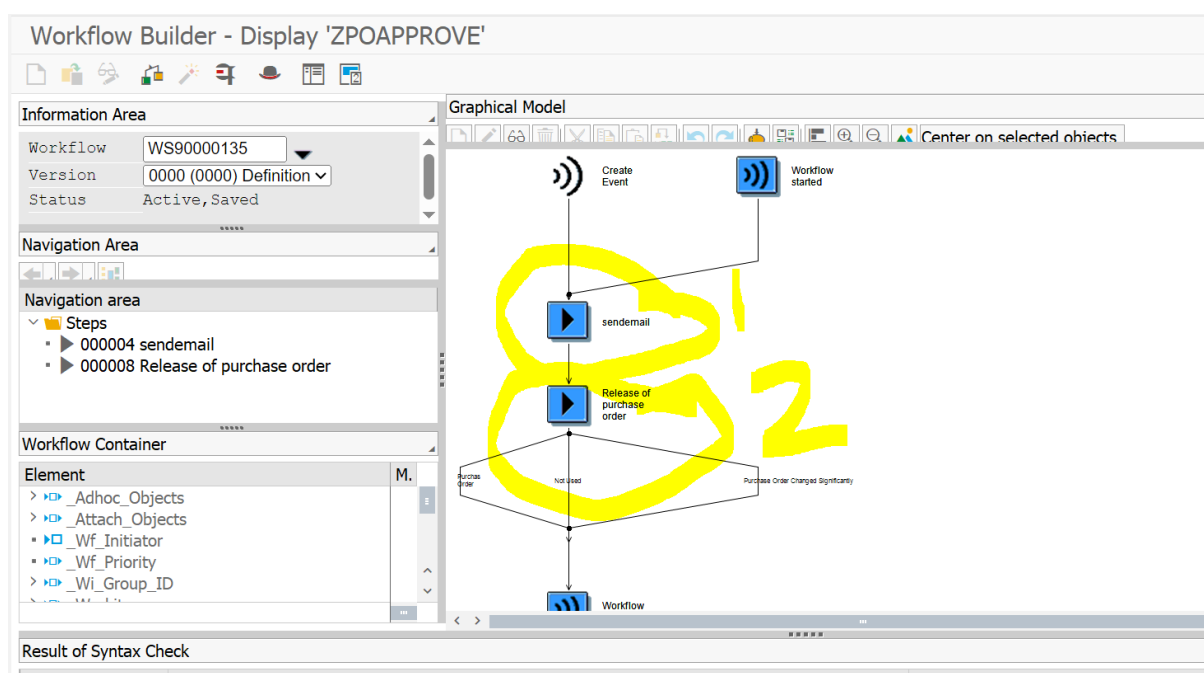
Work Item Text

Work item text PO Approve

Workflow Definition

☒ Workflow definition **Workflow Builder**

We have to create 2 activities only for simplicity and proper understanding.



Activity 1:

We have to create this Task in PFTC T-Code.

Workflow Builder - Display 'ZPOAPPROVE'

Information Area

Workflow: WS90000135
Version: 0000 (0000) Definition
Status: Active, Saved

Navigation Area

Navigation area

- Steps
 - 000004 sendmail
 - 000008 Release of purchase order

Workflow Container

Element

- Adhoc_Objects
- Attach_Objects
- Wf_Initiator
- Wf_Priority
- Wf_Group_ID

Activity: 000004 sendmail

Control

Task: TS90000200 sendmail

Step Name: sendmail

Binding (Exists)

Agents (not for workflows and background tasks)

Expression

Excluded

Task Properties

- Agent Assignment
- Task Complete
- Background Processing
- Confirm End of Processing

Task description - click here to go to task...

No text exists

Result of Syntax Check

Step Number	Message
-------------	---------

Standard Task: Display

Standard task: 90000200 sendmail

Name: sendmail

Package: Z_ASAD

Applic. Component

Basic data

Name

Abbr.: sendmail

Name: sendmail

Release status: Not defined

Work Item Text

Work item text: Send Email To Approve PO & _WI_OBJECT_ID.PURCHASINGDOCUMENT&

Object method

Object Category: BOR Object Type

Object Type: ZEMAPBOB_T ZEMAPBOB_T

Method: SENDEMAIL SendEmail

☒ Synchronous object method

☐ Object method with dialog

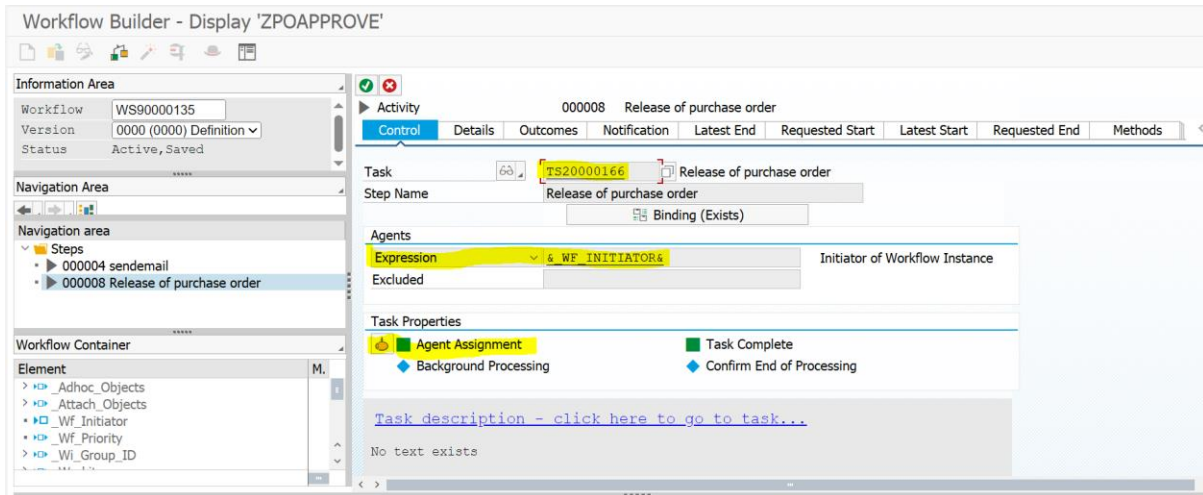
Execution

☒ Background processing

☐ Executable with SAPforms

Activity 2:

We have to use standard task here for sending notification in Inbox.



Step 11:

Let's Run the Workflow and Test it.

Workflow Builder - Display 'ZPOAPPROVE'

Information Area

Workflow: WS90000135
Version: 0000 (0000) Definition
Status: Active, Saved

Navigation Area

Navigation area

- Steps
 - 000004 sendemail

Graphical Model

Create Event

Workflow started

sendemail

Test Workflow

Refresh Organizational Environment Graphical workflow log Workflow Log Bu

Workflow: WS90000135 ZPOAPPROVE
Type: Workflow Template
Name: Approve PO from Email
Validity: 01.01.1900 To 31.12.9999

Input Data Ad Hoc Agents DeadlineData Outcome

Test Data Load Save Initial Data

Expression	M. Values
Wf_Initiator	USASAD_EXD
Wf_Priority	5
Wi_Group_ID	< No Instance >
ZEMAPBOB_T	ZEMAPBOB_T:4500000085

Object Type Key

ZEMAPBOB_T 4500000085

Test Workflow

Refresh Organizational Environment

Graphical workflow log

Workflow Log

Business Workplac

Workflow

WS90000135

ZPOAPPROVE

Type

Workflow Template

Name

Approve PO from Email

Validity

01.01.1900

To

31.12.9999

Input Data

Ad Hoc Agents

DeadlineData

Outcome

Test Data

Load

Save

Initial Data

Expression

M. Values

_Wf_Initiator

_Wf_Priority

_Wi_Group_ID

ZEMAPBOB_T

USASAD_EXD

5

< No Instance >

ZEMAPBOB_T:4500000085

Object Type

Key

ZEMAPBOB_T

4500000085

Task started under work item ID 260275 (current state: In Process)

SAP

We can see our inbox the notification arrived to approve Purchase Order.

Business Workplace of ASAD_EXD

New message Find folder Find document Appointment calendar Distribution lists

Workplace: ASAD_EXD

- Inbox
 - Unread Documents 10,373
 - Documents 10,397
 - Workflow 33
 - Overdue entries 0
 - Deadline Messages 0
 - Entries with Errors 10
- Outbox

Inbox

Class	Title	Date received	Author
▶	Please release purchase order	16.10.2024	
▶	TESTING	05.10.2024	
▶	TESTING	05.10.2024	
▶	Please approve Budget Request 0000090083	05.10.2024	

Also, see in Transaction **SOST** the Email arrived to send.

Refresh

Send Requests (Waiting, Sent and Errors)

Status	Send Method	Doc. Title	Sender	Recipient
▲	By E-Mail	Release PO Notification	saptechall@gmail.com	saptechall@gmail.com
▲	By E-Mail	Release PO Notification	saptechall@gmail.com	m.uzairkoreja@gmail.com

Select Email line and Click on Display to see Email Body.

We can see Approve and Reject buttons.

Display Document: Release PO Notification

Doc. contents Attributes Recipient list

Release PO Notification

Created SAP WorkFlow

Dear Sir,

Purchase number 4500000085 is waiting for your decision, kindly login the following applications to approve the document:

- [Approve](#)
- [Reject](#)

Regards

This e-mail was generated automatically from SAP; please do not reply.

When we click on Approve or Reject Button.

The notification will be removed from inbox.

