

Seitenüberschriften, Seitenvorschübe und Leerzeilen

Dieses Kapitel beschreibt verschiedene Möglichkeiten, wie Sie den Seitenumbruch in einem Report sowie die Ausgabe von Seitenüberschriften am Anfang jeder Seite des Reports und die Erzeugung von Leerzeilen in einem Ausgabe-Report beeinflussen können.

Folgende Themen werden behandelt:

- Standard-Seitenüberschrift
 - Seitenüberschrift unterdrücken — die NOTITLE-Option
 - Eigene Seitenüberschrift definieren — das WRITE TITLE-Statement
 - Logische Seite und physische Seite
 - Seitenlänge — der PS-Parameter
 - Seitenvorschub
 - Neue Seite mit Titel
 - Seiten-Fußzeile — das WRITE TRAILER-Statement
 - Leerzeilen erzeugen — das SKIP-Statement
 - AT TOP OF PAGE-Statement
 - AT END OF PAGE-Statement
 - Weiteres Beispiel
-

Standard-Seitenüberschrift

Natural generiert für jede über ein DISPLAY- oder WRITE-Statement erzeugte Ausgabeseite automatisch eine Standard-Kopfzeile. Diese Kopfzeile enthält die Seitennummer sowie Datum und Uhrzeit.

Beispiel:

```
WRITE 'HELLO'  
END
```

Das obige Programm erzeugt folgende Ausgabe mit einer Standard-Kopfzeile:

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

Seitenüberschrift unterdrücken — die NOTITLE-Option

Falls Sie Ihren Report ohne Kopfzeile ausgeben möchten, geben Sie im DISPLAY- bzw. WRITE-Statement das Schlüsselwort NOTITLE an.

Beispiel - DISPLAY mit 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
```

Ausgabe des Programms DISPLX20:

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

Beispiel - WRITE mit NOTITLE:

```
WRITE NOTITLE 'HELLO'
END
```

Das obige Programm erzeugt folgende Ausgabe ohne Kopfzeile:

```
HELLO
```

Eigene Seitenüberschrift definieren — das WRITE TITLE-Statement

Wenn Sie statt der Natural-Standard-Kopfzeile eine eigene Kopfzeile ausgeben möchten, verwenden Sie dazu das Statement WRITE TITLE.

In diesem Abschnitt werden folgende Themen behandelt:

- Text für Ihre Überschrift angeben
- Leerzeilen nach der Überschrift angeben

- Überschriften-Ausrichtung und/oder -Unterstreichung
- Überschrift mit Seitenzahl

Text für Ihre Überschrift angeben

Mit dem Statement `WRITE TITLE` geben Sie (in Apostrophen) den Text Ihrer Kopfzeile an.

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

Mit dem obigen Programm wird die folgende Ausgabe erzeugt:

```
HELLO  
THIS IS MY PAGE TITLE
```

Leerzeilen nach der Überschrift angeben

Mit der `SKIP`-Option des `WRITE TITLE`-Statements können Sie bestimmen, wieviele Leerzeilen unter der Kopfzeile ausgegeben werden sollen. Nach dem Schlüsselwort `SKIP` geben Sie die Anzahl der gewünschten Leerzeilen an:

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

Mit dem obigen Programm wird die folgende Ausgabe erzeugt:

```
HELLO  
THIS IS MY PAGE TITLE
```

`SKIP` kann nicht nur in einem `WRITE TITLE`-Statement, sondern auch als eigenständiges Statement verwendet werden.

Überschriften-Ausrichtung und/oder -Unterstreichung

Standardmäßig wird die Kopfzeile zentriert und ohne Unterstreichung ausgegeben.

Das `WRITE TITLE`-Statement bietet Ihnen aber auch die Möglichkeit, eine Seitenüberschrift linksbündig (`LEFT JUSTIFIED`) und/oder unterstrichen (`UNDERLINED`) auszugeben.

Option	Auswirkung
LEFT JUSTIFIED	Bewirkt, dass die Überschrift linksbündig angezeigt wird.
UNDERLINED	Bewirkt, dass die Überschrift unterstrichen angezeigt wird. Das Unterstreichen legt die Breite der Zeile fest (siehe auch Natural Profil- und Session-Parameter LS). Standardmäßig werden Überschriften mit einem Bindestrich (-) unterstrichen. Mit dem Session-Parameter UC können Sie aber ein anderes Zeichen angeben, das als Zeichen zum Unterstreichen benutzt werden soll (siehe <i>Unterstreichungszeichen für Überschriften</i>).

Das folgende Beispiel zeigt die Auswirkungen der Optionen LEFT JUSTIFIED und UNDERLINED:

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

Mit dem obigen Programm wird die folgende Ausgabe erzeugt:

```
THIS IS MY PAGE TITLE
-----

HELLO
```

Das WRITE TITLE-Statement wird jedesmal ausgeführt, wenn eine neue Reportseite initiiert wird.

Überschrift mit Seitenzahl

In den folgenden Beispielen wird die Systemvariable *PAGE-NUMBER in Verbindung mit dem Statement WRITE TITLE zur Ausgabe der Seitenzahl in der Überschriftenzeile benutzt.

```
** 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
```

Ausgabe des Programms WTITLX01:

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

Logische Seite und physische Seite

Eine *logische Seite* ist die von einem Natural-Programm erzeugte Ausgabe.

Eine *physische Seite* ist Ihr Bildschirm, auf dem die Ausgabe angezeigt wird; oder es ist das Stück Papier, auf dem die Ausgabe ausgedruckt wird.

Falls mehr Zeilen ausgegeben werden als auf einen Bildschirm passen, ist die logische Seite länger als die physische Seite, und die restlichen Zeilen werden auf dem nächsten Schirm angezeigt.

Physische Seite (Bildschirm)

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

Logische Seite

Anmerkung:

Falls Informationen, die Sie unten auf dem Schirm anzeigen möchten (z.B. mit einem `WRITE TRAILER-` oder `AT END OF PAGE-` Statement erzeugte Ausgaben), erst auf dem nächsten Schirm ausgegeben werden, verkleinern Sie die logische Seitenlänge entsprechend (mit dem Session-Parameter `PS`, wie unten beschrieben).

Seitenlänge — der PS-Parameter

Mit dem Parameter `PS` (Page Size for Natural Reports) bestimmen Sie die maximale Anzahl der Zeilen einer (logischen) Ausgabeseite.

Wenn die Anzahl der mit dem PS-Parameter angegebenen Zeilen erreicht ist, dann erfolgt ein Seitenvorschub (es sei denn, der Seitenvorschub wird über ein NEWPAGE- oder ein EJECT-Statement gesteuert; siehe unten).

Der PS-Parameter kann entweder auf Session-Ebene mit dem Systemkommando GLOBALS gesetzt werden oder innerhalb eines Programms mit den folgenden Statements:

Auf Report-Ebene:

- `FORMAT PS=nn`

Auf Statement-Ebene:

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

Seitenvorschub

Ein Seitenvorschub kann durch eine der folgenden Methoden erreicht werden:

- Seitenvorschub durch EJ-Parameter
- Seitenvorschub durch EJECT oder NEWPAGE-Statement
- EJECT/NEWPAGE bei weniger als *n* restlichen Zeilen auf der Seite

Diese Methoden werden im Folgenden erörtert.

Seitenvorschub durch EJ-Parameter

Mit dem Session-Parameter EJ (Page Eject) bestimmen Sie, ob Seitenvorschübe ausgeführt werden sollen oder nicht. Standardmäßig gilt EJ=ON, d.h. Seitenvorschübe werden wie angegeben ausgeführt.

Wenn Sie EJ=OFF angeben, werden Seitenvorschub-Informationen ignoriert. Dies kann bei Testläufen, bei denen Seitenumbrüche keine Rolle spielen, sinnvoll sein, um Papier zu sparen.

Der Seitenvorschub-Parameter EJ kann auf Session-Ebene mit dem Systemkommando GLOBALS gesetzt werden:

```
GLOBALS EJ=OFF
```

Die Einstellung des EJ-Parameters wird durch das EJECT-Statement überschrieben.

Seitenvorschub durch EJECT oder NEWPAGE-Statement

Folgende Themen werden behandelt:

- Seitenvorschub ohne Überschrift/Fußzeile auf der nächsten Seite
- Seitenvorschub mit Verarbeitung am Ende/Anfang der Seite

Seitenvorschub ohne Überschrift/Fußzeile auf der nächsten Seite

Das EJECT-Statement bewirkt einen Seitenvorschub, *ohne* dass auf der neuen Seite eine Kopfzeile oder Standard-Seitenüberschrift generiert wird. An Seitenanfang und Seitenende gebundene Verarbeitungen wie WRITE TRAILER oder AT END OF PAGE, WRITE TITLE, AT TOP OF PAGE oder *PAGE-NUMBER werden nicht ausgeführt.

Das EJECT-Statement hat Priorität vor dem EJ-Parameter.

Seitenvorschub mit Verarbeitung am Ende/Anfang der Seite

Das NEWPAGE-Statement hingegen bewirkt einen Seitenvorschub *mit* Ausführung der für Seitenanfang und Seitenende festgelegten Verarbeitungen. Eine Fußzeile wird ausgegeben, falls spezifiziert; eine standardmäßige oder benutzerdefinierte Kopfzeile wird auf der neuen Seite ausgegeben (es sei denn, das betreffende DISPLAY- bzw. WRITE-Statement enthält die Option NOTITLE).

Wird kein NEWPAGE-Statement verwendet, so ergibt sich der Seitenvorschub aus der mit dem Parameter PS definierten Seitenlänge (siehe *Seitenlänge - der PS-Parameter* oben).

EJECT/NEWPAGE bei weniger als n restlichen Zeilen auf der Seite

Das NEWPAGE- wie das EJECT-Statement erlauben es, eine WHEN LESS THAN n LINES LEFT-Klausel anzugeben. Mit dieser Klausel geben Sie eine Zeilenanzahl n an; das NEWPAGE- bzw. EJECT-Statement wird dann nur ausgeführt, wenn zum Zeitpunkt der Verarbeitung des Statements weniger als n Zeilen auf der aktuellen Seite zur Verfügung stehen.

Beispiel 1:

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

In diesem Beispiel ist die Seitenlänge mit 55 Zeilen angegeben.

Sind zu dem Zeitpunkt, zu dem das NEWPAGE-Statement verarbeitet wird, auf der aktuellen Seite nur noch 6 oder weniger Zeilen übrig, wird das NEWPAGE-Statement ausgeführt. Sind 7 oder mehr übrig, wird es nicht ausgeführt, und der Seitenvorschub erfolgt in Abhängigkeit vom PS-Parameter, also nach 55 Zeilen.

Beispiel 2:

```
** Example 'NEWPA02': 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 '%%'                               47T '%%' /
            20T '%%' 3X 'REPORT OF EMPLOYEES' 47T '%%' /
            20T '%%' 3X '  SORTED BY CITY ' 47T '%%' /
            20T '%%'                               47T '%%' /
            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

```

Neue Seite mit Titel

Das **NEWPAGE**-Statement bietet darüber hinaus eine **WITH TITLE**-Option. Ohne diese Option wird entweder die Standard-Kopfzeile ausgegeben oder ein **WRITE TITLE**-Statement bzw. eine **NOTITLE**-Option ausgeführt.

Mit der **WITH TITLE**-Option können Sie für einen mit **NEWPAGE** ausgelösten Seitenvorschub eine eigene Kopfzeile ausgeben, die dann Priorität vor allen anderen Seitenüberschrift-Anweisungen hat. Die Syntax der **WITH TITLE**-Klausel entspricht der des **WRITE TITLE**-Statements.

Beispiel:

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

Das folgende Beispielprogramm zeigt die Verwendung des **PS**-Parameters und des **NEWPAGE**-Statements. Außerdem wird hier die Natural-Systemvariable ***PAGE-NUMBER** verwendet, die jeweils die aktuelle Seitenzahl enthält.

```

** 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

```

Ausgabe des Programms NEWPAX01:

Beachten Sie, wann der Seitenvorschub erfolgt, sowie die Kopfzeile der neuen Seite:

```

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

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

```

Seite 2:

```

EMPLOYEES BY CITY - PAGE:      2
      NAME          DEPT      LOCATION
-----
DE JUAN        SALE03  MADRID
DE LA MADRID   PROD01  MADRID

```

Seite 3:

```

EMPLOYEES BY CITY - PAGE: 3

```

Seiten-Fußzeile — das WRITE TRAILER-Statement

Folgende Themen werden behandelt:

- Seiten-Fußzeile angeben
- Logische Seitenlänge berücksichtigen
- Ausrichtung und/oder Unterstreichung der Seiten-Fußzeile

Seiten-Fußzeile angeben

Mit dem Statement `WRITE TRAILER` können Sie einen Text in Apostrophen (') angeben, der als Fußzeile am Ende jeder Seite ausgegeben werden soll.

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

Das Statement wird ausgeführt vor einem `SKIP`- oder `NEWPAGE`-Statement oder am Ende einer logischen Seite.

Logische Seitenlänge berücksichtigen

Die Prüfung, ob das Ende einer logischen Seite erreicht ist, erfolgt erst, *nachdem* ein WRITE- oder DISPLAY-Statement vollständig ausgeführt ist. Daher kann es vorkommen, dass der Umfang einer logischen Seite (d.h. die Anzahl der mit einem DISPLAY- bzw. WRITE-Statement ausgegebenen Zeilen) eine physische Seite überschreitet, bevor das WRITE TRAILER-Statement ausgeführt wird.

Um sicherzustellen, dass die Fußzeilen jeweils am Ende einer physischen Seite erscheinen, sollten Sie die logische Seitenlänge (Session-Parameter PS) so festlegen, dass sie entsprechend kleiner als die physische Seitenlänge ist.

Ausrichtung und/oder Unterstreichung der Seiten-Fußzeile

Standardmäßig wird die Seiten-Fußzeile zentriert auf der Seite und nicht unterstrichen ausgegeben.

Das WRITE TRAILER-Statement bietet Ihnen aber auch die Möglichkeit, eine Fußzeile linksbündig (LEFT JUSTIFIED) und/oder unterstrichen (UNDERLINED) auszugeben:

Option	Auswirkung
LEFT JUSTIFIED	Bewirkt, dass die Fußzeile linksbündig angezeigt wird.
UNDERLINED	Bewirkt, dass die Fußzeile unterstrichen angezeigt wird. Das Unterstreichen erfolgt über die ganze Breite der Zeile fest (siehe auch Natural Profil- und Session-Parameter LS). Standardmäßig werden Überschriften mit einem Bindestrich (–) unterstrichen. Mit dem Session-Parameter UC können Sie aber ein anderes Zeichen angeben, das als Zeichen zum Unterstreichen benutzt werden soll (siehe <i>Unterstreichungszeichen für Überschriften</i>).

Die folgenden Beispiele zeigen die Verwendung der Optionen LEFT JUSTIFIED und UNDERLINED des WRITE TRAILER-Statements:

Beispiel 1:

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

Beispiel 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

```

Leerzeilen erzeugen — das SKIP-Statement

Das SKIP-Statement wird zum Erzeugen von einer oder mehrerer Leerzeilen in einem Ausgabe-Report benutzt.

Beispiel 1 - SKIP in Verbindung mit WRITE und 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

```

Beispiel 2 - SKIP in Verbindung mit 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

Mit dem Statement AT TOP OF PAGE können Sie eine beliebige Verarbeitung angeben, die jedesmal ausgeführt werden soll, wenn eine neue Reportseite beginnt.

Erzeugt das AT TOP OF PAGE-Statement eine Ausgabe, so wird diese unterhalb der Seiten- Kopfzeile (mit einer Leerzeile dazwischen) ausgegeben.

Standardmäßig erfolgt die Ausgabe linksbündig auf der Seite.

Beispiel:

```

** 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

```

Ausgabe des Programms 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
NEEDHAM	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

Mit dem Statement `AT END OF PAGE` können Sie eine beliebige Verarbeitung angeben, die jedesmal ausgeführt werden soll, wenn das Ende einer Reportseite erreicht wird.

Erzeugt das `AT END OF PAGE`-Statement eine Ausgabe, so wird diese unterhalb der (mit dem `WRITE TRAILER`-Statement angegebenen) Seiten-Fußzeile ausgegeben.

Standardmäßig erfolgt die Ausgabe linksbündig auf der Seite.

Dieselben Anmerkungen bezüglich logischer und physischer Seitenlängen, die für das `DISPLAY`- und `WRITE`-Statement gelten (vgl. oben), treffen auch auf das `AT END OF PAGE`-Statement zu.

Beispiel:

```
** 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
```

Ausgabe des Programms 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

Weiteres Beispiel

Siehe das folgende Beispielprogramm:

- *DISPLX21 - DISPLAY-Statement (mit Schrägstrich '/' und zum Vergleich mit WRITE)*