**LIT ASSIGNMENT - DAY 14**

Batch: SAP ABAP ON HANA Date:19th Oct 2022

EMP ID:46255236

CDS-Assignment 1 – Create a Simple CDS view with Date Functions in ABAP CDS Views

**Tables involved** – VBAK

@AbapCatalog.sqlViewName: 'Z36\_SQL\_VIEW'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'CDS VIEW WITH DATE FUNCTIONS'

**define** **view** Z5236\_CDS\_VIEW **as** **select** **from** vbak **{**

vbeln **as** Sales\_Document**,**

auart **as** Sales\_Document\_Type**,**

audat **as** Document\_Date**,**

vdatu **as** Requested\_delivery\_date**,**

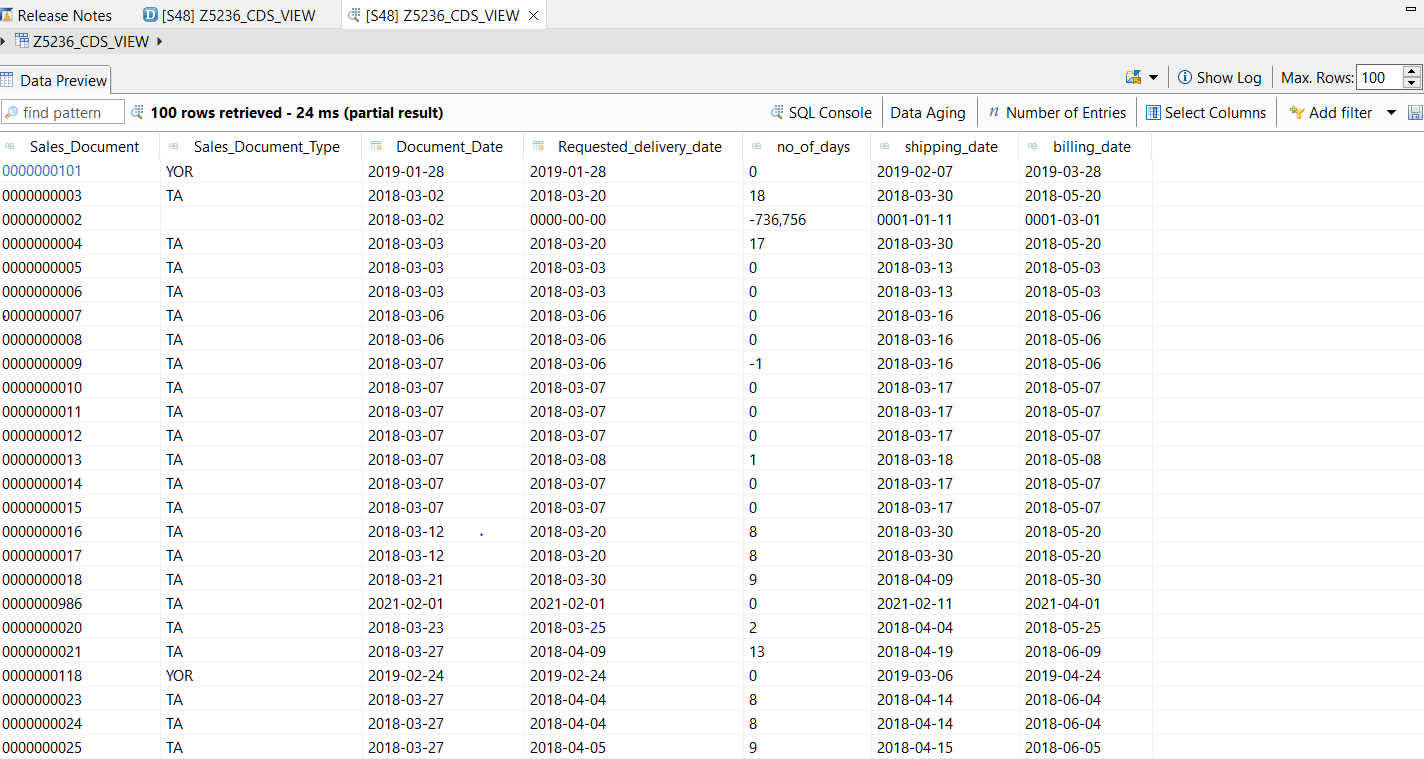
DATS\_DAYS\_BETWEEN**(**audat**,** vdatu**)** **as** no\_of\_days**,**

DATS\_ADD\_DAYS**(**vdatu**,** 10**,** 'NULL'**)** **as** shipping\_date**,**

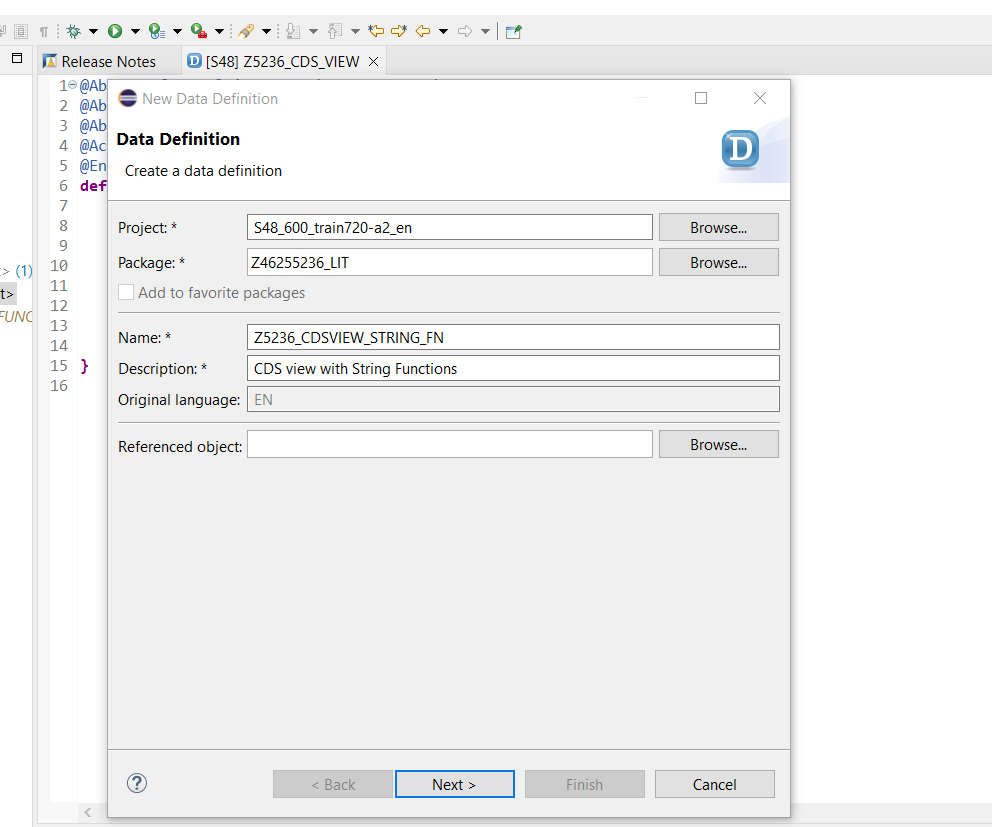
DATS\_ADD\_MONTHS**(**vdatu**,** 2**,** 'NULL'**)** **as** billing\_date

**}**

**OUTPUT:**



CDS-Assignment 2 – Create a Simple CDS view with String Functions in ABAP CDS Views



@AbapCatalog.sqlViewName: 'Z36\_STR\_FN\_VIEW'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'CDS view with String Functions'

**define** **view** Z5236\_CDSVIEW\_STRING\_FN **as** **select** **from** kna1 **{**

kunnr **as** Customer\_Number**,**

land1 **as** Country\_Key**,**

concat**(**kna1**.**name1 **,** kna1**.**name2**)** **as** concat\_fields**,**

concat\_with\_space**(**stras**,**ort01**,**2**)** **as** con\_space **,**

left**(**name1**,**2**)** **as** no\_of\_char\_left**,**

right**(**kna1**.**name1**,**3**)** **as** no\_of\_char\_right**,**

length**(**telf1**)** **as** no\_of\_char**,**

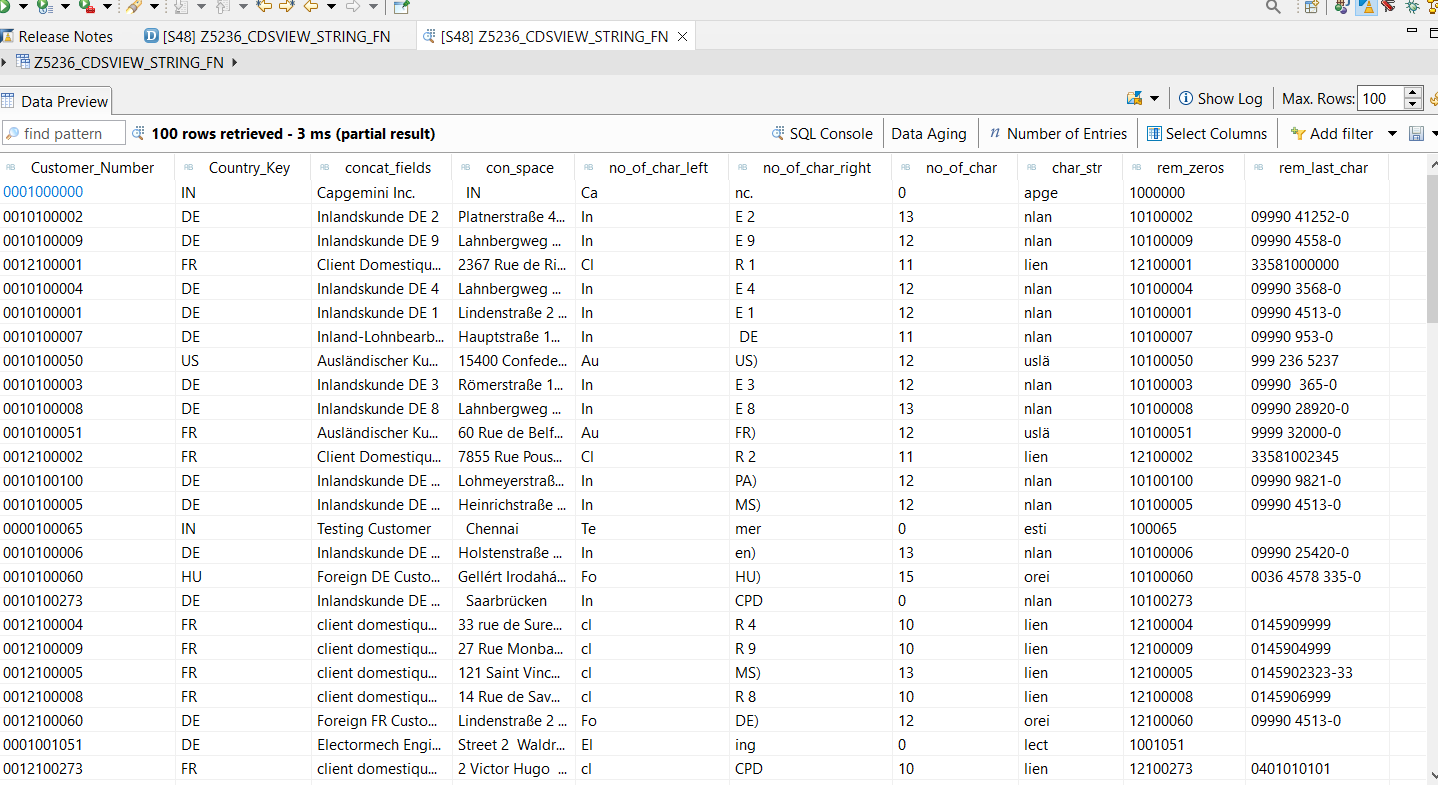
substring**(**name1**,**2**,**4**)** **as** char\_str**,**

ltrim**(**kunnr**,** '0'**)** **as** rem\_zeros**,**

rtrim**(**telf1**,**'4'**)** **as** rem\_last\_char

**}**

OUTPUT:

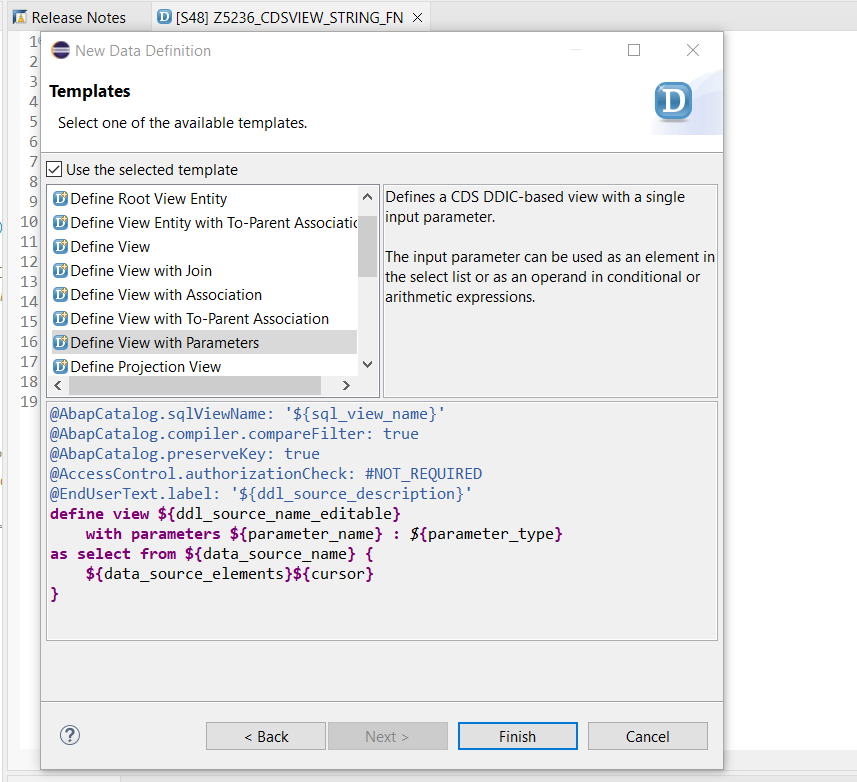


CDS-Assignment 3 – Create a Simple CDS view with Parameters

**Scenario** –

* Create a CDS view with GL currency (PSWSL) as parameter and get the entries form BSEG

**Tables involved** – BSEG



@AbapCatalog.sqlViewName: 'Z36\_SQL\_VIEW1'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'CDS view with Parameters'

**define** **view** Z5236\_CDSVIEW\_PARA

**with** **parameters** in\_curr **:** *abap***.***cuky***(**5**)**

**as** **select** **from** bseg**{**

bukrs**,**

belnr**,**

gjahr**,**

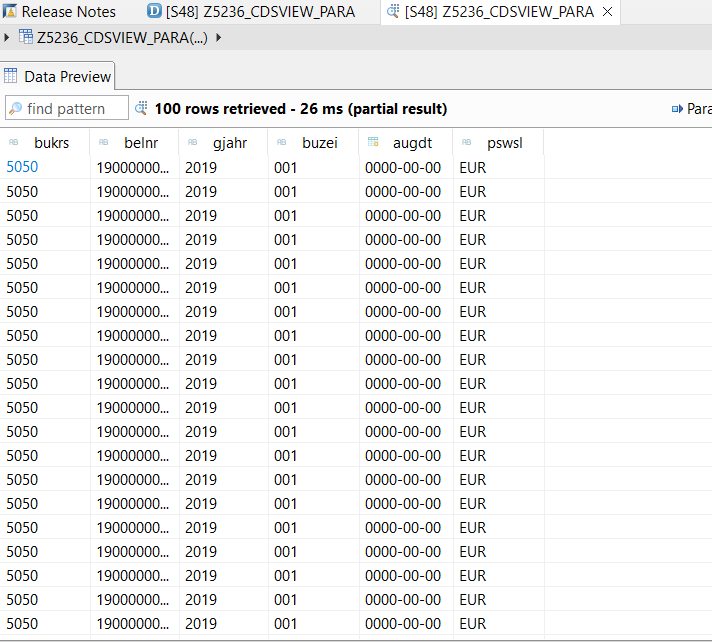
buzei**,**

augdt**,**

pswsl

**}where** pswsl **=** **$parameters.**in\_curr

**output:**



AMDP-Assignment 1 – Procedures - create a basic procedure to fetch data with scalar variable

**Scenario** –

* Print Sales Order number, Material Number, Quantity from VBAP table using scaler variable

**Tables involved** – VBAP

CLASS z5236\_amdp\_scal DEFINITION

PUBLIC

FINAL

CREATE PUBLIC .

PUBLIC SECTION.

INTERFACES :IF\_AMDP\_MARKER\_HDB.

METHODS: get\_vbap\_data

EXPORTING VALUE(lt\_VBAP) TYPE Z5236TTSTU\_VBAP.

PROTECTED SECTION.

PRIVATE SECTION.

ENDCLASS.

CLASS z5236\_amdp\_scal IMPLEMENTATION.

METHOD get\_vbap\_data BY DATABASE PROCEDURE FOR HDB LANGUAGE

SQLSCRIPT OPTIONS READ-ONLY

USING VBAP.

lt\_vbap = select vbeln,matnr,ZMENG from vbap;

endmethod.

ENDCLASS.

**REPORT:**

REPORT z5236\_amdp.

z5236\_amdp\_scal=>get\_vbap\_data( IMPORTING lt\_vbap = DATA(ex\_data) ).

TRY.

CALL METHOD cl\_salv\_table=>factory

IMPORTING

r\_salv\_table = DATA(lr\_salv)

changing

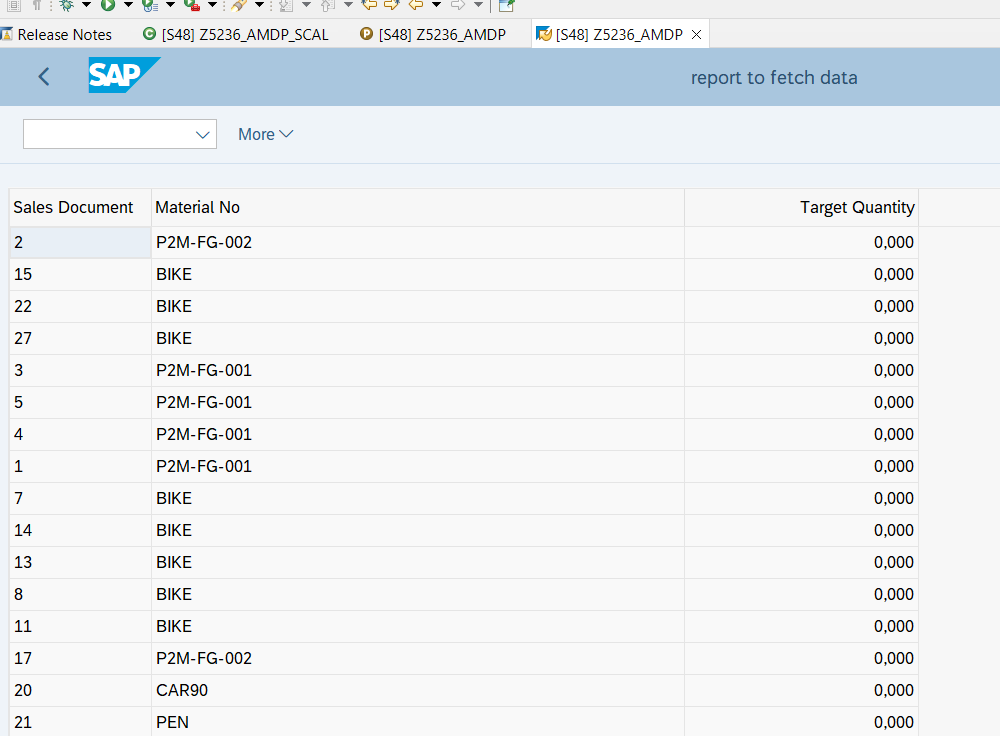
t\_table = ex\_data.

CATCH cx\_salv\_msg.

ENDTRY.

lr\_salv->display( ).

**OUTPUT:**

****

AMDP-Assignment 2 – Procedures - create a basic procedure with Table

**Scenario** –

* Select VBELN, VKORG, MATNR, MENGE from VBAK and VBAP and display the data using table based on material (MATNR).
* MATNR will be input field
* Do inner join

**Tables involved** – VBAK, VBAP

CLASS z5236\_amdp\_join DEFINITION

PUBLIC

FINAL

CREATE PUBLIC .

PUBLIC SECTION.

INTERFACES : IF\_AMDP\_MARKER\_HDB.

CLASS-METHODS:GET\_DATA

IMPORTING VALUE(IM\_MATNR) TYPE VBAP-MATNR

EXPORTING VALUE(ET\_RES) TYPE Z5236TTSRU\_IN.

PROTECTED SECTION.

PRIVATE SECTION.

ENDCLASS.

CLASS z5236\_amdp\_join IMPLEMENTATION.

METHOD get\_data BY DATABASE PROCEDURE FOR HDB LANGUAGE

SQLSCRIPT OPTIONS READ-ONLY

USING VBAP VBAK.

ET\_RES = SELECT A.VBELN,

A.VKORG,

B.MATNR,

B.ZMENG

FROM VBAK AS A

INNER JOIN VBAP AS B

ON A.VBELN = B.VBELN;

ENDMETHOD.

ENDCLASS.

REPORT:

REPORT z5236\_amdp\_join.

PARAMETERS: S\_MATNR TYPE VBAP-MATNR.

z5236\_amdp\_join=>get\_data( EXPORTING IM\_MATNR = S\_MATNR

IMPORTING ET\_RES = DATA(EX\_RES) ).

TRY.

CALL METHOD cl\_salv\_table=>factory

IMPORTING

r\_salv\_table = DATA(lr\_salv)

changing

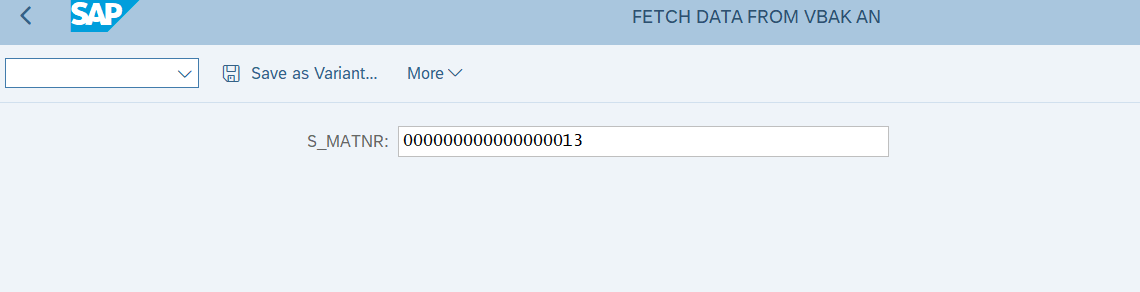
t\_table = ex\_res.

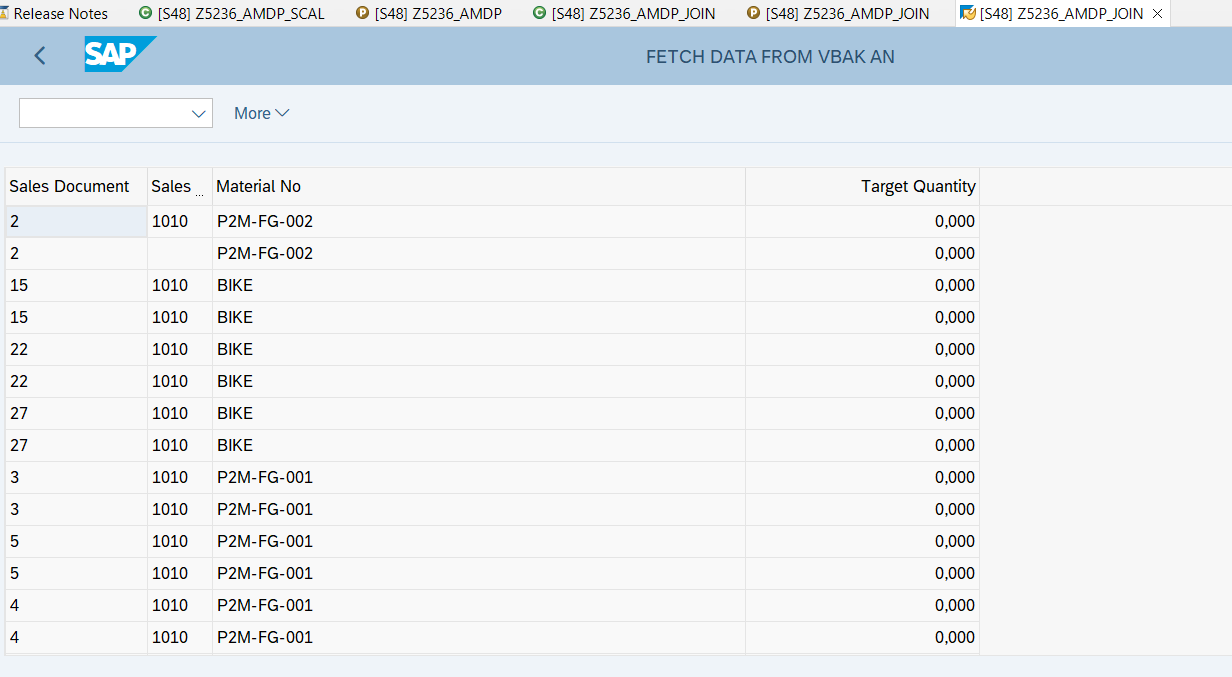
CATCH cx\_salv\_msg.

ENDTRY.

lr\_salv->display( ).

OUTPUT:

****

****