

WebFOCUS

WebFOCUS Quick Reference Card
Version 5 Release 2

WebFOCUS Quick Reference Reporting Handbook

How to Use This Handbook

This handbook presents a summary of reporting tools and syntax in a concise format. For complete information, consult your WebFOCUS documentation set. To help you find the information you need, this handbook includes:

- An overview of the syntax components you can use to create a basic TABLE report request.
- Detailed accounts of the options available for each syntax component in a report request.
- Information on using additional WebFOCUS reporting utilities.

Handbook Conventions

UPPERCASE	For items to be entered as shown.
lowercase	For data that you supply.
[]	For optional items.
{ }	For groups of required items; select one.
. . .	For a continuation.
.	For intervening commands.
.	
.	
underline	For default values.
punctuation	Required as shown.

Note: When a syntax display does not fit on a single line, each subsequent line is indented slightly.

Report Requirements

A data source and corresponding Master File.

WebFOCUS® Information Builders Inc. Copyright © 2003

DN4500512.0103

Table of Contents

Creating a Basic Report

Request: TABLE	3
Overview of Syntax Components	3
File Specification	4
Page Heading	4
Display Commands	5
Manipulating Display Fields With	
Prefix Operators	6
Column Formatting Options	6
Creating a Temporary Field: COMPUTE	7
Numeric Expressions	7
Special Functions	8
Conditional Expressions	8
Alphanumeric Expressions	8
EDIT Function	8
DECODE Function	8
LAST Function	8
Creating a Temporary Field Unrelated to a	
Master File	9
Sort Phrases	10
Selection Phrases	11
Calculating Trend Values and Forecasts	12
Formatting Options	13
Field Formatting: ON Field	13
Report Formatting: ON TABLE	14
Field-Based Reformatting	14
Conditional Formatting	15
Environment Options	15
Output Files	15
Effect of TABLE Syntax Elements	
on Excel PivotTables	16
Page Footing	16
Terminating Commands	16

Using Additional Reporting

Features	17
Creating a Temporary Field: DEFINE	17
Joining Data Sources	18
Using a Conditional Join	19
Combining Data Sources: MATCH	20
Concatenating Data Sources: MORE	21
Creating Financial Reports	22
Reporting Dynamically From an	
FML Hierarchy	23
Using StyleSheets	24
Identifying Report Component Subtypes	26
Linking Reports to Other Resources	
Using StyleSheets	28
Using Cascading Style Sheets	29
Configuring Your Environment: SET	30
SET Commands for StyleSheets	34
Querying Your Environment	34
WebFOCUS Reporting Tools	35

Creating a Basic Report Request: TABLE

Overview of Syntax Components

Each syntax component shown below is described in more detail on the following pages. The minimum requirements for a report request are a TABLE[F] FILE command, a display command or sort phrase, and a terminating command. All other components are optional.

File Specification	TABLE[F] FILE filename
Page Heading	[HEADING [option]] "text"
Display Commands	PRINT LIST COUNT display_field options SUM WRITE
Temporary Field Command	COMPUTE temp_field[/format] = expression;
Sort Phrases	{BY ACROSS} sort_field option
Financial Modeling Language (FML)	FOR fml option
Selection Phrases	{WHERE IF} selection_criteria
Formatting Options	ON {TABLE field} options
Environment Options	ON TABLE SET parameter value
Output Files	ON TABLE output_format options
Page Footing	FOOTING [options] "text"
StyleSheet	ON TABLE SET STYLE[SHEET] * stylesheet declarations [ENDSTYLE]
Concatenating Data	MORE FILE filename subrequest
Terminating Commands	END RUN QUIT

File Specification

The WebFOCUS reporting language is very flexible, requiring only one command to mark the beginning of a request and identify the data source.

TABLE[F] FILE filename

where:

TABLE FILE	Is a required command.
filename	Identifies the data source from which you wish to report. If the data source has already been specified with a SET FILE command, then only TABLE[F] is required.
TABLEF	Is an optional command variation to speed data retrieval if a data source is presorted for printing and requires no additional sorting.

Page Heading

A page heading is text that appears on the top of every page of the report. In general, the heading is composed of text which you supply in your report request, between double quotation marks.

[HEADING [CENTER]]

"text <[±]n <field[>] more text</n"

"text <prefix.field[>][<OX[>]]

text"

"text <TABPAGENO [of <TABLASTPAGE]"

where:

HEADING	Is optional if text is on the next line after the TABLE command.
CENTER	Centers text over the report; requires use of the HEADING option.
"text"	Text for your heading; may contain spot markers and variable field values.
<[±]n	Positions text in an absolute or relative column (n); called spot markers.
</n	Skips "n" line(s); must be put at the start or end of a line.
<field[>	Inserts field values. Add > to retain trailing blanks.
<prefix.field[>	Inserts calculated field value. (See <i>Manipulating Display Fields With Prefix Operators</i> .)
<OX[>	Enables multiple lines of text to be combined into a single line of heading, subheading, footing, or subfooting when the report is displayed. If you supply two or more lines between quotation marks without using the <OX[>] option, the lines are automatically adjusted into pairs to provide coverage across the printed page.
<TABPAGENO	Displays the current page number(s) in text. Overrides the default page number position of top left corner.
<TABLASTPAGE	Displays the last page number of the report in the heading (or footing) text.

Display Commands

Display commands enable you to report from several fields using a single command. You can include several different display commands in a single report request.

`display_command [prefixoperator.] field [/alignment] [/format] [AS 'title'] [NOPRINT] [IN [+]n]`

`[WITHIN by_field] [WITHIN across_field]`
`[[AND] ROW-TOTAL [/alignment] [/format] [AS 'title']]`
`[[AND] COLUMN-TOTAL [/alignment] [AS 'title']]`

The `display_command` is one of the following:

- PRINT [*]** Displays values in columns using field names as headings. Use * to display all of the fields in the left path of the data source.
- LIST** Is the same as PRINT, but also numbers each line. Numbering is reinitialized when the major sort value changes.
- SUM** Adds values in numeric fields or displays the last alphanumeric value. WRITE and ADD are synonyms.
- COUNT [*]** Counts the instances of data in a data source, not the distinct values. Use * to count all instances at the lowest segment level in a path. In a single segment data source, this effectively counts all instances in the data source.

Note:

- The maximum number of display fields is 1024 (495 for MATCH requests). This includes all named fields, whether printed or not, including data source fields, temporary fields (virtual fields and calculated values), certain internal fields (for example, TABPAGENO), and fields used in headings and footings.
- The results produced by the * vary slightly for different platforms. Consult your WebFOCUS documentation set for details.
- Field names used with display commands can have a maximum length of 66 characters. A field name may be qualified with its Master File and/or segment name. Any separation characters (., :, !, %, |, \) are included in the 66 character total. The following syntax describes qualification:

`[filename.][segmentname.]fieldname`

Indexed fields in Master Files and text fields have a maximum field name length of 12 characters.

See *Report Formatting: ON TABLE* for details on ROW-TOTAL and COLUMN-TOTAL.

Manipulating Display Fields With Prefix Operators

You can use prefix operators to perform calculations directly on the values of fields. Use with SUM or COUNT. In addition, you can use the ASQ., AVE., CNT., FST., LST., MAX., MIN., and SUM. prefix operators with SUBTOTAL, SUB-TOTAL, SUMMARIZE, and RECOMPUTE.

AVE.	Average	ST.	Subtotal at sort break; use with SUBFOOT text.
MAX.	Maximum	CT.	Cumulative total; use with SUBFOOT text.
MIN.	Minimum	ASQ.	Average sum of squares
FST.	First instance	PCT.	Percent of column total
LST.	Last instance	PCT.CNT.	Percent of count
TOT.	Column total. Use also with PRINT.	RPCT.	Percent of row total; use with ACROSS.
CNT.	Same as COUNT	SEG.	Retrieves all fields in a segment.
SUM.	Same as SUM	DST.	Aggregate and list unique values of any data source field. Use with PRINT and COUNT.
ALL.	Retrieves segment instances, including ones with missing descendants. Use also with PRINT.		

Column Formatting Options

Column formatting options enable you to control a column's title, format, and width. The options are:

/format	Temporarily alters field format for the specified report column only.
/alignment	Justifies column title to right (R), left (L), or center (C).
WITHIN by_field	Qualifies field domain. Specify the field from the BY phrase.
WITHIN across_field	Is a secondary field domain qualifier. Specify the field from the ACROSS phrase. If this is the only qualifier, then it must follow WITHIN TABLE.
AS 'title'	Assigns new column title text. Comma separators between words or phrases produce up to five lines of stacked titles.
NOPRINT	Retrieves the field and performs the required action, but no results are displayed. SUP-PRINT is a synonym for NOPRINT.
IN [+]<i>n</i>	Controls column placement. Overrides the default spacing between columns.
AND	Is an optional keyword provided to enhance readability. It can be used between any two field names and does not affect the report.
OVER	Displays one value above another.

Creating a Temporary Field: COMPUTE

The COMPUTE command creates and displays new temporary field(s) based on existing data. It works on the results of SUM, LIST, PRINT, or COUNT display commands. A computed field lasts for the current request only and cannot be used as a sort field.

COMPUTE

```
tempname1 [/format]=expression; [AS 'title'] [NOPRINT] [IN [+]n]  
[AND COMPUTE] tempname2 [/format]=expression;
```

The AND COMPUTE phrase is required if a prior COMPUTE command used the NOPRINT, AS, or IN options.

Expressions may be numeric, alphanumeric, logical, conditional or dates. The default field format is D12.2. Format types are:

A	Alphanumeric
I	Integer
F	Floating-point single precision
D	Floating-point double precision
P	Packed decimal

See also: *Creating a Temporary Field: DEFINE* for further information about temporary fields.

Numeric Expressions

The basic format is

*n*1 operator *n*2

where “*n*” is a numeric field, a literal, or an expression. Operators are evaluated in the following order:

()	Parentheses
AND	Logical operator
OR	Logical operator
**	Exponentiation
/ *	Division and multiplication
+ -	Addition and subtraction

Special Functions

Special functions act on a specified argument(s). An argument may be an expression or another function. Listed below are arithmetic functions. For more functions, refer to the documentation for your platform.

```
function(argument1[,argument2...]);
```

ABS	Absolute value; acts on one argument.
EXPN	Scientific notation, base E or D; acts on one argument.
INT	Integer part; acts on one argument.
MAX	Maximum value; acts on many arguments.
MIN	Minimum value; acts on many arguments.
LOG	Logarithm, base E; acts on one argument.
SQRT	Square root; acts on one argument.

Conditional Expressions

A conditional expression tests and assigns values according to specified arguments. An argument may be a literal, a field, an arithmetic expression, or a compound expression (for example, expressions containing IF, AND, or OR). Include single quotation marks around alphanumeric values. A conditional expression within a conditional expression requires parentheses and cannot immediately follow IF

```
IF argument THEN argument [ELSE IF...] [ELSE value];
```

or

```
IF (expression) {AND|OR} (expression) THEN... ELSE ... ;
```

Alphanumeric Expressions

An alphanumeric expression uses concatenation to combine alphanumeric values. The double concatenation symbol moves trailing blanks to the end of the result; the single concatenation symbol includes them as is. The symbol is operating system and/or keyboard dependent.

```
value1 | value2[ || ]...;  
'literal1' | 'literal2' |...;
```

EDIT Function

EDIT converts alphanumeric format to integer and vice versa. The mask inserts integers and alphanumeric values: 9 selects a character, \$ ignores it, and any other character is inserted.

```
EDIT(field[, 'mask']);
```

DECODE Function

DECODE converts coded values to associated values. Field contains code1; result1 is a new value; ELSE default is the possible default value.

```
DECODE field(code1 result1 ...[ELSE default]);
```

LAST Function

LAST retrieves the last encountered value of the field in the specified calculation. Usually used in a conditional expression.

```
IF LAST field...THEN...;
```

Creating a Temporary Field Unrelated to a Master File

You can create temporary fields that are independent of either a Master File or a request using the DEFINE FUNCTION command.

All names defined in the body of the function are local to the function. The last field defined before the END command in the function definition must have the same name as the function and represents the return value for the function. Use DEFINE FUNCTION CLEAR to clear a DEFINE function.

```
DEFINE FUNCTION name (argument1/format1,..., argumentn/formatn)
```

```
[tempvariablea/tempformata = expressiona;]
```

```
.  
.  
.
```

```
[tempvariablex/tempformatx = expressionx;]
```

```
name/format = [result_expression];
```

```
END
```

where:

name	Is the name of the function. This must be the last field calculated in the function and is used to return the value of the function to the calling procedure. Function names are limited to 8 characters.
argument1...argumentn	Are the argument names and their formats. Argument names are limited to twelve characters.
format1...formatn	Are the formats of the function arguments. If an argument's format is alphanumeric, the argument value must also be alphanumeric. Shorter arguments are padded on the right with blanks, and longer arguments are truncated. If an argument's format is numeric, the argument value must also be numeric. To prevent unexpected results, you must be consistent in your use of data types.
tempvariablea...tempvariablex	Are temporary fields. Temporary fields hold intermediate values used in the function. You can define as many temporary fields as you need.
tempformata...tempformatx	Are the formats of the temporary fields.
expressiona...expressionx	Are the expressions that calculate the temporary field values. The expressions can use parameters, constants, and other temporary fields defined in the same function.
format	Is the format of the value the function returns.
result_expression	Is the expression that calculates the value returned by the function. The expression can use parameters, constants, and temporary fields defined in the same function.

Sort Phrases

Sorting a report enables you to display the report's information grouped in a particular order.

```
[RANKED] BY [TOP|HIGHEST|LOWEST] [TOTAL] [n] sortfield [/alignment]
  [IN-GROUPS-OF qty [TILES] [TOP limit]] [NOPRINT] [AS 'title' [IN [+]n]
  sortfield ROWS value1 OVER value2 ...

ACROSS [TOP|HIGHEST|LOWEST] [n] sortfield [/alignment] [IN-GROUPS-OF qty
  [TOP limit]] [NOPRINT] [AS 'title' [IN [+]n]
  sortfield COLUMNS value1 OVER value2 ...
```

where:

- BY** Sorts the rows of a report down the page. You specify the field(s) to sort on. Up to 32 BY phrases per request (31 with PRINT or LIST).
- ACROSS** Sorts the columns of a report across the page. You specify the field(s) to sort on. Up to five ACROSS phrases per request and a total of 95 values for each (with a maximum of 256 columns) are allowed.
- BY and ACROSS** When used together in the same request, creates matrix reports; use with the display commands SUM and COUNT.

You can refine sorting using the following parameters:

- RANKED** Indicates a numeric rank for each row, as sorted by the specified field. It is often combined with a sort order designation (see next option).
- HIGHEST [n]** Specifies a sort order. LOWEST (the default) displays values in low-to-high order; HIGHEST and TOP display values in high-to-low order. "n" limits the sort to a specified number of values (for example, the LOWEST 5 or HIGHEST 5). There is a limit of five per request.
- TOP**
- LOWEST**
- TOTAL** Applies aggregation and sorting simultaneously to numeric columns. For BY TOTAL to work correctly, you must have an aggregating display command such as SUM.
- IN-GROUPS-OF qty** Clusters numeric data into groups: "qty" defines the range of each group; "limit" lets you exclude groups that exceed the limit (for example, IN-GROUPS-OF 10 TOP 50). There is a limit of five per request and one per BY field.
- TOP limit**
- TILES** Groups numeric data into any number of tiles (for example, 100 tiles produces percentiles, 10 tiles produces deciles). Grouping data in tiles sorts data instances on a BY field in the request and then apportions them as equally as possible into the number of tiles you specify.
- /alignment** Controls justification (right, left, center) of column titles. Add a justification suffix (R, L, or C) to a sort field to justify the field name for a BY field or the field's value for an ACROSS field.
- ROWS value OVER value** Enables you to use sort phrases to define a specific sequence for a sort field.
- COLUMNS value AND value** You supply the field's values and desired order.

See also: *Display Commands* for descriptions of NOPRINT, AS 'title', and IN [+]n.

Selection Phrases

There are two phrases for selecting records: WHERE and IF. WHERE does everything IF does, and then some; most notably, you can compare a field to another field or a field to an expression. You can write several tests within a single WHERE statement, and tests that could only be accomplished by including several IF statements within a DEFINE request can be accomplished with a single WHERE statement. You may also code values directly in the WHERE phrase or obtain values from a file using the IN FILE operator.

{WHERE|IF} {READLIMIT|RECORDLIMIT} EQ nn

WHERE [TOTAL] {fieldname relation value1 [OR value2] {OR|AND} (expression|fieldname)}

WHERE [NOT] fieldname IN {FILE ddname}('value1','value2','value3')

IF [TOTAL] fieldname relation {(ddname)|value1} [{OR|AND} {(ddname)|value2} ...]

IF fieldname {EQ|IS} {TRUE|FALSE}

where:

TOTAL	Selects data to be printed after data has been retrieved and processed. Use with SUM, COUNT, computed fields, or with any prefix operator. Use parentheses to indicate that you want TOTAL to act on multiple expressions (for example, expression1 AND expression2).
NOT	Selects records for which the specified field's value is not found.
RECORDLIMIT EQ nn	Limits number of records with values that satisfy the request.
READLIMIT EQ nn	Limits number of records read. Use WHERE with READLIMIT and group keys. Not used with FOCUS data sources; use RECORDLIMIT instead.
OR	Use for two test values. Field value(s) must satisfy one or the other test value.

Note:

- WHERE and IF are two different phrases and often use different syntax to achieve the same result.
- The IF..THEN...ELSE logical expression cannot be used with WHERE.

Choices for relations are:

EQ	Equal to. Synonym is IS.
NE	Not equal to. Synonym is IS-NOT.
GE	Greater than or equal to. Synonym is IS-FROM.
GT	Greater than. Synonym is EXCEEDS.
LT	Less than.
LE	Less than or equal to.
[IS-]FROM n1 TO n3	Is within range; "n" can be an expression for WHERE.
NOT-FROM n1 TO n3	Is not within range; "n" can be an expression for WHERE.
IS [-NOT] MISSING	Tests whether value is missing or not.
[NOT] field IN (z, x, y)	Selects values that are or are not in an unordered list of at least one numeric or alphanumeric constant. Use a comma or blank to delimit constants. Enclose alphanumeric constants in single quotation marks.

[NOT] field IN FILE ddname	Compares a data source value to values in a sequential file. The file must be previously allocated using DYNAM ALLOCATE, ALLOCATE, or FILEDEF
field [NOT] LIKE 'mask'	Selects or rejects records similar to a specified mask. Enclose the mask in single quotation marks. Wild cards are underscore (_) for single character, and percent sign (%) for a sequence of characters from 0 to n.
CONTAINS string	Selects the character string that matches the test value. Only alphanumeric fields are accepted.
OMITS string	Selects non-matching values. Only alphanumeric fields are accepted.
INCLUDES string1 AND string2	Tests if a value chain includes the list of literals. Only for multiple segment FOCUS data sources. Retrieved data must satisfy all string parameters.
EXCLUDES string1 AND string2	Tests if a value chain excludes the list of literals. Only for multiple segment FOCUS data sources. Retrieved data must exclude all string parameters.

See also: *Conditional Expressions* for further information about the IF selection phrase.

Calculating Trend Values and Forecasts

You can uncover trends in numeric data using the FORECAST command. Depending on the options you specify, it can also provide predicted values beyond the range of the values stored in the data source.

ON sortfield RECAP field1[/format] = FORECAST (field2, interval, npredict, 'method', npoint);

or

ON sortfield RECAP field1[/format] = FORECAST (field2, interval, npredict, 'REGRESS');

where:

sortfield	Is the last ACROSS field in the request and must be a numeric or smart date field. If the request does not contain an ACROSS phrase, FORECAST works on the last BY field. However, FORECAST is only supported with ACROSS when the display command is SUM, WRITE, ADD, or COUNT.
field1	Is a numeric field. It can be a real field, a virtual field, or a calculated value.
/format	Is the display format for field1.
field2	Is any numeric field.
interval	Is the increment to add to each sortfield value (after the last data point) to get to the next. It must be a positive whole number.
npredict	Is the number of predictions to calculate. It must be a whole number greater than or equal to zero.
method	Is the estimation method. It can be one of the following: MOVAVE - Simple moving average. EXPAVE - Exponentially smoothed moving average.
npoint	Is a positive whole number that specifies the number of values to average for the MOVAVE method. For EXPAVE, this number is used to calculate the weights for each component in the average. This parameter must be specified for MOVAVE and EXPAVE and omitted for REGRESS.
REGRESS	Is the linear regression estimation method.

Formatting Options

Formatting options enable you to reformat an entire report or a specific field.

ON [field|TABLE] option [AND option] [WHEN expression]

Field Formatting: ON Field

Field formatting occurs when a specified field value changes. Use the field from the prior BY phrase. The WHEN clause is an extension of the ON phrase and must follow the ON phrase to which it applies. One WHEN clause can be specified for each option; multiple WHEN clauses are permitted. Asterisked (*) options cannot be used with WHEN. The option can be:

SUBTOTAL	Calculates subtotals and grand totals for a specified BY field.
SUB-TOTAL	Produces subtotals for all numeric values whenever a BY or other high-level sort field changes; gives grand totals.
RECOMPUTE	Use instead of SUBTOTAL to subtotal real fields and recalculate computed fields with these subtotaled values.
SUMMARIZE	Use instead of SUB-TOTAL to subtotal real fields and recalculate computed fields with these subtotaled values.
MULTILINES*	Suppresses subtotal value when only one value exists at a sort break. Use with SUBTOTAL, SUB-TOTAL, RECOMPUTE, and SUMMARIZE.
RECAP	Creates a temporary field and calculates using subtotaled values. Same syntax as COMPUTE, but the expression uses only the subtotaled value.
NOSPLIT*	Forces a page break before or after sorted values; prevents “widow” lines.
PAGE-BREAK	Starts new page when the associated sort value changes.
REPAGE	Resets the page number to 1 after a sort break. Use with PAGE-BREAK.
SUBHEAD	Places text before the changing sort field.
SUBFOOT	Places text after the changing sort field.
UNDER-LINE	Draws a line across the page.
"text"	May contain spot markers and variable field values.

Note: The following three options also may be used with display fields.

SKIP-LINE	Adds a blank line. Only one per request is allowed.
FOLD-LINE*	Causes columns to be placed on a separate line when the value of the field changes in the ON phrase. The second half of the folded line is indented by two spaces from the start of the line. Up to 16 are allowed per request.
NOPRINT*	Field is retrieved and required action is performed, but result(s) may be displayed in headings or footings, or used in calculations or TOTAL screenings.

Report Formatting: ON TABLE

Report formatting occurs when you specify one of the following options:

COLUMN-TOTAL	Produces a final row of column totals for numeric fields.
ROW-TOTAL	Produces a final column of row totals for numeric fields.
ACROSS-TOTAL	Produces row totals for horizontal (ACROSS) sort field values. Row totals for horizontal sort fields are different from standard row totals in that only the values of the horizontal sort fields are included in the total.
NOTOTAL	Suppresses grand total(s), but displays subtotals.
SUMMARIZE	May be used instead of SUB-TOTAL and SUBTOTAL to generate subtotals for the recalculated values of computed fields in COMPUTE commands.
PAGE-BREAK [AND]	Provides a separate page for the report heading (SUBHEAD) or footing (SUBFOOT).
SUBHEAD	Displays report heading text on the first page only.
SUBFOOT	Displays report footing text on the last page only.
"text"	May contain spot markers and variable field values.
SET	Temporarily overrides system or session SET commands for the current report only. Specify parameter and value. Equal sign is not allowed.

See also:

- *Page Heading* for details on spot markers and variable field values.
- *Configuring Your Environment: SET* for a description of SET parameters.

Field-Based Reformatting

Field-based reformatting allows you to apply different formats to each row in a single report column by using a field to identify the format that applies to each row. The field format must be A8.

```
COMPUTE format_field/A8 = expression;
```

Once the format field is defined, you can apply it in a report request:

```
TABLE FILE filename  
display fieldname/format_field[/just]  
.  
.  
.  
END
```

where:

display	Is any valid display command.
fieldname	Is a field in the request to be reformatted.
format_field	Is the name of the field that contains the formats. If the name of the format field is the same as an explicit format, the explicit format will be used.
/just	Is a justification option, L (left), R (right), or C (center).

Conditional Formatting

Conditional formatting executes the specified option when the expression is satisfied. The expression uses the same functions as COMPUTE and acts on subtotaled values.

WHEN expression

Environment Options

These options temporarily override system or session SET commands for the current report only. They enable you to specify parameters and values. The equal sign is not required.

ON TABLE SET parameter value [AND parameter value]

See also: *Configuring Your Environment: SET* for a description of these parameters.

Output Files

Output files store report data in temporary data sources. Use AS to rename data sources for future sessions since subsequent writes to existing data sources overwrite the contents. FILEDEF or ALLOCATE is automatically issued.

ON TABLE output_format [AS [d:]filename [FORMAT option]] [MISSING {ON|OFF}]

where:

output_format	Is one of the following: HOLD. Stores report data and (for some formats) creates a corresponding Master File. The default file name is HOLD. SAVB. Same as HOLD in internal format, but no Master File is created. The default file name is SAVB. SAVE. Translates data and saves it to character format. No Master File is created. The default file name is SAVE.
AS [d:] filename	Names the output data source.
FORMAT option	Formats the report data in a variety of ways, including a FOCUS data source, a relational table, and a spreadsheet work file for use by different software products. Options vary according to operating environment. For a complete list of options, consult your WebFOCUS documentation set. Options include: ALPHA, BINARY, CLIP, COMMA, COM, COMT, DB2, DBASE, DIF, DOC, EXCEL, EXL2K, EXL2K FORMULA, EXL2K PIVOT, EXL97, FOCUS, FUSION, HTML, HTMTABLE, INGRES, INTERNAL, LOTUS, PDF, POSTSCRIPT (PS), RTF, REDBRICK, SQL, SQLDBC, SQLINE, SQLMAC, SQLMSS, SQLODBC, SQLORA, SQLSYB, SYLK, TABT, WK1, WP. See <i>Effect of TABLE Syntax Elements on Excel PivotTables</i> for details on EXL2K PIVOT.
MISSING	Controls whether fields with the attribute MISSING=ON in the Master File are carried over into the HOLD file. If the HOLD command specifies MISSING OFF, the MISSING attribute is not carried over.

Use the following query command to display fields, aliases, and usage formats in the named file:

? HOLD [filename]

Effect of TABLE Syntax Elements on Excel PivotTables

The following table summarizes TABLE syntax elements that are supported in EXL2K PIVOT. For a valid PivotTable, you need at least one sort field or a PAGEFIELD.

Syntax Element	Usage	Effect on PivotTables
PRINT	Required.	Designates the data field in a PivotTable.
BY	Optional.	Designates a row field in a PivotTable.
ACROSS	Optional.	Designates a column field in a PivotTable.
CACHEFIELDS	Optional.	Places fields in the Pivot cache file and makes them available from the Pivot tool bar.
PAGEFIELDS	Optional.	Designates a Page field in a PivotTable.

Page Footing

A page footing is text that appears on the bottom of every page of a report.

FOOTING [CENTER][BOTTOM]

"text <[±]n <field[>] more text </n"

"text <prefixoperator.field[>]"

where:

FOOTING	Is required.
CENTER	Centers text under the report.
BOTTOM	Places text on the bottom of each page. Requires use of FOOTING. If this option is not specified, then the footing is printed two lines beneath the report.
"text"	Text for your footing; may contain spot markers and variable field values.

See also: *Page Heading* for details on spot markers and variable field values.

Terminating Commands

Terminating commands complete the report request. You must include only one per request. END and RUN must be typed on lines by themselves. Commands are:

END	Use if you plan to issue consecutive report requests against different data sources during one session.
RUN	Keeps the TABLE facility and the data source active for the duration of the session. You do not need to repeat the TABLE command to produce another report using the same data source.
QUIT	Discontinues a report request without executing it.

Using Additional Reporting Features

Creating a Temporary Field: DEFINE

The DEFINE command creates a temporary field and assigns it to a specified data source for the current session. Specify the DEFINE before the report request.

You can define as many fields as you need, as long as you have enough memory and the total number of fields referenced in a request is less than 1024.

```
DEFINE FILE file [CLEAR] [ADD]
field [/format] [WITH qualifier.realfield]=expression;
field [/format] REDEFINES qualifier.realfield
  [WITH qualifier.realfield]=expression;
.
.
.
END
```

The DEFINEd field is treated like a real field; it may be a display field, sorted or screened on individual values. The field format default is D12.2.

<u>CLEAR</u>	Is the default; clears existing DEFINES for the specified data source.
ADD	Retains previously DEFINEd fields in subsequent DEFINES specified for the data source.
WITH qualifier.realfield	Assigns a DEFINE field to a logical home with a real data source field. Use for counters.
REDEFINES qualifier.realfield	In cases where the same field name exists in more than one segment, and that field must be redefined or recomputed, the REDEFINES phrase must be used.
expression	Uses same functions as COMPUTE. If a field(s) used in an expression is not in the Master File and is not DEFINEd, use the WITH option.

To list all DEFINEd fields or just those for a specified data source, use the following query command:

```
? DEFINE [filename]
```

For conditional tests, use a DEFINE field as expression in an IF or WHERE test. For example:

```
DEFINE FILE EMPLOYEE
NEWSAL= IF DPT EQ 'MIS' THEN CSAL * 1.05 ELSE CSAL * 1.06;
END
TABLE FILE EMPLOYEE
PRINT CSAL NEWSAL BY EID
END
```

See also: *Creating a Temporary Field: COMPUTE* for further information about temporary fields.

Joining Data Sources

JOIN temporarily connects two data sources. Joined data sources remain physically separate, but are treated as one for the current session. Specify a JOIN before the report request. You can use up to 16 JOINS at any given time. For FOCUS data sources, field2 must be indexed. For complete details on requirements for joining data sources, see your WebFOCUS documentation set.

```
JOIN field1 [WITH rfield] IN hostfile [TAG tag1]
TO [ALL] field2 IN crfile [TAG tag2] [AS joinname]
END
```

Field1 (host) and field2 (target) must have the same field format; names may differ.

WITH rfield	Use only if field1 will be a DEFINEd field; assigns a logical home with a real field in the host data source.
hostfile	Is the host data source.
tag1	Is the tag name for the host data source. Use as a file name qualifier for field names and aliases in the host data source.
ALL	Is used for multiple instances in the target data source.
crfile	Is the target (cross-referenced) data source.
tag2	Is the tag name for the cross-referenced data source. Use as a file name qualifier for field names and aliases in the cross-referenced data source.
AS joinname	Names the JOIN structure.
END	Required when the JOIN command is longer than one line; terminates the statement.

Note:

- For a DEFINE-based JOIN, issue JOIN, then create field1 in a DEFINE command. DEFINES created prior to this JOIN are cleared. Create subsequent DEFINES for the JOIN after the JOIN command is issued.

- To list all JOINS currently in effect, issue the following query command:

```
? JOIN
```

- With a recursive join, you can uniquely refer to fields in cross-referenced segments by prefixing them with the first four letters of the join name or you can use TAG.
- To clear all or specific named JOINS, issue the following:

```
JOIN CLEAR {*[joinname]}
```

- When using FORMAT FOCUS, up to 4 fields in the host file can be joined to a *single* field in FORMAT FOCUS:

```
JOIN field1 [AND field2 ...] IN file,
TO field 5 [AND field6...]
```

- Multiple fields in the host file may be joined to *multiple* fields for relational data sources.

Using a Conditional Join

You can establish joins based on conditions other than equality between fields. In addition, the host and cross-referenced join fields do not have to contain matching formats, and the cross-referenced field does not have to be indexed.

```
JOIN FILE from_file AT from_field [TAG from_tag [WITH fieldname]
TO [ALL|ONE]
FILE to_file AT to_field [TAG to_tag]
[AS as_name]
[WHERE expression1 ;
WHERE expression2 ;
...
END
where:
```

from_file	Is the host Master File.
AT	Links the correct parent segment or host to the correct child or cross-referenced segment. The field values used as the AT parameter are not used to cause the link; they are simply used as segment references.
from_field	Is the field name in the host Master File whose segment will be joined to the cross-referenced data source. The field name must be at the lowest level segment in the data source that will be referenced.
from_tag	Is the optional tag name that is used as a unique qualifier for fields and aliases in the host data source.
fieldname	Is the data source field with which to associate a DEFINE-based conditional join. For a DEFINE-based conditional join, the KEEPDEFINES setting must be on and you must create the virtual fields before issuing the JOIN command.
ALL	Is the one-to-many relationship between the from_file and to_file.
ONE	Is the one-to-one relationship between the from_file and the to_file.
to_file	Is the cross-referenced Master File.
to_field	Is the join field name in the cross-referenced Master File. It can be any field in the segment.
to_tag	Is the optional tag name that is used as a unique qualifier for fields and aliases in the cross-referenced data source.
as_name	Is the name associated with the joined structure.
expression1, expression2	Are any expressions that are acceptable in the DEFINE FILE command. All fields used in the expression must lie on a single path.
END	Required to terminate the statement.

Combining Data Sources: MATCH

The MATCH command merges record subsets from two or more data sources to a HOLD file. Use AS to name the HOLD file for future sessions. You may merge up to six data sources.

PRINT produces an unprinted BY field called LIST, that may affect MATCH output. ACROSS, IF TOTAL, WHERE TOTAL, COMPUTE, and BY HIGHEST commands are not allowed.

```
MATCH FILE file1
display command
BY phrase [AS 'name']
.
.
.
RUN
.
.
.
FILE file2
display command
BY phrase [AS 'name']
.
.
.
[AFTER MATCH HOLD [AS 'name'] merge phrase]
END
```

File1 and file2 must have at least one common BY field with the same format. File1 is OLD; file2 is NEW. The merge phrase specifies the record subset.

The choices for “merge phrase” are:

OLD-OR-NEW	Writes all records. This is the default option if the entire AFTER MATCH line is omitted.
OLD-AND-NEW	Writes only matching records, those occurring in both data sources.
OLD-NOT-NEW	Writes only OLD non-matching records, excluding common ones.
NEW-NOT-OLD	Writes only NEW non-matching records, excluding common ones.
OLD-NOR-NEW	Writes all non-matching records, excluding common ones.
OLD	Writes all OLD records, including common ones.
NEW	Writes all NEW records, including common ones.

Concatenating Data Sources: MORE

The MORE command concatenates data from data sources which may have dissimilar Master Files into a single HOLD file. Concatenation can occur between FOCUS data sources or any other FOCUS-readable data source.

TABLE FILE file1

main request

.

.

.

MORE

FILE file2

subrequest

MORE

.

.

.

END

where:

TABLE FILE

Is a required command. All MORE subrequests can only be issued from within the TABLE command.

file1

Identifies the first data source that you want to report from.

main request

Is an ordinary TABLE request without an END or RUN terminating command. Output can be redirected using the "ON TABLE HOLD AS filename" subcommand.

MORE

Begins a subrequest.

FILE file2

Identifies the next data source for concatenation.

subrequest

Subrequests may only include WHERE or IF clauses. All fields in each of the subrequests must have a corresponding field in the main request. Corresponding fields must have either the same FIELDNAME or ALIAS.

END

Ends a request.

Creating Financial Reports

Financial Modeling Language (FML) creates, calculates, and presents financially oriented data such as balance sheets, consolidations, and budgets.

[SET FORMULTIPLE={ON|OFF}]

TABLE FILE file

display command

[BY sortfield]

FOR field [NOPRINT]

value [AS 'text'] [LABEL label] [WHEN EXISTS][NOPRINT] OVER

value [OR value2...] [AS 'text']... OVER

[value1 TO value3] [AS 'text']... OVER

value [POST TO ddname]... OVER

"text <[±]n <field[>] more text</n" OVER

"text <prefix.field[>]" OVER

PAGE-BREAK OVER

DATA value, [value2,...,\$] [AS 'text']... OVER

DATA PICKUP [FROM ddname] id [OR id]... OVER

BAR [AS 'x'] OVER

RECAP name [/format][([n[,m [,p]])]=expression; [AS 'text']...

[output file phrase]

{END|QUIT}

Value must be a specific value or tag for a given field. You cannot use the same value twice, unless you rename it with RECAP or use the SET parameter FORMULTIPLE.

FORMULTIPLE	Allows you to include the same FOR field value in multiple rows of an FML matrix.
OVER	Stacks values. The last value line should not contain OVER.
AS 'text'	Displays text for a row.
LABEL	Assigns a variable name to a line item for use in a RECAP calculation.
WHEN EXISTS	Displays a line if any value exists on the line. Overrides period default for missing data.
value1 OR value2	Displays the summed value for two or more values.
value1 TO value3	Displays the summed value for a specified range. Value1 is the lower limit; value3 is the upper limit.
POST TO ddname	Saves a data line in a temporary work file (ddname). The default work file is FOCPOST, with a comma-delimited format. See Note below.
"text"	For additional text, place anywhere after FOR; may contain spot markers and variable field values.
PAGE-BREAK	Breaks the page at any point.
DATA value, \$	Enters value(s) directly.
DATA PICKUP	Retrieves posted lines and uses them as if they were provided in a DATA line. Use "id" to retrieve an explicit label or value. See Note below.
BAR AS 'x'	The default character is the hyphen (-). AS 'x' replaces the default with a line of the specified character.
RECAP	Same syntax as COMPUTE. Calculates data to produce new row(s).
n	Calculation done only in column "n".

m Calculation done in columns “n” through “m”.

p Calculation done in columns “n” through “m” in increments of “p”.

expression; Uses same functions as COMPUTE, excluding DECODE and EDIT. Calculates only explicit or default labels, not specific values.

Note: For POST and PICKUP, the data source must be identified (with a FILEDEF, ALLOCATE, or DYNAM ALLOCATE as appropriate to your operating environment) before the request is issued.

See also: *Page Heading* for details on spot markers and variable field values.

Reporting Dynamically From an FML Hierarchy

With FML hierarchies you can use the hierarchical relationship between two fields (defined in the Master File) to *dynamically* construct the rows that represent the fields’ relationship and display them in the report starting at any point in the hierarchy.

This is different from a traditional FML report, where you would have to list the employee IDs in the request syntax in the order in which they should appear on the report. If an employee or account is added, removed, or transferred, you would have to change the report request to reflect this change in organizational structure. Using FML hierarchy, you can do this dynamically.

```
TABLE FILE filename
{PRINT|SUM}
FOR hierarchyfld
parentvalue {GET|WITH} CHILD[REN] [{n|ALL}] [ADD [m|ALL]] [AS {CAPTION'text'}] [LABEL label]
parentvalue ADD [{m|ALL}] [AS {CAPTION'text'}] [LABEL label]
.
.
.
END
```

where:

hierarchyfld Is the hierarchy field name. If the request references a joined structure, the name must be the field name from the host file. The alias name is not supported.

parentvalue Is the parent value for which the children are to be retrieved.

GET CHILD[REN] Prints each child instance over the next child instance. (This corresponds to the FML syntax CHILD1 OVER CHILD2 OVER ...) Successive levels of the hierarchy field are indented two spaces from the previous level.

WITH CHILD[REN] Displays the hierarchy starting from the specified parentvalue. Includes the parent in the display.

ADD Displays all children on one row, summing the numeric data values displayed on the row. (This corresponds to the FML syntax CHILD1 OR CHILD2 OR ...) When you use ADD without GET or without WITH CHILDREN the parent *and* the children values are summed. Also, when you use WITH CHILDREN ADD, the parent *and* the children values are summed.

n|ALL Is a positive integer from 1 to 99 specifying the number of levels of the hierarchy to retrieve (and aggregate if ADD is used in the request). If n is omitted, direct children are retrieved. If n is 2, direct children and grandchildren are retrieved and if n is 99 children to 99 levels are retrieved. ALL is a synonym for 99.

m|ALL Is the number of levels of hierarchy (from 1 to 99) that will be added together on each line. ALL is the default and is equal to 99.

AS CAPTION	Indicates that the caption values to display should be taken from the field defined as the CAPTION in the Master File. The AS CAPTION phrase is supported for rows that do not display hierarchical data. However, the hierarchy must be defined (by specifying the PARENT_OF attribute) in order to load and display the caption values. If the hierarchy is not defined, the AS CAPTION phrase is ignored.
text	Is a text string to use as the caption for the hierarchy field values.
LABEL label	Is an explicit row label. Each generated row is labeled with the specified label text.

Using StyleSheets

The StyleSheet facility enables you to produce presentation quality PDF, PostScript, HTML, and Excel 2000 reports using styles you define. Styles are stored in a StyleSheet file (or directly within the report request) that is used during generation of the report.

ON TABLE SET STYLE[SHEET] *

```
[DEFMACRO = macroname, attribute1 = value1, [attribute2 = value2, ...] $]
TYPE=type[,subtype=value][,MACRO=macroname][,FONT=fontname] [,SIZE=n] [,COLOR=color] [,BACKCOLOR=color]
[,PAGECOLOR=color] [,STYLE=[+|-]style[+style2...]] [,JUSTIFY=option] [,BORDER[-position]=option]
[,BORDER[-position]-STYLE=style][,BORDER[-position]-COLOR={color|RGB(r g b)}][,GRID={ON|OFF|FILL}]
[, {HGRID|VGRID}={ON|OFF|HEAVY}] [,WRAP={ON|OFF|n}][,TITLETEX=title] [,IMAGE=image] [,BACKIMAGE=image]
[,WHEN=column operator {constant|column}][,WHEN=FORECAST],$
[ENDSTYLE]
```

DEFMACRO=macroname	Defines a StyleSheet macro that groups a sequence of attributes together, enabling you to apply them repeatedly throughout a StyleSheet without recoding them.
value1, value2...	Is the value you want to assign to the attribute.
attribute1, attribute2...	Is any StyleSheet attribute, such as an attribute to format a report component, insert a graphic, define a hyper link, or apply a condition for conditional formatting (WHEN).
TYPE=type	Specifies the section of the report to which the designated style will be applied. Options include: REPORT, DATA, ACROSSVALUE, ACROSSTITLE, TITLE, UNDERLINE, SKIPLINE, PAGENUM, HEADING, FOOTING, SUBHEAD, SUBFOOT, TABHEADING, TABFOOTING, RECAP, SUBTOTAL, GRANDTOTAL, FREETEXT.
subtype	Are any additional attributes, such as COLUMN, ACROSS, or ITEM, that are needed to identify the report component. For example, a specific line, text phrase, column, field, or total. (See the <i>Identifying Report Component Subtypes</i> section, which follows.)
MACRO=macroname	Is the name of the macro to apply to the specified report component. The macro must be defined in the same StyleSheet.
FONT=fontname	Designates a font for the report component.
SIZE=n	Designates a font size for the report component.
COLOR=color	Designates a color for the report component.
BACKCOLOR=color	Designates the background color for the report component.

PAGECOLOR=color	Sets page color for HTML reports with internal Cascading Style Sheets.
STYLE=style	Designates one or more text styles for the report component. Options include: BOLD, ITALIC, OUTLINE, UNDERLINE, NORMAL, or a combination of styles. For example, BOLD+ITALIC. Use the minus (-) sign to remove a style.
JUSTIFY=option	Justifies the report component. Options include: LEFT (the default for alphanumeric data), RIGHT (the default for numeric data), and CENTER.
BORDER[-position]=option	Adds a border around the specified report component. Options include: ON, OFF, LIGHT, MEDIUM, HEAVY, or width (in points). You should not use BORDER and GRID (or VGRID/HGRID) in the same StyleSheet. Position specifies which border line to format. Values are TOP, BOTTOM, LEFT, or RIGHT.
BORDER[-position]-STYLE=style	Sets the style of the border line. Those styles noted by asterisks are available only in HTML. Styles include SOLID, DOTTED, DASHED, DOUBLE, GROOVE*, RIDGE*, INSET*, OUTSET*, and NONE.
BORDER[-position]-COLOR={color RGB(r g b)}	Sets the border color to one of the preset color values. The default value is BLACK. If the display or output device does not support colors, it substitutes shades of gray.
GRID={ON OFF FILL}	Applies horizontal and vertical grid lines to an HTML report.
{HGRID VGRID}={ON OFF HEAVY}	Inserts horizontal (HGRID) or vertical (VGRID) grid lines in a PDF report.
WRAP={ON OFF n}	Controls the wrapping of data in a report. ON turns on data wrapping and is the default. OFF turns off data wrapping and n represents a specific numeric value that the column width can be set to. The value represents the measure specified with the UNITS SET parameter.
TITLETEXT=title	In HTML reports and in graphs, inserts the specified title in your browser's title bar. If you are running a report in EXL2K format, the title text is inserted as the worksheet's tab name. Use with TYPE=REPORT.
IMAGE=image	Specifies the file name of a graphical image file. The image must exist as a separate graphic file in a format that your browser supports. You can specify a local image file, or identify an image elsewhere on the network using a URL for HTML reports. For PDF or PS reports, your image must be a GIF file that is on the EDAPATH of your reporting server.
BACKIMAGE=image	Adds a background image to your HTML report. See IMAGE=image definition.
WHEN=column operator {constant column}	Specifies the use of stoplighting, or the styling of report components through WHEN testing. Column identifies the column. See the definition for COLUMN= below. Operator is any relational operator (EQ, NE, GT, LT, GE, or LE). Constant is a numeric constant or an alphanumeric string in single quotes.
WHEN=FORECAST	Specifies styling for forecasted data values.
,\$	Designates termination of the style specification.
ENDSTYLE	Indicates the end of an inline StyleSheet. You can omit ENDSTYLE if it is followed immediately by END in the report request.

See also: *SET Commands for StyleSheets* for a description of StyleSheet SET parameters.

Identifying Report Component Subtypes

The options for “subtype” are:

Subtype	Value	Description
ACROSS=	An Nn fieldname	Identifies a specific ACROSS in a TABLE request that uses multiple ACROSS phrases. Valid for types: ACROSSVALUE, ACROSSTITLE, REPORT.
ACROSSCOLUMN=	Nn Pn fieldname	Specifies the column number within each group of columns printed under each ACROSS value. Valid for types: DATA, GRANDTOTAL, REPORT, SUBTOTAL, TITLE.
BY=	Bn fieldname	Specifies the BY field for which the subtotal lines is being generated. Valid for types: RECAP, SUBTOTAL, SUBHEAD, SUBFOOT.
ITEM=	n	Specifies which item in a multi-item heading or footing is being described. Valid for types: HEADING, FOOTING, SUBHEAD, SUBFOOT, TABHEADING, TABFOOTING.
LINE=	n	Specifies a specific line of any type of multi-line heading or footing. Valid for types: HEADING, FOOTING, SUBHEAD, SUBFOOT, TABHEADING, TABFOOTING.
OBJECT=	TEXT FIELD	Specifies whether the object applies to textual information or embedded fields. Valid for types: HEADING, FOOTING, SUBHEAD, SUBFOOT, TABHEADING, TABFOOTING.
COLUMN=	Cn Bn Pn Nn fieldname ROWTOTAL	Identifies a particular column for stylizing. Valid for types: DATA, REPORT, TITLE, SUBTOTAL, GRANDTOTAL, ACROSS, ACROSSTITLE.
LABEL=	label	Identifies a row in an FML report by its explicit label.

To identify “value” count:

- An To identify the value of n, count horizontal sort (ACROSS) fields, including NOPRINT ACROSS fields, from left to right.
- Nn To identify the value of n, count vertical sort (BY) fields, display fields, and ROW-TOTAL fields, from left to right, including NOPRINT fields.
- Pn To determine the value of n, count vertical sort (BY) fields, display fields, and ROW-TOTAL fields from left to right. Do not count NOPRINT fields.
- Cn To determine the value of n, count only display fields from left to right, including NOPRINT fields. Do not count vertical sort (BY) fields or ROW-TOTAL fields.
To select all display fields use C*.
- Bn To determine the value of n, count only vertical sort (BY) fields, including NOPRINTs, from left to right.
To select all BY fields use B*.
- field Identifies a column by its field name. When a field occurs more than once, use field(n) to select a particular occurrence or field(*) to select all occurrences of the field.
- ROWTOTAL Identifies a column of row totals generated using ROW-TOTAL. When used with ACROSS and multiple display commands, ROWTOTAL generates multiple total columns. Use ROWTOTAL(n) to select a particular total column. Use ROWTOTAL(fieldname) to select the row total column for a particular field.
Use ROWTOTAL(*) to select all row total columns in the report.

Linking Reports to Other Resources Using StyleSheets

You can use StyleSheet declarations to define links from any report component to another report, procedure, URL, or JavaScript function. The maximum length of a link=value argument, including any associated parameters, is 2400 characters, but can span more than one line with the use of the backslash (\) continuation character.

```
TYPE=type, [subtype], [IMAGE=image,] link=value [(parameters ...)], \  
[WHEN=column operator {constant|column},][TARGET=frame,] $
```

where:

IMAGE=image

Specifies the file name of a graphical image file. The image must exist as a separate graphic file in a format that your browser supports. Most browsers support GIF and JPEG file types. You can specify a local image file, or identify an image elsewhere on the network using a URL. You can use IMAGE with HTML reports.

You can link from an image in any of the following TYPE= values: REPORT, TABHEADING, TABFOOTING, HEADING, FOOTING, SUBHEAD, or SUBFOOT.

link=value

Is one of the following:

FOCEXEC=fex

Identifies the file name of the linked report or procedure to run when you select the report component.

URL=url

Identifies any valid URL, including a URL that specifies a CGI program. If the URL refers to a CGI program that takes parameters, the URL must end with a question mark (?).

JAVASCRIPT=function

Identifies the JavaScript function to run when you select the report component.

parameters

Is in the form parameter=value, where parameter is the name of the variable in the linked procedure and value is one of the following:

'constant_value'

Identifies an actual value to be passed. The value must be enclosed in single quotation marks, or you can use an unquoted numeric value.

field

Identifies the field in the report whose value is to be passed to the procedure. You can identify the field using either the field name or the field position.

'&variable'

Identifies an amper variable whose value is to be passed to the procedure. The name of the amper variable must be enclosed in single quotation marks. You can use amper variables only in inline StyleSheets.

The entire string of parameter names and values must be enclosed in parentheses. Each parameter=value pair must be separated by a blank space. You can include multiple parameters in your request but the entire string cannot exceed 2400 characters.

<code>WHEN=column operator {constant column}</code>	Specifies the use of stoplighting, or the styling of report components through WHEN testing. Column identifies the column. See the definition for COLUMN= in the previous section. Operator is any relational operator (EQ, NE, GT, LT, GE, or LE). Constant is a numeric constant or an alphanumeric string in single quotes. You can also use WHEN=FORECAST to identify only forecasted data.
<code>TARGET=frame</code>	Identifies the target frame in the Web page where the output from the drill-down link (either a FOCEXEC or URL) is displayed. If the name of the target frame contains embedded spaces, the name will be correctly interpreted without enclosing the name in quotation marks. You can also use the following standard HTML frame names: <code>_BLANK</code> , <code>_SELF</code> , <code>_PARENT</code> , <code>_TOP</code> . You can use TARGET with HTML reports.

Using Cascading Style Sheets

When you create a report in HTML format, code is generated that specifies how the report is formatted. You can set up your report to generate an internal Cascading Style Sheet (iCSS) as part of the HTML code. This will improve performance by reducing the size of the HTML file and provide more formatting options.

`SET HTMLCSS = {ON|OFF}`

where:

- ON** Generates an internal Cascading Style Sheet in the HTML output to control most aspects of the report's appearance. This is the default for reports created in the Report Painter.
- OFF** Turns off the generation of an internal Cascading Style Sheet. Instead, formatting tags are placed in each HTML table cell used to create the report. This is the default.

You can also use an external Cascading Style Sheet (eCSS) to provide a standard way of formatting HTML documents.

`SET CSSURL=url_to_externalcss_file`

Configuring Your Environment: SET

You can use SET commands to temporarily override system or session SET commands for the current report request. Omit the equal sign when using as a format option with ON TABLE. As a command before and after a report request, an equal sign is required. All parameters except ASNAMES, COMPUTE, FOCSTACK, HOLDATTR, and HOLDLIST may be used as format options.

SET Command	Values	Description
ACCBLN=	<u>ON</u> OFF	Accepts blank or zero values for fields described with ACCEPT in the Master File. This SET command cannot be used in ON TABLE SET.
ALL=	<u>OFF</u> ON PASS	Selects handling of missing segment instances.
ASNAMES=	ON OFF <u>FOCUS</u>	Handles use of literal in AS phrase as field name in HOLD Master Files.
AUTOINDEX=	<u>ON</u> OFF	If your report request includes an IF or WHERE test on an indexed field, the data source is automatically searched using an indexed view.
AUTOPATH=	<u>ON</u> OFF	When accessing a FOCUS data source, retrieval begins from the lowest possible segment.
AUTOTABLEF=	ON <u>OFF</u>	Permits WebFOCUS to convert TABLE requests into TABLEF requests internally.
BYDISPLAY=	ON <u>OFF</u>	Populates every vertical (BY) sort column cell with a value, even if the value is repeating.
CACHE=	nnn	Controls the number of cache pages to be allocated. This SET command cannot be used in ON TABLE SET.
CARTESIAN=	<u>OFF</u> ON	Enables generation of a report that includes all combinations of non-related data instances in the case of a multi-path request.
CDN=	<u>OFF</u> ON SPACE QUOTE	Specifies punctuation used in numerical notation. SPACE sets the decimal point as a comma, and the thousands separator as a space. QUOTE sets the decimal point as a comma and the thousands separator as an apostrophe.
CENTZERO=	ON <u>OFF</u>	Displays a leading zero in decimal-only numbers.
COMPOUND=	OPEN CLOSE	Combines multiple report into a single PDF file. Use COMPOUND OPEN in the first report, and COMPOUND CLOSE in the last report.
COMPUTE=	<u>NEW</u> OLD	Processes DEFINE calculations with new (compilation) or old (interpretive) logic.
COUNTWIDTH=	<u>ON</u> OFF	Increases the maximum value for COUNT and LIST to 999,999,999 from 99,999.

SET Command	Values	Description
DATETIME=	STARTUP CURRENT NOW RESET	Statically or dynamically sets time and date in reports.
DEFCENT=	cc 19	Defines a default century globally or on a field-level for an application that does not contain an explicit century. This SET command cannot be used in ON TABLE SET.
DEFINES=	COMPILED OLD	Compiles virtual fields into machine code to improve performance. This SET command cannot be used in ON TABLE SET.
EMPTYREPORT=	ON OFF	Determines whether a report is displayed when zero records are found.
ERROROUT=	ON OFF	Terminates a request and returns an error message when an error is encountered. This SET command cannot be used in ON TABLE SET.
ESTRECORDS=	n	Used to pass the estimated number of records to be sorted in the request. Used without external sort.
EXTAGGR=	ON OFF NOFLOAT	Uses external sorts to perform aggregation. The NOFLOAT option allows aggregation if there are no floating data fields present.
EXTHOLD=	ON OFF	Enables you to create HOLD files using an external sort.
EXTSORT=	ON OFF	Calls an external sort for use with TABLE, MATCH, and GRAPH when needed.
EXTTERM=	ON OFF	Allows extended terminal attributes. This SET command cannot be used in ON TABLE SET.
FIELDNAME=	OLD NEW NOTRUNC	Controls whether long and qualified field names and aliases are supported. This SET command cannot be used in ON TABLE SET.
FILE=	file	Assigns the specified data source as a default and allows the use of TABLE alone to initiate a request. This SET command cannot be used in ON TABLE SET.
FILTER=	ON OFF	Assigns screening conditions to a data source for reporting purposes.
FIXRETRIEVE=	ON OFF	Enables keyed retrieval for fixed data sources. That is, the retrieval process stops when an equality or range test on a key holds true.
FOCSTACK=	g n	Assigns core for stack of WebFOCUS commands from FOCEXECs awaiting execution. This SET command cannot be used in ON TABLE SET.
FORMULTIPLE=	ON OFF	Allows you to include the same FOR field value in multiple rows of an FML matrix.

SET Command	Values	Description
HDAY=	xxxx	Specifies the holiday file from which to retrieve dates that are considered holidays. xxxx are the letters in the name of the holiday file, named HDAYxxxx, after HDAY. This string must be four characters long.
HOLDATTR=	ON OFF FOCUS	Controls whether TITLE and ACCEPT attributes in the original Master File are used in a HOLD Master File.
HOLDLIST=	PRINTONLY ALL	Controls whether only printed fields, or all fields are included in the HOLD file.
HOLDMISS=	ON OFF	Distinguishes between missing data and default data in a HOLD file.
HOLDSTAT=	ON OFF name	Enables inclusion of comments and DBA information in HOLD Master Files.
HTMLCSS=	ON OFF	Generates an internal Cascading Style Sheet in the HTML output to control most aspects of the report's appearance.
JOINOPT=	NEW OLD	Use when creating a JOIN between two numeric fields of differing formats.
LINES=	57 n	Sets number of printed lines per page.
MSG=	ON OFF	Displays or suppresses messages. This SET command cannot be used in ON TABLE SET.
NODATA=	string	Overrides the period default for missing data. String can be 1 to 11 characters. Used to indicate missing data.
PAGE=	ON OFF NOPAGE	Displays or suppresses page numbers.
PAPER=	66 n	Sets paper length.
PASS=	password [IN file]	Required for protected FOCUS data sources. Use IN to name a specific file. This SET command cannot be used in ON TABLE SET.
PAUSE=	ON OFF	Specifies a pause before report displays.
PCOMMA=	ON OFF	Enables WebFOCUS to read PC type comma-delimited files with double quotation marks around alpha data and a carriage return line feed at the end of each record. This SET command cannot be used in ON TABLE SET.
POOL=	ON OFF	Reads ahead in an application, searching for consecutive TABLE requests that access the same data source using the same access method. This SET command cannot be used in ON TABLE SET.

SET Command	Values	Description
PRINT=	ONLINE OFFLINE	Assigns output to the terminal or printer.
PSPAGESETUP=	ON OFF	Enables you to set the page size for PDF and PostScript reports.
QUALCHAR=	character	Specifies a qualifying character (., :, !, %, , \) for qualified field names. This SET command cannot be used in ON TABLE SET.
QUALTITLES=	ON OFF	Determines whether duplicate field names appear as qualified column titles in reports.
SAVEDMASTERS=	n Q	Saves a Master File to memory after it has been used in a request. n specifies the number of Master Files that can be saved in memory. When set to 0, Master Files are not saved in memory.
SAVEMATRIX=	ON OFF	Saves the last report request that was run to the file allocated to the ddname FOCSORT.
SHADOW=	ON OFF	Activates Absolute File Integrity feature.
SPACES=	n AUTO	Controls the number of spaces between columns in a report: n is from 1 to 8; AUTO uses 1 or 2 spaces.
SQLOPTTF=	ON OFF	Enables the SQL Translator to generate TABLEF commands instead of TABLE commands.
TEMP[Disk]=	disk	Available in CMS only. Specifies disk for temporary work space.
TITLE=	ON OFF	Allows column titles from Master File or field names to be used as column headings.
USER=	password	Required for protected FOCUS data sources.
XRETRIEVAL=	ON OFF	Previews the report format before performing data retrieval.
YRTHRESH=	yy Q	Defines the start of a 100-year window globally or on a field-level. If yy is a positive number, that number is the start of the 100-year window. If yy is a negative number, the start date of the window is derived by subtracting that number from the current year, and the default century is automatically calculated. This SET command cannot be used in ON TABLE SET.

SET Commands for StyleSheets

SET Command	Values	Description
STYLE[SHEET]=	OFF <u>ON</u> stylesheetname	Determines whether StyleSheet parameters are respected or specifies the StyleSheet to be applied to the request.
SQUEEZE=	OFF <u>ON</u>	Removes excess space in report columns.
PAGESIZE=	LETTER A0 A1 A2 A3 A4 A B C D E LEDGER LEGAL	Specifies paper size for report pages.
ORIENTATION=	<u>PORTRAIT</u> LANDSCAPE	Specifies page orientation.
UNITS=	<u>INCHES</u> CM PTS	Specifies unit of measure for page margins.
TOPMARGIN= BOTTOMMARGIN= LEFTMARGIN= RIGHTMARGIN=	n n n n	Specifies top and bottom, left and right boundaries of report contents on printed reports. The value of “n” is a positive number set for each margin.

Querying Your Environment

? DEFINE [filename]	Lists all DEFINE fields or only those for a specified data source.
? F [filename]	Displays current fields from a data source being used in the TABLE environment and for a specified data source.
? FF [filename]	Displays current fields, with alias and format information grouped by segment, from a data source being used in the TABLE environment and for a specified data source.
? FILTER [{filename}*] [SET] [ALL]	Determines the status of existing filters.* displays filters for all Master Files for which filters have been declared. SET displays only active filters and ALL displays all information about the filter, including its description and the exact WHERE/IF definition.
? FUNCTION	Displays all DEFINE functions.
? HOLD [filename]	Displays fields, aliases, and usage formats in the named data source.
? JOIN [filename]	Lists all currently JOINed data sources.
? SET	Displays current SET parameters.
? STYLE	Displays the current settings for page parameters and StyleSheet files.

WebFOCUS Reporting Tools

Tool	Description	Available in
Report Painter	Provides you with many powerful reporting features that enable you to create and style complex reports. You can graphically paint the report on the Report Painter window, a representation of the report page.	Developer Studio
Report Assistant	A graphical tool that allows you to create a styled report and display the report in your browser or another desktop application.	Developer Studio WebFOCUS Dashboard
Graph Assistant	A graphical tool that allows you to easily transform almost any type of data into an effective graph that you can customize to suit your needs.	Developer Studio WebFOCUS Dashboard
HTML Report Assistant	An HTML version of the applet Report Assistant. Functionality is the same.	WebFOCUS Dashboard
HTML Graph Assistant	An HTML version of the applet Graph Assistant. Functionality is the same.	WebFOCUS Dashboard

Information Builders

Corporate Headquarters Two Penn Plaza, New York, NY 10121-2898 (212) 736-4433 Fax (212) 967-0460
World Wide Web: www.informationbuilders.com E-mail: info@informationbuilders.com

Sales and Consulting Offices

- **Atlanta***, GA (770) 395-9913
- **Baltimore**, MD Consulting: (703) 247-5565
- **Boston***, MA (781) 224-7660
- **Charlotte***, NC Consulting: (704) 334-7440
- **Chicago***, IL (630) 971-6700
- **Cincinnati***, OH (513) 891-2338
- **Cleveland***, OH (216) 520-1333
- **Dallas***, TX (972) 490-1300
- **Denver***, CO (303) 770-4440
- **Detroit***, MI (248) 743-3030
- **Federal Systems***, DC (703) 276-9006
- **Hartford**, CT Consulting: (860) 249-7229
- **Houston***, TX (713) 952-4800
- **Kansas City**, MO (816) 471-3320
- **Los Angeles***, CA (310) 615-0735
- **Metropolitan***, NY Sales: (212) 736-7928
Consulting: (212) 736-4433, ext. 4443
- **Minneapolis***, MN (612) 338-5181
- **New Jersey***, (973) 593-0022
- **Orlando***, FL (407) 804-8000
- **Philadelphia***, PA (610) 940-0790
- Pittsburgh, PA (412) 494-9699
- **St. Louis***, MO (636) 519-1411
- **San Jose***, CA (408) 453-7600
- **Seattle***, WA (425) 861-9400
- **Washington***, DC Sales: (703) 276-9006
Consulting: (703) 247-5565

Subsidiaries

- **Australia** Information Builders Pty. Ltd.
Melbourne 61-3-9631-7900
Sydney 61-2-8223-0600
- **Belgium** Information Builders Belgium
Brussels 32-2-7430240
- **Canada** Information Builders (Canada) Inc.
Calgary (403) 538-5415
Montreal* (514) 630-1134
Ottawa (613) 233-0865
Toronto* (416) 364-2760
Vancouver* (604) 688-2499
Victoria (250) 995-8674
- **France** Information Builders France S.A.
Paris 33-14-507-6600
- **Germany** Information Builders (Deutschland)
Dusseldorf 49-211-522877-0
Eschborn 49-6196-77576-0
Munich 49-89-35489-0
Stuttgart 49-711-7287288-0
- **Netherlands** Information Builders (Netherlands) B.V.
Amsterdam 31-20-4563333
- **Portugal** Information Builders Portugal
Lisbon 351-217-230-720
- **Spain** Information Builders Iberica S.A.
Barcelona 34-3-344-32-70
Bilbao 34-4-425-72-24
Madrid 34-1-710-22-75

- Sevilla 34-5-448-83-63
- **Switzerland** Information Builders Switzerland A G
Wallisellen 41-1-8394949
- **United Kindom** Information Builders (UK) Ltd.
London 44-208-9824700
Warrington 44-1925-820111

Representatives

- **Argentina** Ditra S.A., Buenos Aires 54-114-3071850
- **Austria** FOCUS Informationstechnologie GmbH.
Vienna 43-12-1136-3870
- **Colombia Coransa**, Bogota 571-635-0273
- **Finland**, InfoBuild Oy, Helinki 358-9-7250-2250
- **Gulf States** Bahrain, Kuwait, Oman, Qatar, Saudi Arabia, Yemen, United Arab Emirates
Al-Gosaibi Information Systems 973-274-090
- **Hong Kong**
Spherion Ltd. (HK) 852-2834-4788
Digital Empires Co. Ltd. 852-2780-4949
- **Ireland** ACSIS Technologies (Ireland) Ltd.
Dublin 353-1-8725233
- **Israel** NESS A.T. Ltd., Tel Aviv 972-3-5483638
- **Italy** Selesta G C Applications S.P.A.
Genova 39-010-64201-224
Milan 39-02-2515181
Torino 39-011-5513-211
- **Japan** K.K. Ashisuto
Osaka 81-6-6373-7113
Tokyo 81-3-3437-0651
- **Simplex Solutions**
Tokyo 81-3-5652-3180/81
- **Korea** Unitech Infocam Co. Ltd.
Seoul 82-2-3477-4456
- **Malaysia** Optegra Sdn Bhd, Selangor 60-3-80240188
- **Norway** iSolutions A S, Stavanger 47-81544011
- **Philippines** Corporate Information Solution, 63-2-633-1321
- **Poland** Comfort/Meridian Polska SP
Warsaw 4822-630-2660
- **Singapore** Spherion Ltd. (Singapore)
Singapore 65-2255077
- **South Africa** International Computers S.A. (Pty.) Ltd.
Johannesburg 27-11-2335911
- **Sweden** Cybernetics Business Solutions A B
Solna 46-7539900
- **Taiwan** Galaxy Software Services,
Taipei 886-22-3897722
- **Turkey** Istanbul
Erdemsoft 90-212-257-5555
Key Soft Ltd. 90-216-428-5933
- **Venezuela** InfoServices Consulting
Caracas 58-212-763-1653

Toll-Free Numbers

- **Sales and Information** (800) 969-INFO
- **VAR and Reseller Information** (800) 969-4636

*Training facilities are located at these branches; additional locations are available.