**OOPS ABAP Concepts**

1. MVC(Model View Controller) is nothing but Isolate the ***business logic*** from the ***user interface*** to provide the***flexibility*** to change business logic independently.

The advantage of using this approach is to **reuse the existing logic** in multiple similar requirements.   Let’s say we have a sales order report to prepare.  We can design our **Model** (business logic classes) in such a way that the **same Class** (model) can be used for creating the different views like report, smart forms. This approach gives us the flexibility to reuse that logic as it is part of the Model class.

Example program:-

*DATA : lv\_vbeln TYPE vbap-vbeln  .*

*\* Create selection screen*

*SELECTION-SCREEN BEGIN OF BLOCK b1 WITH FRAME TITLE text-000.*

*PARAMETERS : p\_werks TYPE vbap-werks OBLIGATORY.*

*SELECT-OPTIONS : s\_vbeln1 FOR lv\_vbeln .*

*SELECTION-SCREEN END OF BLOCK b1 .*

*\* Create selection screen class*

*CLASS cl\_sel DEFINITION FINAL .*

*PUBLIC SECTION .*

*TYPES : t\_vbeln  TYPE RANGE OF vbeln .*

*DATA : s\_vbeln TYPE t\_vbeln .*

*DATA : s\_werks TYPE werks\_ext  .*

*METHODS : get\_screen IMPORTING lp\_werks TYPE werks\_ext*

*ls\_vbeln TYPE t\_vbeln .*

*ENDCLASS .*

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

*\*&       CLASS (IMPLEMENTATION)  SEL*

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

*\*        Get Selection Screen Information*

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

*CLASS cl\_sel IMPLEMENTATION.*

*METHOD get\_screen .*

*me->s\_werks = lp\_werks.*

*me->s\_vbeln = ls\_vbeln[] .*

*ENDMETHOD .*

*ENDCLASS.               "SEL*

*\* Create DATA MODEL to  fetch  recordes*

*CLASS cl\_fetch DEFINITION  .*

*PUBLIC SECTION .*

*DATA : sel\_obj TYPE REF TO cl\_sel .*

*DATA : it\_vbap TYPE STANDARD TABLE OF vbap .*

*METHODS constructor IMPORTING ref\_sel TYPE REF TO cl\_sel .*

*METHODS : fetch\_data .*

*ENDCLASS .*

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

*\*&       CLASS (IMPLEMENTATION)  FETCH*

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

*\*        Fetch Sales Info*

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

*CLASS cl\_fetch IMPLEMENTATION.*

*METHOD constructor.*

*me->sel\_obj = ref\_sel .*

*ENDMETHOD .*

*METHOD fetch\_data .*

*SELECT \* FROM vbap INTO TABLE me->it\_vbap UP TO 10 ROWS WHERE vbeln*

*IN me->sel\_obj->s\_vbeln AND werks EQ me->sel\_obj->s\_werks .*

*ENDMETHOD .*

*ENDCLASS.               "FETCH*

*\* Display data class*

*CLASS cl\_alv DEFINITION .*

*PUBLIC SECTION .*

*DATA : fetch\_obj  TYPE REF TO cl\_fetch .*

*METHODS : constructor IMPORTING ref\_fetch TYPE REF TO cl\_fetch.*

*METHODS : display\_alv .*

*ENDCLASS .*

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

*\*&       Class (Implementation)  CL\_ALV*

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

*\*        Display the Output*

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

*CLASS cl\_alv IMPLEMENTATION.*

*METHOD constructor .*

*me->fetch\_obj = ref\_fetch .*

*ENDMETHOD .*

*METHOD display\_alv .*

*DATA: lx\_msg TYPE REF TO cx\_salv\_msg.*

*DATA: o\_alv TYPE REF TO cl\_salv\_table.*

*TRY.*

*cl\_salv\_table=>factory(*

*IMPORTING*

*r\_salv\_table = o\_alv*

*CHANGING*

*t\_table      = me->fetch\_obj->it\_vbap ).*

*CATCH cx\_salv\_msg INTO lx\_msg.*

*ENDTRY.*

*o\_alv->display( ).*

*ENDMETHOD.*

*ENDCLASS.               "CL\_ALV*

*\* Declare TYPE REF TO class objects*

*DATA: o\_sel     TYPE REF TO cl\_sel,*

*o\_fetch   TYPE REF TO cl\_fetch,*

*o\_display TYPE REF TO cl\_alv.*

*INITIALIZATION .*

*\* Creating Objects of the Class*

*CREATE OBJECT : o\_sel,*

*o\_fetch   EXPORTING ref\_sel = o\_sel, " ref\_sel is in Constructor*

*o\_display EXPORTING ref\_fetch = o\_fetch. " ref\_fetch is in Constructor*

*START-OF-SELECTION .*

*\* Import screen data to class screen*

*o\_sel->get\_screen( EXPORTING lp\_werks = p\_werks ls\_vbeln = s\_vbeln1[] ) .*

*\* Call fetch data to fetch the records*

*o\_fetch->fetch\_data( ).*

*END-OF-SELECTION .*

*\* Display data*

*o\_display->display\_alv( ).*