

# Referenced Example Programs

This chapter contains some additional example programs that are referenced in the *Programming Guide*.

The following topics are covered:

- READ Statement
- FIND Statement
- Nested READ and FIND Statements
- ACCEPT and REJECT Statements
- AT START OF DATA and AT END OF DATA Statements
- DISPLAY and WRITE Statements
- DISPLAY Statement
- Column Headers
- Field-Output-Relevant Parameters
- Edit Masks
- DISPLAY VERT with WRITE Statement
- AT BREAK Statement
- COMPUTE, MOVE and COMPRESS Statements
- System Variables
- System Functions

## READ Statement

The following example is referenced in the section *Statements for Database Access*.

### READX03 - READ statement (with LOGICAL clause)

```
** Example 'READX03': READ (with LOGICAL clause)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 PERSONNEL-ID
  2 JOB-TITLE
END-DEFINE
*
LIMIT 8
READ EMPLOY-VIEW LOGICAL BY PERSONNEL-ID
  DISPLAY NOTITLE *ISN      NAME
```

```
'PERS-NO' PERSONNEL-ID
'POSITION' JOB-TITLE
```

```
END-READ
END
```

Output of Program READX03:

ISN	NAME	PERS-NO	POSITION
204	SCHINDLER	11100102	PROGRAMMIERER
205	SCHIRM	11100105	SYSTEMPROGRAMMIERER
206	SCHMITT	11100106	OPERATOR
207	SCHMIDT	11100107	SEKRETAERIN
208	SCHNEIDER	11100108	SACHBEARBEITER
209	SCHNEIDER	11100109	SEKRETAERIN
210	BUNGERT	11100110	SYSTEMPROGRAMMIERER
211	THIELE	11100111	SEKRETAERIN

## FIND Statement

The following examples are referenced in the section *Statements for Database Access*.

### FINDX07 - FIND statement (with several clauses)

```
** Example 'FINDX07': FIND (with several clauses)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 CITY
  2 SALARY (1)
  2 CURR-CODE (1)
END-DEFINE
*
FIND EMPLOY-VIEW WITH PHONETIC-NAME = 'JONES' OR = 'BECKR'
                      AND CITY      = 'BOSTON' THRU 'NEW YORK'
                      BUT NOT       'CHAPEL HILL'
                      SORTED BY NAME
                      WHERE SALARY (1) < 28000
  DISPLAY NOTITLE NAME FIRST-NAME CITY SALARY (1)
END-FIND
END
```

Output of Program FINDX07:

NAME	FIRST-NAME	CITY	ANNUAL SALARY
BAKER	PAULINE	DERBY	4450
JONES	MARTHA	KALAMAZOO	21000
JONES	KEVIN	DERBY	7000

**FINDX08 - FIND statement (with LIMIT)**

```

** Example 'FINDX08': FIND (with LIMIT)
**           Demonstrates FIND statement with LIMIT option to
**           terminate program with an error message if the
**           number of records selected exceeds a specified
**           limit (no output).
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 JOB-TITLE
END-DEFINE
*
FIND EMPLOY-VIEW WITH LIMIT (5) JOB-TITLE = 'SALES PERSON'
  DISPLAY NAME JOB-TITLE
END-FIND
END
    
```

Runtime Error Caused by Program FINDX08:

NAT1005 More records found than specified in search limit.

**FINDX09 - FIND statement (using \*NUMBER, \*COUNTER, \*ISN)**

```

** Example 'FINDX09': FIND (using *NUMBER, *COUNTER, *ISN)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 DEPT
  2 NAME
END-DEFINE
*
FIND EMPLOY-VIEW WITH CITY = 'BOSTON'
  WHERE DEPT = 'TECH00' THRU 'TECH10'
  DISPLAY NOTITLE
  'COUNTER' *COUNTER NAME DEPT 'ISN' *ISN
  AT START OF DATA
  WRITE '(TOTAL NUMBER IN BOSTON:' *NUMBER ')' /
  END-START
END-FIND
END
    
```

Output of Program FINDX09:

COUNTER	NAME	DEPARTMENT CODE	ISN
(TOTAL NUMBER IN BOSTON:		7 )	
1	STANWOOD	TECH10	782
2	PERREAULT	TECH10	842

**FINDX10 - FIND statement (combined with READ)**

```

** Example 'FINDX10': FIND (combined with READ)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
    
```

```

2 NAME
2 FIRST-NAME
1 VEHIC-VIEW VIEW OF VEHICLES
2 PERSONNEL-ID
2 MAKE
END-DEFINE
*
LIMIT 15
*
EMP. READ EMPLOY-VIEW BY NAME STARTING FROM 'JONES'
  VEH. FIND VEHIC-VIEW WITH PERSONNEL-ID = PERSONNEL-ID (EMP.)
    IF NO RECORDS FOUND
      MOVE '*** NO CAR ***' TO MAKE
    END-NOREC
    DISPLAY NOTITLE
      NAME (EMP.) (IS=ON)
      FIRST-NAME (EMP.) (IS=ON)
      MAKE (VEH.)
    END-FIND
  END-READ
END

```

Output of Program FINDX10:

NAME	FIRST-NAME	MAKE
JONES	VIRGINIA	CHRYSLER
	MARSHA	CHRYSLER
		CHRYSLER
	ROBERT	GENERAL MOTORS
	LILLY	FORD
		MG
	EDWARD	GENERAL MOTORS
	MARTHA	GENERAL MOTORS
	LAUREL	GENERAL MOTORS
	KEVIN	DATSUN
	GREGORY	FORD
JOPER	MANFRED	*** NO CAR ***
JOUSSELIN	DANIEL	RENAULT
JUBE	GABRIEL	*** NO CAR ***
JUNG	ERNST	*** NO CAR ***
JUNKIN	JEREMY	*** NO CAR ***
KAISER	REINER	*** NO CAR ***

**FINDX11 - FIND NUMBER statement (with \*NUMBER)**

```

** Example 'FINDX11': FIND NUMBER (with *NUMBER)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
2 FIRST-NAME
2 NAME
2 CITY
2 JOB-TITLE
2 SALARY          (1)
*
1 #PERCENT          (N.2)
1 REDEFINE #PERCENT
2 #WHOLE-NBR       (N2)
1 #ALL-BOST        (N3.2)
1 #SECR-ONLY       (N3.2)

```

```

1 #BOST-NBR      (N3)
1 #SECR-NBR      (N3)
END-DEFINE
*
F1. FIND NUMBER EMPLOY-VIEW WITH CITY = 'BOSTON'
F2. FIND NUMBER EMPLOY-VIEW WITH CITY = 'BOSTON'
      AND JOB-TITLE = 'SECRETARY'
*
MOVE *NUMBER(F1.) TO #ALL-BOST #BOST-NBR
MOVE *NUMBER(F2.) TO #SECR-ONLY #SECR-NBR
DIVIDE #ALL-BOST INTO #SECR-ONLY GIVING #PERCENT
*
WRITE TITLE LEFT JUSTIFIED UNDERLINED
  'There are' #BOST-NBR 'employees in the Boston offices.' /
  #SECR-NBR '(=' #WHOLE-NBR (EM=99%')) 'are secretaries.'
*
SKIP 1
FIND EMPLOY-VIEW WITH CITY      = 'BOSTON'
      AND JOB-TITLE = 'SECRETARY'
  DISPLAY NAME FIRST-NAME JOB-TITLE SALARY (1)
END-FIND
END

```

Output of Program FINDX11:

There are 7 employees in the Boston offices.  
 3 (= 42%) are secretaries.

```

-----

```

NAME	FIRST-NAME	CURRENT POSITION	ANNUAL SALARY
SHAW	LESLIE	SECRETARY	18000
CREMER	WALT	SECRETARY	20000
COHEN	JOHN	SECRETARY	16000

```

-----

```

## Nested READ and FIND Statements

The following examples are referenced in the section *Database Processing Loops*.

### READX04 - READ statement (in combination with FIND and the system variables \*NUMBER and \*COUNTER)

```

** Example 'READX04': READ (in combination with FIND and the system
**                       variables *NUMBER and *COUNTER)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 FIRST-NAME
1 VEHIC-VIEW VIEW OF VEHICLES
  2 PERSONNEL-ID
  2 MAKE
END-DEFINE
*
LIMIT 10
RD. READ EMPLOY-VIEW BY NAME STARTING FROM 'JONES'
FD. FIND VEHIC-VIEW WITH PERSONNEL-ID = PERSONNEL-ID (RD.)

```

```

IF NO RECORDS FOUND
  ENTER
  END-NOREC
  /*
  DISPLAY NOTITLE
    *COUNTER (RD.)(NL=8) NAME           (AL=15) FIRST-NAME (AL=10)
    *NUMBER  (FD.)(NL=8) *COUNTER (FD.)(NL=8) MAKE
  END-FIND
END-READ
END
    
```

Output of Program READX04:

CNT	NAME	FIRST-NAME	NMBR	CNT	MAKE
1	JONES	VIRGINIA	1	1	CHRYSLER
2	JONES	MARSHA	2	1	CHRYSLER
2	JONES	MARSHA	2	2	CHRYSLER
3	JONES	ROBERT	1	1	GENERAL MOTORS
4	JONES	LILLY	2	1	FORD
4	JONES	LILLY	2	2	MG
5	JONES	EDWARD	1	1	GENERAL MOTORS
6	JONES	MARTHA	1	1	GENERAL MOTORS
7	JONES	LAUREL	1	1	GENERAL MOTORS
8	JONES	KEVIN	1	1	DATSUN
9	JONES	GREGORY	1	1	FORD
10	JOPER	MANFRED	0	0	

**LIMITX01 - LIMIT statement (for READ, FIND loop processing)**

```

** Example 'LIMITX01': LIMIT (for READ, FIND loop processing)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
1 VEH-VIEW VIEW OF VEHICLES
  2 PERSONNEL-ID
  2 MAKE
END-DEFINE
*
LIMIT 4
*
READ EMPLOY-VIEW BY NAME STARTING FROM 'A'
  FIND VEH-VIEW WITH PERSONNEL-ID = EMPLOY-VIEW.PERSONNEL-ID
  IF NO RECORDS FOUND
    MOVE 'NO CAR' TO MAKE
  END-NOREC
  DISPLAY PERSONNEL-ID NAME FIRST-NAME MAKE
END-FIND
END-READ
END
    
```

Output of Program LIMITX01:

Page 1

04-12-13 14:01:57

PERSONNEL-ID	NAME	FIRST-NAME	MAKE
	ABELLAN	KEPA	NO CAR
30000231	ACHIESON	ROBERT	FORD
	ADAM	SIMONE	NO CAR
20008800	ADKINSON	JEFF	GENERAL MOTORS

## ACCEPT and REJECT Statements

The following examples are referenced in the section *Selecting Records Using ACCEPT/REJECT*.

### ACCEPX04 - ACCEPT IF ... LESS THAN ... statement

```

** Example 'ACCEPX04': ACCEPT IF ... LESS THAN ...
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 JOB-TITLE
  2 SALARY (1)
END-DEFINE
*
LIMIT 20
READ EMPLOY-VIEW BY PERSONNEL-ID = '20017000'
  ACCEPT IF SALARY (1) LESS THAN 38000
  DISPLAY NOTITLE PERSONNEL-ID NAME JOB-TITLE SALARY (1)
END-READ
END

```

Output of Program ACCEPX04:

PERSONNEL ID	NAME	CURRENT POSITION	ANNUAL SALARY
20017000	CREMER	ANALYST	34000
20017100	MARKUSH	TRAINEE	22000
20017400	NEEDHAM	PROGRAMMER	32500
20017500	JACKSON	PROGRAMMER	33000
20017600	PIETSCH	SECRETARY	22000
20017700	PAUL	SECRETARY	23000
20018000	FARRIS	PROGRAMMER	30500
20018100	EVANS	PROGRAMMER	31000
20018200	HERZOG	PROGRAMMER	31500
20018300	LORIE	SALES PERSON	28000
20018400	HALL	SALES PERSON	30000
20018500	JACKSON	MANAGER	36000
20018800	SMITH	SECRETARY	24000
20018900	LOWRY	SECRETARY	25000

**ACCEPX05 - ACCEPT IF ... AND ... statement**

```

** Example 'ACCEPX05': ACCEPT IF ... AND ...
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 CITY
  2 JOB-TITLE
  2 SALARY (1:2)
END-DEFINE
*
LIMIT 6
READ EMPLOY-VIEW PHYSICAL WHERE SALARY(2) > 0
  ACCEPT IF SALARY(1) > 10000
    AND SALARY(1) < 50000
  DISPLAY (AL=15) 'SALARY I' SALARY (1) 'SALARY II' SALARY (2)
    NAME JOB-TITLE CITY
END-READ
END
    
```

Output of Program ACCEPX05:

Page 1 04-12-13 14:05:28

SALARY I	SALARY II	NAME	CURRENT POSITION	CITY
48000	46000	SPENGLER	SACHBEARBEITER	DARMSTADT
45000	40000	SPECK	SACHBEARBEITER	DARMSTADT
48000	46000	SCHINDLER	PROGRAMMIERER	HEPPENHEIM
36000	32000	SCHMIDT	SEKRETAERIN	HEPPENHEIM

**ACCEPX06 - REJECT IF ... OR ... statement**

```

** Example 'ACCEPX06': REJECT IF ... OR ...
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 SALARY (1)
  2 JOB-TITLE
  2 CITY
  2 NAME
END-DEFINE
*
LIMIT 20
READ EMPLOY-VIEW LOGICAL BY PERSONNEL-ID = '20017000'
  REJECT IF SALARY (1) < 20000
    OR SALARY (1) > 26000
  DISPLAY NOTITLE SALARY (1) NAME JOB-TITLE CITY
END-READ
END
    
```

Output of Program ACCEPX06:



ANNUAL SALARY	NAME	CURRENT POSITION	CITY
22000	MARKUSH	TRAINEE	LOS ANGELES
22000	PIETSCH	SECRETARY	VISTA
23000	PAUL	SECRETARY	NORFOLK
24000	SMITH	SECRETARY	SILVER SPRING
25000	LOWRY	SECRETARY	LEXINGTON

## AT START OF DATA and AT END OF DATA Statements

The following examples are referenced in the section *AT START/END OF DATA Statements*.

### ATENDX01 - AT END OF DATA statement

```

** Example 'ATENDX01': AT END OF DATA
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 JOB-TITLE
END-DEFINE
*
READ (6) EMPLOY-VIEW BY PERSONNEL-ID FROM '20017000'
  DISPLAY NOTITLE NAME JOB-TITLE
  AT END OF DATA
    WRITE / 'LAST PERSON SELECTED:' OLD(NAME)
  END-ENDDATA
END-READ
END
    
```

Output of Program ATENDX01:

NAME	CURRENT POSITION
CREMER	ANALYST
MARKUSH	TRAINEE
GEE	MANAGER
KUNEY	DBA
NEEDHAM	PROGRAMMER
JACKSON	PROGRAMMER

LAST PERSON SELECTED: JACKSON

### ATSTAX02 - AT START OF DATA statement

```

** Example 'ATSTAX02': AT START OF DATA
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 SALARY (1)
  2 CURR-CODE (1)
  2 BONUS (1,1)
    
```

```

END-DEFINE
*
LIMIT 3
FIND EMPLOY-VIEW WITH CITY = 'MADRID'
  DISPLAY NAME FIRST-NAME SALARY(1) BONUS(1,1) CURR-CODE (1)
  /*
  AT START OF DATA
  WRITE NOTITLE *DAT4E /
  END-START
END-FIND
END
    
```

Output of Program ATSTAX02:

NAME	FIRST-NAME	ANNUAL SALARY	BONUS	CURRENCY CODE
-----				
13/12/2004				
DE JUAN	JAVIER	1988000		0 PTA
DE LA MADRID	ANSELMO	3120000		0 PTA
PINERO	PAULA	1756000		0 PTA

**WRITEX09 - WRITE statement (in combination with AT END OF DATA)**

```

** Example 'WRITEX09': WRITE (in combination with AT END OF DATA )
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 NAME
  2 BIRTH
  2 JOB-TITLE
  2 DEPT
END-DEFINE
*
READ (3) EMPLOY-VIEW BY CITY
  DISPLAY NOTITLE NAME BIRTH (EM=YYYY-MM-DD) JOB-TITLE
  WRITE 38T 'DEPT CODE:' DEPT
  /*
  AT END OF DATA
  WRITE / 'LAST PERSON SELECTED:' OLD(NAME)
  END-ENDDATA
  SKIP 1
END-READ
END
    
```

Output of Program WRITEX09:

NAME	DATE OF BIRTH	CURRENT POSITION
-----		
SENKO	1971-09-11	PROGRAMMER DEPT CODE: TECH10
GODEFROY	1949-01-09	COMPTABLE DEPT CODE: COMP02
CANALE	1942-01-01	CONSULTANT

DEPT CODE: TECH03

LAST PERSON SELECTED: CANALE

## DISPLAY and WRITE Statements

The following examples are referenced in the section *Statements DISPLAY and WRITE*.

### DISPLX13 - DISPLAY statement (compare with WRITEX08 using WRITE)

```

** Example 'DISPLX13': DISPLAY (compare with WRITEX08 using WRITE)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 SALARY (2)
  2 BONUS (1,1)
  2 CITY
END-DEFINE
*
LIMIT 2
READ EMPLOY-VIEW WITH CITY = 'CHAPEL HILL' WHERE BONUS(1,1) NE 0
/*
  DISPLAY 'PERS/ID' PERSONNEL-ID NAME / FIRST-NAME
          '***' '=' SALARY(1:2) 'BONUS' BONUS(1,1) CITY (AL=15)
/*
SKIP 1
END-READ
END
    
```

Output of Program DISPLX13:

```

Page          1                                04-12-13  14:11:28

  PERS          NAME          ANNUAL          BONUS          CITY
  ID           FIRST-NAME        SALARY
-----
20027000 CUMMINGS          **          41000          1500 CHAPEL HILL
          PUALA                38900
20000200 WOOLSEY          **          26000          3000 CHAPEL HILL
          LOUISE                24700
    
```

### WRITEX08 - WRITE statement (compare with DISPLX13 using DISPLAY)

```

** Example 'WRITEX08': WRITE (compare with DISPLX13 using DISPLAY)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 SALARY (2)
  2 BONUS (1,1)
  2 CITY
END-DEFINE
    
```

```

*
LIMIT 2
READ EMPLOY-VIEW WITH CITY = 'CHAPEL HILL' WHERE BONUS(1,1) NE 0
/*
WRITE 'PERS/ID' PERSONNEL-ID NAME / FIRST-NAME
      '*** '=' SALARY(1:2) 'BONUS' BONUS(1,1) CITY (AL=15)
/*
SKIP 1
END-READ
END
    
```

Output of Program WRITEX08:

```

Page      1                                04-12-13  14:12:43

PERS/ID 20027000 CUMMINGS
PUALA          ** ANNUAL SALARY:      41000      38900 BONUS      1500
CHAPEL HILL

PERS/ID 20000200 WOOLSEY
LOUISE         ** ANNUAL SALARY:      26000      24700 BONUS      3000
CHAPEL HILL
    
```

**DISPLX14 - DISPLAY statement (with AL, SF and nX)**

```

** Example 'DISPLX14': DISPLAY (with AL, SF and nX)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 FIRST-NAME
  2 NAME
  2 ADDRESS-LINE (1)
  2 TELEPHONE
  3 AREA-CODE
  3 PHONE
  2 CITY
END-DEFINE
*
READ (3) EMPLOY-VIEW BY NAME STARTING FROM 'W'
  DISPLAY (AL=15 SF=5) NAME CITY / ADDRESS-LINE(1) 2X TELEPHONE
SKIP 1
END-READ
END
    
```

Output of Program DISPLX14:

```

Page      1                                04-12-13  14:14:00

      NAME                CITY                TELEPHONE
                ADDRESS
                AREA
                CODE                TELEPHONE
-----
WABER           HEIDELBERG           06221           456452
                ERBACHERSTR. 78

WADSWORTH       DERBY                0332           515365
                56 PINECROFT CO

WAGENBACH       FRANKFURT            069            983218
                BECKERSTR. 4
    
```

**WRITEX09 - WRITE statement (in combination with AT END OF DATA)**

```

** Example 'WRITEX09': WRITE (in combination with AT END OF DATA )
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 NAME
  2 BIRTH
  2 JOB-TITLE
  2 DEPT
END-DEFINE
*
READ (3) EMPLOY-VIEW BY CITY
  DISPLAY NOTITLE NAME BIRTH (EM=YYYY-MM-DD) JOB-TITLE
  WRITE 38T 'DEPT CODE:' DEPT
  /*
  AT END OF DATA
  WRITE / 'LAST PERSON SELECTED:' OLD(NAME)
  END-ENDDATA
  SKIP 1
END-READ
END
    
```

Output of Program WRITEX09:

NAME	DATE OF BIRTH	CURRENT POSITION
SENKO	1971-09-11	PROGRAMMER DEPT CODE: TECH10
GODEFROY	1949-01-09	COMPTABLE DEPT CODE: COMP02
CANALE	1942-01-01	CONSULTANT DEPT CODE: TECH03

LAST PERSON SELECTED: CANALE

**DISPLAY Statement**

The following example is referenced in the section *Page Titles, Page Breaks, Blank Lines*.

**DISPLX21 DISPLAY statement (with slash '/' and compare with WRITE)**

```

** Example 'DISPLX21': DISPLAY (usage of slash '/' in DISPLAY and WRITE)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 NAME
  2 FIRST-NAME
  2 ADDRESS-LINE (1)
END-DEFINE
*
WRITE TITLE LEFT JUSTIFIED UNDERLINED
    
```

```

*TIME
5X 'PEOPLE LIVING IN SALT LAKE CITY'
21X 'PAGE:' *PAGE-NUMBER /
15X 'AS OF' *DAT4E //
*
WRITE TRAILER UNDERLINED 'REGISTER OF' / 'SALT LAKE CITY'
*
READ (2) EMPLOY-VIEW WITH CITY = 'SALT LAKE CITY'
  DISPLAY NAME /
    FIRST-NAME
    'HOME/CITY' CITY
    'STREET/OR BOX NO.' ADDRESS-LINE (1)
  SKIP 1
END-READ
END

```

Output of Program DISPLX21:

```

14:15:50.1      PEOPLE LIVING IN SALT LAKE CITY      PAGE:      1
                AS OF 13/12/2004

```

NAME FIRST-NAME	HOME CITY	STREET OR BOX NO.
ANDERSON JENNY	SALT LAKE CITY	3701 S. GEORGE MASON
SAMUELSON MARTIN	SALT LAKE CITY	7610 W. 86TH STREET

REGISTER OF  
SALT LAKE CITY

## Column Headers

The following example is referenced in the section *Column Headers*.

### DISPLX15 - DISPLAY statement (with FC, UC)

```

** Example 'DISPLX15': DISPLAY (with FC, UC)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 FIRST-NAME
  2 NAME
  2 ADDRESS-LINE (1)
  2 CITY
  2 TELEPHONE
  3 AREA-CODE
  3 PHONE
END-DEFINE
*
FORMAT AL=12 GC== UC=%
*
READ (3) EMPLOY-VIEW BY NAME STARTING FROM 'R'
  DISPLAY NOTITLE (FC=*)

```

```

NAME FIRST-NAME CITY (FC=- UC=-) /
ADDRESS-LINE(1) TELEPHONE

SKIP 1
END-READ
END

```

Output of Program DISPLX15:

```

****NAME**** *FIRST-NAME* ----CITY---- =====TELEPHONE=====
**ADDRESS**
***AREA*** *TELEPHONE**
***CODE***
%%%%%%%%%%%%% %%%%%%%%%%%%%% ----- %%%%%%%%%%%%%% %%%%%%%%%%%%%%

RACKMANN      MARIAN      FRANKFURT  069          375849
              FINKENSTR. 1

RAMAMOORTHY   TY           SEPULVEDA  209          175-1885
              12018 BROOKS

RAMAMOORTHY   TIMMIE       SEATTLE    206          151-4673
              921-178TH PL

```

**DISPLX16 - DISPLAY statement (with '/', 'text', 'text/text')**

```

** Example 'DISPLX16': DISPLAY (with '/', 'text', 'text/text')
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 FIRST-NAME
  2 NAME
  2 ADDRESS-LINE (1)
  2 CITY
  2 TELEPHONE
  3 AREA-CODE
  3 PHONE
END-DEFINE
*
READ (5) EMPLOY-VIEW BY NAME STARTING FROM 'E'
  DISPLAY NOTITLE
  '/'          NAME          (AL=12) /* suppressed header
  'FIRST/NAME' FIRST-NAME (AL=10) /* two-line user-defined header
  'ADDRESS'    CITY /        /* user-defined header
  ' '          ADDRESS-LINE(1) /* 'blank' header
              TELEPHONE (HC=L) /* default header

SKIP 1
END-READ
END

```

Output of Program DISPLX16:

```

              FIRST          ADDRESS          TELEPHONE
              NAME
              AREA    TELEPHONE
              CODE
-----
EAVES      TREVOR      DERBY          0332    657623
              17 HARTON ROAD

ECKERT     KARL        OBERRAMSTADT  06154   99722
              FORSTWEG 22

```

```

ECKHARDT      RICHARD      DARMSTADT
                BRESLAUERPL. 4

EDMUNDSON     LES          TULSA          918      945-4916
                2415 ALSOP CT.

EGGERT        HERMANN     STUTTGART      0711     981237
                RABENGASSE 8
    
```

## Field-Output-Relevant Parameters

The following examples are referenced in the section *Parameters to Influence the Output of Fields*.

They are provided to demonstrate the use of the parameters LC, IC, TC, AL, NL, IS, ZP and ES, and the SUSPEND IDENTICAL SUPPRESS statement:

### DISPLX17 - DISPLAY statement (with NL, AL, IC, LC, TC)

```

** Example 'DISPLX17': DISPLAY (with NL, AL, IC, LC, TC)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 FIRST-NAME
  2 NAME
  2 SALARY (1)
  2 BONUS (1,1)
END-DEFINE
*
READ (3) EMPLOY-VIEW BY NAME STARTING FROM 'JONES'
  DISPLAY NOTITLE (IS=ON NL=15)
    NAME
    '- ' '=' FIRST-NAME (AL=12)
    'ANNUAL SALARY' SALARY(1) (LC=USD TC=.00) /
    '+ BONSES' BONUS(1,1) (IC='+ ' TC=.00)
  SKIP 1
END-READ
END
    
```

Output of Program DISPLX17:

NAME	FIRST-NAME	ANNUAL SALARY + BONSES
JONES	- VIRGINIA	USD 46000.00 + 9000.00
	- MARSHA	USD 50000.00 + 0.00
	- ROBERT	USD 31000.00 + 0.00



## DISPLX18 - DISPLAY statement (using default settings for SF, AL, UC, LC, IC, TC and compare with DISPLX19)

```

** Example 'DISPLX18': DISPLAY (using default settings for SF, AL, UC,
**                          LC, IC, TC and compare with DISPLX19)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 CITY
  2 SALARY (1)
  2 BONUS (1,1)
END-DEFINE
*
FIND (6) EMPLOY-VIEW WITH CITY = 'CHAPEL HILL'
  DISPLAY NAME FIRST-NAME SALARY(1) BONUS(1,1)
END-FIND
END

```

### Output of Program DISPLX18:

```

Page      1                                04-12-13  14:20:48

      NAME                FIRST-NAME          ANNUAL      BONUS
                        FIRST-NAME          SALARY
-----
KESSLER                CLARE                41000        0
ADKINSON               DAVID                24000        0
GEE                    TOMMIE               39500        0
HERZOG                 JOHN                 31500        0
QUILLION               TIMOTHY              30500        0
CUMMINGS               PUALA                41000        1500

```

## DISPLX19 - DISPLAY statement (with SF, AL, LC, IC, TC and compare with DISPLX18)

```

** Example 'DISPLX19': DISPLAY (with SF, AL, LC, IC, TC and compare
**                          with DISPLX19)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 CITY
  2 SALARY (1)
  2 BONUS (1,1)
END-DEFINE
*
FORMAT SF=3 AL=15 UC==
*
FIND (6) EMPLOY-VIEW WITH CITY = 'CHAPEL HILL'
  DISPLAY (NL=10)
    NAME
    FIRST-NAME (LC='- ' UC=-)
    SALARY (1) (LC=USD)
    BONUS (1,1) (IC='*** ' TC=' ***')
END-FIND
END

```

Output of Program DISPLX19:

Page 1

04-12-13 14:21:57

NAME	FIRST-NAME	ANNUAL SALARY	BONUS
KESSLER	- CLARE	USD 41000	*** 0 ***
ADKINSON	- DAVID	USD 24000	*** 0 ***
GEE	- TOMMIE	USD 39500	*** 0 ***
HERZOG	- JOHN	USD 31500	*** 0 ***
QUILLION	- TIMOTHY	USD 30500	*** 0 ***
CUMMINGS	- PUALA	USD 41000	*** 1500 ***

**SUSPEX01 - SUSPEND IDENTICAL SUPPRESS statement (in conjunction with parameters IS, ES, ZP in DISPLAY)**

```

** Example 'SUSPEX01': SUSPEND IDENTICAL SUPPRESS (in conjunction with
** parameters IS, ES, ZP in DISPLAY)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 CITY
1 VEH-VIEW VIEW OF VEHICLES
  2 PERSONNEL-ID
  2 MAKE
END-DEFINE
*
LIMIT 15
RD. READ EMPLOY-VIEW BY NAME STARTING FROM 'JONES'
  SUSPEND IDENTICAL SUPPRESS
  FD. FIND VEH-VIEW WITH PERSONNEL-ID = PERSONNEL-ID (RD.)
  IF NO RECORDS FOUND
    MOVE '*****' TO MAKE
  END-NOREC
  DISPLAY NOTITLE (ES=OFF IS=ON ZP=ON AL=15)
    NAME (RD.)
    FIRST-NAME (RD.)
    MAKE (FD.) (IS=OFF)
  END-FIND
END-READ
END

```

Output of Program SUSPEX01:

NAME	FIRST-NAME	MAKE
JONES	VIRGINIA	CHRYSLER
JONES	MARSHA	CHRYSLER
JONES	ROBERT	CHRYSLER
JONES	LILLY	GENERAL MOTORS
JONES		FORD
JONES		MG
JONES	EDWARD	GENERAL MOTORS
JONES	MARTHA	GENERAL MOTORS
JONES	LAUREL	GENERAL MOTORS

JONES	KEVIN	DATSUN
JONES	GREGORY	FORD
JOPER	MANFRED	*****
JOUSSELIN	DANIEL	RENAULT
JUBE	GABRIEL	*****
JUNG	ERNST	*****
JUNKIN	JEREMY	*****
KAISER	REINER	*****

**SUSPEX02 - SUSPEND IDENTICAL SUPPRESS statement (in conjunction with parameters IS, ES, ZP in DISPLAY) Identical to SUSPEX01, but with IS=OFF.**

```

** Example 'SUSPEX02': SUSPEND IDENTICAL SUPPRESS (in conjunction with
**                      parameters IS, ES, ZP in DISPLAY)
**                      Identical to SUSPEX01, but with IS=OFF.
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 CITY
1 VEH-VIEW VIEW OF VEHICLES
  2 PERSONNEL-ID
  2 MAKE
END-DEFINE
*
LIMIT 15
RD. READ EMPLOY-VIEW BY NAME STARTING FROM 'JONES'
  SUSPEND IDENTICAL SUPPRESS
  FD. FIND VEH-VIEW WITH PERSONNEL-ID = PERSONNEL-ID (RD.)
    IF NO RECORDS FOUND
      MOVE '*****' TO MAKE
    END-NOREC
  DISPLAY NOTITLE (ES=OFF IS=OFF ZP=ON AL=15)
    NAME          (RD.)
    FIRST-NAME    (RD.)
    MAKE          (FD.) (IS=OFF)
  END-FIND
END-READ
END

```

Output of Program SUSPEX02:

NAME	FIRST-NAME	MAKE
-----	-----	-----
JONES	VIRGINIA	CHRYSLER
JONES	MARSHA	CHRYSLER
JONES	MARSHA	CHRYSLER
JONES	ROBERT	GENERAL MOTORS
JONES	LILLY	FORD
JONES	LILLY	MG
JONES	EDWARD	GENERAL MOTORS
JONES	MARTHA	GENERAL MOTORS
JONES	LAUREL	GENERAL MOTORS
JONES	KEVIN	DATSUN
JONES	GREGORY	FORD
JOPER	MANFRED	*****
JOUSSELIN	DANIEL	RENAULT

```
JUBE           GABRIEL           *****
JUNG           ERNST             *****
JUNKIN         JEREMY            *****
KAISER         REINER            *****
```

### COMPRX03 - COMPRESS statement

```
** Example 'COMPRX03': COMPRESS (using parameters LC and TC)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 SALARY      (1)
  2 CURR-CODE   (1)
  2 LEAVE-DUE
  2 NAME
  2 FIRST-NAME
  2 JOB-TITLE
*
1 #SALARY      (N9)
1 #FULL-SALARY (A25)
1 #VACATION    (A11)
END-DEFINE
*
READ (3) EMPLOY-VIEW WITH CITY = 'BOSTON'
  MOVE SALARY(1) TO #SALARY
  COMPRESS 'SALARY :' CURR-CODE(1) #SALARY INTO #FULL-SALARY
  COMPRESS 'VACATION:' LEAVE-DUE      INTO #VACATION
  /*
  DISPLAY NOTITLE NAME FIRST-NAME
           'JOB DESCRIPTION' JOB-TITLE (LC='JOB      : ') /
           '/'                #FULL-SALARY                /
           '/'                #VACATION (TC='DAYS' )
  SKIP 1
END-READ
END
```

#### Output of Program COMPRX03:

NAME	FIRST-NAME	JOB DESCRIPTION
SHAW	LESLIE	JOB      : SECRETARY SALARY   : USD 18000 VACATION: 2DAYS
STANWOOD	VERNON	JOB      : PROGRAMMER SALARY   : USD 31000 VACATION: 1DAYS
CREMER	WALT	JOB      : SECRETARY SALARY   : USD 20000 VACATION: 3DAYS

## Edit Masks

The following examples are referenced in the section *Edit Masks - EM Parameter*.

### EDITMX03 - Edit mask (different EM for alpha-numeric fields)

```

** Example 'EDITMX03': Edit mask (different EM for alpha-numeric fields)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 CITY
  2 SALARY(1)
END-DEFINE
*
LIMIT 3
READ EMPLOY-VIEW BY PERSONNEL-ID FROM '20018000'
      WHERE SALARY(1) = 28000 THRU 30000
      DISPLAY 'N A M E'   NAME           (EM=X^X^X^X^X^X^X^X^X^X^X^X^X^X^X^X^X) /
      'NAME HEX'   NAME           (EM=H^H^H^H^H^H^H^H^H^H^H^H^H^H^H^H)
      FIRST-NAME   (EM=' - 'X(15)*)
      CITY         (EM=X..X(10))

      SKIP 1
END-READ
END
    
```

Output of Program EDITMX03:

Page 1 04-12-13 14:26:57

N A M E NAME HEX	FIRST-NAME	CITY
L O R I E D3 D6 D9 C9 C5 40 40 40 40 40 40	- JEAN-PAUL	* C..LEVELAND
H A L L C8 C1 D3 D3 40 40 40 40 40 40 40	- ARTHUR	* A..NN ARBER
V A S W A N I E5 C1 E2 E6 C1 D5 C9 40 40 40 40	- TOMMIE	* M..ONTERREY

### EDITMX04 - Edit mask (different EM for numeric fields)

```

** Example 'EDITMX04': Edit mask (different EM for numeric fields)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 FIRST-NAME
  2 NAME
  2 SALARY (1)
  2 BONUS (1,1)
  2 LEAVE-DUE
END-DEFINE
*
LIMIT 2
READ EMPLOY-VIEW BY PERSONNEL-ID = '20018000'
      WHERE SALARY(1) = 28000 THRU 30000
      DISPLAY (SF=4)
      'N A M E'   NAME
      'SALARY'   SALARY(1) (EM=*USD^ZZZ,999)
      'BONUS (ZZ)' BONUS(1,1) (EM=S*ZZZ,999) /
    
```

```

'BONUS (Z9)' BONUS(1,1) (EM=SZ99,999+) /
'->' '=' BONUS(1,1) (EM=-999,999)
'VAC/DUE' LEAVE-DUE (EM=+999)

SKIP 1
END-READ
END

```

Output of Program EDITMX04:

```

Page      1                                04-12-13  14:27:43

      N A M E          SALARY      BONUS (ZZ)      VAC
                                BONUS (Z9)      DUE
                                BONUS
-----
LORIE          USD *28,000      +**4,000      +13
                                + 04,000+
                                -> 004,000

HALL           USD *30,000      +**5,000      +14
                                + 05,000+
                                -> 005,000

```

**EDITMX05 - Edit mask (EM for date and time system variables)**

```

** Example 'EDITMX05': Edit mask (EM for date and time system variables)
*****
WRITE NOTITLE //
' DATE INTERNAL : ' *DATX (DF=L) /
'                : ' *DATX (EM=N(9)' 'ZW.'WEEK 'YYYY) /
'                : ' *DATX (EM=ZZJ'.DAY 'YYYY) /
'   ROMAN       : ' *DATX (EM=R) /
'   AMERICAN    : ' *DATX (EM=MM/DD/YYYY)      12X 'OR ' *DAT4U /
'   JULIAN      : ' *DATX (EM=YYYYJJJ)        15X 'OR ' *DAT4J /
'   GREGORIAN   : ' *DATX (EM=ZD.' 'L(10)' 'YYYY) 5X 'OR ' *DATG ///
' TIME INTERNAL : ' *TIMX                      14X 'OR ' *TIME /
'                : ' *TIMX (EM=HH.II.SS.T) /
'                : ' *TIMX (EM=HH.II.SS' 'AP) /
'                : ' *TIMX (EM=HH)
END

```

Output of Program EDITMX05:

```

DATE INTERNAL : 2004-12-13
                : Monday 51.WEEK 2004
                : 348.DAY 2004
   ROMAN       : MMIV
   AMERICAN    : 12/13/2004      OR   12/13/2004
   JULIAN      : 2004348         OR   2004348
   GREGORIAN   : 13.December2004 OR   13December 2004

TIME INTERNAL  : 14:28:49          OR   14:28:49.1
                : 14.28.49.1
                : 02.28.49 PM
                : 14

```

# DISPLAY VERT with WRITE Statement

## WRITEX10 - WRITE statement (with nT, T\*field and P\*field)

```

** Example 'WRITEX10': WRITE (with nT, T*field and P*field)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 JOB-TITLE
  2 NAME
  2 SALARY (1)
  2 BONUS (1,1)
END-DEFINE
*
READ (3) EMPLOY-VIEW WITH JOB-TITLE FROM 'SALES PERSON'
  DISPLAY NOTITLE NAME 30T JOB-TITLE
    VERT AS 'SALARY/BONUS' SALARY(1) BONUS(1,1)
  AT BREAK OF JOB-TITLE
    WRITE 20T 'AVERAGE' T*JOB-TITLE OLD(JOB-TITLE) (AL=15)
      '(SAL)' P*SALARY AVER(SALARY(1)) /
      46T '(BON)' P*BONUS AVER(BONUS(1,1)) /
  END-BREAK
  SKIP 1
END-READ
END

```

Output of Program WRITEX10:

NAME	CURRENT POSITION	SALARY BONUS
SAMUELSON	SALES PERSON	32000 6000
PAPAYANOPOULOS	SALES PERSON	34000 7000
HELL	SALES PERSON	38000 9000
AVERAGE	SALES PERSON (SAL)	34666
	(BON)	7333

## AT BREAK Statement

The following example is referenced in the section *Control Breaks*.

## ATBREX06 - AT BREAK OF statement (comparing NMIN, NAVER, NCOUNT with MIN, AVER, COUNT)

```

** Example 'ATBREX06': AT BREAK OF (comparing NMIN, NAVER, NCOUNT with
**
**
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 SALARY (1:2)
END-DEFINE

```

```

*
WRITE TITLE '-- SALARY STATISTICS BY CITY --' /
*
READ (2) EMPLOY-VIEW WITH CITY = 'NEW YORK'
  DISPLAY CITY 'SALARY (1)' SALARY(1) 15X 'SALARY (2)' SALARY(2)
  AT BREAK OF CITY
  WRITE /
    14T 'S A L A R Y   (1)'           39T 'S A L A R Y   (2)'           /
    13T '-   MIN:' MIN(SALARY(1))    38T '-   MIN:' MIN(SALARY(2))    /
    13T '-  AVER:' AVER(SALARY(1))   38T '-  AVER:' AVER(SALARY(2))   /
    16T COUNT(SALARY(1)) 'RECORDS'  41T COUNT(SALARY(2)) 'RECORDS' //
    13T '-  NMIN:' NMIN(SALARY(1))  38T '-  NMIN:' NMIN(SALARY(2))  /
    13T '- NAVER:' NAVER(SALARY(1)) 38T '- NAVER:' NAVER(SALARY(2)) /
    16T NCOUNT(SALARY(1)) 'RECORDS' 41T NCOUNT(SALARY(2)) 'RECORDS'
  END-BREAK
END-READ
END
  
```

Output of Program ATBREX06:

```

-- SALARY STATISTICS BY CITY --

-----
CITY          SALARY (1)          SALARY (2)
-----
NEW YORK          17000          16100
NEW YORK          38000          34900

          S A L A R Y   (1)          S A L A R Y   (2)
-   MIN:          17000          -   MIN:          16100
-  AVER:          27500          -  AVER:          25500
          2 RECORDS          2 RECORDS

-   NMIN:          17000          -   NMIN:          16100
- NAVER:          27500          - NAVER:          25500
          2 RECORDS          2 RECORDS
  
```

## COMPUTE, MOVE and COMPRESS Statements

The following examples are referenced in the section *Data Computation*.

### WRITEX11 - WRITE statement (with nX, n/n and COMPRESS)

```

** Example 'WRITEX11': WRITE (with nX, n/n and COMPRESS)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 SALARY      (1)
  2 FIRST-NAME
  2 NAME
  2 CITY
  2 ZIP
  2 CURR-CODE  (1)
  2 JOB-TITLE
  2 LEAVE-DUE
  2 ADDRESS-LINE (1)
*
1 #SALARY      (A8)
1 #FULL-NAME   (A25)
  
```



```

1 #FULL-CITY      (A25)
1 #FULL-SALARY   (A25)
1 #VACATION      (A16)
END-DEFINE
*
READ (3) EMPLOY-VIEW LOGICAL BY PERSONNEL-ID = '2001800'
  MOVE SALARY(1) TO #SALARY
  COMPRESS FIRST-NAME NAME                INTO #FULL-NAME
  COMPRESS ZIP      CITY                  INTO #FULL-CITY
  COMPRESS 'SALARY  :'  CURR-CODE(1) #SALARY INTO #FULL-SALARY
  COMPRESS 'VACATION:'  LEAVE-DUE      'DAYS' INTO #VACATION
/*
  DISPLAY NOTITLE 'NAME AND ADDRESS' NAME
                5X 'PERS-NO.'        PERSONNEL-ID
                3X 'JOB TITLE'       JOB-TITLE (LC='JOB      : ')
  WRITE  1/5 #FULL-NAME      1/37 #FULL-SALARY
        2/5 ADDRESS-LINE(1)  2/37 #VACATION
        3/5 #FULL-CITY
  SKIP 1
END-READ
END

```

Output of Program WRITEX11:

NAME AND ADDRESS	PERS-NO.	JOB TITLE
FARRIS JACKIE FARRIS 918 ELM STREET 32306 TALLAHASSEE	20018000	JOB      : PROGRAMMER SALARY   : USD 30500 VACATION: 10 DAY
EVANS JO EVANS 1058 REDSTONE LANE 68508 LINCOLN	20018100	JOB      : PROGRAMMER SALARY   : USD 31000 VACATION: 11 DAY
HERZOG JOHN HERZOG 255 ZANG STREET #253 27514 CHAPEL HILL	20018200	JOB      : PROGRAMMER SALARY   : USD 31500 VACATION: 12 DAY

**IFX03 - IF statement**

```

** Example 'IFX03': IF
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 CITY
  2 BONUS (1,1)
  2 SALARY (1)
*
1 #INCOME (N9)
1 #TEXT   (A26)
END-DEFINE
*
WRITE TITLE '-- DISTRIBUTION OF CATALOGS I AND II --' /
*
READ (3) EMPLOY-VIEW BY CITY = 'SAN FRANCISCO'
  COMPUTE #INCOME = BONUS(1,1) + SALARY(1)
/*

```

```

IF #INCOME > 40000
  MOVE 'CATALOGS I AND II' TO #TEXT
ELSE
  MOVE 'CATALOG I'          TO #TEXT
END-IF
/*
DISPLAY NAME 5X 'SALARY' SALARY(1) / BONUS(1,1)
WRITE T*SALARY '-'(10) /
      16X 'INCOME:' T*SALARY #INCOME 3X #TEXT /
      16X '='(19)
SKIP 1
END-READ
END

```

Output of Program IFX03:

```

-- DISTRIBUTION OF CATALOGS I AND II --

      NAME                SALARY
                        BONUS
-----
COLVILLE JR                56000
                        0
                        -----
                        INCOME:    56000   CATALOGS I AND II
                        =====

RICHMOND                    9150
                        0
                        -----
                        INCOME:    9150   CATALOG I
                        =====

MONKTON                     13500
                        600
                        -----
                        INCOME:    14100  CATALOG I
                        =====

```

**COMPRX03 - COMPRESS statement (using parameters LC and TC)**

```

** Example 'COMPRX03': COMPRESS (using parameters LC and TC)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 SALARY      (1)
  2 CURR-CODE  (1)
  2 LEAVE-DUE
  2 NAME
  2 FIRST-NAME
  2 JOB-TITLE
*
1 #SALARY      (N9)
1 #FULL-SALARY (A25)
1 #VACATION   (A11)
END-DEFINE
*
READ (3) EMPLOY-VIEW WITH CITY = 'BOSTON'
  MOVE SALARY(1) TO #SALARY
  COMPRESS 'SALARY  :' CURR-CODE(1) #SALARY INTO #FULL-SALARY

```

```

COMPRESS 'VACATION:' LEAVE-DUE          INTO #VACATION
/*
DISPLAY NOTITLE NAME FIRST-NAME
          'JOB DESCRIPTION' JOB-TITLE (LC='JOB      : ' ) /
          '/'                #FULL-SALARY                /
          '/'                #VACATION (TC='DAYS' )
SKIP 1
END-READ
END

```

Output of Program COMPRX03:

NAME	FIRST-NAME	JOB DESCRIPTION
SHAW	LESLIE	JOB      : SECRETARY SALARY   : USD 18000 VACATION: 2DAYS
STANWOOD	VERNON	JOB      : PROGRAMMER SALARY   : USD 31000 VACATION: 1DAYS
CREMER	WALT	JOB      : SECRETARY SALARY   : USD 20000 VACATION: 3DAYS

## System Variables

The following examples are referenced in the section *System Variables and System Functions*.

### EDITMX05 - Edit mask (EM for date and time system variables)

```

** Example 'EDITMX05': Edit mask (EM for date and time system variables)
*****
WRITE NOTITLE //
  'DATE INTERNAL : ' *DATX (DF=L) /
  '              : ' *DATX (EM=N(9)' 'ZW.'WEEK 'YYYY) /
  '              : ' *DATX (EM=ZZJ'.DAY 'YYYY) /
  '    ROMAN     : ' *DATX (EM=R) /
  '    AMERICAN  : ' *DATX (EM=MM/DD/YYYY)      12X 'OR ' *DAT4U /
  '    JULIAN    : ' *DATX (EM=YYYYJJJ)        15X 'OR ' *DAT4J /
  '    GREGORIAN: ' *DATX (EM=ZD.' 'L(10)' 'YYYY) 5X 'OR ' *DATG ///
  'TIME INTERNAL : ' *TIMX                      14X 'OR ' *TIME /
  '              : ' *TIMX (EM=HH.II.SS.T) /
  '              : ' *TIMX (EM=HH.II.SS' 'AP) /
  '              : ' *TIMX (EM=HH)
END

```

Output of Program EDITMX05:

```

DATE INTERNAL : 2004-12-13
              : Monday 51.WEEK 2004
              : 348.DAY 2004
    ROMAN     : MMIV
    AMERICAN  : 12/13/2004      OR  12/13/2004
    JULIAN    : 2004348        OR  2004348
    GREGORIAN: 13.December2004 OR  13December 2004

```

```

TIME INTERNAL : 14:36:58          OR   14:36:58.8
              : 14.36.58.8
              : 02.36.58 PM
              : 14
    
```

**READX04 - READ statement (in combination with FIND and the system variables \*NUMBER and \*COUNTER)**

```

** Example 'READX04': READ (in combination with FIND and the system
**                      variables *NUMBER and *COUNTER)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 FIRST-NAME
1 VEHIC-VIEW VIEW OF VEHICLES
  2 PERSONNEL-ID
  2 MAKE
END-DEFINE
*
LIMIT 10
RD. READ EMPLOY-VIEW BY NAME STARTING FROM 'JONES'
  FD. FIND VEHIC-VIEW WITH PERSONNEL-ID = PERSONNEL-ID (RD.)
    IF NO RECORDS FOUND
      ENTER
    END-NOREC
  /*
  DISPLAY NOTITLE
    *COUNTER (RD.)(NL=8) NAME           (AL=15) FIRST-NAME (AL=10)
    *NUMBER  (FD.)(NL=8) *COUNTER (FD.)(NL=8) MAKE
  END-FIND
END-READ
END
    
```

Output of Program READX04:

CNT	NAME	FIRST-NAME	NMBR	CNT	MAKE
1	JONES	VIRGINIA	1	1	CHRYSLER
2	JONES	MARSHA	2	1	CHRYSLER
2	JONES	MARSHA	2	2	CHRYSLER
3	JONES	ROBERT	1	1	GENERAL MOTORS
4	JONES	LILLY	2	1	FORD
4	JONES	LILLY	2	2	MG
5	JONES	EDWARD	1	1	GENERAL MOTORS
6	JONES	MARTHA	1	1	GENERAL MOTORS
7	JONES	LAUREL	1	1	GENERAL MOTORS
8	JONES	KEVIN	1	1	DATSUN
9	JONES	GREGORY	1	1	FORD
10	JOPER	MANFRED	0	0	

**WTITLX01 - WRITE TITLE statement (with \*PAGE-NUMBER)**

```

** Example 'WTITLX01': WRITE TITLE (with *PAGE-NUMBER)
*****
DEFINE DATA LOCAL
1 VEHIC-VIEW VIEW OF VEHICLES
  2 MAKE
  2 YEAR
  2 MAINT-COST (1)
END-DEFINE
*
LIMIT 5
*
READ VEHIC-VIEW
END-ALL
SORT BY YEAR USING MAKE MAINT-COST (1)
  DISPLAY NOTITLE YEAR MAKE MAINT-COST (1)
  AT BREAK OF YEAR
    MOVE 1 TO *PAGE-NUMBER
    NEWPAGE
  END-BREAK
/*
  WRITE TITLE LEFT JUSTIFIED
    'YEAR:' YEAR 15X 'PAGE' *PAGE-NUMBER
END-SORT
END

```

Output of Program WTITLX01:

```

YEAR:  1980                PAGE      1
YEAR      MAKE              MAINT-COST
-----
1980 RENAULT                20000
1980 RENAULT                20000
1980 PEUGEOT                20000

```

**System Functions**

The following examples are referenced in the section *System Variables and System Functions*.

**ATBREX06 - AT BREAK OF statement (comparing NMIN, NAVER, NCOUNT with MIN, AVER, COUNT)**

```

** Example 'ATBREX06': AT BREAK OF (comparing NMIN, NAVER, NCOUNT with
**                               MIN, AVER, COUNT)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 SALARY (1:2)
END-DEFINE
*
WRITE TITLE '-- SALARY STATISTICS BY CITY --' /
*
READ (2) EMPLOY-VIEW WITH CITY = 'NEW YORK'
  DISPLAY CITY 'SALARY (1)' SALARY(1) 15X 'SALARY (2)' SALARY(2)
  AT BREAK OF CITY
    WRITE /
      14T 'S A L A R Y   (1)'           39T 'S A L A R Y   (2)'           /

```

```

13T '- MIN:' MIN(SALARY(1)) 38T '- MIN:' MIN(SALARY(2)) /
13T '- AVER:' AVER(SALARY(1)) 38T '- AVER:' AVER(SALARY(2)) /
16T COUNT(SALARY(1)) 'RECORDS' 41T COUNT(SALARY(2)) 'RECORDS' //
13T '- NMIN:' NMIN(SALARY(1)) 38T '- NMIN:' NMIN(SALARY(2)) /
13T '- NAVER:' NAVER(SALARY(1)) 38T '- NAVER:' NAVER(SALARY(2)) /
16T NCOUNT(SALARY(1)) 'RECORDS' 41T NCOUNT(SALARY(2)) 'RECORDS'
END-BREAK
END-READ
END

```

Output of Program ATBREX06:

```

-- SALARY STATISTICS BY CITY --

```

CITY	SALARY (1)	SALARY (2)
NEW YORK	17000	16100
NEW YORK	38000	34900

  

S A L A R Y (1)		S A L A R Y (2)	
- MIN:	17000	- MIN:	16100
- AVER:	27500	- AVER:	25500
	2 RECORDS		2 RECORDS
- NMIN:	17000	- NMIN:	16100
- NAVER:	27500	- NAVER:	25500
	2 RECORDS		2 RECORDS

**ATENPX01 - AT END OF PAGE statement (with system function available via GIVE SYSTEM FUNCTIONS in DISPLAY)**

```

** Example 'ATENPX01': AT END OF PAGE (with system function available
** via GIVE SYSTEM FUNCTIONS in DISPLAY)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
2 PERSONNEL-ID
2 NAME
2 JOB-TITLE
2 SALARY (1)
END-DEFINE
*
READ (10) EMPLOY-VIEW BY PERSONNEL-ID = '20017000'
DISPLAY NOTITLE GIVE SYSTEM FUNCTIONS
NAME JOB-TITLE 'SALARY' SALARY(1)
/*
AT END OF PAGE
WRITE / 24T 'AVERAGE SALARY: ...' AVER(SALARY(1))
END-ENDPAGE
END-READ
END

```

Output of Program ATENPX01:

NAME	CURRENT POSITION	SALARY
CREMER	ANALYST	34000
MARKUSH	TRAINEE	22000

GEE	MANAGER	39500
KUNEY	DBA	40200
NEEDHAM	PROGRAMMER	32500
JACKSON	PROGRAMMER	33000
PIETSCH	SECRETARY	22000
PAUL	SECRETARY	23000
HERZOG	MANAGER	48500
DEKKER	DBA	48000
	AVERAGE SALARY: ...	34270