**LIT ASSIGNMENT**

**Day - 14**

**Emp Id: 46255265**

**Name: Pathe Kavya Malathi**

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

**Scenario** –

* Create a CDS view with 7 fields from VBAK using date functions

1. VBELN (Sales Document),
2. AUART (Sales Document Type),
3. AUDAT (Document Date),
4. VDATU (Requested delivery date)
5. Days between Document Date & Requested delivery date as Processing days
6. Delivery date + 10 days as shipping date
7. Delivery Date + 2 months as billing date

**Tables involved** – VBAK

Cd's View Code:

@AbapCatalog.sqlViewName: 'Z265\_SQLDAY14Q1'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'CDs view day14 q1 vbak table'

**define** **view** z265\_cds\_date\_day14\_q1 **as** **select** **from** vbak **{**

vbeln**,**

auart**,**

audat**,**

vdatu**,**

dats\_days\_between**(**audat**,**vdatu**)** **as** Processing\_days**,**

dats\_add\_Days**(**vdatu**,**10**,**'FAIL'**)** **as** Shipping\_date**,**

dats\_add\_months**(**vdatu**,**2**,**'FAIL'**)** **as** Billing\_date

**}**

**output:**

Table

Description automatically generated

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

**Scenario** –

* Create a CDS view with 10 fields from KNA1 using string functions

1. KUNNR (Customer Number),
2. LAND1 (Country Key),
3. Concatenate NAME1 & NAME2
4. Concatenate STRAS, ORT01 & REGIO & PSTLZ with space in between.
5. Get first 2 characters of NAME1
6. Get last 3 characters of NAME2
7. No of characters of TELF1
8. NAME1+2(4). 4 characters starting from 2nd character.
9. Remove the leading zeros of KUNNR
10. Remove last 4 characters of TELF1.

**Tables involved** – KNA1

cds view code

@AbapCatalog.sqlViewName: 'Z265SQLDATE'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'cds view day14 q2 kna1 table'

**define** **view** Z265\_cds\_date\_day14\_q2 **as** **select** **from** kna1 **{**

kunnr**,**

land1**,**

concat**(**name1**,**name2**)** **as** name**,**

concat\_with\_space**(**concat\_with\_space**(**stras**,**ort01**,**1**),**concat\_with\_space**(**regio**,**pstlz**,**1**),**1**)** **as** address**,**

left**(**name1**,**2**)** **as** l2\_name**,**

right**(**name2**,**3**)** **as** r3\_name**,**

length**(**telf1**)** **as** length\_telf1**,**

substring**(**name1**,**2**,**4**)** **as** sub\_name**,**

ltrim**(**kunnr**,**'0'**)** **as** remove\_kunnr**,**

replace**(**telf1**,**right**(**telf1**,**4**),**''**)** **as** replace\_telf1

**}**

**output:**

Table

Description automatically generated

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

CDS view Code:

@AbapCatalog.sqlViewName: 'Z265SQLPARAV'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'parameter view day14 q3'

**define** **view** z265\_cds\_para\_day14\_q3

**with** **parameters** gl\_currency **:** *pswsl*

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

bukrs**,**

belnr**,**

gjahr**,**

buzei**,**

pswsl

**}** **where** pswsl **=** **$parameters.**gl\_currency

output:

Table

Description automatically generated

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

Amdp Class :

class Z265\_AMDP\_DAY14Q4 definition

public

final

create public .

public section.

TYPES:BEGIN OF TY\_vbap,

vbeln TYPE vbeln\_va,

matnr TYPE matnr,

zmeng TYPE dzmeng,

END OF TY\_vbap.

TYPES: it\_vbap TYPE STANDARD TABLE OF ty\_vbap WITH EMPTY KEY.

INTERFACES IF\_AMDP\_MARKER\_HDB.

class-METHODs get\_details

IMPORTING

VALUE(iv\_vbeln) TYPE vbeln

EXPORTING

VALUE(et\_order) TYPE it\_vbap.

protected section.

private section.

ENDCLASS.

CLASS Z265\_AMDP\_DAY14Q4 IMPLEMENTATION.

METHOD get\_details BY DATABASE PROCEDURE

FOR HDB

LANGUAGE SQLSCRIPT

USING vbap.

et\_order = SELECT vbeln,

matnr,

zmeng

From vbap where vbeln = iv\_vbeln;

endmethod.

ENDCLASS.

**Execution Code:**

\*&---------------------------------------------------------------------\*

\*& Report z265\_amdp\_day14q4

\*&---------------------------------------------------------------------\*

\* Description: amdp program day14 q4 \*

\* \*

\* Author: Pathe kavya Malathi \*

\* \*

\* Create date: 20-10-2022. \*

\* \*

\*&---------------------------------------------------------------------\*

\* Modification log: \*

\* -------------------------------------------------------------------- \*

\* Date User name ID Change Request Case ref. \*

\* Description \*

\*&---------------------------------------------------------------------\*

REPORT z265\_amdp\_day14q4.

PARAMETERS : p\_vbeln type vbeln.

z265\_amdp\_day14q4=>GET\_DETAILS(

exporting

IV\_VBELN = p\_vbeln

importing

ET\_ORDER = data(ta\_vbap)

).

cl\_demo\_output=>display\_data(

EXPORTING

value = ta\_vbap

name = 'Sales Details'

).

**output:**

**Graphical user interface, application, Word

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

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.

Code:

class Z265\_AMDP\_DAY14Q5 definition

public

final

create public .

public section.

TYPES:BEGIN OF TY\_SALES,

VBELN TYPE VBELN\_VA,

VKORG TYPE VKORG,

MATNR TYPE MATNR,

MATKL TYPE MATKL,

END OF TY\_SALES.

TYPES IT\_SALES TYPE TABLE OF TY\_SALES.

INTERFACES IF\_AMDP\_MARKER\_HDB.

CLASS-METHODS

GET\_SALES

IMPORTING VALUE(IM\_MATNR) TYPE MATNR

EXPORTING VALUE(EX\_SALES) TYPE IT\_SALES.

protected section.

private section.

ENDCLASS.

CLASS Z265\_AMDP\_DAY14Q5 IMPLEMENTATION.

METHOD GET\_SALES BY DATABASE PROCEDURE

FOR HDB

LANGUAGE SQLSCRIPT

USING VBAK VBAP.

EX\_SALES = SELECT V1.VBELN,

V1.VKORG,

V2.MATNR,

V2.MATKL

FROM VBAK AS V1 INNER JOIN VBAP AS V2

ON V1.VBELN = V2.VBELN

WHERE MATNR = IM\_MATNR;

endmethod.

ENDCLASS.

**Program Code:**

\*&---------------------------------------------------------------------\*

\*& Report z265\_amdp\_day14q5

\*&---------------------------------------------------------------------\*

\* Description: amdp program day14q5 \*

\* \*

\* Author:Pathe Kavya Malathi \*

\* \*

\* Create date: 20-10-2022 \*

\* \*

\*&---------------------------------------------------------------------\*

\* Modification log: \*

\* -------------------------------------------------------------------- \*

\* Date User name ID Change Request Case ref. \*

\* Description \*

\*&---------------------------------------------------------------------\*

REPORT z265\_amdp\_day14q5.

Parameters : p\_matnr type matnr.

z265\_amdp\_day14q5=>GET\_SALES(

exporting

IM\_MATNR = p\_matnr

importing

EX\_SALES = DATA(RS\_VBAK)

).

CL\_DEMO\_OUTPUT=>display\_data(

EXPORTING

value = RS\_VBAK

name = 'Sales Order Details using Join'

).

**output:**

**Graphical user interface, application

Description automatically generated**

**Table

Description automatically generated**