

Create Report - Tables

When creating reports in the Report Writer, you need to know how data is stored in the database. This is a guide to commonly used tables.

Table Name	Purpose	Keys	Joining	Comments
Product	Storing product data integrated from ERP with additional fields that might only be stored online	Product Code		<p>To get all items that are shown on the website use the where clause:</p> <p>itemvalid = 1</p> <p>Generally field names from ERP will be the same online but without the dashes.</p>
ProntoStockWarehouse	A mirror of the PRONTO ERP table stock-warehouse-detail	StockCode WhseCode	<p>Product:</p> <pre>Product LEFT OUTER JOIN ProntoStockWarehouse on ProductCode = StockCode</pre>	
ProntoSalesOrder	Sales Order header table. Stores both carts online and integrated orders from ERP	SoOrderNo SoBoSuffix	<p>Customer:</p> <pre>ProntoSalesOrder LEFT OUTER JOIN Customer on ProntoSalesOrder. socustcode = Customer. Code</pre> <p>User:</p> <pre>ProntoSalesOrder LEFT OUTER JOIN [User] on ProntoSalesOrder. souuseridcode= [User]. EmailAddress</pre>	<p>Filter on orders that were created online and have been integrated to ERP:</p> <p>ConfirmedbyERP = 1</p> <p>Filter on orders that got created in ERP:</p> <p>fwcreatedby = 'WS'</p> <p>Filter on carts in progress:</p> <p>fwcreatedby != 'WS' and soordertotalpackages > 0</p> <p>Filter on Ready to Integrate:</p> <p>sendrequired = -1</p>

ProntoSalesOrderLine	Sales order lines	SoOrderNo SoBoSuffix SolLineSeq	<p>ProntoSalesOrder:</p> <pre> ProntoSalesOrderline LEFT OUTER JOIN ProntoSalesOrder on ProntoSalesOrderline. SoOrderNo = ProntoSalesOrder. SoOrderNo AND ProntoSalesOrderline. SoBoSuffix = ProntoSalesOrder. SoBoSuffix </pre> <p>Product:</p> <pre> ProntoSalesOrderline LEFT OUTER JOIN product on ProntoSalesOrderline. StockCode = Product.ProductCode </pre>	
CustomerPayments	Payments received either associated with a order or could also be associated with account payments	RowId	<p>ProntoSalesOrder:</p> <pre> CustomerPayments LEFT OUTER JOIN Prontosalesorder on CustomerPayments. OrderNumber = ProntoSalesOrder. SoOrderNo </pre> <p>User:</p> <pre> CustomerPayments LEFT OUTER JOIN [User] on CustomerPayments. UserName = [user].emailAddress </pre>	<p>Status:</p> <p>1 = ready to integrate</p> <p>2 = integrated</p> <p>9 = payment failed</p> <p>Also note for account payments the OrderNumber will be 0 so if you are joining to ProntoSalesOrder suggest to use a LEFT OUTER JOIN instead of LEFT OUTER JOIN</p>
User	All users	EmailAddress	<p>ProntoSalesOrder:</p> <pre> ProntoSalesOrder LEFT OUTER JOIN [User] on ProntoSalesOrder. souseridcode= [User]. EmailAddress </pre> <p>Customer:</p> <pre> [User] LEFT OUTER JOIN Customer on [User].CustomerCode = Customer.Code </pre>	<p>Note that all use of User table must use [User] and not User</p> <p>Best to ignore guest users which will have emailaddress like 'guest%'</p>

UserSession	Top level details of every session	EmailAddress Session Start	User: <pre>UserSession LEFT OUTER JOIN [User] on UserSession.EmailAddress = [User].Emailaddress</pre>	Encourage to put a date range on the extract of this as it can be large file
UserSessionLog	Detailed information on user session	EmailAddress Session Start Seq	UserSession: <pre>UserSessionLog LEFT OUTER JOIN UserSession on UserSession.EmailAddress = UserSessionLog.Emailaddress</pre>	This table is only stored for maximum of 4 weeks due to its size Filter on EventType to get details. See below for UserSessionLog Event Types

UserSessionLog Event Types

Event	Description	Other Data
CategoryView	A view of a category	EventKey1 = CategoryCode Description = Category Name
OrderPlaced	The placing of an order	EventAmount = Order Value
PageView	The viewing of a page	EventKey1 = Pagename and role E.G: (PUBLICR) ProductDisplay.aspx means PUBLICR role and productdisplay.aspx is the name of the page
ProductInCart	A product being placed in the cart	EventKey1 = ProductCode EventQty = Qty EventAmount = Qty * Price
ProductOutOfCart	A product being taken out of the cart	EventKey1 = ProductCode
ProductSearch	A search of products	EventKey1 = Search Key EventQty = Number of results returned
ProductSold	Sale of a product	EventKey1 = ProductCode EventQty = Qty EventAmount = Qty * Price
ProductView	Detailed view of a product	EventKey1 = ProductCode EventAmount = Price
Trace	Trace of milliseconds to create the page	Description = PageName EventQty = Milliseconds to create the page

Related Resources

- [Report Subscriptions \(Scheduled Reporting\)](#)
- [Configure Scheduled Reporting](#)
- [Report Generator \(Scheduled Reporting\)](#)
- [Create Users to Approve Report](#)
- [Google Tag Manager with Universal Analytics](#)