

## 0710 | ABAP

Z\_PACKAGE\_05010 | Z\_0707\_0503 | 요구사항

TABSTRIP을 이용하여 항공기 운항정보 조회 화면을 구성해 보자. 사용자에게 보이는 화면은 INPUT과 OUTPUT 필드들, 그리고 VIEWFLIGHT | VIEW AIRCRAFT | MAINTAIN BOOKING 탭을 가진 TAB STRIP, 그리고 각각 VIEW1 | MAINTAIN FLIGHTS | MAINTAIN BOOKING 에 해당하는 RADIO BUTTON으로 이루어져 있다. INPUT과 OUTPUT 필드들에 대해서는 운항 일정 테이블 (SDYN\_CONN) 에서 운항 일정이 존재하는 항공편에 대한 AIRLINE | FLIGHT NUMBER | DATE 를 입력했을 때 항공편 테이블 (SFLIGHT) 에서 AIRFARE | AIRLINE CURRENCY | PLANE TYPE | MAX. CAPACITY ECON. | OCCUPIED ECON. | TOTAL 정보를 조회하고자 한다. 만약 사용자가 VIEW 라디오 버튼을 누르면, 입력 조건에 따른 OUPUT을 조회하며, MAINTAIN FLIGHTS 라디오 버튼을 누르면 좌석 OCC 가 MAX보다 적을 경우 STATUS의 저장 버튼과 PLANE TYPE필드를 활성화하며, 변경한 PLANE TYPE을 ZSFLIGHT05에 반영한다. TABSTRIP은 초기화면을 VIEWFLIGHT 로 하며 SPFLI 테이블에서 COUNTRYFR | COUNTRYTO | DEPART.CITY | ARRIVAL CITY | DEP. AIRPORT | DEST. AIRPORT | DEPARTURE | ARRIVAL TIME 을 조회한다. VIEW AIRCRAFT 탭에서는 SAPLANE 테이블에서 PLANE TYPE | MANUFACTURER | MAX. CAPACITY ECON | FUEL | UNIT | WEIGHT | UNIT OF MEASURE | SPEED | UNIT 을 조회하며 STATUS BAR에 EXIT BACK CANC 를 선택했을 시 프로그램 종료 또는 호출된 시점에서의 복귀를 수 행할 수 있도록 구현한다.

TABLE & STR	SDYN_CONN   SFLIGHT   SAPLANE   SPFLI   ZSFLIGHT05
SCREEN	100 110 120 130 150
INPUT	AIRLINE   FLIGHT NUMBER   DATE

OUTPUT	AIRFARE   AIRLINE CURRENCY   PLANE TYPE   MAX. CAPACITY ECON.   OCCUPIED ECON.   TOTAL COUNTRYFR   COUNTRYTO   DEPART.CITY   ARRIVAL CITY   DEP. AIRPORT   DEST. AIRPORT   DEPARTURE   ARRIVAL TIME PLANE TYPE   MANUFACTURER   MAX. CAPACITY ECON   FUEL   UNIT   WEIGHT   UNIT OF MEASURE   SPEED   UNIT
RADIO BTN	VIEW1   MAINTAIN FLIGHTS   MAINTAIN BOOKING
TABSTRIP	MY_TAB
SUBSCREEN	SUB
COMMAND	SAVE   BACK   EXIT   CANC

MOD.POOL |

PROGRAM Z\_0707\_0503.

INCLUDE Z\_0707\_0503\_TOP

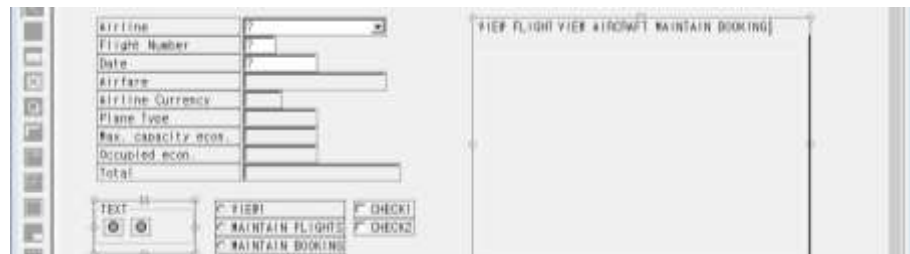
INCLUDE Z\_0707\_0503\_O01

INCLUDE Z\_0707\_0503\_I01

INCLUDE Z\_0707\_0503\_E01.

. " Global Data  
. " PBO-Modules  
. " PAI-Modules

LAYOUT | 100



TAB	FC1   FC2   FC3
SUBSCREEN	SUB
REF.	SUB

## SCREEN | 100

OK_CODE	OK	0	0	20	20	1	OK
---------	----	---	---	----	----	---	----

## SCREEN | 100 | FLOW LOGIC

PROCESS BEFORE OUTPUT.

MODULE STATUS\_0100.

MODULE MOVE\_TO\_DYNP.

MODULE CLEAR\_OKCODE.

MODULE SET\_CURSOR.

MODULE SET\_ICON.

MODULE MODIFY\_SCREEN.

MODULE FILL\_DYNNR. "STRUCTURE구조를 채워주는 MODULE

CALL SUBSCREEN SUB INCLUDING SY-

CPRG DYNNR. "SUBSCREEN 호출 "현재 프로그램

\*CPRG는 CALLING C PROGRAM, 즉 DYNNR STRUCTURE을 포함  
한 SUBSCREEN

PROCESS AFTER INPUT.

MODULE EXIT AT EXIT-COMMAND.

CALL SUBSCREEN SUB1. "SUBSCREEN 호출

CALL SUBSCREEN SUB2.

CALL SUBSCREEN SUB3.

CHAIN.

FIELD: SDYN\_CONN-CARRID,  
SDYN\_CONN-CONNID,  
SDYN\_CONN-FLDATE.

MODULE CHECK\_SPFLIGHT ON CHAIN-REQUEST.  
ENDCHAIN.

CHAIN.

FIELD: SDYN\_CONN-SEATSMAX,  
SDYN\_CONN-PLANETYPE.

MODULE CHECK\_PLANETYPE ON CHAIN-REQUEST.  
ENDCHAIN.

MODULE USER\_COMMAND\_0100.

## LAYOUT | 110

Country	
Country	
Depart. city	
Arrival city	
Dep. airport	
Dest. airport	
Departure	
Arrival Time	

## SCREEN | 110 | FLOW LOGIC

PROCESS BEFORE **OUTPUT**.

**MODULE** GET\_SPFLI.

PROCESS AFTER **INPUT**.

*\* MODULE USER\_COMMAND\_0110.*

## LAYOUT | 120

Plane Type	
Manufacturer	
Max. capacity econ.	
Fuel	. . . . .
Unit	
Weight	. . . . .
Unit of measure	
Speed	. . . . .
Unit	

## SCREEN | 120 | FLOW LOGIC

PROCESS BEFORE **OUTPUT**.

*\* MODULE STATUS\_0120.*

**MODULE** GET\_SAPLANE.

PROCESS AFTER **INPUT**.

*\* MODULE USER\_COMMAND\_0120.*

## SCREEN | 130 | FLOW LOGIC

PROCESS BEFORE **OUTPUT**.

*\* MODULE STATUS\_0130.*

PROCESS AFTER **INPUT**.

*\* MODULE USER\_COMMAND\_0130.*

## LAYOUT | 150

Time	
------	--

## SCREEN | 150 | FLOW LOGIC

PROCESS BEFORE **OUTPUT**.

**MODULE** STATUS\_0150.

*\**

PROCESS AFTER **INPUT**.

*\* MODULE USER\_COMMAND\_0150.*

## STATUS | 100

Standard Toolbar					
	SAVE	BACK	EXIT	CANC	

Z\_0707\_0503\_TOP

PROGRAM Z\_0707\_0504.

TABLES SDYN\_CONN.

TABLES SAPLANE.

CONTROLS: MY\_TAB TYPE TABSTRIP.

*\*SCREEN에서 입력 받은 값을 SELECT 해오기 위해 선언*

DATA: WA\_SFLIGHT TYPE SFLIGHT.

DATA DYNNR TYPE SY-DYNNR.

DATA IO\_COMMAND.

DATA OK\_CODE TYPE SY-UCOMM.

DATA: ICON1 TYPE ICON-NAME,  
      ICON2 TYPE ICON-NAME,  
      ICONTEXT TYPE ICONS-TEXT.

DATA: CH1,  
      CH2.

DATA: VIEW1 VALUE 'X',  
      MAINTAIN\_FLIGHTS,  
      MAINTAIN\_BOOKING.

Z\_0707\_0503\_O01

MODULE STATUS\_0100 OUTPUT.

CASE 'X'.

    WHEN VIEW1.

        SET TITLEBAR '0100' WITH TEXT-VIE .

    WHEN MAINTAIN\_FLIGHTS.

        SET TITLEBAR '0100' WITH TEXT-FLI .

    WHEN MAINTAIN\_BOOKING.

        SET TITLEBAR '0100' WITH TEXT-BOO.

ENDCASE.

ENDMODULE.

\*&-----

MODULE MOVE\_TO\_DYNP OUTPUT.

    MOVE-CORRESPONDING WA\_SFLIGHT TO SDYN\_CONN.

ENDMODULE.

\*&-----

MODULE SET\_CURSOR OUTPUT.

    SET CURSOR FIELD 'SDYN\_CONN-CONNID'.

ENDMODULE.

\*&-----

MODULE CLEAR\_OKCODE OUTPUT.

    CLEAR OK\_CODE.

ENDMODULE.

```
MODULE STATUS_0150 OUTPUT.  
  SET PF-STATUS '0150'.  
  SET TITLEBAR '0150' WITH TEXT-VIE .  
ENDMODULE.
```

\*&-----

```
MODULE SET_ICON OUTPUT.  
CALL FUNCTION 'ICON_CREATE'  
  EXPORTING  
    NAME                = 'ICON_CHECKED'  
  IMPORTING  
    RESULT              = ICON1.
```

```
CALL FUNCTION 'ICON_CREATE'  
  EXPORTING  
    NAME                = 'ICON_INCOMPLETE'  
  IMPORTING  
    RESULT              = ICON2.
```

```
ENDMODULE.
```

\*&-----

```
MODULE MODIFY_SCREEN OUTPUT.  
  IF MAINTAIN_FLIGHTS = 'X'.  
    SET PF-STATUS 'STATUS_100'.  
    LOOP AT SCREEN.
```

```
    IF SCREEN-NAME = 'SDYN_CONN-PLANETYPE'.  
      SCREEN-INPUT = 1 .  
      MODIFY SCREEN.
```

```
    ENDIF.
```

```
  ENDLOOP.
```

```
ELSE.
```

```
  SET PF-STATUS 'STATUS_100' EXCLUDING 'SAVE'.
```

```
ENDIF.
```

```
ENDMODULE.
```

\*&-----

```
MODULE FILL_DYNNR OUTPUT.
```

```
  CASE MY_TAB-ACTIVETAB.
```

```
    WHEN 'FC1'.
```

```
      DYNNR = 110.
```

```
    WHEN 'FC2'.
```

```
      DYNNR = 120.
```

```
    WHEN 'FC3'.
```

```
      DYNNR = 130.
```

```
    WHEN OTHERS.
```

```
      MY_TAB-ACTIVETAB = 'FC1'.
```

```
      DYNNR = 110.
```

```
  ENDCASE.
```

```
ENDMODULE.
```

*\*VIEW는 WHEN에 VIEW NAME , 수행에 DYNNR = SCREEN NUMBER.*

*\*&-----*

MODULE GET\_SPFLI OUTPUT.

ON CHANGE OF WA\_SFLIGHT-CARRID OR WA\_SFLIGHT-  
CONNID.

SELECT SINGLE \*

FROM SPFLI

INTO CORRESPONDING FIELDS OF SDYN\_CONN

WHERE CARRID = WA\_SFLIGHT-CARRID

AND CONNID = WA\_SFLIGHT-CONNID.

ENDON.

ENDMODULE.

*\*&-----*

MODULE GET\_SAPLANE OUTPUT.

ON CHANGE OF WA\_SFLIGHT-PLANETYPE.

SELECT SINGLE \*

FROM SAPLANE

INTO SAPLANE

WHERE PLANETYPE = WA\_SFLIGHT-PLANETYPE.

ENDON.

ENDMODULE.

Z\_0707\_0503\_I01

MODULE USER\_COMMAND\_0100 INPUT.

CASE OK\_CODE.

WHEN 'BACK'.

LEAVE TO SCREEN 0.

WHEN 'TIME'.

CALL SCREEN 150

STARTING AT 10 10

ENDING AT 50 20.

WHEN 'SAVE'.

UPDATE ZSFLIGHT05 FROM WA\_SFLIGHT.

IF SY-SUBRC <> 0.

MESSAGE A008(ZBC4100) .

ENDIF.

MESSAGE S009(ZBC4100) .

WHEN 'FC1' OR 'FC2' OR 'FC3'.

MY\_TAB-ACTIVETAB = OK\_CODE.

ENDCASE.

ENDMODULE.

*\*&-----*

MODULE CHECK\_SPFLIGHT INPUT.

SELECT SINGLE \*

FROM SFLIGHT

```

        INTO WA_SFLIGHT
        WHERE CARRID = SDYN_CONN-CARRID
              AND CONNID = SDYN_CONN-CONNID
              AND FLDATE = SDYN_CONN-FLDATE.

        IF SY-SUBRC <> 0.
            CLEAR WA_SFLIGHT.
            MESSAGE E007(ZBC4100) .
        ENDIF.

```

ENDMODULE.

\*&-----

MODULE EXIT INPUT.

CASE OK\_CODE.

WHEN 'CANC'.

CLEAR WA\_SFLIGHT.

SET PARAMETER ID:

'CAR' FIELD WA\_SFLIGHT-CARRID,

'CON' FIELD WA\_SFLIGHT-CONNID,

'DAY' FIELD WA\_SFLIGHT-FLDATE.

LEAVE TO SCREEN 100.

WHEN 'EXIT'.

LEAVE PROGRAM.

ENDCASE.

ENDMODULE.

\*&-----

MODULE CHECK\_PLANETYPE INPUT.

IF SDYN\_CONN-PLANETYPE IS INITIAL.

MESSAGE E555(BC410) WITH 'PLANE TYPE NEEDED'(PLT) .

ENDIF.

SELECT SINGLE SEATSMAX

FROM SAPLANE

INTO SDYN\_CONN-SEATSMAX

WHERE PLANETYPE = SDYN\_CONN-PLANETYPE.

IF SDYN\_CONN-SEATSOCC > SDYN\_CONN-SEATSMAX.

MESSAGE E109(ZBC410).

ELSE.

MOVE-

CORRESPONDING SDYN\_CONN TO WA\_SFLIGHT.

ENDIF.

ENDMODULE.

Z\_0707\_0503\_E01

LOAD-OF-PROGRAM.

GET PARAMETER ID: "PARAMETER RECENT RECORD CALL

'CAR' FIELD SDYN\_CONN-CARRID,

'CON' FIELD SDYN\_CONN-CONNID,

'DAY' FIELD SDYN\_CONN-FLDATE.

SELECT SINGLE \*

FROM SFLIGHT

INTO WA\_SFLIGHT

WHERE CARRID = SDYN\_CONN-CARRID

AND CONNID = SDYN\_CONN-CONNID

AND FLDATE = SDYN\_CONN-FLDATE.

VIEW FLIGHT VIEW AIRCRAFT MAIN...

Country DE  
Country US  
Depart.city FRANKFURT  
Arrival city NEW YORK  
Dep. airport FRA  
Dest. airport JFK  
Departure 13:30:00  
Arrival Time 15:05:00

\*LH 0402 2020.12.24 실습 진행

VIEW FLIGHT VIEW AIRCRAFT MAIN...

Country DE  
Country US  
Depart.city FRANKFURT  
Arrival city NEW YORK  
Dep. airport FRA  
Dest. airport JFK  
Departure 13:30:00  
Arrival Time 15:05:00

VIEW FLIGHT VIEW AIRCRAFT MAIN...

Country DE  
Country US  
Depart.city FRANKFURT  
Arrival city NEW YORK  
Dep. airport FRA  
Dest. airport JFK  
Departure 13:30:00  
Arrival Time 15:05:00

SE11 |

Data Browser: Table SFLIGHT: Selection Screen

Number of Entries

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE	SEATSMAX	SEATSOCC	PAYMENTSUM	SEATSMAX_B	SEATSOCC_B	SEAT
400	LH	0402 2020.12.24	666.00	EUR	A380-800	475	94	72,247.68	30	6	

SE80 |

VIEW FLIGHT VIEW AIRCRAFT MAIN...

Country DE  
Country US  
Depart.city FRANKFURT  
Arrival city NEW YORK  
Dep. airport FRA  
Dest. airport JFK  
Departure 13:30:00  
Arrival Time 15:05:00



SE11 | ZSFLIGHT05

Data Browser: Table ZSFLIGHT05: Selection Screen

Number of Entries

MANDT	CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE	SEATSMAX	SEATSOCC	PAYMENTSUM	SEATSMAX_B	SEATSOCC_B	SEAT
400	LH	0402	2020.12.24	666.00	EUR	A340-600	330	94	72,247.68	30	6	

Z\_PACKAGE\_05010 | Z\_0707\_0504 | 요구사항

TABSTRIP을 이용하여 항공기 운항정보 조회 화면을 구성해 보자. 위의 Z\_0707\_0503 파일과 요구사항은 동일하나 이번에는 각 탭마다 서브스크린을 구성해 구현해보자. 실행 결과는 Z\_0707\_0503 과 동일하다

TABLE & STR	SDYN_CONN   SFLIGHT   SAPLANE   SPFLI   ZSFLIGHT05
SCREEN	100 110 120 130 150
INPUT	AIRLINE   FLIGHT NUMBER   DATE
OUTPUT	AIRFARE   AIRLINE CURRENCY   PLANE TYPE   MAX. CAPACITY ECON.   OCCUPIED ECON.   TOTAL COUNTRYFR   COUNTRYTO   DEPART.CITY   ARRIVAL CITY   DEP. AIRPORT   DEST. AIRPORT   DEPARTURE   ARRIVAL TIME
RADIO BTN	VIEW1   MAINTAIN FLIGHTS   MAINTAIN BOOKING
TABSTRIP	MY_TAB
SUBSCREEN	SUB
COMMAND	SAVE   BACK   EXIT   CANC

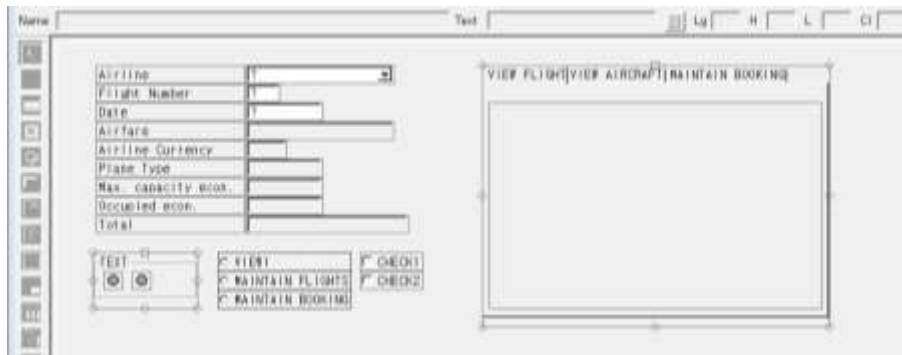
MOD.POOL |

INCLUDE Z\_0707\_0504\_TOP.

INCLUDE Z\_0707\_0504\_O01.

INCLUDE Z\_0707\_0504\_IO1.

INCLUDE Z\_0707\_0504\_E01.



Name	FC1
Text	VIEW_FLIGHT_____

Ref, Field	SUB1
------------	------

Name	FC2
Text	VIEW_AIRCRAFT_____

Ref, Field	SUB2
------------	------

Name	FC3
Text	MAINTAIN_BOOKING_____

Ref, Field	SUB3
------------	------

TAB	FC1   FC2   FC3
SUBSCREEN	SUB
REF.	SUB1   SUB2   SUB3

PROCESS BEFORE **OUTPUT**.

**MODULE** STATUS\_0100.

**MODULE** MOVE\_TO\_DYNP.

**MODULE** CLEAR\_OKCODE.

**MODULE** SET\_CURSOR.

**MODULE** SET\_ICON.

**MODULE** MODIFY\_SCREEN.

**MODULE** FILL\_DYNNR. "STRUCTURE구조를 채워주는 MODULE

**CALL** SUBSCREEN SUB1 INCLUDING SY-

CPROG '0110'. "SUBSCREEN 호출

**CALL** SUBSCREEN SUB2 INCLUDING SY-CPROG '0120'.

**CALL** SUBSCREEN SUB3 INCLUDING SY-CPROG '0130'.

\*CPROG는 CALLING PROGRAM, 즉 DYNNR STRUCTURE을 포함  
한 SUBSCREEN

PROCESS AFTER **INPUT**.

\*사용자가 입력한 값을 가지고 SELECT

\*PBO 화면이 보여지기 이전

\*PAI 화면에 입력된 값을 처리하기 위한 로직

**MODULE** EXIT AT EXIT-COMMAND.

**CALL** SUBSCREEN SUB1. "SUBSCREEN 호출

**CALL** SUBSCREEN SUB2.

**CALL** SUBSCREEN SUB3.

CHAIN.

FIELD: SDYN\_CONN-CARRID,  
SDYN\_CONN-CONNID,  
SDYN\_CONN-FLDATE.

MODULE CHECK\_SPFLIGHT ON CHAIN-REQUEST.  
ENDCHAIN.

CHAIN.

FIELD: SDYN\_CONN-SEATSMAX,  
SDYN\_CONN-PLANETYPE.

MODULE CHECK\_PLANETYPE ON CHAIN-REQUEST.  
ENDCHAIN.

MODULE USER\_COMMAND\_0100.

Z\_0707\_0504\_TOP

PROGRAM Z\_0707\_0504.

TABLES SDYN\_CONN.

TABLES SAPLANE.

CONTROLS: MY\_TAB TYPE TABSTRIP.

DATA: WA\_SFLIGHT TYPE SFLIGHT.

DATA DYNNR TYPE SY-DYNNR.

DATA IO\_COMMAND.

DATA OK\_CODE TYPE SY-UCOMM.

DATA: ICON1 TYPE ICON-NAME,  
ICON2 TYPE ICON-NAME,  
ICONTEXT TYPE ICONS-TEXT.

DATA: CH1,  
CH2.

DATA: VIEW1 VALUE 'X',  
MAINTAIN\_FLIGHTS,  
MAINTAIN\_BOOKING.

Z\_0707\_0504\_O01

MODULE STATUS\_0100 OUTPUT.

CASE 'X'.

WHEN VIEW1.

SET TITLEBAR '0100' WITH TEXT-VIE .

WHEN MAINTAIN\_FLIGHTS.

SET TITLEBAR '0100' WITH TEXT-FLI .

WHEN MAINTAIN\_BOOKING.

SET TITLEBAR '0100' WITH TEXT-BOO.

```

ENDCASE.
ENDMODULE.
*&-----
MODULE MOVE_TO_DYNP OUTPUT.
    MOVE-CORRESPONDING WA_SFLIGHT TO SDYN_CONN.
ENDMODULE.
*&-----
MODULE SET_CURSOR OUTPUT.
    SET CURSOR FIELD 'SDYN_CONN-CONNID'.
ENDMODULE.
*&-----
MODULE CLEAR_OKCODE OUTPUT.
    CLEAR OK_CODE.
ENDMODULE.

MODULE STATUS_0150 OUTPUT.
    SET PF-STATUS '0150'.
    SET TITLEBAR '0150' WITH TEXT-VIE .
ENDMODULE.
*&-----
MODULE SET_ICON OUTPUT.
CALL FUNCTION 'ICON_CREATE'
EXPORTING
    NAME                = 'ICON_CHECKED'
IMPORTING
    RESULT              = ICON1.

```

```

CALL FUNCTION 'ICON_CREATE'
EXPORTING
    NAME                = 'ICON_INCOMPLETE'
IMPORTING
    RESULT              = ICON2.

ENDMODULE.
*&-----
MODULE MODIFY_SCREEN OUTPUT.
    IF MAINTAIN_FLIGHTS = 'X'.
        SET PF-STATUS '0100'.
        LOOP AT SCREEN.
            IF SCREEN-NAME = 'SDYN_CONN-PLANETYPE'.
                SCREEN-INPUT = 1 .
                MODIFY SCREEN.
            ENDIF.
        ENDLOOP.
    ELSE.
        SET PF-STATUS '0100' EXCLUDING 'SAVE'.
    ENDIF.
ENDMODULE.
*&-----
MODULE FILL_DYNNR OUTPUT.
    CASE MY_TAB-ACTIVETAB.

```

```

WHEN 'FC1'.
    DYNNR = 110.
WHEN 'FC2'.
    DYNNR = 120.
WHEN 'FC3'.
    DYNNR = 130.
WHEN OTHERS.
    MY_TAB-ACTIVETAB = 'FC1'.
    DYNNR = 110.

```

```
ENDCASE.
```

```
ENDMODULE.
```

*\*VIEW는 WHEN에 VIEW NAME , 수행에 DYNNR = SCREEN NUMBER.*

*\*&-----*

```
MODULE GET_SPFLI OUTPUT.
```

```
    ON CHANGE OF WA_SFLIGHT-CARRID OR WA_SFLIGHT-
    CONNID.
```

```
        SELECT SINGLE *
```

```
            FROM SPFLI
```

```
            INTO CORRESPONDING FIELDS OF SDYN_CON
```

```
N
```

```
            WHERE CARRID = WA_SFLIGHT-CARRID
```

```
                AND CONNID = WA_SFLIGHT-CONNID.
```

```
ENDON.
```

```
ENDMODULE.
```

*\*&-----*

```
MODULE GET_SAPLANE OUTPUT.
```

```
    ON CHANGE OF WA_SFLIGHT-PLANETYPE.
```

```
        SELECT SINGLE *
```

```
            FROM SAPLANE
```

```
            INTO CORRESPONDING FIELDS OF SAPLANE
```

```
            WHERE PLANETYPE = WA_SFLIGHT-PLANETYPE.
```

```
ENDON.
```

```
ENDMODULE.
```

Z\_0707\_0504\_I01

```
MODULE USER_COMMAND_0100 INPUT.
```

```
    CASE OK_CODE.
```

```
        WHEN 'BACK'.
```

```
            LEAVE TO SCREEN 0.
```

```
        WHEN 'TIME'.
```

```
            CALL SCREEN 150
```

```
                STARTING AT 10 10
```

```
                ENDING AT 50 20.
```

```
        WHEN 'SAVE'.
```

```
            UPDATE ZSFLIGHT05 FROM WA_SFLIGHT.
```

```
            IF SY-SUBRC <> 0.
```

```

        MESSAGE A008(ZBC4100) .
    ENDIF.
    MESSAGE S009(ZBC4100) .
    WHEN 'FC1' OR 'FC2' OR 'FC3'.
        MY_TAB-ACTIVETAB = OK_CODE.
    ENDCASE.
ENDMODULE.
*&-----
MODULE CHECK_SPFLIGHT INPUT.
    SELECT SINGLE *
        FROM SFLIGHT
        INTO WA_SFLIGHT
        WHERE CARRID = SDYN_CONN-CARRID
            AND CONNID = SDYN_CONN-CONNID
            AND FLDATE = SDYN_CONN-FLDATE.

    IF SY-SUBRC <> 0.
        CLEAR WA_SFLIGHT.
        MESSAGE E007(ZBC4100) .
    ENDIF.
ENDMODULE.
*&-----
MODULE EXIT INPUT.
    CASE OK_CODE.
        WHEN 'EXIT'.

```

```

        LEAVE PROGRAM.
    WHEN 'CANC'.
        CLEAR WA_SFLIGHT.
        SET PARAMETER ID:
        'CAR' FIELD WA_SFLIGHT-CARRID,
        'CON' FIELD WA_SFLIGHT-CONNID,
        'DAY' FIELD WA_SFLIGHT-FLDATE.
        LEAVE TO SCREEN 100.

    ENDCASE.
ENDMODULE.

MODULE CHECK_PLANETYPE INPUT.
    IF SDYN_CONN-PLANETYPE IS INITIAL.
        MESSAGE E555(BC410) WITH 'PLANE TYPE NEEDED'(PLT) .
    ENDIF.
    SELECT SINGLE SEATSMAX
        FROM SAPLANE
        INTO SDYN_CONN-SEATSMAX
        WHERE PLANETYPE = SDYN_CONN-PLANETYPE.
    IF SDYN_CONN-SEATSOCC > SDYN_CONN-SEATSMAX.
        MESSAGE E109(BC410).
    ELSE.
        MOVE-

```

CORRESPONDING SDYN\_CONN TO WA\_SFLIGHT.

ENDIF.

ENDMODULE.

Z\_0707\_0504\_E01

LOAD-OF-PROGRAM.

GET PARAMETER ID: "PARAMETER RECENT RECORD CALL

'CAR' FIELD SDYN\_CONN-CARRID,

'CON' FIELD SDYN\_CONN-CONNID,

'DAY' FIELD SDYN\_CONN-FLDATE.

SELECT SINGLE \*

FROM SFLIGHT

INTO WA\_SFLIGHT

WHERE CARRID = SDYN\_CONN-CARRID

AND CONNID = SDYN\_CONN-CONNID

AND FLDATE = SDYN\_CONN-FLDATE.

FLIGHT ( DISPLAY )

VIEW FLIGHT VIEW AIRCRAFT

Airline	
Flight Number	
Date	
Airfare	0.00
Airline Currency	
Plane Type	
Max. capacity econ.	0
Occupied econ.	0
Total	0.00

TEXT

VIEW1

CHECK1

CHECK2

MAINTAIN FLIGHTS

MAINTAIN BOOKING

Country	
Country	
Depart.city	
Arrival city	
Dep. airport	
Dest. airport	
Departure	00:00:00
Arrival Time	00:00:00