**1. Network participation fees**  
Issuing banks (the cardholder's bank) pay a service fee that is a percentage of the payments value processed on cards (debit, credit, or prepaid) issued on that network.    
  
Acquiring banks (the merchants bank) also pay a service fee that is a percentage of the payments value processed through that network.    
  
**2. Data processing fees**  
Issuing and acquiring banks also pay small fees for each transaction that is authorized or settled on its network.  These fee are not related to the value of the transaction.    
  
**3. International transaction fees**  
Revenue from currency conversion and other international transaction fees.  
  
**4. Other**  
Revenue from operating rewards programs, concierge services, marketing promotions for certain merchants, and other programs that issuing banks offer to cardholders.

What you do ?

Enterprise level reporting for enterprsie applications

prepared various dashboard on enterprise cirtical reports and published them

Google charts and intermediate D# library for visulization\

Coordination with IB Team for various issues

spear headed migration project from 8009 to 8105

Traineed team

SQL Server procuderes

Dynamic master files

Faced various difficulties

Developed reusuable compoenent for charts

Developed workaround for gantt chart

developed PMF dashboard where we calculate stat formulas optimized way

Optimization in CI search dashbaord

recent issues

x axis triming

focexeec not working correctly

debuggin migration issues with Sys admin

Filedef and match:

SUBTOTAL: Only mentioned sort field subtotal - > Recompute(only that field recomputed

SUB-TOTAL : Displays all hierarchical subtotals - >Summarize (Every break it recomputes)

Define and compute:Never use LAST keyword with define

**Join and Match:**

The WebFOCUS manual called "Creating Reports With WebFOCUS Language" has two chapters dedicated to "Joining Data Sources" and "Merging Data Sources".  
  
The two Information Builders training courses that explain JOIN and MATCH in a fair amount of detail are:  
Course 351: Building Reports with Report Painter (Part 1)  
Course 354: Building Reports with Report Painter (Part 2)  
The 2nd course compares JOIN ands MATCH.  
  
A (somewhat long-winded) helicopter view:  
A JOIN is a temporary, virtual connection between two or more physical data sources. The data in the host file determines which records are retrieved from the cross-reference file. When a record from the host file of a one-to-many join has no corresponding record(s) in the cross-reference file, the record is automatically excluded from the report output. To control this, you could SET ALL to ON, in which case records from the host file, without corresponding records in the cross-reference file, will be displayed in the report output.  
  
If you JOIN two files: WebFOCUS retrieves a record from the first data source (host file) and then searches for a corresponding record (or records) in the second file (cross-reference file), and stores the results in an Internal Table, which is eventually formatted to produce the report. NB: Only matching records are retrieved from the second file.  
  
MATCH allows you to merge information from two or more data sources and capture the results (selected records and requested columns) in a HOLD file. If a report needs to be produced, then it must run as a subsequent request off the HOLD file.  
  
If you MATCH two files: WebFOCUS retrieves the requested records from the first data source and writes the output to a temporary work area. WebFOCUS then retrieves the requested records from the second data source and writes the output to a temporary work area. WebFOCUS then merges the results based on the high-order sort fields and writes the merged results to an Internal Table. The results in the Internal Table are then written to a HOLD file.  
If a report is required, you must then run a TABLE request against the resultant HOLD file.  
  
So there is a lot more processing, and potentially Disk IO, that goes on in a MATCH process than in a JOIN process!  
  
If the required indexes etc. are in place, and if you need an Inner Join or a Left Outer Join, then generally your starting point would be a JOIN.  
However, MATCH is ideally suited if you need more capabilities than JOIN – such as merging data sources which do not have indexes, or such as merging two unsorted flat files, or such as merging ALL the requested records from the first data source with ALL the requested records from the second data source.

**FOC\_NONE and FOC\_NULL**

FOC\_NONE causes the line of code to be ignored, as if that line were blank. The line need not be a WHERE phrase -- for example, it can be employed to eliminate a BY clause or a verb object.

\_FOC\_NULL forces a WHERE condition to evaluate as "True", regardless of the incoming data values.

**1) Explain the Master file Approach of generating reports?**

**2) Explain the webfocus Architecture.**

**3) What are all the Types of Joins available in Webfocus?**

**4) How Right Outer Join can be implemented in Webfocus?**

(Match Command) Match command with New phrase

**5) What is segment, Root Segment in a master file?**

A segment is a group of fields that have a one-to-one correspondence with each other and usually describe a group of related characteristics. In a relational data source, a segment is equivalent to a table. Segments are the building blocks of larger data structures.

**6) What is HTML Painter? Report Painter...? Report Assistant?**

**7) What are Dialogue parameter variables?**

**8) How to execute a stored procedure in FEX code?**

ENGINE SQLORA SET VARCHAR OFF

ENGINE SQLORA SET DEFAULT\_CONNECTION dev1

ENGINE SQLORA

EX FRD\_MAIN.SOLV\_UPD\_RPT\_DT '&ACCT','&REQDATE','&V\_OUT';

**10) What is am Accordin Report, Cross Tab Report, Drill down**

**Report?**

ON TABLE SET EXPANDABLE ON

**11) Is it possible to include input elements in a report output?**

**12) What is MRE?**

**13) While using master file to generate a report, it takes 5**

**minutes for you to fetch the output... What are all you have to**

**do to improve the performance?**

**14) What is Report Caster? What are the various options**

**available in Report Caster to deliver reports?**

**15) What are the different Kind of Mails that can be used to**

**deliver reports in Report Caster?**

**16) What is Report Bursting? Why is it used for?**

**17) What are the various scheduling Frequencies available in**

**Report Caster?**

**18) What is Error code you have faced to the most while**

**implementing FEX code?**

**19) Architecture of Developer Studio?**

**20) What are all the directories that are available while you**

**create a new project using dev studio?**

**21) What you have to do to get alternate row color on a Table**

**output using FEX?**

BACKCOLOR=( 'WHITE' RGB(238 238 238) )

**22) What is the concept of Hold File Reuse? Advantages...**

**23) How webfocus is superior / worse compared to other**

**Reporting tools?**

**24) Explain the iWay flow of ETL?**

**25) What is an Active HTML Report?**

**26) What are kind of Charts / Graphs that can be generated**

**using webfocus?**

**27) What is a Compound Report?**

-\* File multi\_page3.fex

-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-MULTI\_PAGE3

-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-SET &HDR\_PASS = 0;

SET HTMLARCHIVE=ON

COMPOUND LAYOUT PCHOLD FORMAT PDF

UNITS=IN, $

SECTION=section1, LAYOUT=ON, METADATA='0.5^0.5^0.5^0.5', MERGE=OFF, ORIENTATION=PORTRAIT, PAGESIZE=Letter, $

PAGELAYOUT=1, NAME='Page layout 1', text='Page layout 1', TOC-LEVEL=1, BOTTOMMARGIN=0.0, TOPMARGIN=0.0, METADATA='0^0.5^0^0.5', $

-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-\* LEFT TOP WIDTH HEIGHT

-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

COMPONENT='report8', TEXT='report8', TOC-LEVEL=2, POSITION=(0.167 0.156), DIMENSION=(8.177 0.552), METADATA='Z-INDEX: 100; LEFT: 0.167in; WIDTH: 8.177in; POSITION: absolute; TOP: 0.156in; HEIGHT: 0.552in', $

COMPONENT='report1', TEXT='report1', TOC-LEVEL=2, POSITION=(0.140 0.923), DIMENSION=(4.250 6.700), METADATA='Z-INDEX: 100; LEFT: 0.140in; WIDTH: 4.250in; POSITION: absolute; TOP: 0.923in; HEIGHT: 6.700in', $

COMPONENT='report2', TEXT='report2', TOC-LEVEL=2, POSITION=(4.427 0.923), DIMENSION=(4.250 6.750), METADATA='Z-INDEX: 100; LEFT: 4.427in; WIDTH: 4.250in; POSITION: absolute; TOP: 0.923in; HEIGHT: 6.750in', $

PAGELAYOUT=2, NAME='Page layout 2', text='Page layout 2', TOC-LEVEL=1, BOTTOMMARGIN=0.0, TOPMARGIN=0.0, METADATA='0^0.5^0^0.5', $

COMPONENT='report9', TEXT='report9', TOC-LEVEL=2, POSITION=(0.167 0.156), DIMENSION=(8.177 0.552), METADATA='Z-INDEX: 100; LEFT: 0.167in; WIDTH: 8.177in; POSITION: absolute; TOP: 0.156in; HEIGHT: 0.552in', $

COMPONENT='report3', TEXT='report3', TOC-LEVEL=2, POSITION=(0.167 0.923), DIMENSION=(4.250 6.700), METADATA='Z-INDEX: 103; LEFT: 0.167in; WIDTH: 4.250in; POSITION: absolute; TOP: 3.350in; HEIGHT: 6.700in', $

COMPONENT='report5', TEXT='report5', TOC-LEVEL=2, POSITION=(4.427 0.923), DIMENSION=(4.250 6.750), METADATA='Z-INDEX: 104; LEFT: 4.427in; WIDTH: 4.250in; POSITION: absolute; TOP: 3.350in; HEIGHT: 6.750in', $

PAGELAYOUT=3, NAME='Page layout 3', text='Page layout 3', TOC-LEVEL=1, BOTTOMMARGIN=0.0, TOPMARGIN=0.0, METADATA='0^0.5^0^0.5', $

COMPONENT='report10', TEXT='report10', TOC-LEVEL=2, POSITION=(0.167 0.156), DIMENSION=(8.177 0.552), METADATA='Z-INDEX: 100; LEFT: 0.167in; WIDTH: 8.177in; POSITION: absolute; TOP: 0.156in; HEIGHT: 0.552in', $

COMPONENT='report7', TEXT='report7', TOC-LEVEL=2, POSITION=(0.140 0.923), DIMENSION=(4.250 6.000), METADATA='Z-INDEX: 100; LEFT: 0.140in; WIDTH: 4.250in; POSITION: relative; TOP: 0.923in; HEIGHT: 6.000in', $

COMPONENT='report4', TEXT='report4', TOC-LEVEL=2, POSITION=(4.427 0.923), DIMENSION=(3.250 6.500), METADATA='Z-INDEX: 100; LEFT: 4.427in; WIDTH: 3.250in; POSITION: relative; TOP: 0.923in; HEIGHT: 6.500in', $

COMPONENT='report6', TEXT='report6', TOC-LEVEL=2, POSITION=(0.538 9.479), DIMENSION=(6.875 0.938), METADATA='Z-INDEX: 100; LEFT: 0.538in; WIDTH: 6.875in; POSITION: absolute; TOP: 9.479in; HEIGHT: 0.938in', $

END

SET COMPONENT='report8'

-\*component\_type report

-INCLUDE &SHRPATH.EVAL/xrpt\_main\_hdr

SET COMPONENT='report1'

-\*component\_type report

-INCLUDE &SHRPATH.EVAL/xrpt00012

SET COMPONENT='report2'

**28) What is a PDF Layout Painter, what for it is used for?**

**29) What’s that you can do to segregate the Common Style**

**information that is used across many reports?**

**30) How logging / Debugging can be done in webfocus?**

This very much depends on the type of error you get.  
  
If the agent crashes, then you will probably have to turn tracing on, on the server.  
  
If you get an error, always check the first error found, not the last.  
  
There should be a line number with the error, it should be the line the error is on. Sometimes it can be the previous error.  
  
Always research the error, understand what the message means, and any hints as to what the error is about, like the field name or column number, etc.  
  
With difficult errors, I always start breaking the report down, to find the cause, Add a breakpoint (-EXIT), and then test to see if the report works to this point correctly.  
  
Turn code echoing on with -SET &ECHO=ALL;, this can highlight issues.  
  
Turn SQL tracing on with:

SET TRACEOFF = ALL

SET TRACEON = SQLAGGR//CLIENT

SET TRACEON = STMTRACE//CLIENT

SET TRACEON = STMTRACE/2/CLIENT

SET TRACEUSER = ON

This could highlight issues with SQL generation against the tables.  
  
There are many more things that could be done, but as mentioned before, it is relative to the type of debugging needed.

* + **In TABLE:**
    - **Use LET NOPRINT = ; to show complete request**
    - **Use SET DEFECHO to show variable substitution**

**31) What are the Standards that need to be followed while**

**developing a Report?**

**New features:**

* Responsive designs in bi portal
* Annotation in JS Charts
* Too tip arrangement
* Prompt behavior
* Parameters are by default passed with quotes

**Architecture:**

WebFOCUS unobtrusively integrates into your existing network by connecting your Web server to your data. End users access WebFOCUS applications through a Web browser, so they need only the following elements:

**Web browser.** To access WebFOCUS applications, users simply need a browser and a TCP/IP connection to a Web server.

**Web server.** Web servers handle requests by returning files to a browser or by executing processes that provide additional functionality. You can provide WebFOCUS functionality by connecting to the Web server using either the Java servlets, CGI, or ISAPI.

**Data.** WebFOCUS can access data from almost anywhere. Once data access has been configured and your data is described, reporting on it is simple.

WebFOCUS Components

There are two main WebFOCUS components.

**WebFOCUS Client.** The WebFOCUS Client resides on the Web server and connects WebFOCUS to the Web through either the Java servlets, CGI, or ISAPI. When a user makes a request from Developer Studio or a browser, the WebFOCUS Client receives and processes the request by passing it to the WebFOCUS Reporting Server.

**Note:** When you perform a full Developer Studio installation for stand-alone development, you do not have to install the WebFOCUS Client separately because it is packaged with the WebFOCUS Reporting Server.

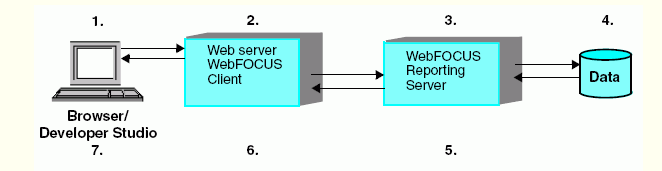
A stand-alone development environment is typically one in which all software components (the Web server, WebFOCUS Client, and WebFOCUS Reporting Server) are installed on the same local machine. This configuration gives you access to all your application files and data from a single machine. You do not need a physical network connection to access any other machine in order to accomplish your development tasks.

**WebFOCUS Reporting Server.** The WebFOCUS Reporting Server resides on machines that can access your data. The WebFOCUS Reporting Server provides data access, number crunching, and report generation functionality.

WebFOCUS Processing

The following steps accompany the figure and describe how WebFOCUS processes requests:

1. A user requests a report and passes parameters by calling a WebFOCUS Servlet, CGI, or ISAPI through links and forms on a Web page or through Developer Studio.
2. The request and parameters come to the WebFOCUS Client on the Web server, which processes the parameters and creates a request for the WebFOCUS Reporting Server.
3. The WebFOCUS Reporting Server receives the request, processes it, and accesses any necessary data.
4. Data is retrieved from data sources to process the request.
5. The WebFOCUS Reporting Server processes the user's request using the retrieved data.
6. The response is returned to the WebFOCUS Client on the Web server.
7. The response is returned to the user.



WebFOCUS Configuration

WebFOCUS employs a distributed architecture, so the WebFOCUS Client, WebFOCUS Reporting Server, and your data can be located on any platform, anywhere in your network. You can easily connect an Apache Web server running on UNIX to SQL Server data on Windows or DB2 data on z/OS. There can be any number of WebFOCUS Reporting Servers connected to the WebFOCUS Client. WebFOCUS can report on all of them. The requirements are the following:

The WebFOCUS Client must reside on a machine with a Web server.

An instance of the WebFOCUS Reporting Server must be installed on machines with your data or machines that have access to your data.

For more information on WebFOCUS configuration options, see the WebFOCUS and ReportCaster installation manuals and the server configuration manuals.

**&ECHO:**

{-DEFAULT|-SET|EX} &ECHO = {ON|ALL|OFF}

where:

ON

Displays FOCUS commands that are expanded and stacked for execution.

ALL

Displays Dialogue Manager commands and FOCUS commands that are expanded and stacked for execution.

OFF

Suppresses the display of both stacked commands and Dialogue Manager commands. This value is the default.

* The &ECHO variable controls the display of command lines as they execute so you can test and debug procedures.

The &STACK variable enables you to test the logic of Dialogue Manager commands. Setting this variable to OFF lets you run the procedure while preventing the execution of stacked commands. This gives you the ability to view the sequence of commands and see how the variable values are resolved.

The &RETCODE variable returns a code after a procedure is executed. If the procedure results in normal output or no records are retrieved, the value of &RETCODE is 1. If an error occurs while parsing the procedure, the value of &RETCODE is 8.

&RETCODE can be used to test the result of an operating system command. This retrieves the return code from the operating system.

The &IORETURN variable tests the result of Dialogue Manager -READ and -WRITE commands. After a -READ or -WRITE operation, a non-zero return code indicates an error such as end-of-file being reached.

&IORETURN can be used to test the result of the following:

A -READ command. If &IORETURN equals zero, a value was successfully read from the external file.

A -WRITE command. If &IORETURN equals zero, a value was successfully written to the external file.

**Managed Reporting Concepts**

Managed Reporting includes the following components:

**Domains.** Domains are the highest level of organization. Domains provide data on a particular topic (such as sales, inventory, or personnel). The data is stored in different forms in the following domain components: predefined reports (Standard Reports), data sources used to create reports (Reporting Objects), and reports created and saved by users (My Reports, Custom Reports, and Shared Reports).

**Standard Reports.** A Standard Report is a pre-defined procedure that your Administrator creates and stores in a group folder or subgroup folder. You use Standard Reports to retrieve data that changes on a regular basis, for example, monthly inventory reports or weekly sales reports. Each time you run a Standard Report the output reflects the most current data, while the format of the report remains constant.

**Reporting Objects.** A Reporting Object is a tailored view of a set of data that your Administrator creates and saves to a group folder. You use the data contained in a Reporting Object to create personal reports quickly and in compliance with the reporting rules and guidelines of your company.

**My Reports.** A My Report is a personal report you save while working in a domain. Once you access a Reporting Object and create a report, you can save the report as a My Report. Once saved, you can run or edit these reports. No other user has access to your reports.

**Custom Reports.** A Custom Report is a report that you create and edit using Report Assistant, Graph Assistant, or the Editor. Custom Reports are located in the Custom Reports folder located under the My Reports tab in the Domains environment. Custom Reports are available to users who have been granted the Advanced privilege in Managed Reporting. Administrators automatically have the Advanced privilege and they can assign this privilege to other users and roles.

**Shared Reports.** A Shared Report is a My Report or Custom Report that another user has prepared and saved with the Shared Report capability. You can run a Shared Report from the Shared Reports tab. You can also copy it to your My Reports tab and then modify it without affecting the original report.

**Note:** Although you can share a Custom Report with other users, if those other users do not have the Advanced privilege, those users can only run and run deferred the report. They cannot save the report. If those users has the Advanced privilege, then those users can run, run deferred, save, and edit the saved copy.

**Static Reports.** A Static Report is a type of Standard Report in which the output never changes. Unlike a regular Standard Report, which always reflects current data, a Static Report delivers a snapshot of data from a specific time. For example, a Static Report may be a Web page that contains a report.

**Help System.** Each domain may also contain a customized help system that you can access for specific information about your implementation of Managed Reporting.

**SUB-TOTAL and SUBTOTAL:**

You can use the SUBTOTAL and SUB-TOTAL commands to sum individual values, such as columns of numbers, each time a named sort field changes value.

SUB-TOTAL displays a subtotal for all numeric values when the sort field changes value, and for any higher-level sort fields when their values change.

SUBTOTAL displays a subtotal only when the specified sort field changes value. It does not give subtotals for higher-level fields.

Both SUB-TOTAL and SUBTOTAL produce grand totals. You can suppress grand totals using the NOTOTAL command. See [Suppressing Grand Totals](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic72.htm#dtot1011025).

The subtotal is calculated every time the sort field value changes or, if WHEN criteria are applied to the sort field, every time the WHEN conditions are met.

A BY, ACROSS, or ON phrase is required to initialize the syntax.

When using a SUM or COUNT command with only one sort phrase in the request, SUB-TOTAL and SUBTOTAL produce the same result as the value of the SUM or COUNT command. However, when using a PRINT command with one sort phrase, SUBTOTAL is useful because there can be many values within a sort break.

All SUB-TOTALs display up to and including the point where the sort break occurs, so only the innermost point of subtotaling should be requested. For instance, if the BY fields are

BY AREA

BY PROD\_CODE

BY DATE SUB-TOTAL

then, when AREA changes, subtotals are displayed for DATE, PROD\_CODE, and AREA on three lines (one under the other).

If you use a WHERE TOTAL or IF TOTAL test, the display of the sort field value for the subtotal line is suppressed unless PRINTPLUS is ON. For details about using PRINTPLUS in WebFOCUS, see [Using PRINTPLUS](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic213.htm#dcr28_HeadFootTitleLabel1454997).

Subtotals display on the next line if the subtotal text does not fit on the line prior to the displayed field columns.

**SUMMARIZE AND RECOMPUTE:**

You can use the SUMMARIZE and RECOMPUTE commands instead of SUB-TOTAL and SUBTOTAL to recalculate the result of a COMPUTE command. SUMMARIZE is similar to SUB-TOTAL in that it recomputes values at every sort break. RECOMPUTE is similar to SUBTOTAL in that it recalculates only at the specified sort break.

SUMMARIZE

Recomputes values at every sort break.

RECOMPUTE

Recalculates values only at the specified sort break.

SUMMARIZE recomputes grand totals for the entire report. If you wish to suppress grand totals, you can include the NOTOTAL command in your request.

**Specifying a Target Frame**

You can use frames to subdivide application HTML pages into separate scrollable sections. Frames enable users to explore various information items on a page by scrolling through a section, instead of linking to a separate page. When defining a link from a report component to a report procedure or URL, you can specify that the results of the drill-down link be displayed in a target frame on a Web page.

There are two ways to specify a target frame. You can specify:

A target frame in a StyleSheet declaration using the TARGET attribute. You can use StyleSheets to specify that drill-down links from a report or graph are displayed in a target frame on the Web page displaying the report or graph. However, using StyleSheets to specify target frames adds extra HTML syntax to every HREF that is generated.

**Note:** When specifying a target frame from the Report Painter, manually added commands in the StyleSheet are not recognized. The Report Painter removes commands that it does not generate itself.

A default target frame with a SET command. SET TARGETFRAME puts the HTML code <BASE TARGET="framename"> into the header of the HTML file that WebFOCUS displays. All drill-down links from the base report or graph are directed to the specified frame, unless overridden by the TARGET attribute in the StyleSheet.

To use the TARGET attribute or the SET TARGETFRAME command, you must create multiple frames on the Web page.

**Note:** You cannot specify a target frame if you are executing a JavaScript function. However, the JavaScript function itself can specify a target frame for its results.

*Syntax:* How to Specify a Target Frame

To specify a target frame in a report or procedure use:

TYPE=type, [subtype], FOCEXEC=fex[(parameters ...)], [TARGET=frame,] $

To specify a target frame for an URL use:

TYPE=type, [subtype], URL=url[(parameters ...)], [TARGET=frame,] $

where:

type

Identifies the report component that the user selects in the Web browser to execute the link. The TYPE attribute and its value must appear at the beginning of the declaration.

subtype

Are any additional attributes, such as COLUMN, LINE, or ITEM, that are needed to identify the report component that you are formatting. See [Identifying a Report Component in a WebFOCUS StyleSheet](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic169.htm#dcr25_Identifying1019919), for information on identifying report components.

fex

Identifies the file name of the linked procedure to run when the user selects the report component. For details about linking to another procedure, see [Linking to Another Report](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic109.htm#dcr11_Linking1008780).

url

Identifies any valid URL, or the name of a report column enclosed in parentheses whose value is a valid URL to which the link will jump. For details about linking to an URL, see [Linking to a URL](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic110.htm#dcr11_Linking1008865).

parameters

Are values being passed to the procedure or URL. You can pass one or more parameters. The entire string of values must be enclosed in parentheses, and separated from each other by a blank space. For details, see [Creating Parameters](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic114.htm#dcr11_Linking1086634).

frame

Identifies the target frame in the Web page in which 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. For example:

TYPE=DATA, COLUMN=N1,

FOCEXEC=MYREPORT, TARGET=MY FRAME, $

The name of the target frame is correctly interpreted to be MY FRAME.

You can also use the following standard HTML frame names: \_BLANK, \_SELF, \_PARENT, \_TOP.

**Note:** When specifying a target frame from the Report Painter, manually added commands in the StyleSheet are not recognized. For example, entering TARGET=NEW is removed when the procedure is run in the Report Painter. The Report Painter removes commands that it does not generate itself. For this example, set the target \_BLANK to open a drill down report in a new window.

*Syntax:* How to Specify a Default Target Frame

SET TARGETFRAME=frame

where:

frame

Identifies the target frame in the Web page in which the output from the drill-down link (either a FOCEXEC or URL) is displayed.

*Example:* Specifying a Target Frame

The following illustrates how to specify a default target frame:

SET TARGETFRAME=\_SELF

The following illustrates how to specify a target frame in a request. The relevant StyleSheet declaration is highlighted in the request.

TABLE FILE EMPLOYEE

PRINT CURR\_SAL

BY DEPARTMENT

ON TABLE SET STYLE \*

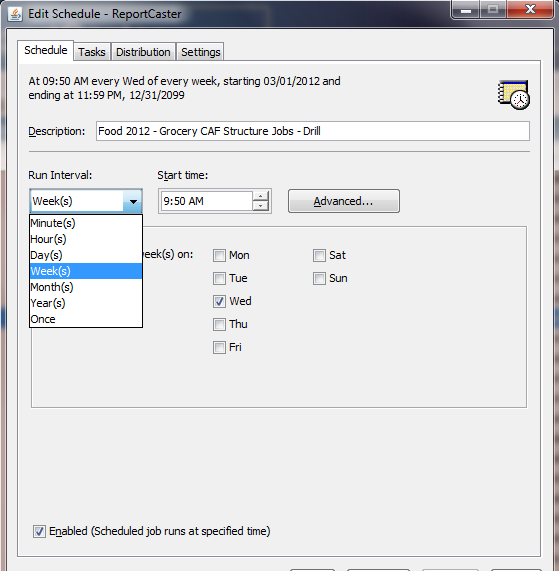
TYPE=DATA, COLUMN=N1, URL=http:\\www.informationbuilders.com,

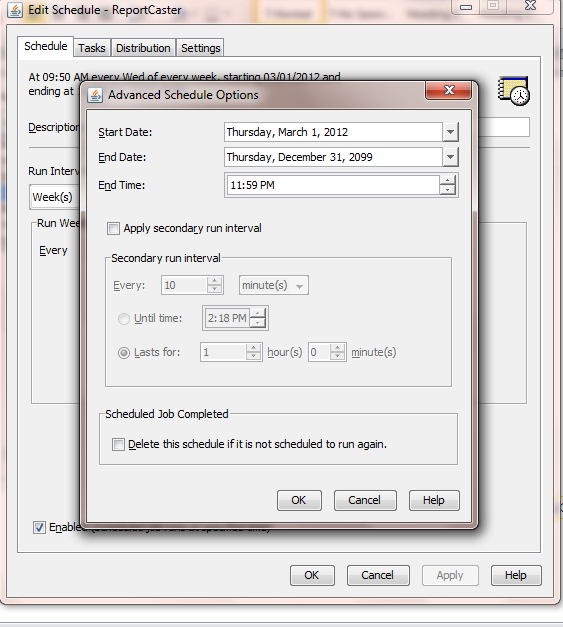
TARGET=\_SELF, $

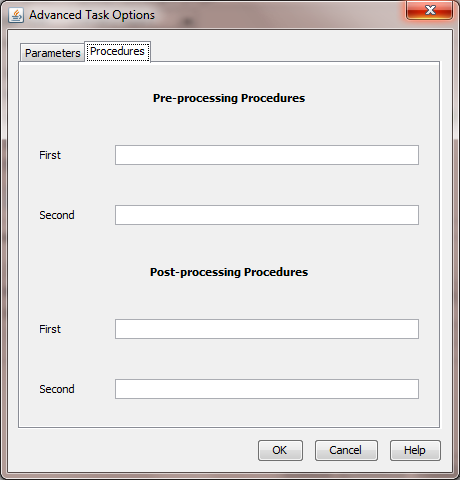
ENDSTYLE

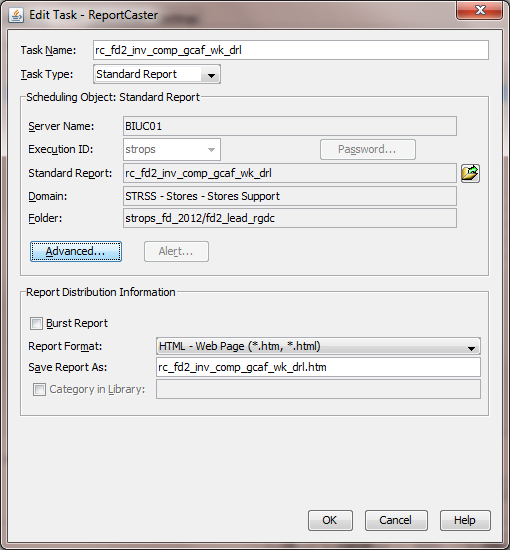
END

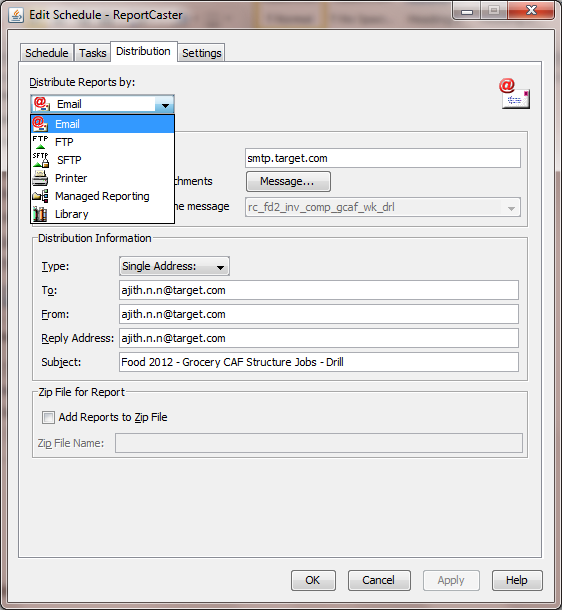
**Report Caster:**

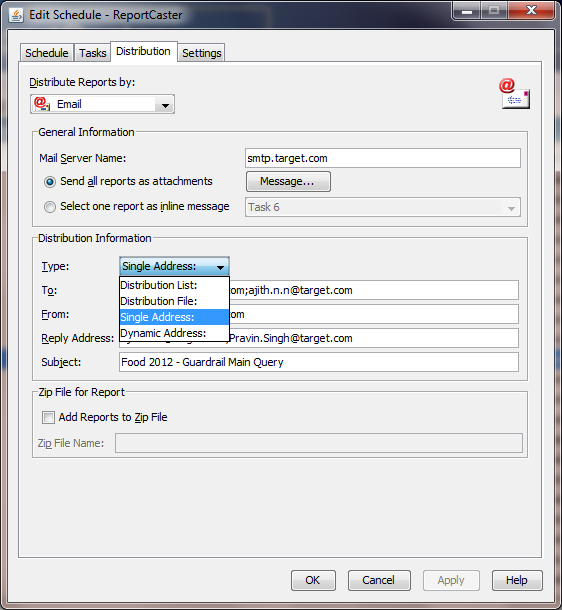


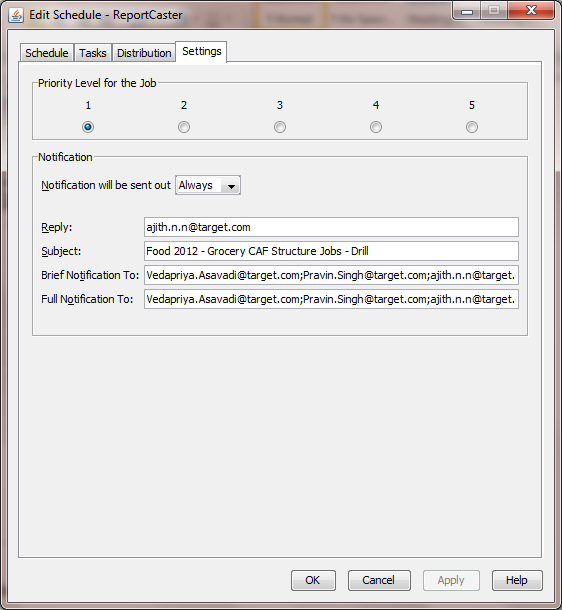












**LAST command:**

Available Languages: reporting

The LAST function retrieves the preceding value for a field.

The effect of LAST depends on whether it appears in a DEFINE or COMPUTE command:

In a DEFINE command, the LAST value applies to the previous record retrieved from the data source before sorting takes place.

In a COMPUTE command, the LAST value applies to the record in the previous line of the internal matrix.

Do not use LAST with the -SET command in Dialogue Manager.

Code to use Excel Template:

FILEDEF TEMP1 DISK temp1.mht

ON TABLE HOLD AS 'temp1' FORMAT EXL2K TEMPLATE 'store\_template\_fs' SHEETNUMBER 1

ON TABLE HOLD AS 'temp1' FORMAT EXL2K TEMPLATE 'store\_template\_fs' SHEETNUMBER 1

**You can test and debug your procedure with the following.**

The &ECHO variable controls the display of command lines as they execute so you can test and debug procedures.

The &STACK variable enables you to test the logic of Dialogue Manager commands. Setting this variable to OFF lets you run the procedure while preventing the execution of stacked commands. This gives you the ability to view the sequence of commands and see how the variable values are resolved.

The &RETCODE variable returns a code after a procedure is executed. If the procedure results in normal output or no records are retrieved, the value of &RETCODE is 1. If an error occurs while parsing the procedure, the value of &RETCODE is 8.

&RETCODE can be used to test the result of an operating system command. This retrieves the return code from the operating system.

The &IORETURN variable tests the result of Dialogue Manager -READ and -WRITE commands. After a -READ or -WRITE operation, a non-zero return code indicates an error such as end-of-file being reached.

&IORETURN can be used to test the result of the following:

A -READ command. If &IORETURN equals zero, a value was successfully read from the external file.

A -WRITE command. If &IORETURN equals zero, a value was successfully written to the external file.

* *The .EXIST suffix tests the presence of a value.*
* *The .EVAL suffix tests the value of a variable.*
* *The .LENGTH suffix tests the length of a value.*
* *The .TYPE suffix tests the type of a value.*

***Syntax:* How to Display Explanations of Error Messages**

? n

where:

n

Is the error message number.

*Example:* Displaying Explanations of Error Messages

If you receive the message

(FOC125) RECAP CALCULATIONS MISSING

issuing the command

? 125

produces the following message:

(FOC125) RECAP CALCULATIONS MISSING

The word RECAP is not followed by a calculation. Either the RECAP should

be removed, or a calculation provided.

**You can perform a function** repeatedly by using looping in your procedure with the -REPEAT command. Looping can be used for many tasks. For example, files can be named and renamed by embedding operating system calls within a Dialogue Manager loop. Indexed variables can also be populated using a loop, or the output of a request can be used for a second request.

A process loop can be executed a designated number of times or until a condition is met. A loop ends when any of the following occurs:

It is executed in its entirety.

A -QUIT or -EXIT command is issued.

A -GOTO is issued to a label outside of the loop.

**Note:** If you issue another -GOTO later in the procedure to return to the loop, the loop proceeds from the point it left off.

You can also limit the repetition of a loop by incrementing a variable with the -SET command.

*Syntax:* How to Specify a Loop

-REPEAT label n TIMES

or

-REPEAT label WHILE condition

or

-REPEAT label FOR &variable [FROM fromval] [TO toval] [STEP s]

where:

label

Identifies the code to be repeated (the loop). A label can include another loop if the label for the second loop has a different name than the first.

n TIMES

Specifies the number of times to execute the loop. The value of n can be a local variable, a global variable, or a constant. If it is a variable, it is evaluated only once, so you cannot change the number of times to execute the loop. The loop can only be ended early using -QUIT or -EXIT.

WHILE condition

Specifies the condition under which to execute the loop. The condition is any logical expression that can be true or false. The loop executes if the condition is true.

&variable

Is a variable that is tested at the start of each execution of the loop. It is compared with the value of fromval and toval, if supplied. The loop is executed only if &variable is less than or equal to toval when s is positive, or greater than or equal to toval when s is negative.

fromval

Is the minimum value of &variable that will execute a loop. 1 is the default value.

toval

Is the maximum value of &variable that will execute a loop. 1,000,000 is the default value.

STEP s

Increments the value of &variable by a constant, s. It may be positive or negative. The default increment is 1.

**Note:** The parameters FROM, TO, and STEP can appear in any order.

***Example:* Listing Records**

To number the records in a report, use the LIST command.

TABLE FILE EMPLOYEE

LIST LAST\_NAME AND FIRST\_NAME

END

The following shows the report output.

|  |  |  |
| --- | --- | --- |
| LIST  ----    1  2  3  4  5  6  7  8  9  10  11  12 | LAST\_NAME  ---------    STEVENS  SMITH  JONES  SMITH  BANNING  IRVING  ROMANS  MCCOY  BLACKWOOD  MCKNIGHT  GREENSPAN  CROSS | FIRST\_NAME  ----------    ALFRED  MARY  DIANE  RICHARD  JOHN  JOAN  ANTHONY  JOHN  ROSEMARIE  ROGER  MARY  BARBARA |

**Ranking Sort Field Values**

When you sort report rows using the BY phrase, you can indicate the numeric rank of each row. Ranking sort field values is frequently combined with restricting sort field values by rank.

Note that it is possible for several report rows to have the same rank if they have identical sort field values.

The default column title for RANKED BY is RANK. You can change the title using an AS phrase. The RANK field has format I7. Therefore, the RANK column in a report can be up to seven digits. For more information, see [Using Headings, Footings, Titles, and Labels](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\cr28_headfoottitlelabel.htm#dcr28_HeadFootTitleLabel1012308).

You can rank aggregated values using the syntax RANKED BY TOTAL. For details, see [Sorting and Aggregating Report Columns](file:///C:\Users\dt206898\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\U80U5YPG\wf761crlang_topic36.htm#WS3AB29FEC-286A-4315-8B86-AC4AAC621EC9).

*Syntax:* How to Rank Sort Field Values

RANKED [AS 'name'] BY sortfield

where:

sortfield

Is the name of the sort field. The field can be numeric or alphanumeric.

name

Is the new name for the RANK column title.

*Example:* Ranking Sort Field Values

Issue the following request to display a list of employee names in salary order, indicating the rank of each employee by salary. Note that employees Jones and McCoy have the same rank since their current salary is the same.

TABLE FILE EMPLOYEE

PRINT LAST\_NAME

RANKED AS 'Sequence' BY CURR\_SAL

END

The output is:

Ranking Sort Field Values

How to:

Rank Sort Field Values

When you sort report rows using the BY phrase, you can indicate the numeric rank of each row. Ranking sort field values is frequently combined with restricting sort field values by rank.

Note that it is possible for several report rows to have the same rank if they have identical sort field values.

The default column title for RANKED BY is RANK. You can change the title using an AS phrase. The RANK field has format I7. Therefore, the RANK column in a report can be up to seven digits. For more information, see Using Headings, Footings, Titles, and Labels.

You can rank aggregated values using the syntax RANKED BY TOTAL. For details, see Sorting and Aggregating Report Columns.

Syntax: How to Rank Sort Field Values

RANKED [AS 'name'] BY sortfield

where:

sortfield

Is the name of the sort field. The field can be numeric or alphanumeric.

name

Is the new name for the RANK column title.

Example: Ranking Sort Field Values

Issue the following request to display a list of employee names in salary order, indicating the rank of each employee by salary. Note that employees Jones and McCoy have the same rank since their current salary is the same.

TABLE FILE EMPLOYEE

PRINT LAST\_NAME

RANKED AS 'Sequence' BY CURR\_SAL

END

The output is:

Example: Ranking and Restricting Sort Field Values

Ranking sort field values is frequently combined with restricting sort field values by rank, as in the following example.

TABLE FILE EMPLOYEE

PRINT LAST\_NAME

RANKED BY HIGHEST 5 CURR\_SAL

END

The output is:

RANK CURR\_SAL LAST\_NAME

---- -------- ---------

1 $29,700.00 BANNING

2 $27,062.00 CROSS

3 $26,862.00 IRVING

4 $21,780.00 BLACKWOOD

|  |  |  |
| --- | --- | --- |
| 5 $21,120.00 ROMANS |  |  |

*Example:* Ranking and Restricting Sort Field Values

Ranking sort field values is frequently combined with restricting sort field values by rank, as in the following example.

TABLE FILE EMPLOYEE

PRINT LAST\_NAME

RANKED BY HIGHEST 5 CURR\_SAL

END

The output is:

RANK CURR\_SAL LAST\_NAME

---- -------- ---------

1 $29,700.00 BANNING

2 $27,062.00 CROSS

3 $26,862.00 IRVING

4 $21,780.00 BLACKWOOD

1. $21,120.00 ROMANS

**Drill down menu:**

TYPE=DATA,

COLUMN=N1,

DRILLMENUITEM='Get Bar Graph',

FOCEXEC=wf201\_dm\_graph(PDIV=N1),

TARGET='\_blank',

DRILLMENUITEM='Get Pie Chart',

FOCEXEC=wf201\_dm\_graph2(PDIV=N1),

TARGET='\_blank',

$

**HTMLCSS**

The HTMLCSS parameter creates an internal Cascading Style Sheets command in the HTML display page.

The syntax is:

SET HTMLCSS = {ON|OFF}

where:

ON

Creates an internal CSS command in the HTML page that displays the report output.

OFF

Does not create an internal CSS command in the HTML page that displays the report output. OFF is the default value.

**HOLDLIST:**

The HOLDLIST parameter controls whether only displayed fields or all fields are included in the HOLD or PCHOLD file.

The syntax is:

SET HOLDLIST = {PRINTONLY|ALL|ALLKEYS}

where:

PRINTONLY

Includes only those fields in the HOLD or PCHOLD file that are specified in the report request.

ALL

Includes all fields referenced in a request in the HOLD or PCHOLD file, including both computed fields and fields referenced in a COMPUTE command. ALL is the default value. (OLD may be used as a synonym for ALL.)

**Note:** Vertical sort (BY) fields specified in the request with the NOPRINT option are not included in the HOLD file, even with SET HOLDLIST=ALL.

ALLKEYS

Propagates all fields to the HOLD file, including NOPRINTed BY fields.

**HOLDMISS**:

The HOLDMISS parameter enables you to distinguish between missing data and default values of blank (for character data) or zero (for numeric data) in a HOLD file.

The syntax is:

SET HOLDMISS = {OFF|ON}

where:

OFF

Does not allow you to store missing data in a HOLD file. OFF is the default value.

ON

Enables you to store missing data in a HOLD file. When TABLE generates a default value for data not found, it generates missing values.

***Example:* Creating an Accordion Report**:

This example shows how to use an EXPANDABLE command to create an Accordion Report.

TABLE FILE GGSALES

SUM UNITS DOLLARS

BY REGION BY ST BY CITY BY CATEGORY

ON TABLE SET EXPANDABLE ON

END

**Reference: Usage Notes for Match Requests**

The ACROSS and WHERE TOTAL phrases, and the COMPUTE command, are not permitted in a MATCH request. You can, however, use the DEFINE command.

A total of 32 BY phrases and the maximum number of display fields can be used in each MATCH request. The maximum number of display fields is determined by a combination of factors.

Up to 32 sort sets are supported, including the number of common sort fields.

You must specify at least one BY field for each file used in the MATCH request.

When used with MATCH, the SET HOLDLIST parameter behaves as if HOLDLIST were set to ALL.

You cannot use BY HIGHEST in a MATCH request.

The following prefix operators are not supported in MATCH requests: DST., DST.CNT., RNK., ST., and CT.

The WebFOCUS manual called "Creating Reports With WebFOCUS Language" has two chapters dedicated to "Joining Data Sources" and "Merging Data Sources".  
  
The two Information Builders training courses that explain JOIN and MATCH in a fair amount of detail are:  
Course 351: Building Reports with Report Painter (Part 1)  
Course 354: Building Reports with Report Painter (Part 2)  
The 2nd course compares JOIN ands MATCH.  
  
A (somewhat long-winded) helicopter view:  
A JOIN is a temporary, virtual connection between two or more physical data sources. The data in the host file determines which records are retrieved from the cross-reference file. When a record from the host file of a one-to-many join has no corresponding record(s) in the cross-reference file, the record is automatically excluded from the report output. To control this, you could SET ALL to ON, in which case records from the host file, without corresponding records in the cross-reference file, will be displayed in the report output.  
  
If you JOIN two files: WebFOCUS retrieves a record from the first data source (host file) and then searches for a corresponding record (or records) in the second file (cross-reference file), and stores the results in an Internal Table, which is eventually formatted to produce the report. NB: Only matching records are retrieved from the second file.  
  
MATCH allows you to merge information from two or more data sources and capture the results (selected records and requested columns) in a HOLD file. If a report needs to be produced, then it must run as a subsequent request off the HOLD file.  
  
If you MATCH two files: WebFOCUS retrieves the requested records from the first data source and writes the output to a temporary work area. WebFOCUS then retrieves the requested records from the second data source and writes the output to a temporary work area. WebFOCUS then merges the results based on the high-order sort fields and writes the merged results to an Internal Table. The results in the Internal Table are then written to a HOLD file.  
If a report is required, you must then run a TABLE request against the resultant HOLD file.  
  
So there is a lot more processing, and potentially Disk IO, that goes on in a MATCH process than in a JOIN process!  
  
If the required indexes etc. are in place, and if you need an Inner Join or a Left Outer Join, then generally your starting point would be a JOIN.  
However, MATCH is ideally suited if you need more capabilities than JOIN – such as merging data sources which do not have indexes, or such as merging two unsorted flat files, or such as merging ALL the requested records from the first data source with ALL the requested records from the second data source.

1. **Dialogue Managers: It is used to control execution of flow in the fex file. Basically everything starting with ‘-‘ is dialogue manager. Ex: -SET, -IF, -RUN,-EXIT,-REPEAT etc and many more.**
2. **SQL pass through and Master file Approach: Mas file approach is more structured way to create the report. SQL is basically used when the query is very complex and doing it using master file creates query performance.**
3. **?FF filename 🡪 For structure of mas/hold file**
4. **WHENCE CAR FOCUS 🡪 For finding location of a file**
5. **&&DEFECHO 🡪 && is used for global variables**
6. **Define and compute🡪 Define is at row by row level and compute is at aggregate level**
7. **Methods to improve the performance of a report:**
8. **Use least possible define**
9. **Lesser number of possible hold files**
10. **Check the joins sometimes it does cross join**
11. **Add filters at starting for data pulling if data is hold for few times.**
12. **If joining 2 hold files, use index on joining fields.**
13. **Most common types of error in webFOCUS**
14. **Value is missing for &var1**
15. **Syntax error in if then else**
16. **Operation is missing an argument**
17. **SQL cursor error. Table or view does not exist.**
18. **Unknown keyword in stylesheet at line 37**
19. **Difference between –RUN, -EXIT, AND –QUIT.**
20. **&LINES and &RECORDS returns the numbers of lines from last table file query and how records are the after aggregation and filters respectively.**
21. **Basic functions used:**

**A)**

**Use of User Defined functions:**

**DEFINE FUNCTION ADDNO(NO1/D20.2, NO2/D20.2)**

**ADDNO/D20.2 = NO1+NO2;**

**END**

**-RUN**

**-DEFAULT &CNT='\_FOC\_NULL';**

**TABLE FILE CAR**

**PRINT COUNTRY**

**COMPUTE NO\_SUM/D20.2 = ADDNO(SALES, DEALER\_COST);**

**WHERE COUNTRY EQ '&CNT'**

**ON TABLE PCHOLD FORMAT HTML**

**END**

**Implementing sorting on data in ascending/descending order by clicking on column title:**

**WebFOCUS report column sorting using javascript:**

**In our day to day reporting projects, we need to give column sorting on the reports. Traditional method is to give the recursive drill down on the title. In that case clicking on the drilldown will open the same .fex file with some parameters to identify sort logic and we get the sort logic done. The below code does the same thing without reloading the report page:**

**-\* File test\_sort.fex**

**-SET &ECHO=ALL;**

**-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*for sorting\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**-DEFAULT &sort\_col = 'COUNTRY' ;**

**-DEFAULT &sort\_type = 0 ;**

**-SET &sort = IF &sort\_type EQ 1 THEN 'LOWEST' ELSE IF &sort\_type EQ 2 THEN 'HIGHEST' ELSE '' ;**

**-DEFAULT &sort\_col1 = 'COUNTRY' ;**

**-DEFAULT &sort\_type1 = 0 ;**

**-SET &sort1 = IF &sort\_type1 EQ 1 THEN 'LOWEST' ELSE IF &sort\_type1 EQ 2 THEN 'HIGHEST' ELSE '' ;**

**-DEFAULT &sort\_col2 = 'COUNTRY' ;**

**-DEFAULT &sort\_type2 = 0 ;**

**-SET &sort2 = IF &sort\_type2 EQ 1 THEN 'LOWEST' ELSE IF &sort\_type2 EQ 2 THEN 'HIGHEST' ELSE '' ;**

**-DEFAULT &sort\_col3 = 'COUNTRY' ;**

**-DEFAULT &sort\_type3 = 0 ;**

**-SET &sort3 = IF &sort\_type3 EQ 1 THEN 'LOWEST' ELSE IF &sort\_type3 EQ 2 THEN 'HIGHEST' ELSE '' ;**

**-DEFAULT &sort\_col4 = 'COUNTRY' ;**

**-DEFAULT &sort\_type4 = 0 ;**

**-SET &sort4 = IF &sort\_type4 EQ 1 THEN 'LOWEST' ELSE IF &sort\_type4 EQ 2 THEN 'HIGHEST' ELSE '' ;**

**-SET &col1 = IF &sort\_col EQ 'COUNTRY' THEN &sort\_col | &sort\_type ELSE 'COUNTRY0' ;**

**-SET &col2 = IF &sort\_col EQ 'CAR' THEN &sort\_col | &sort\_type ELSE 'CAR0' ;**

**-SET &col3 = IF &sort\_col EQ 'DEALER\_COST' THEN &sort\_col | &sort\_type ELSE 'DEALER\_COST0' ;**

**-SET &col4 = IF &sort\_col EQ 'RETAIL\_COST' THEN &sort\_col | &sort\_type ELSE 'RETAIL\_COST0' ;**

**-\*-SET &J='&J';**

**TABLE FILE CAR**

**PRINT COUNTRY AS '<A href="javascript:sortReport1( ''&col1'' ); "TITLE="Click to sort in ascending or descending order">Country</A>'**

**CAR AS '<A href="javascript:sortReport1( ''&col2'' ); "TITLE="Click to sort in ascending or descending order">Car</A>'**

**DEALER\_COST AS '<A href="javascript:sortReport1( ''&col3'' ); "TITLE="Click to sort in ascending or descending order">Dealer cost</A>'**

**RETAIL\_COST AS '<A href="javascript:sortReport1( ''&col4'' ); "TITLE="Click to sort in ascending or descending order">Retail cost</A>'**

**-IF &sort EQ '' THEN SKL1 ELSE RPT1;**

**-RPT1**

**BY &sort.EVAL &sort\_col.EVAL NOPRINT**

**-SKL1**

**HEADING**

**"Sorting logic using javascript(domain)"**

**"Note: The report is POC for sorting logic using javascript and will not work for other than HTML format"**

**ON TABLE PCHOLD FORMAT HTML**

**ON TABLE SET PAGE-NUM OFF**

**ON TABLE SET BYDISPLAY ON**

**ON TABLE SET LINES 99999**

**ON TABLE SET HTMLCSS ON**

**ON TABLE SET EMPTYREPORT OFF**

**ON TABLE SET STYLE \***

**ENDSTYLE**

**ON TABLE HOLD AS SORT\_TEST FORMAT HTMTABLE**

**END**

**-RUN**

**-HTMLFORM BEGIN**

**<HTML>**

**<SCRIPT>**

**function sortReport1( col ) {**

**col = String( col ) ;**

**var len = col.length ;**

**var sort = col.substring( len - 1 , len ) ;**

**if ( sort == 0 ) {**

**FRM1.sort\_type.value = 1 ;**

**} else if ( sort == 1 ) {**

**FRM1.sort\_type.value = 2 ;**

**} else {**

**FRM1.sort\_type.value = 1 ;**

**}**

**FRM1.sort\_col.value = col.substring( 0, len - 1 ) ;**

**-\*alert( FRM1.sort\_col.value );**

**FRM1.submit();**

**return 0 ;**

**}**

**</SCRIPT>**

**<form method="POST" action="/ibi\_apps/WFServlet" NAME=FRM1 target=\_self >**

**-\*<input type="hidden" name="IBIF\_ex" value="test\_sort.fex">**

**-\*<input type="hidden" name="IBIC\_server" value="EDASERVE">**

**-\*<input type="hidden" name="IBIAPP\_app" value="psingh">**

**<input type="hidden" name="IBIC\_server" value="EDASERVE">**

**<input type="hidden" name="IBIMR\_drill" value="X,reportte/reportte.htm">**

**<input type="hidden" name="IBIF\_ex" value="app/test\_sort1.fex">**

**<input type="hidden" name="sort\_col" value=!IBI.AMP.sort\_col;>**

**<input type="hidden" name="sort\_type" value=!IBI.AMP.sort\_type;>**

**<input type="hidden" name="sort\_col1" value=!IBI.AMP.sort\_col1;>**

**<input type="hidden" name="sort\_type1" value=!IBI.AMP.sort\_type1;>**

**<input type="hidden" name="sort\_col2" value=!IBI.AMP.sort\_col2;>**

**<input type="hidden" name="sort\_type2" value=!IBI.AMP.sort\_type2;>**

**</head>**

**<body>**

**<table border="0" id="table1">**

**<tr>**

**<td>!IBI.FIL.SORT\_TEST;</td>**

**</tr>**

**</table>**

**</body>**

**</form>**

**<STYLE>**

**IMG {BORDER:0}**

**a:link {color: blue;}**

**a:visited {color: blue;}**

**a:hover {color: blue;}**

**</STYLE>**

**</HTML>**

**-EXIT**

This could be a very handy technique.   
For instance, when you click on a hyperlink (drilldown) with many many parameters which will generate an URL   
that will exceed the maximum length for an URL.  
  
Just write (all) ampers to a file in the FOCCACHE directory with an unique name and add this filename   
to the parameters in the drilldown definition in your fex.  
  
In the fex that is called by the hyperlink, include this fex and all parameters are set.  
  
  
-Fred-

-\* --- Define some temp-variables

-SET &PA\_TEMP1 = 'This';

-SET &PA\_TEMP2 = 'That';

-SET &PA\_TEMP3 = 'Wendy's';

-SET &PA\_COUNTRY = 'ENGLAND';

-SET &PA\_QWERTY = 'Netherlands';

-RUN

-? & SAVE FOCCACHE/inc\_amp.fex

-RUN

The file inc\_amp.fex in the FOCCACHE directory looks like this:

-\*CURRENTLY DEFINED VARIABLES:

-DEFAULT &ECHO = 'OFF ';

-SET &EXCELSERVURL = 'SET EXCELSERVURL=http://localhost:80/ibi\_apps';

-SET &FOCEXURL = '/ibi\_apps/WFServlet?IBIF\_webapp=/ibi\_apps&IBIC\_server=EDASERVE&IBIWF\_msgviewer=OFF&';

-SET &FOCHTMLURL = '/ibi\_html';

-SET &FOCREL = M727705D;

-SET &GOOGLEMAPSAPIKEY = ' ';

-SET &IBIMR\_domain = ' ';

-SET &PA\_COUNTRY = 'ENGLAND';

-SET &PA\_QWERTY = 'Netherlands';

-SET &PA\_TEMP1 = 'This';

-SET &PA\_TEMP2 = 'That';

-SET &PA\_TEMP3 = 'Wendy''s';

-SET &WFDESCRIBE = XMLRUN;

**Concatenation with space:**

**COUNTRY1/A50=CAR||(' - ' |COUNTRY);**

I don't believe there is way to do it after the report generates but you could use RPCT in the ACROSS then use the stylesheet command HIDE=ON to hide all of the RPCT columns before the report generates. Then it's up to the user to unhide/hide the desired RPCT columns using ACTIVE Tech.  
  
Here's some code to play with that works in 7.6.11:

TABLE FILE CAR

SUM SALES

RPCT.SALES AS PCT

ACROSS COUNTRY AS ''

BY SEATS

ON TABLE PCHOLD FORMAT AHTML

ON TABLE SET STYLE \*

TYPE=REPORT,COLUMN=RPCT.SALES(\*),HIDE=ON,$

ENDSTYLE

END

On report generation, hit any Active drop-arrow and select Show Columns then select Show All. POW! All of the RPCT columns display nicely. To hide all of the RPCT columns, select Restore Original from the bottom of the Active drop-arrow menu.

**Popup Code:**

****

**Code logic to Print \* except the first column:**

TABLE FILE CAR

PRINT

COUNTRY

CAR

MODEL

BODYTYPE

ON TABLE HOLD AS TMP\_CAR

END

-RUN

CHECK FILE TMP\_CAR HOLD

-RUN

TABLE FILE HOLD

PRINT FIELDNAME

WHERE FLDNO GT 1

ON TABLE HOLD AS FEX\_COLUMNS

END

-RUN

TABLE FILE TMP\_CAR

PRINT

-INCLUDE FEX\_COLUMNS

END

**Using SQL Pass through approach to generate the report:**

ENGINE DB2 SET DEFAULT\_CONNECTION SSQDTA

SQL DB2 PREPARE SQLOUT FOR

SELECT \* FROM SSQDTA.ECRMAIN

END

TABLE FILE SQLOUT

PRINT \*

WHERE RECORDLIMIT EQ 10

END

## Q. Tell Me About Yourself.

This is a commonly asked interview question designed to break the ice. A strong, succinct answer will quickly gain the interviewer’s attention and separate you from other candidates who may be tempted to divulge their life story. Give a brief, concise description of who you are and your key qualifications, strengths and skills. Tailoring your answer to the role offered by declaring the strongest benefit that you offer an employer will leave the interviewer compelled to know more.

## Q. Why Do You Want to Work Here?

-Will help me to expand my skillset at a global level.

-Will help me to introduce with new world of challenges

## Q. What Are Your Strengths?

-Problem-solving and coding

-Working under pressure

-Helping Team

## Q. What Are Your Greatest Weaknesses?

-Finds difficult to tell NO

-Push Back

## Q. What Have Been Your Achievements to Date?

-Created XML parser in webfocus to read Google Datatable

-Trained 20 plus team and found 30 plus work around where webfocus was technical limited.

-Gantt Chart

Q. What is the Most Difficult Situation You Have Faced at Work?

-Coordination of prject resulting delay

The interviewer is trying to find out your definition of ‘difficult’ and whether you can show a logical approach to problem solving. Select a tough work situation that was not caused by you. Explain the way you approached the problem, including the actions you took and the solution you applied to overcome the problem. Give your answer with the air of someone who takes setbacks and frustrations in stride, as part of the job.

## Q. What Did You Like/Dislike About Your Last Role?

The interviewer is trying to find out your key interests and whether the job offered has responsibilities you will dislike. Focus on what you particularly enjoyed in your last role and what you learned from it, drawing parallels to the new role. When addressing what you disliked, be conscious not to criticize your last employer. Choose an example that does not reflect on your skills (such as company size) or which reveals a positive trait (such as your dislike for prolonged decision making).

## Q. Why Do You Want to Leave Your Current Employer?

This should be straightforward. Reflect positively on your current employer but state how you are looking for more challenge, responsibility, experience and a change of environment. Explain how your current role can no longer provide you with these things, but how you believe the role offered presents an opportunity for growth that will make full use of your strengths and potential.

## Q. What Are Your Goals For The Future?

A sense of purpose is an attractive feature in an applicant, so this question is designed to probe your ambition and the extent of your career planning. Your commitment is also under question, but avoid blankly stating, “I want to be with your company.” Instead, describe how your goal is to continue to grow, learn, add value and take on new responsibilities in the future that build on the role for which you are applying.

## Q. How Do you Respond to Working Under Pressure?

The interviewer wants to see that you have composure, problem solving skills and can stay focused in difficult conditions. Give an example of a time when you were faced with a stressful situation (not caused by you) and how you handled it with poise. Describe the context, how you approached the situation, the actions you took and the positive outcome. Demonstrate how you remained calm, in control and got the job done.

## Other Questions to Consider

* Tell me about a successful team project that you have been involved in. What was your role and what made it a success?
* What do you enjoy about this industry?
* Give me an example of when your work was criticized.
* Give me an example of when you haven't gotten along with co-workers or bosses. What did you do?
* Do you prefer to work alone or in a group? Why?
* What are you looking for in a company?
* How do you measure your own performance?
* What kind of pressures have you encountered at work?
* Are you a self-starter? Give me examples to demonstrate this.
* What can you bring to this organization?

Strength:

Coding and Problem Solving

Keeping calm under pressure

Weakness: