

Ostendo

And

Freeway

Custom

Inquiries

(Update 230)

Contents

Ostendo - Freeway Inquiries Tutorial	2
Overview:	2
Creating a Warehouse Inquiry Pack:	3
Linking An Inquiry Pack to an Employee:	5
Freeway Inquiries - Listing Information:	6
Freeway Inquiries - Charts	15
Freeway Inquiries - Scorecard Statistics	18

Ostendo - Freeway Inquiries Tutorial

Overview:

Ostendo allows you to create Custom Inquiries to be run directly from Freeway. These Inquiries run real-time, therefore you must have a live connection back to the Ostendo database to run them.

Inquiry Types define what you will be inquiring on eg:

- Assembly Orders - Inquiry by selecting an Assembly Order
- Customer - Inquiry by selecting a Customer
- Delivery - Inquiry by selecting a Sales Delivery
- Job - Inquiry by selecting a Job Order
- Supplier - Inquiry by selecting a Supplier
- Product - Inquiry by selecting an Item
- Internal - General Inquiry non-specific to any above. Used for Company-wide reporting

NB: The above inquiry types rely on the selection data to be available to the Freeway user. Eg: the selection of a Job relies on that Job being available to the device at the time the inquiry is run.

Inquiry Packs are a way of bundling one or more Inquiries together into a 'Pack'. These Packs are linked to Mobility Employees, from the Employee Mobility Settings screen. This way you can create targeted Packs for either specific Employees or Groups of Employees. Eg:

- Warehouse Inquiry Pack
- Service Inquiry Pack
- Management Inquiry Pack
- Production Inquiry Pack etc....

Inquiry Styles allow each Inquiry to be configured to present information in a number of different formats. Eg:

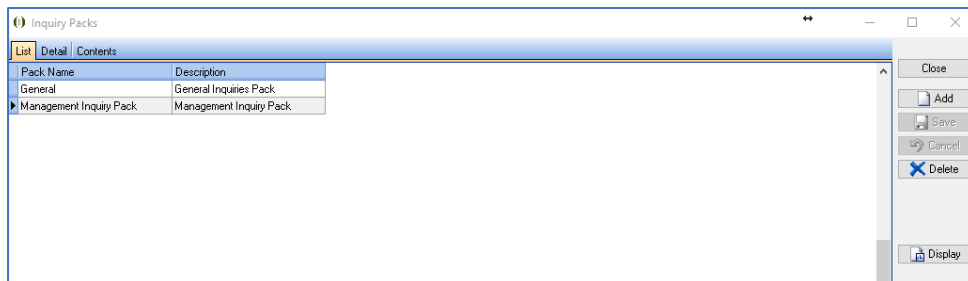
- Listing
- Chart
- Scorecard

Standard Functions are optional pre-developed routines that can be incorporated into a custom inquiry. Eg:

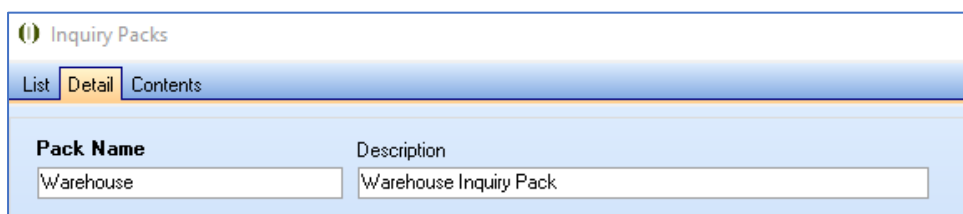
- Display Customer Aging
- Display Stock Levels

Creating a Warehouse Inquiry Pack:

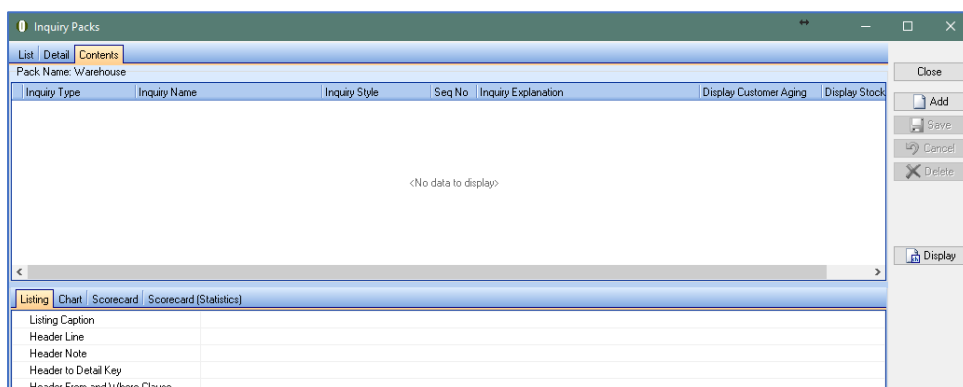
From within Ostendo go to Mobility -> Inquiry Packs



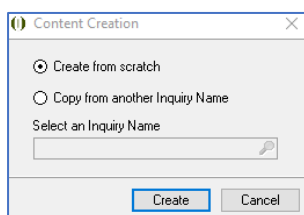
Press Add to create a new Inquiry Pack. Give it a Pack Name and Description, then Press Save



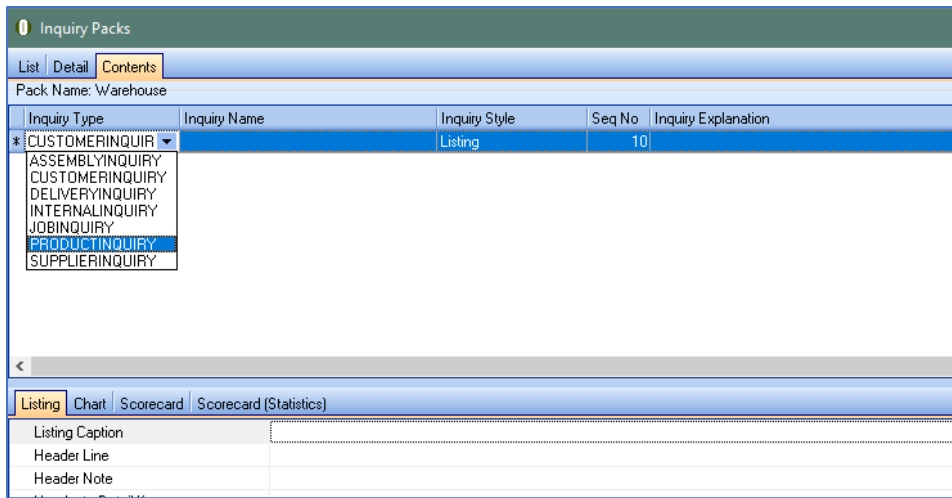
Select the Contents Tab to start defining your Inquiry. This screen is split into two sections (Upper & Lower).



Press the Add button to add an Inquiry to this Pack. The Content Creation screen allows you to create a new inquiry from scratch or copy an existing Inquiry from the current Pack or even a different pack. For this Tutorial, select the 'Create from Scratch' option



An Inquiry record will be created in the upper portion of the screen. Now select the PRODUCTINQUIRY Inquiry Type.



Inquiry Name:

This is the name of the Inquiry '**General Item Inquiry**'

Inquiry Style:

From here you can select the appropriate style format for this Inquiry. Select **Chart-Scorecard-Listing** combination. Ultimately at the conclusion of this Tutorial, this inquiry will represent a combination of Chart / Scorecard Information and Listing data

Inquiry Explanation:

This explanation appears at the top of the screen on Freeway when the inquiry is run. Use this to describe what the inquiry does as well as explaining colour coding combinations used.

Display Stock Levels:

Tick this to enable Ostendo to run a Standard function to display Inventory Qty's. NB: The detail of what is displayed is controlled by the 'Stock Inquiry Level' on the Materials Tab of the Employee Mobility Settings screen (This function is only to be used for Product Inquiries)

At this point you can configure the Mobility Employee settings in order to run this inquiry

The next section details how to link this inquiry to an Employee Mobility user.

Linking An Inquiry Pack to an Employee:

- Go to Mobility -> Employee Mobility Settings and select the Detail Tab of the Employee
- On the Device Tab browse and Select the '**Warehouse**' Inquiry Pack
- From the Materials Tab tick '**Product Inquiry Available**'
- Select the Stock Inquiry Level eg:

All Site Totals

This summarises all inventory records together for all sites ie: one total for each site

Mobility Site Totals

This summarises all inventory records together for this Employees site ie: one total for this Employees site.

All Site/Warehouse Totals

This summarises all inventory records together by warehouse within each site Ie: one record for each Warehouse within each site

Mobility Site/Warehouse Totals

This summarises all inventory records together by warehouse within the Employees site Ie: one record for each Warehouse within the site

All Site/Details

This shows detailed inventory records (showing variants eg: Expiry / Batch etc..) together by warehouse/Location within each site
Ie: multiple records for each Warehouse Location within each site

Mobility Site/Details

This shows detailed inventory records (showing variants eg: Expiry / Batch etc..) together by warehouse/Location within the Employees site
Ie: multiple records for each Warehouse Location within the Employees site

- Select the '**General Item Inquiry**' from the 'Drill Down Product Inquiry Name'

You are now ready to run the Inquiry directly from Freeway:

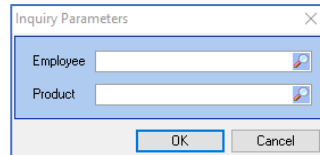
- Launch Freeway
- Perform a **Re-Initialise** Sync Update for the Freeway User
- Run the Product Inquiry -> General Item Inquiry from within Freeway

Alternatively, you can run this Inquiry from within Ostendo by pressing the



button on the Inquiry Packs screen

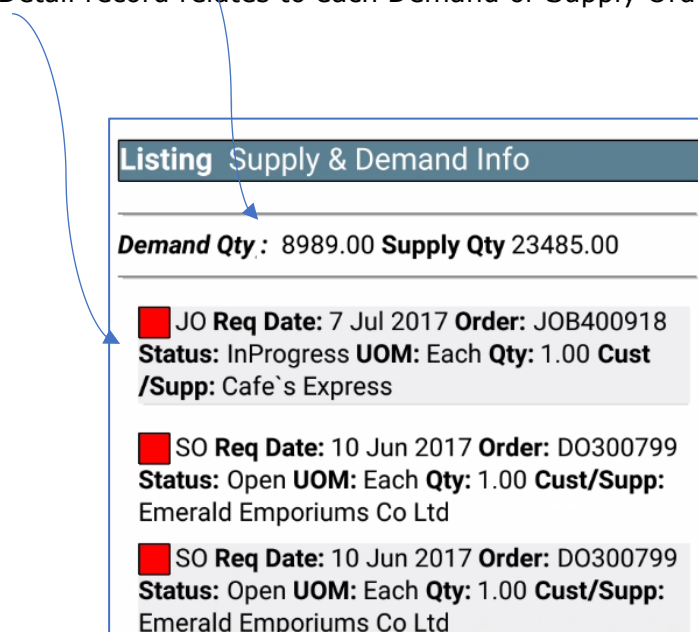
Select the Employee and Product to Inquire on.



Freeway Inquiries - Listing Information:

Listing information can be added in the form of Header records only or both Header and Detail records together.

In this next example the Header record relates to Total Supply and Demand orders for an item. The Detail record relates to each Demand or Supply Order for that item



Hint:

Have a clear idea of exactly what you want to report and how you want it laid out.

Construct an SQL statement to present the data you wish to display. This will help you define the inquiry later and also ensure your SQL syntax is correct. It is suggested you use EMS or IB Expert to do this. In this example the SQL would be:

Select cast(demandqty as decimal(9,2)), cast(supplyqty as decimal(9,2)) from itemmaster where :passedselectedvalue = itemcode

The Listing screen has been designed to accept elements of SQL and HTML. Some standard syntax is not required. Eg: 'Select'

From the Inquiry Pack Listing Tab enter the following information:

Listing Caption:

Enter a meaningful name for this Listing '**Supply & Demand Info**'

Header Line:

This information is a combination of SQL and HTML coding. Copy in the following coding into this field. Think of the Header Line as the Select part of the Stmt.

```
'<b> Demand Qty: </b>' || cast(demandqty as decimal(9,2)) || '<b> Supply Qty: </b>' || cast(supplyqty as decimal(9,2))
```

Explanation of Coding:

Let's now breakdown and explain each component of this code.

Basic HTML coding:

	indicates the start of HTML for Bold
	indicates the end of HTML for Bold
<i>	indicates the start of HTML for <i>Italics</i>
</i>	indicates the end of HTML for <i>Italics</i>
<i> 	indicates the start of HTML for <i>Italics Bold</i>
</i> 	indicates the end of HTML for <i>Italics Bold</i>
<u>	indicates the start of HTML <u>underline</u>
</u>	indicates the end of HTML for <u>underline</u>
<h style="color:green">	indicates the start of HTML font colour green
</h>	indicates the end of HTML font colour green
 	indicates a carriage return (new line)

Below is an example of the above commands nested together to provide the following example:

```
'<h style="color:green"> <b><u><i> Example Text </i></u></b></h>' || '<br> '
```

Result: **Example Text**

```
'<b> Demand Qty: </b>'
```

This Header line above includes field headings and field names. The field heading is always enclosed in single quotes (one at the start and one at the end).

The wording '**Demand Qty :**' is the actual heading that will appear next to the DemandQty field value

```
||
```

This symbol is the SQL Pipe symbol. It is used to concatenate multiple elements of data together. In this case it will join the heading of Demand Qty with the DemandQty field

```
cast(demandqty as decimal(9,2))
```

This is the actual field in the database to report on. In this case we are forcing the number of decimal places to 2. The cast function is optional here but allows us to display the data in a more presentable form.

```
'<b> Supply Qty: </b>'
```

This is a heading called **Supply Qty** with HTML formatting

||

Pipe symbol to join the Supply Qty heading with the SupplyQty field

`cast(supplyqty as decimal(9,2))`

Field SupplyQty to be presented to 2 decimals places by using the cast function.

Header Note:

Most Header records also have a Notes field in the same record. This entry allows you to specify this. If you type OrderNotes here you will display the Order notes field relating to the Header Line. In this case the Header Lines doesn't relate to an Order but instead relates to the Item, therefore we could define the ItemNotes field here if we wish, otherwise leave it blank.

Header to Detail Key:

This is the field relating to the Header Line Query that will link to the Detail Line Query. In this case enter **ItemCode**. However if the Header Line was related to the Order Header and the Detail Line related to the Order Lines, you would enter OrderNumber as the field that joins the Header with the Detail (Line) record.

Header From and Where Clause:

This is where we define the table(s) the Header Query relates to, as well and any conditions relating to it.

From itemmaster where :passedselectedvalue = itemcode

Explanation of Coding:

Let's now breakdown and explain each component of this code.

From itemmaster

Standard SQL syntax stating from which Table the data is to be retrieved from.

where :passedselectedvalue = itemcode

This is where we place the conditioning statements and table joins. The :passedselectedvalue statement in this case will be the ItemCode that has been selected by the user from within Freeway. This is also the section you could insert the Order By function to controlling sorting.

Header Results Restriction Count:

This allows you to restrict the number of records returned from Ostendo to Freeway. If the volume of records returned is substantial, performance will be affected. Therefore as an example if your query could potentially return 100 records, you should perhaps consider restricting it to 20 or so records. (Ensure your Listing Caption conveys this to the user) (Leave blank for this tutorial)

Header Indicator Logic:

This allows you to control the colour of an indicator optionally displayed to the side of the Header Record. The colour of this indicator can be controlled by either an SQL Case statement, or by the Workflow Tracking Status colour of the Header Record. In this case simply insert the Tracking field of TRACKINGCOLOUR. (Leave blank for this tutorial)

Detail Line:

Just like the Header Query, you can specify a Detail record Query.

```
case orderclass when 'Job Order' then 'JO'
  when 'Assembly Receipt' then 'AR'
  when 'Assembly Issue' then 'AI'
  when 'POS Order' then 'PS'
  when 'Purchase Order' then 'PO'
  when 'Sales Order' then 'SO'
  when 'Rental Issue' then 'RI'
  when 'Rental Return' then 'RR'
end
|| '<b> Req Date: </b>' || (select * from RETURN_DATEASTEXT(requireddate)) ||
'<b> Order: </b>' || OrderNumber || '<b> Status: </b>' || orderstatus || '<b> UOM:
</b>' || originalunit || '<b> Qty: </b>' || cast(originalremainingqty as Decimal(13,2))
|| '<b> Cust/Supp: </b>' || CustomerOrSupplier
```

Explanation of Coding:

Let's now breakdown and explain each component of this code.

Like the Header Query you do not need to specify the Select statement.

```
case orderclass when 'Job Order' then 'JO'
  when 'Assembly Receipt' then 'AR'
  when 'Assembly Issue' then 'AI'
  when 'POS Order' then 'PS'
  when 'Purchase Order' then 'PO'
  when 'Sales Order' then 'SO'
  when 'Rental Issue' then 'RI'
  when 'Rental Return' then 'RR'
end
```

This SQL code inserts a two character transaction type indicator. This indicator is based on the OrderClass.

Eg: When the OrderClass = 'Job Order' then insert 'JO'
When the OrderClass = 'Assembly Order' then insert 'AO' and so on for all OrderClasses

```
||
```

Pipe symbol to join the Transaction Type field together with the next field

```
'<b> Req Date: </b>'
```

Bold the Label Heading **Req Date:**

```
||
```

Pipe symbol to join the label **Req Date:** together with the next field

```
(select * from RETURN_DATEASTEXT(requireddate))
```

A standard Ostendo function called 'Return_DateAsText' has been provided to convert date fields into a more readable format. This function is being used on the RequiredDate field. Eg: '2017/07/31' becomes '31 Jul 2017'

Hint:

When querying on fields that could contain a null value you should include a coalesce statement that will check for nulls and replace the null with a character. If you do not do this, nothing will be returned in the Listing Queries.

Eg: `coalesce(orderdescription,'No Desc')`

The above example will return 'No Desc' if the OrderDescription field is null. You could equally specify the following to return a blank value. Eg: `coalesce(orderdescription,"")`

The rest of the code is in a similar format to that previously shown above.

Detail Note:

Detail records may also have a Notes field in the same record. This entry allows you to specify this. If you type relevant notes fields name here you will display the Order or Line notes field relating to the Detail Line. In this case we do not need the Notes shown, therefore we will leave this blank for this Tutorial

Detail From and Where Clause:

`FROM RETURN_DEMANDANDSUPPLYNOP (:Headerkey,"0,") order by requireddate desc, demandorsupply`

Explanation of Coding:

Let's now breakdown and explain each component of this code.

In this Tutorial we are returning the data from an existing Stored Procedure called Return_DemandandSupply.

Data Being Retrieved from Tables

Example: If the data was being retrieved directly from within a table eg joblines you would simply enter **from joblines**

`(:Headerkey,"0,")`

The open and close brackets and the ``0`` are required here simply because we are using a Stored Procedure to retrieve the records.

Data Being Retrieved from Tables

Example: If you were using a table you would simply type **where ordernumber = :HeaderKey**

:HeaderKey is a reserved word that holds the value of the fieldname indicated in Header to Detail Key.

`order by requireddate desc, demandorsupply`

This SQL controls the sort sequence of the data being returned. In this case by Required Date in Descending Sequence, then by DemandorSupply type.

Detail Results Restriction Count:

Just like the Header Query you are able to control the number of records returned. Use this to enhance performance of the inquiry if required. (NB: Ensure you convey this restriction to the user by way of an explanation or the Listing Caption)

Detail Indicator Logic:

```
case demandorsupply  
  when 'Supply' then 16760576  
  when 'Demand' then 255  
end
```

Explanation of Coding:

Let's now breakdown and explain each component of this code.

This SQL controls the colour of the Indicator against the Detail record being returned. We are hardcoding the colour based on whether the DemandorSupply field = 'Demand' or 'Supply'

The Colour numbers represented are 16760576 = Blue and 255 = Red.

Hint:

*To determine the colour number, set the desired colour of a Job Tracking Code, then query the **JOBWORKFLOWSTATUSES** TrackingColour field.*

Finally give this Inquiry an Explanation in the Contents screen. This explanation will be displayed to the user as indicated.

Inquiry Packs					
List Detail Contents					
Pack Name: Warehouse					
pe	Inquiry Name	Inquiry Style	Seq No	Inquiry Explanation	Disp
▶TINQUIRY	General Item Inquiry	Chart-Scorecard-Listing	10	Demand / Supply Orders. Indicator Colours: Re	
				Demand / Supply Orders. Indicator Colours: Red = Demand, Blue = Supply Sorted by Latest Required Date back	

← Warehouse
PDF
✉

General Item Inquiry

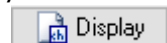
Explanation: Demand / Supply Orders. Indicator Colours: Red = Demand, Blue = Supply Sorted by Latest Required Date back

Product : 100-2000
Washer-Mild Steel-8mm

You are now ready to run the Inquiry directly from Freeway:

- Launch Freeway
- Perform a **Re-Initialise** Sync Update for the Freeway User
- Run the Product Inquiry -> General Item Inquiry from within Freeway

Alternatively, you can run this Inquiry from within Ostendo by pressing the



Select the Employee and Product to Inquire on.

Inquiry Parameters

Employee

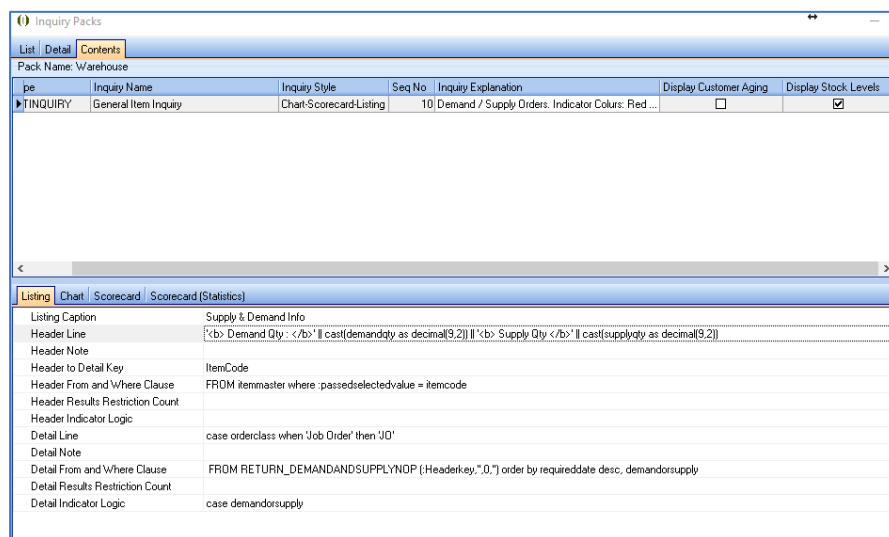
Product

Your inquiry should look something like this



Hint:

Further development of the queries & formatting can be done from within the Lower area of the Inquiry Pack Content screen as indicated without the necessity for a full Re-Initialisation of Freeway as the SQL code is read each time the Inquiry is run, however if you add or amend anything in the upper section of this screen you will need to perform a full Re-Initialisation.



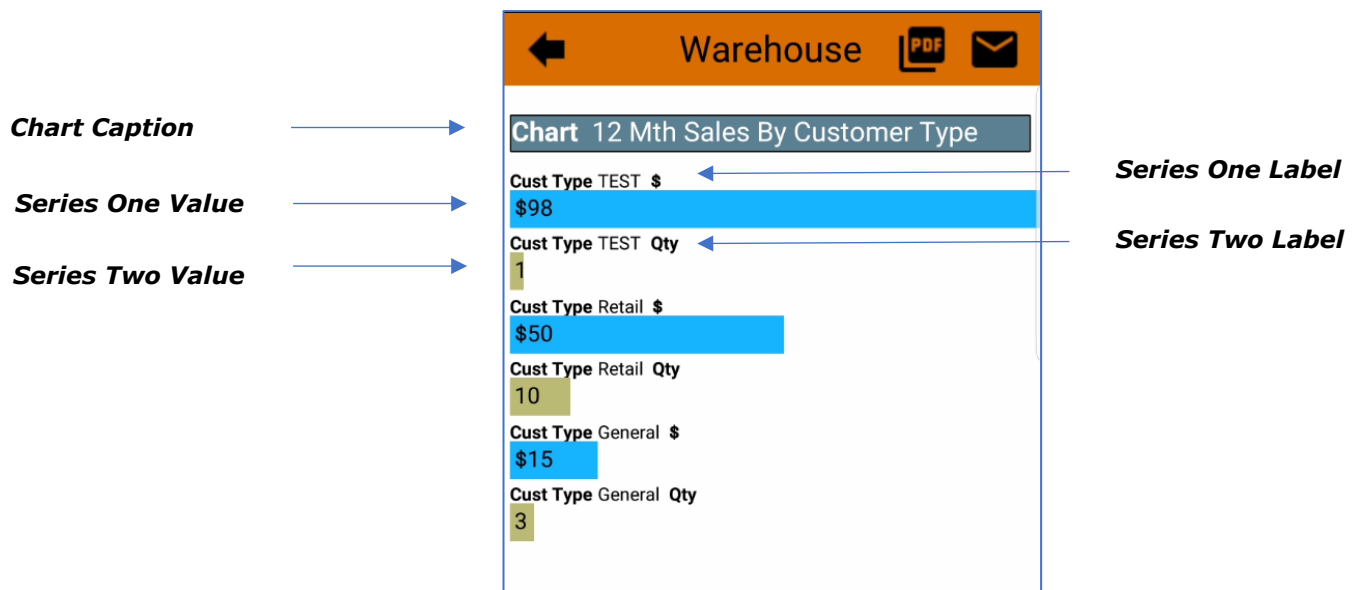
Freeway Inquiries – Charts

A chart can be added to your inquiry to graphically represent information. In this case we will insert 2 charts, one showing 12 Month Sales Value by Customer Type and the other showing 12 Month Sales Qty by Customer Type

Freeway Inquiry Charts are always represented as horizontal bar charts. What is optional, is whether you show one series or two.

A single series can optionally be presented with an overall percentage for each value.

In this tutorial, we will create a Two series chart. Series One will show Total Value sold over the last 12 months, whilst Series Two will show Total Qty sold over 12 months.



From the Chart Tab of the Contents screen

Property	Value
Chart Caption	12 Mths Sales By Customer Type
Chart Style	Two Series
Series One Value	sum[invoiceextendedprice]
Series Two Value	sum[invoiceQty]
Series One Label	' Cust Type ' CustomerMaster.CustomerType ' \$ '
Series Two Label	' Cust Type ' CustomerMaster.CustomerType ' Qty '
Series One Format	Local Currency
Series Two Format	None
Chart From and Where Clause	from SalesHistory ,customermaster where SalesHistory.LineCode = :passedselectedvalue and SalesHistory.OrderCustomer = CustomerMaster.Customer
Chart Results Restriction Count	

Chart Caption:

Enter a Chart Caption of '**12 Mth Sales By Customer Type**'

Chart Style:

Select **Two Series**

Series One Value:

This is the first series value we will report on. In this case we need to total the SalesHistory.InvoiceExtendedPrice field. Therefore type **sum(InvoiceExtendedPrice)**

Series Two Value:

This is the second series value we will report on. In this case we need to total the SalesHistory.InvoiceQty. Therefore type **sum(InvoiceQty)**

Series One Label:

The label appears immediately above the bar chart. HTML can also be used in this field. Note the || symbol used here to concatenate this information together as one field

```
' <b> Cust Type </b>' || CustomerMaster.CustomerType || ' <b> $ </b>'
```

Series Two Label:

The label appears immediately above the bar chart. HTML can also be used in this field. Note the || symbol used here to concatenate this information together as one field

```
' <b> Cust Type </b>' || CustomerMaster.CustomerType || ' <b> Qty </b>'
```

Series One Format:

Local Currency will be used to represent the Value of Sales for this item

Series Two Format:

Select **None** here as no special formatting is required for Qty

Chart From and Where Clause:

This is where we define the tables, joins and conditioning for the SQL required to generate the Sales Value and Qty sold in the last 12 months. Also, note the Group By clause.

The "Group By" clause is only used when Series One Value or Series Two Value is an aggregated value.

The number 1 refers to Series One Value;

2 refers to Series Two Value,

3 refers to Series One Label, and 4 refers to Series Two Label

```
from SalesHistory, customermaster where SalesHistory.LineCode = :passedselectedvalue and SalesHistory.OrderCustomer = CustomerMaster.Customer and (SalesHistory.InvoiceDate >= (Current_Date - 364)) and (SalesHistory.InvoiceDate <= (Current_Date)) group by 3,4 order by 1 desc
```

Hint:

Ensure there is a space before the 'from' statement

Chart Results Restriction Count:

Just like the other Restrictions in Listings, you are able to control the number of records that are returned. In this example, if you had many Customer Types, you could use this to limit the Number of Customer Types returned. Use this to enhance performance of the inquiry if required. (NB: Ensure you convey this restriction to the user by way of an explanation or the Caption)

Freeway Inquiries - Scorecard Statistics

Freeway inquiries allow you to add multiple queries to return results. These could be statistical results or general information

For example

- Total Qty sold in the last 12 months
- Standard Sell Price
- Available Qty

From the Scorecard Tab on the Contents screen

Seq No	Statistic Name	Statistic Value Field	Statistic From and Where Clause	Statistic Format	Highlight Conditional Type	Highlight Conditional Value	Highlight Colour
10	12 Mth Sales	sum(invoicedqty)	from saleshistory where linecode ...	Decimal	Less Than	0	Red
20	Selling Price	stdsellprice	from itemmaster where itemcode ...	Local Currency	No Highlight		Blue

Scorecard Caption:

Enter a Scorecard Caption of **'Item Information & Statistics'**

Now select the Scorecard (Statistics) Tab on the Contents screen to define your queries.

Statistic Name:

Enter a meaningful Statistic name for this query **'12 Mth Sales'**

Statistic Value Field:

This is the field you are reporting on. In this case we will sum the InvoiceQty field to get this statistic. Type **sum(InvoiceQty)**

Statistic From and Where Clause:

This is where we define the tables, joins and conditioning for the SQL required to generate the Qty sold in the last 12 months.

```
from saleshistory where linecode = :passedselectedvalue and ((invoicedate >= (current_date - 364)) and (invoicedate <= Current_date))
```

Hint:

Ensure there is a space before the 'from' statement

Statistic Format:

This defines how the returned data will be displayed. In this case we will select **'Decimals'**

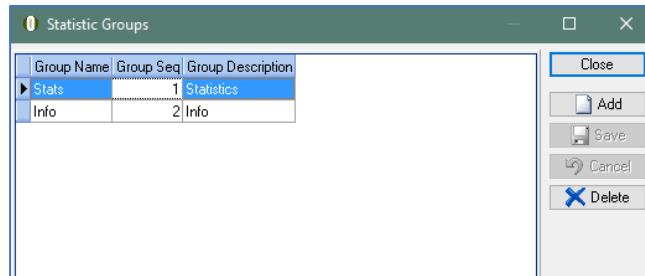
Highlight Conditional Type / Highlight Conditional Value / Highlight Colour:

This allows you to highlight the result (by changing the colour of the font) when a specified condition is met. Eg: in this case we could specify **'Less Than'** in the Conditional Type and **'0'** in the Conditional Value and **Red** in Highlight Colour. This would have the effect of changing the result to Red if the 12 Mths Sales Qty was Less than Zero.

Group Seq and Group Name:

If you have multiple statistics you need to display, a Grouping mechanism is available allowing you to group these together into logical groups instead of displaying one large list of Statistics.

To create Groups for your Inquiry Statistics, go to Mobility->Settings->Statistic Groups and specify your groups



Now link these defined groups to each Statistic. (The Group Seq is read from the Statistics Groups screen.)

For the purpose of this Tutorial, create the following Groups

Stats
Info

- Link the Stats Group to the 12 Mth Sales Statistic
- Repeat this process as follows for each statistic:

Statistic Name

Selling Price

Statistic Value Field

StdSellPrice

Statistic From & Where Clause

from ItemMaster where ItemCode = :passedselectedvalue

Statistic Format

Local Currency

Highlight Conditional Type

No Highlight

Group Name

Info

Statistic Name

Available Qty

Statistic Value Field

AvailableQty

Statistic From & Where Clause

from ItemMaster where ItemCode = :passedselectedvalue

Statistic Format

Decimal

Highlight Conditional Type

No Highlight

Group Name

Info

Your completed inquiry should now look like this

Scorecard Caption →

Group Name →

Statistic Name →

Scorecard Item Information & Statistics	
Statistics	Value
12 Mth Sales	14.00

Info	Value
Selling Price	\$5.00
Avaiable Qty	36,305.00

Listing Supply & Demand Info	
Demand Qty: 8990.00	Supply Qty 23505.00

Adsgf adf hga dfhadhzdhdgah
adjg

■ **JO Req Date:** 9 Oct 2017 **Order:** 400955
Status: Open **UOM:** Each **Qty:** 1.00 **Cust/Supp:**
Auckland

■ **PO Req Date:** 14 Sep 2017 **Order:** PO100480
Status: Receipts **UOM:** Each **Qty:** 20.00 **Cust**
/Supp: Bruce Wilson