

Page Titles, Page Breaks, Blank Lines

This chapter describes various ways of controlling page breaks in a report, the output of page titles at the top of each report page and the generation of empty lines in an output report.

The following topics are covered:

- Default Page Title
 - Suppress Page Title - NOTITLE Option
 - Define Your Own Page Title - WRITE TITLE Statement
 - Logical Page and Physical Page
 - Page Size - PS Parameter
 - Page Advance
 - New Page with Title
 - Page Trailer - WRITE TRAILER Statement
 - Generating Blank Lines - SKIP Statement
 - AT TOP OF PAGE Statement
 - AT END OF PAGE Statement
 - Further Example
-

Default Page Title

For each page output via a DISPLAY or WRITE statement, Natural automatically generates a single default title line. This title line contains the page number, the date and the time of day.

Example:

```
WRITE 'HELLO'  
END
```

The above program produces the following output with default page title:

```
Page          1                               04-12-14  13:19:33  
  
HELLO
```

Suppress Page Title - NOTITLE Option

If you wish your report to be output without page titles, you add the keyword NOTITLE to the statement DISPLAY or WRITE.

Example - DISPLAY with NOTITLE:

```

** Example 'DISPLX20': DISPLAY (with NOTITLE)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 NAME
  2 FIRST-NAME
END-DEFINE
*
READ (5) EMPLOY-VIEW BY CITY FROM 'BOSTON'
  DISPLAY NOTITLE NAME FIRST-NAME CITY
END-READ
END

```

Output of Program DISPLX20:

NAME	FIRST-NAME	CITY
SHAW	LESLIE	BOSTON
STANWOOD	VERNON	BOSTON
CREMER	WALT	BOSTON
PERREAULT	BRENDA	BOSTON
COHEN	JOHN	BOSTON

Example - WRITE with NOTITLE:

```

WRITE NOTITLE 'HELLO'
END

```

The above program produces the following output without page title:

HELLO

Define Your Own Page Title - WRITE TITLE Statement

If you wish a page title of your own to be output instead of the Natural default page title, you use the statement `WRITE TITLE`.

The following topics are covered below:

- Specifying Text for Your Title
- Specifying Empty Lines after the Title
- Title Justification and/or Underlining
- Title with Page Number

Specifying Text for Your Title

With the statement `WRITE TITLE`, you specify the text for your title (in apostrophes).

```
WRITE TITLE 'THIS IS MY PAGE TITLE'
WRITE 'HELLO'
END
```

The above program produces the following output:

```
                THIS IS MY PAGE TITLE

HELLO
```

Specifying Empty Lines after the Title

With the `SKIP` option of the `WRITE TITLE` statement, you can specify the number of empty lines to be output immediately below the title line. After the keyword `SKIP`, you specify the number of empty lines to be inserted.

```
WRITE TITLE 'THIS IS MY PAGE TITLE' SKIP 2
WRITE 'HELLO'
END
```

The above program produces the following output:

```
                THIS IS MY PAGE TITLE

                HELLO
```

`SKIP` is not only available as part of the `WRITE TITLE` statement, but also as a stand-alone statement.

Title Justification and/or Underlining

By default, the page title is centered on the page and not underlined.

The `WRITE TITLE` statement provides the following options which can be used independent of each other:

Option	Effect
LEFT JUSTIFIED	Causes the page trailer to be displayed left-justified.
UNDERLINED	Causes the title to be displayed underlined. The underlining runs the width of the line size (see also Natural profile and session parameter <code>LS</code>). By default, titles are underlined with a hyphen (-). However, with the <code>UC</code> session parameter you can specify another character to be used as underlining character (see <i>Underlining Character for Titles and Headers</i>).

The following example shows the effect of the `LEFT JUSTIFIED` and `UNDERLINED` options:

```
WRITE TITLE LEFT JUSTIFIED UNDERLINED 'THIS IS MY PAGE TITLE'
SKIP 2
WRITE 'HELLO'
END
```

The above program produces the following output:

```
THIS IS MY PAGE TITLE
```

```
HELLO
```

The WRITE TITLE statement is executed whenever a new page is initiated for the report.

Title with Page Number

In the following examples, the system variable *PAGE-NUMBER is used in conjunction with the WRITE TITLE statement to output the page number in the title line.

```
** 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	

Logical Page and Physical Page

A *logical page* is the output produced by a Natural program. A *physical page* is your terminal screen on which the output is displayed; or it may be the piece of paper on which the output is printed.

The size of the logical page is determined by the number of lines output by the Natural program.

If more lines are output than fit onto one screen, the logical page will exceed the physical screen, and the remaining lines will be displayed on the next screen.

Physical Page (Screen)

The diagram shows two overlapping rectangular areas representing pages. The top area, labeled 'Physical Page (Screen)', is a green-bordered rectangle containing a blue area with a table. The bottom area, labeled 'Logical Page', is a blue area with a table, partially overlapping the bottom of the Physical Page.

NAME	FIRST-NAME
ALHAZRED	ABDUL
BEAR	EDWARD
BROWN	HOLLIS
CARTER	RANDOLPH
DUNSON	THOMAS
HARGREAVES	ALICE
INNES	DAVID
MORESBY	KATHERINE
PERRY	ABNER
WARD	CHARLES
WILDE	IRENE
ZIMMERMANN	ROBERT

Note:

If information you wish to appear at the bottom of the screen (for example, output created by a WRITE TRAILER or AT END OF PAGE statement) is output on the next screen instead, reduce the logical page size accordingly (with the session parameter PS, which is discussed below).

Page Size - PS Parameter

With the parameter PS (Page Size for Natural Reports), you determine the maximum number of lines per (logical) page for a report.

When the number of lines specified with the PS parameter is reached, a page advance occurs (unless page advance is controlled with a NEWPAGE or EJECT statement; see *Page Advance Controlled by EJ Parameter* below).

The PS parameter can be set either at session level with the system command GLOBALS, or within a program with the following statements:

At report level:

- `FORMAT PS=nn`

At statement level:

- `DISPLAY (PS=nn)`
- `WRITE (PS=nn)`
- `WRITE TITLE (PS=nn)`
- `WRITE TRAILER (PS=nn)`
- `INPUT (PS=nn)`

Page Advance

A page advance can be triggered by one of the following methods:

- Page Advance Controlled by EJ Parameter
- Page Advance Controlled by EJECT or NEWPAGE Statements
- Eject/New Page when less than n Lines Left

These methods are discussed below.

Page Advance Controlled by EJ Parameter

With the session parameter EJ (Page Eject), you determine whether page ejects are to be performed or not. By default, EJ=ON applies, which means that page ejects will be performed as specified.

If you specify EJ=OFF, page break information will be ignored. This may be useful to save paper during test runs where page ejects are not needed.

The EJ parameter can be set at session level with the system command GLOBALS; for example:

```
GLOBALS EJ=OFF
```

The EJ parameter setting is overridden by the EJECT statement.

Page Advance Controlled by EJECT or NEWPAGE Statements

The following topics are covered below:

- Page Advance without Title/Header on Next Page
- Page Advance with End/Top-of-Page Processing

Page Advance without Title/Header on Next Page

The EJECT statement causes a page advance *without* a title or header line being generated on the next page. A new physical page is started *without* any top-of-page or end-of-page processing being performed (for example, no WRITE TRAILER or AT END OF PAGE, WRITE TITLE, AT TOP OF PAGE or *PAGE-NUMBER processing).

The EJECT statement overrides the EJ parameter setting.

Page Advance with End/Top-of-Page Processing

The NEWPAGE statement causes a page advance *with* associated end-of-page and top-of-page processing. A trailer line will be displayed, if specified. A title line, either default or user-specified, will be displayed on the new page, unless the NOTITLE option has been specified in a DISPLAY or WRITE statement (as described above).

If the NEWPAGE statement is not used, page advance is automatically controlled by the setting of the PS parameter; see *Page Size - PS Parameter* above).

Eject/New Page when less than *n* Lines Left

Both the NEWPAGE statement and the EJECT statement provide a WHEN LESS THAN *n* LINES LEFT option. With this option, you specify a number of lines (*n*). The NEWPAGE/EJECT statement will then be executed if - at the time the statement is processed - less than *n* lines are available on the current page.

Example 1:

```
FORMAT PS=55
...
NEWPAGE WHEN LESS THAN 7 LINES LEFT
...
```

In this example, the page size is set to 55 lines.

If only 6 or less lines are left on the current page at the time when the NEWPAGE statement is processed, the NEWPAGE statement is executed and a page advance occurs. If 7 or more lines are left, the NEWPAGE statement is not executed and no page advance occurs; the page advance then occurs depending on the session parameter PS (Page Size for Natural Reports), that is, after 55 lines.

Example 2:

```
** Example 'NEWPAX02': NEWPAGE (in combination with EJECT and
**                          parameter PS)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 CITY
  2 NAME
  2 JOB-TITLE
END-DEFINE
*
FORMAT PS=15
```

```

*
READ (9) EMPLOY-VIEW BY CITY STARTING FROM 'BOSTON'
  AT START OF DATA
    EJECT
    WRITE /// 20T '%' (29) /
              20T '%%' /
              20T '%%' 3X 'REPORT OF EMPLOYEES' 47T '%%' /
              20T '%%' 3X ' SORTED BY CITY ' 47T '%%' /
              20T '%%' /
              20T '%' (29) /

    NEWPAGE
  END-START
  AT BREAK OF CITY
    NEWPAGE WHEN LESS 3 LINES LEFT
  END-BREAK
  DISPLAY CITY (IS=ON) NAME JOB-TITLE
END-READ
END

```

New Page with Title

The NEWPAGE statement also provides a WITH TITLE option. If this option is not used, a default title will appear at the top of the new page or a WRITE TITLE statement or NOTITLE clause will be executed.

The WITH TITLE option of the NEWPAGE statement allows you to override these with a title of your own choice. The syntax of the WITH TITLE option is the same as for the WRITE TITLE statement.

Example:

```
NEWPAGE WITH TITLE LEFT JUSTIFIED 'PEOPLE LIVING IN BOSTON:'
```

The following program illustrates the use of the session parameter PS (Page Size for Natural Reports) and the NEWPAGE statement. Moreover, the system variable *PAGE-NUMBER is used to display the current page number.

```

** Example 'NEWPAX01': NEWPAGE
*****
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 NAME
  2 CITY
  2 DEPT
END-DEFINE
*
FORMAT PS=20
READ (5) VIEWEMP BY CITY STARTING FROM 'M'
  DISPLAY NAME 'DEPT' DEPT 'LOCATION' CITY
  AT BREAK OF CITY
    NEWPAGE WITH TITLE LEFT JUSTIFIED
      'EMPLOYEES BY CITY - PAGE:' *PAGE-NUMBER
  END-BREAK
END-READ
END

```

Output of Program NEWPAX01:

Note the position of the page breaks and the title line:

Page 1 04-11-11 14:15:54

NAME	DEPT	LOCATION
FICKEN	TECH10	MADISON
KELLOGG	TECH10	MADISON
ALEXANDER	SALE20	MADISON

Page 2:

EMPLOYEES BY CITY - PAGE: 2

NAME	DEPT	LOCATION
DE JUAN	SALE03	MADRID
DE LA MADRID	PROD01	MADRID

Page 3:

EMPLOYEES BY CITY - PAGE: 3

Page Trailer - WRITE TRAILER Statement

The following topics are covered below:

- Specifying a Page Trailer
- Considering Logical Page Size
- Page Trailer Justification and/or Underlining

Specifying a Page Trailer

The WRITE TRAILER statement is used to output text (in apostrophes) at the bottom of a page.

```
WRITE TRAILER 'THIS IS THE END OF THE PAGE'
```

The statement is executed when an end-of-page condition is detected, or as a result of a SKIP or NEWPAGE statement.

Considering Logical Page Size

As the end-of-page condition is checked only *after* an entire DISPLAY or WRITE statement has been processed, it may occur that the logical page size (that is, the number of lines output by a DISPLAY or WRITE statement) causes the physical size of the output page to be exceeded before the WRITE TRAILER statement is executed.

To ensure that a page trailer actually appears at the bottom of a physical page, you should set the logical page size (with the PS session parameter) to a value less than the physical page size.

Page Trailer Justification and/or Underlining

By default, the page trailer is displayed centered on the page and not underlined.

The WRITE TRAILER statement provides the following options which can be used independent of each other:

Option	Effect
LEFT JUSTIFIED	Causes the page trailer to be displayed left justified.
UNDERLINED	The underlining runs the width of the line size (see also Natural profile and session parameter LS). By default, titles are underlined with a hyphen (-). However, with the UC session parameter you can specify another character to be used as underlining character (see <i>Underlining Character for Titles and Headers</i>).

The following examples show the use of the LEFT JUSTIFIED and UNDERLINED options of the WRITE TRAILER statement:

Example 1:

```
WRITE TRAILER LEFT JUSTIFIED UNDERLINED 'THIS IS THE END OF THE PAGE'
```

Example 2:

```
** Example 'WTITLX02': WRITE TITLE AND WRITE TRAILER
*****
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' *DATE //
*
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
```

Generating Blank Lines - SKIP Statement

The SKIP statement is used to generate one or more blank lines in an output report.

Example 1 - SKIP in conjunction with WRITE and DISPLAY:

```
** Example 'SKIPX01': SKIP (in conjunction with WRITE and DISPLAY)
*****
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
  'PEOPLE LIVING IN SALT LAKE CITY AS OF' *DAT4E 7X
  'PAGE:' *PAGE-NUMBER
SKIP 3
*
READ (2) EMPLOY-VIEW WITH CITY = 'SALT LAKE CITY'
  DISPLAY NAME / FIRST-NAME CITY ADDRESS-LINE (1)
  SKIP 1
END-READ
END
```

Example 2 - SKIP in conjunction with DISPLAY VERT:

```
** Example 'SKIPX02': SKIP (in conjunction with DISPLAY VERT)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 CITY
  2 JOB-TITLE
END-DEFINE
*
READ (2) EMPLOY-VIEW WITH JOB-TITLE = 'SECRETARY'
  DISPLAY NOTITLE VERT
    NAME FIRST-NAME / CITY
  SKIP 3
END-READ
*
NEWPAGE
*
READ (2) EMPLOY-VIEW WITH JOB-TITLE = 'SECRETARY'
  DISPLAY NOTITLE
    NAME FIRST-NAME / CITY
  SKIP 3
END-READ
END
```

AT TOP OF PAGE Statement

The AT TOP OF PAGE statement is used to specify any processing that is to be performed whenever a new page of the report is started.

If the AT TOP OF PAGE processing produces any output, this will be output below the page title (with a skipped line in between).

By default, this output is displayed left-justified on the page.

Example:

```
** Example 'ATTOPX01': AT TOP OF PAGE
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 MAR-STAT
  2 BIRTH
  2 CITY
  2 JOB-TITLE
  2 DEPT
END-DEFINE
*
LIMIT 10
READ EMPLOY-VIEW BY PERSONNEL-ID FROM '20017000'
  DISPLAY NOTITLE (AL=10)
    NAME DEPT JOB-TITLE CITY 5X
    MAR-STAT 'DATE OF/BIRTH' BIRTH (EM=YY-MM-DD)
  /*
  AT TOP OF PAGE
    WRITE / '-BUSINESS INFORMATION-'
      26X '-PRIVATE INFORMATION-'
  END-TOPPAGE
END-READ
END
```

Output of Program ATTOPX01:

-BUSINESS INFORMATION-				-PRIVATE INFORMATION-	
NAME	DEPARTMENT CODE	CURRENT POSITION	CITY	MARITAL STATUS	DATE OF BIRTH
-----				-----	
CREMER	TECH10	ANALYST	GREENVILLE	S	70-01-01
MARKUSH	SALE00	TRAINEE	LOS ANGELE	D	79-03-14
GEE	TECH05	MANAGER	CHAPEL HIL	M	41-02-04
KUNEY	TECH10	DBA	DETROIT	S	40-02-13
NEDHAM	TECH10	PROGRAMMER	CHATTANOOG	S	55-08-05
JACKSON	TECH10	PROGRAMMER	ST LOUIS	D	70-01-01
PIETSCH	MGMT10	SECRETARY	VISTA	M	40-01-09
PAUL	MGMT10	SECRETARY	NORFOLK	S	43-07-07
HERZOG	TECH05	MANAGER	CHATTANOOG	S	52-09-16
DEKKER	TECH10	DBA	MOBILE	W	40-03-03

AT END OF PAGE Statement

The AT END OF PAGE statement is used to specify any processing that is to be performed whenever an end-of-page condition occurs.

If the AT END OF PAGE processing produces any output, this will be output after any page trailer (as specified with the WRITE TRAILER statement).

By default, this output is displayed left-justified on the page.

The same considerations described above for page trailers regarding physical and logical page sizes and the number of lines output by a DISPLAY or WRITE statement also apply to AT END OF PAGE output.

Example:

```

** 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	

Further Example

See the following example program:

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