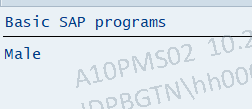
**SAP Basic programs**

|  |  |  |
| --- | --- | --- |
| S.NO | Contents | Page NO |
| 1 | Case statement |  |
| 1.1 | Case statement use OR |  |
| 2 | Concatenate, Condense, Split, Replace |  |
| 2.1 | Concatenate separated by and date concatenate |  |
| 2.2 | Palindrome using reverse |  |
| 2.3 | Translate |  |
| 2.4 | Condense and condense no gaps |  |
| 2.5 | Find, Find all occurrences, Replace |  |
| 3 | Data type declare in begin of and end of |  |
| 4 | Do & Uline |  |
| 4.1 | Do and use the end of selection |  |
| 4.2 | Do using Exit and check |  |
| 5 | Odd and even number using MOD |  |
| 6 | Factorial numbers using while |  |
| 6.1 | Offset using while, add, skip |  |
| 6.2 | While simple program |  |
| 6.3 | While using continue |  |
| 7 | If condition using between |  |
| 8 | Global message class declaration |  |
| 8.1 | Message using global class |  |
| 9 | Parameters additions |  |
| 10 | Report additions |  |
| 11 | SWAP 3rd variables |  |
| 11.1 | SWAP without use 3rd variables |  |
| 12 | Write additions |  |
| 13 | ABS constant value how to use |  |
| 14 | How to square root the numbers |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

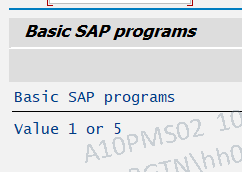
1.PARAMETERS P\_GENDER TYPE C LENGTH 10.  
  
CASE P\_GENDER.  
  WHEN 'MALE'.  
    WRITE:/ 'MALE'.  
  WHEN 'FEMALE'.  
    WRITE:/ 'FEMALE'.  
  WHEN OTHERS.  
    WRITE:/ 'TRANSGENDER'.  
ENDCASE.

Output: Input is male

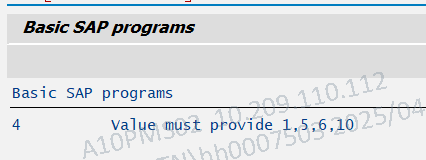


1.1 PARAMETERS P\_INPUT TYPE C LENGTH 10.  
  
  *" CASE OR STATEMENT*  
CASE P\_INPUT.  
  WHEN 1 OR 5.  
    WRITE 'VALUE 1 OR 5'.  
  WHEN 6 OR 10.  
    WRITE 'VALUE 6 OR 10'.  
  WHEN OTHERS.  
    WRITE: P\_INPUT, 'VALUE MUST PROVIDE 1,5,6,10'.  
ENDCASE.

Output: When input is 1

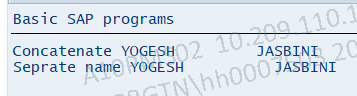


When input is 4.



2. PARAMETERS: P\_NAME1 TYPE C LENGTH 15,  
             P\_NAME2 TYPE C LENGTH 15.  
  
DATA: LV\_CONC TYPE STRING,  
      LV\_SPL1 TYPE C LENGTH 15,  
      LV\_SPL2 TYPE C LENGTH 15,  
      SEP.  
  
START-OF-SELECTION.  
  
CONCATENATE P\_NAME1 P\_NAME2 INTO LV\_CONC RESPECTING BLANKS.  
WRITE: 'CONCATENATE :', LV\_CONC.  
  
SPLIT LV\_CONC AT SEP INTO LV\_SPL1 LV\_SPL2.  
CONDENSE LV\_SPL2 NO-GAPS.  
WRITE:/ 'SEPRATE NAME :', LV\_SPL1, LV\_SPL2.  
  
REPLACE 'JASBINI' IN LV\_SPL2 WITH 'SAP'.  
WRITE:/ 'REPLACE :', LV\_SPL2.

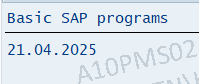
Output:



2.1 DATA: P\_DATE TYPE DATS VALUE '20250421',  
       P\_DATE1(10).  
  
CONSTANTS: C(1) VALUE '.'.  
P\_DATE1 = P\_DATE.  
  
CONCATENATE P\_DATE+6(2) P\_DATE+4(2) P\_DATE+0(4) INTO P\_DATE1 SEPARATED BY C.

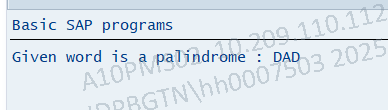
*"or*  
  
CONCATENATE P\_DATE+6(2) P\_DATE+4(2) P\_DATE+0(4) INTO P\_DATE1 SEPARATED BY '/'.  
  
WRITE: P\_DATE1.

Output:



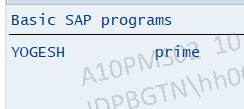
2.2 PARAMETERS P\_INPUT TYPE STRING.  
DATA: LV\_REV TYPE STRING.  
  
LV\_REV = REVERSE( P\_INPUT ).  
IF LV\_REV = P\_INPUT.  
  WRITE: 'GIVEN WORD IS A PALINDROME :', LV\_REV.  
ELSE.  
  WRITE: 'GIVEN WORD IS NOT A PALINDROME:', LV\_REV.  
ENDIF.

Output:



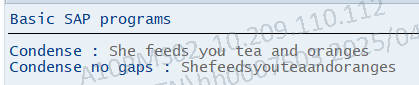
2.3 PARAMETERS: P\_INPUT1 TYPE C LENGTH 15,  
            P\_INPUT2 TYPE C LENGTH 15.  
  
DATA LV\_CONDENSE TYPE STRING.  
  
TRANSLATE P\_INPUT1 TO UPPER CASE.  
TRANSLATE P\_INPUT2 TO LOWER CASE.  
  
WRITE: P\_INPUT1,  
       P\_INPUT2.

Output:



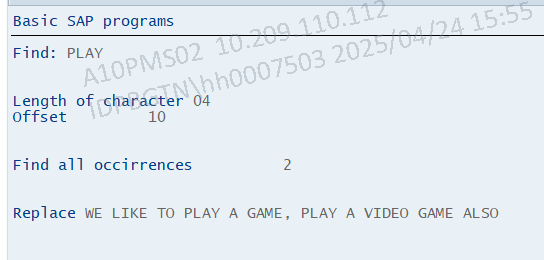
2.4 DATA: BEGIN OF SENTENCE,  
        WORD1 TYPE C LENGTH 30 VALUE 'SHE',  
        WORD2 TYPE C LENGTH 30 VALUE 'FEEDS',  
        WORD3 TYPE C LENGTH 30 VALUE 'YOU',  
        WORD4 TYPE C LENGTH 30 VALUE 'TEA',  
        WORD5 TYPE C LENGTH 30 VALUE 'AND',  
        WORD6 TYPE C LENGTH 30 VALUE 'ORANGES',  
      END OF SENTENCE,  
      TEXT TYPE STRING.  
  
TEXT = SENTENCE.  
CONDENSE TEXT.  
WRITE: / 'CONDENSE :', TEXT INVERSE COLOR 2.  
  
CONDENSE TEXT NO-GAPS.  
WRITE:/ 'CONDENSE NO GAPS :', TEXT INVERSE COLOR 2.

Output:



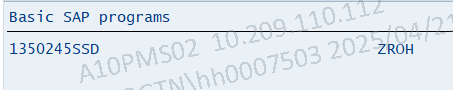
2.5 DATA: LV\_TEXT TYPE STRING,  
      LV\_OFFSET TYPE I,  
      LV\_COUNT  TYPE I,  
      LV\_LENGTH TYPE N LENGTH 2.  
  
LV\_TEXT = 'I LIKE TO PLAY A GAME, PLAY A VIDEO GAME ALSO'.  
  
FIND 'PLAY' IN LV\_TEXT MATCH OFFSET LV\_OFFSET MATCH LENGTH LV\_LENGTH.  
IF SY-SUBRC = 0.  
  WRITE: 'FIND:', LV\_TEXT+LV\_OFFSET(LV\_LENGTH) INVERSE COLOR 2.  
ENDIF.  
SKIP 2.  
WRITE:/ 'LENGTH OF CHARACTER', LV\_LENGTH INVERSE COLOR 2,/ 'OFFSET', LV\_OFFSET INVERSE COLOR 2.  
  
CLEAR LV\_COUNT.  
FIND ALL OCCURRENCES OF 'PLAY' IN LV\_TEXT MATCH COUNT LV\_COUNT.  
SKIP 2.  
WRITE:/ 'FIND ALL OCCIRRENCES', LV\_COUNT INVERSE COLOR 2.  
  
REPLACE 'I' IN LV\_TEXT WITH 'WE'.  
SKIP 2.  
WRITE:/ 'REPLACE', LV\_TEXT INVERSE COLOR 2.

Output:



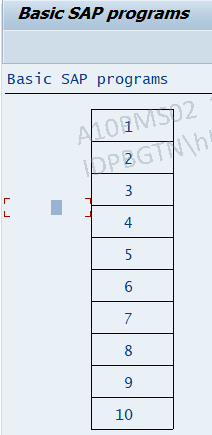
3. DATA: BEGIN OF TY\_MARA,  
        MATNR TYPE MATNR,  
        MATKL TYPE MATKL,  
      END OF TY\_MARA.  
  
START-OF-SELECTION.  
  TY\_MARA-MATNR = '1350245SSD'.  
  TY\_MARA-MATKL = 'ZROH'.  
  
  WRITE: / TY\_MARA-MATNR, TY\_MARA-MATKL.

Output:



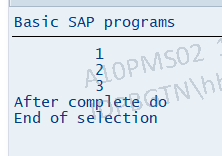
4. PARAMETERS P\_TIMES TYPE I.  
DATA ADD TYPE I.  
DO P\_TIMES TIMES.  
  IF SY-INDEX = 1.  
    WRITE:/ ADD.  
  ENDIF.  
  ADD = ADD + 1.  
  ULINE AT 10(10). *" START AT (LENGTH)*  
  WRITE: /10 '|', 11(8) ADD CENTERED, 19 '|'.  
  ULINE AT /10(10).  
ENDDO.

Output when 10



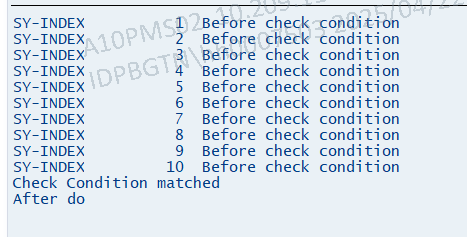
4.1 PARAMETERS P\_TIMES TYPE I.  
DATA ADD TYPE I.  
  
DO P\_TIMES TIMES.  
  WRITE:/ SY-INDEX.  
ENDDO.  
WRITE: / 'AFTER COMPLETE DO'.  
  
END-OF-SELECTION.  
  WRITE: / 'END OF SELECTION'.

Output When 2



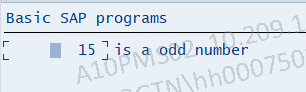
4.2 DO.  
  WRITE: / 'SY-INDEX', SY-INDEX, 'BEFORE CHECK CONDITION'.  
  CHECK SY-INDEX = 10.  
  WRITE: / 'CHECK CONDITION MATCHED'.  
  EXIT. *"If you remove exit you get dump in why it will create infinite loop*  
ENDDO.  
  
WRITE: / 'AFTER DO'.

Output



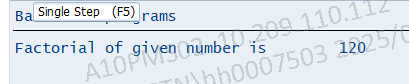
5. PARAMETERS P\_VALUE TYPE I.  
  
IF P\_VALUE MOD 2 = 0.  
  WRITE: P\_VALUE, 'IS A EVEN NUMBER'.  
ELSE.  
  WRITE: P\_VALUE, 'IS A ODD NUMBER'.  
ENDIF.

Output when 15.



6. PARAMETERS P\_VALUE TYPE I.  
DATA LV\_FACT TYPE I.  
LV\_FACT = 1.  
WHILE P\_VALUE NE 0.  
  LV\_FACT = LV\_FACT \* P\_VALUE.  
  P\_VALUE = P\_VALUE - 1.  
ENDWHILE.  
  
WRITE: 'FACTORIAL OF GIVEN NUMBER IS', LV\_FACT.

Output when 5 : Fact formula 5! = 5 \* 4\* 3 \* 2 \* 1 = 120



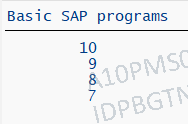
6.1 PARAMETERS P\_INPUT TYPE I.  
  
DATA: A TYPE INT1,  
      B TYPE INT1.  
  
WHILE B <= P\_INPUT.  
  A = 1.  
  WHILE A <= B.  
    WRITE A.  
    ADD 1 TO A.  
  ENDWHILE.  
  SKIP.  
  ADD 1 TO B.  
ENDWHILE.

Output when 5.



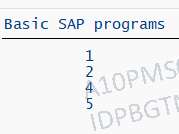
6.2 DATA LV\_A TYPE I VALUE 10.  
  
START-OF-SELECTION.  
  
  WHILE LV\_A > 6.  
    WRITE / LV\_A.  
    LV\_A = LV\_A - 1.  
  ENDWHILE.

Output



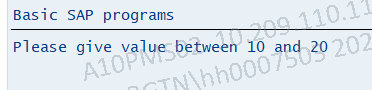
6.3 DATA LV\_A TYPE I VALUE 0.  
  
START-OF-SELECTION.  
  
  WHILE LV\_A < 5.  
    LV\_A = LV\_A + 1.  
    IF LV\_A = 3.  
      CONTINUE.  
    ENDIF.  
    WRITE / LV\_A.  
  ENDWHILE.

Output



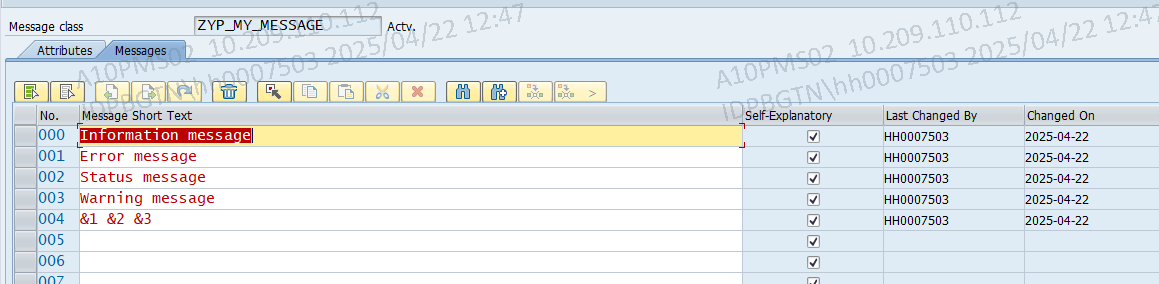
7. PARAMETERS P\_INPUT TYPE I.  
  
IF P\_INPUT BETWEEN 10 AND 20.  
  WRITE: 'VALUE BETWEEN 10 & 20'.  
ELSE.  
  WRITE: 'PLEASE GIVE VALUE BETWEEN 10 AND 20'.  
ENDIF.

Output when 9



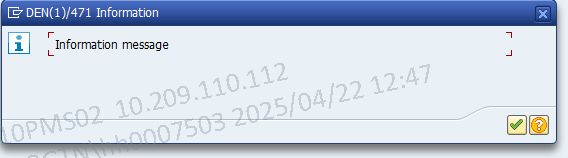
8. PARAMETERS: R\_INFO RADIOBUTTON GROUP G1,  
            R\_ERR  RADIOBUTTON GROUP G1,  
            R\_STA  RADIOBUTTON GROUP G1,  
            R\_WAR  RADIOBUTTON GROUP G1,  
            R\_ABO  RADIOBUTTON GROUP G1.  
DATA: LV\_MSG1 TYPE C LENGTH 10 VALUE 'YOGESH'.  
  
IF R\_INFO = 'X'.  
  MESSAGE I000(ZYP\_MY\_MESSAGE).  
ELSEIF R\_ERR = 'X'.  
  MESSAGE E001.  
ELSEIF R\_STA = 'X'.  
  MESSAGE S002.  
ELSEIF R\_WAR = 'X'.  
  MESSAGE W003.  
ELSEIF R\_ABO = 'X'.  
  MESSAGE A004(ZYP\_MY\_MESSAGE) WITH LV\_MSG1 'PRIME' 'JASBINI'.  
ENDIF.

Message ID screen



Output

Information



Error



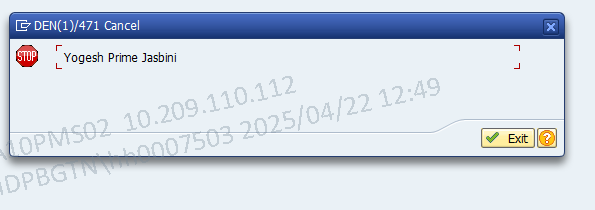
Status



Warning

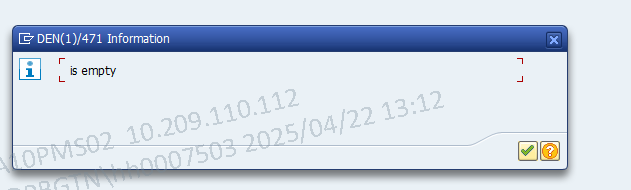


About

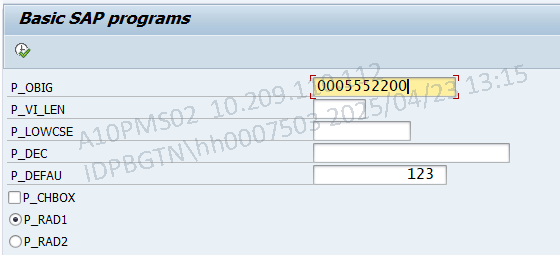


8.1 PARAMETERS P\_INPUT TYPE I.  
  
START-OF-SELECTION.  
  
  IF P\_INPUT IS INITIAL.  
    MESSAGE I007(00).  
  ELSE.  
    MESSAGE S007(045).  
  ENDIF.

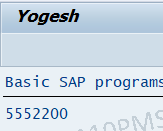
Output



9 PARAMETERS: P\_OBIG(15) TYPE C OBLIGATORY,  
            P\_VI\_LEN(25)  TYPE C VISIBLE LENGTH 5,  
            P\_LOWCSE(10)  TYPE C LOWER CASE,  
            P\_DEC         TYPE P DECIMALS 2,  
            P\_NO\_DIS      TYPE I NO-DISPLAY,  
            P\_DEFAU       TYPE I DEFAULT '123',  
            P\_CHBOX       AS CHECKBOX,  
            P\_RAD1        RADIOBUTTON GROUP G1,  
            P\_RAD2        RADIOBUTTON GROUP G1.  
  
DATA LV\_A TYPE I VALUE 10.  
  
SY-TITLE = 'YOGESH'.  
  
SHIFT P\_OBIG LEFT DELETING LEADING '0'. *" IT REMOVE THE PREFIX EXTRA ZERO'S*  
  
WRITE P\_OBIG.



Output



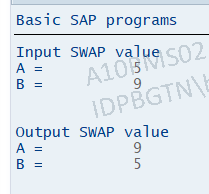
10 REPORT ZYP\_BASIC\_SAP\_PROGRAMS LINE-SIZE 20 LINE-COUNT 10(5).  
  
DATA: LV\_DOJ TYPE D,  
      LV\_TOJ TYPE T,  
      LV\_A TYPE I VALUE 10.  
  
START-OF-SELECTION.  
  
SY-TITLE = 'REPORT ADDISION'.  
LV\_DOJ = SY-DATUM.  
LV\_TOJ = SY-UZEIT.  
  
WRITE: / LV\_DOJ, / LV\_TOJ.

Output



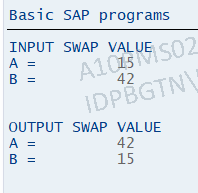
11 PARAMETERS: P\_A TYPE I,  
             P\_B TYPE I.  
  
DATA: P\_C TYPE I.  
  
WRITE:/ 'INPUT SWAP VALUE',  
      / 'A =',P\_A INVERSE COLOR 2,  
      / 'B =',P\_B INVERSE COLOR 2.  
  
P\_C = P\_A.  
P\_A = P\_B.  
P\_B = P\_C.  
SKIP 2.  
WRITE:/ 'OUTPUT SWAP VALUE',  
      / 'A =',P\_A INVERSE COLOR 2,  
      / 'B =',P\_B INVERSE COLOR 2.

Output



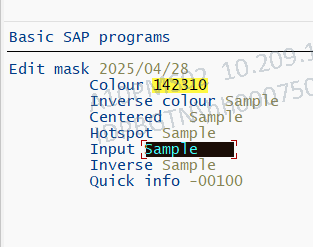
11.1 PARAMETERS: P\_A TYPE I,  
             P\_B TYPE I.  
  
WRITE:/ 'INPUT SWAP VALUE',  
      / 'A =',P\_A INVERSE COLOR 2,  
      / 'B =',P\_B INVERSE COLOR 2.  
  
P\_B = P\_B - P\_A.  
P\_A = P\_A + P\_B.  
P\_B = P\_A - P\_B.  
  
SKIP 2.  
WRITE:/ 'OUTPUT SWAP VALUE',  
      / 'A =',P\_A INVERSE COLOR 2,  
      / 'B =',P\_B INVERSE COLOR 2.

Output



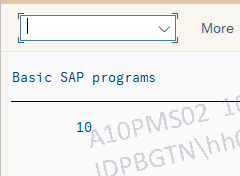
12 DATA: LV\_DOJ TYPE C LENGTH 10,  
       LV\_TOJ TYPE T,  
       LV\_A   TYPE C LENGTH 10 VALUE 'SAMPLE',  
       LV\_SIGN TYPE C LENGTH 10 VALUE '-00100'.  
  
START-OF-SELECTION.  
  
  SY-TITLE = 'YOGESH'.  
  LV\_DOJ  = SY-DATUM.  
  LV\_TOJ  = SY-UZEIT.  
  
  WRITE: / 'EDIT MASK', LV\_DOJ  INVERSE COLOR 3 USING EDIT MASK '\_\_\_\_/\_\_/\_\_',  
         /10 'COLOUR', LV\_TOJ  COLOR 3,  
         /10 'INVERSE COLOUR', LV\_A INVERSE COLOR 3,  
         /10 'CENTERED', LV\_A  CENTERED INVERSE COLOR 3,  
         /10 'HOTSPOT', LV\_A  HOTSPOT INVERSE COLOR 3,  
         /10 'INPUT', LV\_A  INPUT INVERSE COLOR 3,  
         /10 'INVERSE', LV\_A  LEFT-JUSTIFIED INVERSE COLOR 3,  
         /10 'QUICK INFO', LV\_SIGN QUICKINFO 'QUICK INFO' INVERSE COLOR 3.

Output:



13. DATA: LV\_NUMBER   TYPE I VALUE -10,  
       LV\_ABSOLUTE TYPE I.  
  
LV\_ABSOLUTE = ABS( LV\_NUMBER ). *" THE ABS() FUNCTION RETURNS THE ABSOLUTE VALUE OF THE INPUT, WHICH IS THE VALUE WITHOUT ITS SIGN.*  
  
WRITE LV\_ABSOLUTE.

Output:



14. DATA: LV\_SQU\_ROOT TYPE I VALUE 15,  
      LV\_RESULT   TYPE I.  
  
START-OF-SELECTION.  
  
  LV\_RESULT = SQRT( LV\_SQU\_ROOT ).  
  WRITE LV\_RESULT.

Output:

