1. Create 3 Data elements

ZATS\_XX\_DTE\_ID CHAR32

ZATS\_XX\_DTE\_BPTYP DOMAIN CHAR2 01 CUSTOMER 02 SUPPLIER

ZATS\_XX\_DTE\_REGION CHAR4

2. Create a reusable structure

@EndUserText.label : 'Administration Data'

@AbapCatalog.enhancement.category : #NOT\_EXTENSIBLE

define structure zats\_xx\_str\_admin\_data {

created\_by : abap.char(16);

created\_on : timestamp;

changed\_by : abap.char(16);

changed\_on : timestamp;

}

3. Create DB Tables as below

@EndUserText.label : 'Region master data'

@AbapCatalog.enhancement.category : #NOT\_EXTENSIBLE

@AbapCatalog.tableCategory : #TRANSPARENT

@AbapCatalog.deliveryClass : #A

@AbapCatalog.dataMaintenance : #RESTRICTED

define table zats\_xx\_region {

key client : abap.clnt not null;

key region : zats\_xx\_dte\_region not null;

regionname : abap.char(16);

}

@EndUserText.label : 'Business partner Master data'

@AbapCatalog.enhancement.category : #NOT\_EXTENSIBLE

@AbapCatalog.tableCategory : #TRANSPARENT

@AbapCatalog.deliveryClass : #A

@AbapCatalog.dataMaintenance : #RESTRICTED

define table zats\_xx\_bpa {

key client : abap.clnt not null;

key bp\_id : zats\_xx\_dte\_id not null;

bp\_role : zats\_xx\_dte\_bptype;

@EndUserText.label : 'Company Name'

company\_name : abap.string(256);

@EndUserText.label : 'Street'

street : abap.string(256);

@EndUserText.label : 'Country'

country : abap.string(256);

region : zats\_xx\_dte\_region;

@EndUserText.label : 'City'

city : abap.char(100);

}

@EndUserText.label : 'Products master data'

@AbapCatalog.enhancement.category : #NOT\_EXTENSIBLE

@AbapCatalog.tableCategory : #TRANSPARENT

@AbapCatalog.deliveryClass : #A

@AbapCatalog.dataMaintenance : #RESTRICTED

define table zats\_xx\_product {

key client : abap.clnt not null;

key product\_id : zats\_xx\_dte\_id not null;

name : abap.string(256);

category : abap.char(40);

@Semantics.amount.currencyCode : 'zats\_xx\_product.currency'

price : abap.curr(10,2);

currency : abap.cuky;

discount : abap.int4;

}

@EndUserText.label : 'Sales order header transaction'

@AbapCatalog.enhancement.category : #NOT\_EXTENSIBLE

@AbapCatalog.tableCategory : #TRANSPARENT

@AbapCatalog.deliveryClass : #A

@AbapCatalog.dataMaintenance : #RESTRICTED

define table zats\_xx\_so\_hdr {

key client : abap.clnt not null;

key order\_id : zats\_xx\_dte\_id not null;

order\_no : int4;

@AbapCatalog.foreignKey.screenCheck : true

buyer : zats\_xx\_dte\_id not null

with foreign key [0..\*,1] zats\_xx\_bpa

where bp\_id = zats\_xx\_so\_hdr.buyer;

@Semantics.amount.currencyCode : 'zats\_xx\_so\_hdr.currency\_code'

gross\_amount : abap.curr(10,2);

currency\_code : abap.cuky;

include zats\_xx\_str\_admin\_data;

}

@EndUserText.label : 'Sales order header transaction'

@AbapCatalog.enhancement.category : #NOT\_EXTENSIBLE

@AbapCatalog.tableCategory : #TRANSPARENT

@AbapCatalog.deliveryClass : #A

@AbapCatalog.dataMaintenance : #RESTRICTED

define table zats\_xx\_so\_item {

key client : abap.clnt not null;

key item\_id : zats\_xx\_dte\_id not null;

order\_id : zats\_xx\_dte\_id not null;

@AbapCatalog.foreignKey.screenCheck : true

product : zats\_xx\_dte\_id not null

with foreign key [0..\*,1] zats\_xx\_product

where product\_id = zats\_xx\_so\_item.product;

@Semantics.amount.currencyCode : 'zats\_xx\_so\_item.currency'

amount : abap.curr(10,2);

currency : abap.cuky;

@Semantics.quantity.unitOfMeasure : 'zats\_xx\_so\_item.uom'

qty : abap.quan(5,2);

uom : abap.unit(3);

include zats\_xx\_str\_admin\_data;

}

4. Create data builder class to upload sample data

CLASS zats\_xx\_data\_builder DEFINITION

PUBLIC

FINAL

CREATE PUBLIC .

PUBLIC SECTION.

INTERFACES if\_oo\_adt\_classrun .

PROTECTED SECTION.

PRIVATE SECTION.

METHODS fill\_transaction\_data.

METHODS fill\_master\_data.

METHODS flush.

ENDCLASS.

CLASS zats\_xx\_data\_builder IMPLEMENTATION.

METHOD if\_oo\_adt\_classrun~main.

flush( ).

fill\_master\_data( ).

fill\_transaction\_data( ).

out->write(

EXPORTING

data = 'processing is completed successfully!'

\* name =

\* RECEIVING

\* output =

).

ENDMETHOD.

METHOD fill\_master\_data.

data : lt\_bp type table of zats\_xx\_bpa,

lt\_prod type table of zats\_xx\_product.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'TACUM'

street = 'Victoria Street'

city = 'Kolkatta'

country = 'IN'

region = 'APJ'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'SAP'

street = 'Rosvelt Street Road'

city = 'Walldorf'

country = 'DE'

region = 'EMEA'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'Asia High tech'

street = '1-7-2 Otemachi'

city = 'Tokyo'

country = 'JP'

region = 'APJ'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'AVANTEL'

street = 'Bosque de Duraznos'

city = 'Maxico'

country = 'MX'

region = 'NA'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'Pear Computing Services'

street = 'Dunwoody Xing'

city = 'Atlanta, Georgia'

country = 'US'

region = 'NA'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'PicoBit'

street = 'Fith Avenue'

city = 'New York City'

country = 'US'

region = 'NA'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'TACUM'

street = 'Victoria Street'

city = 'Kolkatta'

country = 'IN'

region = 'APJ'

)

to lt\_bp.

append value #(

bp\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

bp\_role = '01'

company\_name = 'Indian IT Trading Company'

street = 'Nariman Point'

city = 'Mumbai'

country = 'IN'

region = 'APJ'

)

to lt\_bp.

append value #(

product\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

name = 'Blaster Extreme'

category = 'Speakers'

price = 1500

currency = 'INR'

discount = 3

)

to lt\_prod.

append value #(

product\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

name = 'Sound Booster'

category = 'Speakers'

price = 2500

currency = 'INR'

discount = 2

)

to lt\_prod.

append value #(

product\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

name = 'Smart Office'

category = 'Software'

price = 1540

currency = 'INR'

discount = 32

)

to lt\_prod.

append value #(

product\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

name = 'Smart Design'

category = 'Software'

price = 2400

currency = 'INR'

discount = 12

)

to lt\_prod.

append value #(

product\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

name = 'Transcend Carry pocket'

category = 'PCs'

price = 14000

currency = 'INR'

discount = 7

)

to lt\_prod.

append value #(

product\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

name = 'Gaming Monster Pro'

category = 'PCs'

price = 15500

currency = 'INR'

discount = 8

)

to lt\_prod.

insert zats\_xx\_bpa from table @lt\_bp.

insert zats\_xx\_product from table @lt\_prod.

ENDMETHOD.

METHOD fill\_transaction\_data.

data : o\_rand type REF TO cl\_abap\_random\_int,

n type i,

seed type i,

lv\_date type timestamp,

lv\_ord\_id type zats\_xx\_dte\_id,

lt\_so type table of zats\_xx\_so\_hdr,

lt\_so\_i type table of zats\_xx\_so\_item.

seed = cl\_abap\_random=>seed( ).

cl\_abap\_random\_int=>create(

EXPORTING

seed = seed

min = 1

max = 7

RECEIVING

prng = o\_rand

).

get time stamp FIELD lv\_date.

select \* from zats\_xx\_bpa into table @data(lt\_bpa).

select \* from zats\_xx\_product into table @data(lt\_prod).

do 50 times.

lv\_ord\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( ).

n = o\_rand->get\_next( ).

read table lt\_bpa into data(ls\_bp) index n.

append value #(

order\_id = lv\_ord\_id

order\_no = sy-index

buyer = ls\_bp-bp\_id

gross\_amount = 10 \* n

currency\_code = 'EUR'

created\_by = sy-uname

created\_on = lv\_date

changed\_by = sy-uname

changed\_on = lv\_date

) to lt\_so.

do 2 times.

read table lt\_prod into data(ls\_prod) index n.

append value #(

item\_id = cl\_uuid\_factory=>create\_system\_uuid( )->create\_uuid\_c32( )

order\_id = lv\_ord\_id

product = ls\_prod-product\_id

qty = n

uom = 'EA'

amount = n \* ls\_prod-price

currency = ls\_prod-currency

) to lt\_so\_i.

enddo.

enddo.

insert zats\_xx\_so\_hdr from table @lt\_so.

insert zats\_xx\_so\_item from table @lt\_so\_i.

ENDMETHOD.

METHOD flush.

delete from : zats\_xx\_bpa, zats\_xx\_product, zats\_xx\_so\_hdr, zats\_xx\_so\_item.

ENDMETHOD.

ENDCLASS.