

# Table of Contents

[Table of Contents 2](#__RefHeading__27796_411373615)

[AMAZONA-HLGH6D4 3](#__RefHeading__27798_411373615)

[User databases 4](#__RefHeading__27800_411373615)

[CampaignTrack\_Dev Database 4](#__RefHeading__27802_411373615)

[Tables 6](#__RefHeading__27804_411373615)

[[dbo].[Calendars] 7](#__RefHeading__27806_411373615)

[[dbo].[CalendarUser] 8](#__RefHeading__27808_411373615)

[[dbo].[CancelledOrders] 10](#__RefHeading__27810_411373615)

[[dbo].[ColorMaster] 11](#__RefHeading__27812_411373615)

[[dbo].[Company] 12](#__RefHeading__27814_411373615)

[[dbo].[CompanyMail] 14](#__RefHeading__27816_411373615)

[[dbo].[CompanyOrders] 15](#__RefHeading__27818_411373615)

[[dbo].[Contacts] 17](#__RefHeading__27820_411373615)

[[dbo].[Events] 18](#__RefHeading__27822_411373615)

[[dbo].[EventTracking] 20](#__RefHeading__27824_411373615)

[[dbo].[Mail] 22](#__RefHeading__27826_411373615)

[[dbo].[OrderContacts] 23](#__RefHeading__27828_411373615)

[[dbo].[OrderEvents] 25](#__RefHeading__27830_411373615)

[[dbo].[OrderItems] 26](#__RefHeading__27832_411373615)

[[dbo].[Orders] 28](#__RefHeading__27834_411373615)

[[dbo].[OrderType] 30](#__RefHeading__27836_411373615)

[[dbo].[Organisation] 31](#__RefHeading__27838_411373615)

[[dbo].[ProductGroups] 32](#__RefHeading__27840_411373615)

[[dbo].[Products] 33](#__RefHeading__27842_411373615)

[[dbo].[ProductSchedules] 35](#__RefHeading__27844_411373615)

[[dbo].[ProductUser] 36](#__RefHeading__27846_411373615)

[[dbo].[Property] 37](#__RefHeading__27848_411373615)

[[dbo].[PropertyInvoices] 39](#__RefHeading__27850_411373615)

[[dbo].[User] 40](#__RefHeading__27852_411373615)

[[dbo].[UserTypes] 42](#__RefHeading__27854_411373615)

[Stored Procedures 43](#__RefHeading__27856_411373615)

[[dbo].[GetDataforCalenderEntry] 44](#__RefHeading__27858_411373615)

[[dbo].[GetProductSchedules] 47](#__RefHeading__27860_411373615)

[[dbo].[InsertCalendars] 48](#__RefHeading__27862_411373615)

[[dbo].[InsertCancelledOrder] 48](#__RefHeading__27864_411373615)

[[dbo].[InsertEvents] 50](#__RefHeading__27866_411373615)

[[dbo].[InsertOtherEvents] 51](#__RefHeading__27868_411373615)

[[dbo].[InsertPropertyInvoiceNo] 54](#__RefHeading__27870_411373615)

[[dbo].[Processmail] 55](#__RefHeading__27872_411373615)

[[dbo].[SyncCompany] 62](#__RefHeading__27874_411373615)

[[dbo].[SyncProducts] 63](#__RefHeading__27876_411373615)

[[dbo].[UpdateOrderEvent] 64](#__RefHeading__27878_411373615)

[Table-valued Functions 65](#__RefHeading__27880_411373615)

[[dbo].[QFN\_SPLIT] 66](#__RefHeading__27882_411373615)

[[dbo].[split] 67](#__RefHeading__27884_411373615)

[Database Roles 68](#__RefHeading__27886_411373615)

[db\_accessadmin 68](#__RefHeading__27888_411373615)

[db\_backupoperator 69](#__RefHeading__27890_411373615)

[db\_datareader 69](#__RefHeading__27892_411373615)

[db\_datawriter 69](#__RefHeading__27894_411373615)

[db\_ddladmin 69](#__RefHeading__27896_411373615)

[db\_denydatareader 70](#__RefHeading__27898_411373615)

[db\_denydatawriter 70](#__RefHeading__27900_411373615)

[db\_owner 70](#__RefHeading__27902_411373615)

[db\_securityadmin 71](#__RefHeading__27904_411373615)

[public 71](#__RefHeading__27906_411373615)

|  |
| --- |
| AMAZONA-HLGH6D4 |

Databases(1)

* [CampaignTrack\_Dev](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/index)

Server Properties

|  |  |
| --- | --- |
| Property | Value |
| Product | Microsoft SQL Server |
| Version | 11.0.2218.0 |
| Language | English (United States) |
| Platform | NT x64 |
| Edition | Standard Edition (64-bit) |
| Processors | 1 |
| OS Version | 6.2 (9200) |
| Physical Memory | 3840 |
| Is Clustered | False |
| Root Directory | C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |

Server Settings

|  |  |
| --- | --- |
| Property | Value |
| Default data file path | C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\ |
| Default backup file path | C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Backup |
| Default log file path | C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\ |
| Recovery Interval (minutes) | 0 |
| Default index fill factor | 0 |
| Default backup media retention | 0 |
| Compress Backup | False |

Advanced Server Settings

|  |  |
| --- | --- |
| Property | Value |
| Locks | 0 |
| Nested triggers enabled | True |
| Allow triggers to fire others | True |
| Default language | English |
| Network packet size | 4096 |
| Default fulltext language LCID | 1033 |
| Two-digit year cutoff | 2049 |
| Remote login timeout | 10 |
| Cursor threshold | -1 |
| Max text replication size | 65536 |
| Parallelism cost threshold | 5 |
| Scan for startup procs | False |
| Transform noise words | False |
| Blocked process threshold | 0 |
| Filestream access level | False |
| Optimize for ad hoc workloads | False |

|  |
| --- |
| User databases |

Databases(1)

* [CampaignTrack\_Dev](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/index)

|  |
| --- |
| CampaignTrack\_Dev Database |

Database Properties

|  |  |
| --- | --- |
| Property | Value |
| SQL Server Version | SQL Server 2012 |
| Compatibility Level | SQL Server 2012 |
| Database Encryption Enabled | False |
| Last backup time | - |
| Last log backup time | - |
| Creation date | Jun 17 2014 |
| Users | 4 |
| Database size | 204.13 MB |
| Unallocated space | 0.86 MB |

Database Options

|  |  |
| --- | --- |
| Property | Value |
| Compatibility Level | 110 |
| Database collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Restrict access | MULTI\_USER |
| Is read-only | False |
| Auto close | False |
| Auto shrink | False |
| Database status | ONLINE |
| In standby | False |
| Cleanly shutdown | False |
| Supplemental logging enabled | False |
| Snapshot isolation state | OFF |
| Read committed snapshot on | False |
| Recovery model | FULL |
| Page verify option | CHECKSUM |
| Auto create statistics | True |
| Auto update statistics | True |
| Auto update statistics asynchronously | False |
| ANSI NULL default | False |
| ANSI NULL enabled | False |
| ANSI padding enabled | False |
| ANSI warnings enabled | False |
| Arithmetic abort enabled | False |
| Concatenating NULL yields NULL | False |
| Numeric roundabort enabled | False |
| Quoted Identifier On | False |
| Recursive triggers enabled | False |
| Close cursors on commit | False |
| Local cursors by default | False |
| Fulltext enabled | True |
| Trustworthy | False |
| Database chaining | False |
| Forced parameterization | False |
| Master key encrypted by server | False |
| Published | False |
| Subscribed | False |
| Merge published | False |
| Is distribution database | False |
| Sync with backup | False |
| Service broker GUID | 679ff0df-e758-4b27-b8e9-baad7d845014 |
| Service broker enabled | False |
| Log reuse wait | LOG\_BACKUP |
| Date correlation | False |
| CDC enabled | False |
| Encrypted | False |
| Honor broker priority | False |
| Default language | English |
| Default fulltext language LCID | 1033 |
| Nested triggers enabled | True |
| Transform noise words | False |
| Two-digit year cutoff | 2049 |
| Containment | NONE |
| Target recovery time | 0 |
| Database owner | AMAZONA-HLGH6D4\Administrator |

Files

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Size | File Name |
| CampaignTrack\_Prod | Data | 6.06 MB | D:\Sachin\CampaignTrack\_Prod.mdf |
| CampaignTrack\_Prod\_log | Log | 198.06 MB | D:\Sachin\CampaignTrack\_Prod\_log.ldf |

|  |
| --- |
| Tables |

Objects

|  |
| --- |
| Name |
| [dbo.Calendars](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Calendars) This table stores all of the Google Calendars found linked to the primary Google Google account. |
| [dbo.CalendarUser](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CalendarUser) This table stores the relationship between a User and a Google Calendar. |
| [dbo.CancelledOrders](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CancelledOrders) This table stores the Cancelled Orders that have been received via email. |
| [dbo.ColorMaster](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/ColorMaster) This table stores the colours available for Google Calendar Events. |
| [dbo.Company](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company) This table stores the Companies received from the XERO API Contact Method. |
| [dbo.CompanyMail](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CompanyMail) This table stores the Order Emails that have been received against this company. |
| [dbo.CompanyOrders](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CompanyOrders) This table stores the relationship between the Company and Order Record. |
| [dbo.Contacts](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Contacts) |
| [dbo.Events](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Events) This table stores the Events that have been sychronised from each of the Google Calendars. |
| [dbo.EventTracking](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/EventTracking) This table tracks movements from one Google Calendar to another shared google Calendar, the date it was moved on, the new start date and end time etc. |
| [dbo.Mail](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Mail) This table stores the Order emails that have been received. |
| [dbo.OrderContacts](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderContacts) This table stores the Contacts that were specified in the Order Email. These are people that are relating to a specific order. |
| [dbo.OrderEvents](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderEvents) This table stores the Events that were created resulting from an Order. |
| [dbo.OrderItems](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderItems) This table stores the Order Items that have been sourced from the Order Email. |
| [dbo.Orders](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders) This table Stores the Orders that have been received. |
| [dbo.OrderType](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderType) This Procedure Stores the Type of Orders used by the system. |
| [dbo.Organisation](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Organisation) This table will store the internal organisations using this software service. The solution will be multi-tennant so this will store companies using it. |
| [dbo.ProductGroups](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/ProductGroups) This table will store the Product Groups. |
| [dbo.Products](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Products) This table stores the Products that are synchronised from XERO Items method. |
| [dbo.ProductSchedules](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/ProductSchedules) This table stores the Product Configuration Settings. |
| [dbo.ProductUser](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/ProductUser) This Table is Not used. |
| [dbo.Property](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Property) This table stores the properties received from Order Emails |
| [dbo.PropertyInvoices](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/PropertyInvoices) This table stores the Invoices that have been created in XERO against Property records resulting from Order Emials. |
| [dbo.User](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/User) This table will contact the Users of the system. |
| [dbo.UserTypes](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/UserTypes) This table will contain the different types of Users configured within the system. |

|  |
| --- |
| [dbo].[Calendars] |

MS\_Description

This table stores all of the Google Calendars found linked to the primary Google Google account.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 49 |
| Created | 1:42:56 PM Thursday, 19 June 2014 |
| Last Modified | 1:42:56 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | Org\_Id Multi-Org Identifier | int | 4 | True |  |  |
|  | Name Google Calendar Name | varchar(2000) | 2000 | True |  |  |
|  | Created Date Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Calendars | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Calendars]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Org\_Id] [int] NULL,  [Name] [varchar] (2000) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_Calendars\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[Calendars] ADD CONSTRAINT [PK\_Calendars] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores all of the Google Calendars found linked to the primary Google Google account. ', 'SCHEMA', N'dbo', 'TABLE', N'Calendars', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date Created', 'SCHEMA', N'dbo', 'TABLE', N'Calendars', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Google Calendar Name', 'SCHEMA', N'dbo', 'TABLE', N'Calendars', 'COLUMN', N'Name'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Multi-Org Identifier', 'SCHEMA', N'dbo', 'TABLE', N'Calendars', 'COLUMN', N'Org\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'Calendars', 'COLUMN', N'Row\_Id'  GO |

Used By

[[dbo].[InsertCalendars]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertCalendars)

[[dbo].[InsertEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertEvents)

[[dbo].[InsertOtherEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertOtherEvents)

|  |
| --- |
| [dbo].[CalendarUser] |

MS\_Description

This table stores the relationship between a User and a Google Calendar.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Row Count (~) | 0 |
| Created | 11:13:08 AM Wednesday, 28 May 2014 |
| Last Modified | 11:13:32 AM Wednesday, 28 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unqiue Id | int | 4 | False | 1 - 1 |  |
|  | UserId FK to User.Row\_Id | int | 4 | True |  |  |
|  | CalendarId FK to Calendar.RowId | int | 4 | True |  |  |
|  | Created Date Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_CalendarUser | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[CalendarUser]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [UserId] [int] NULL,  [CalendarId] [int] NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_CalendarUser\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[CalendarUser] ADD CONSTRAINT [PK\_CalendarUser] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the relationship between a User and a Google Calendar. ', 'SCHEMA', N'dbo', 'TABLE', N'CalendarUser', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Calendar.RowId', 'SCHEMA', N'dbo', 'TABLE', N'CalendarUser', 'COLUMN', N'CalendarId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date Created', 'SCHEMA', N'dbo', 'TABLE', N'CalendarUser', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unqiue Id', 'SCHEMA', N'dbo', 'TABLE', N'CalendarUser', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to User.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'CalendarUser', 'COLUMN', N'UserId'  GO |

|  |
| --- |
| [dbo].[CancelledOrders] |

MS\_Description

This table stores the Cancelled Orders that have been received via email.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 16 |
| Created | 2:54:08 PM Thursday, 19 June 2014 |
| Last Modified | 2:54:08 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |
|  | OrgId FK to Organisation.Row\_Id | int | 4 | True |  |
|  | Property Property Address | varchar(max) | max | True |  |
|  | Service Product Line | varchar(max) | max | True |  |
|  | RequiredDate Required Date of the Service | varchar(50) | 50 | True |  |
|  | FileName File Name as Stored on the Filesystem | varchar(50) | 50 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_CancelledOrders | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[CancelledOrders]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [OrgId] [int] NULL,  [Property] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Service] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [RequiredDate] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [FileName] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[CancelledOrders] ADD CONSTRAINT [PK\_CancelledOrders] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Cancelled Orders that have been received via email. ', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'File Name as Stored on the Filesystem', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', 'COLUMN', N'FileName'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Organisation.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', 'COLUMN', N'OrgId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Property Address', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', 'COLUMN', N'Property'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Required Date of the Service', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', 'COLUMN', N'RequiredDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Product Line', 'SCHEMA', N'dbo', 'TABLE', N'CancelledOrders', 'COLUMN', N'Service'  GO |

Used By

[[dbo].[InsertCancelledOrder]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertCancelledOrder)

|  |
| --- |
| [dbo].[ColorMaster] |

MS\_Description

This table stores the colours available for Google Calendar Events.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 11 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |
|  | Color Google Cal Hex Code | varchar(50) | 50 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_ColorMaster | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[ColorMaster]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Color] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[ColorMaster] ADD CONSTRAINT [PK\_ColorMaster] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the colours available for Google Calendar Events. ', 'SCHEMA', N'dbo', 'TABLE', N'ColorMaster', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Google Cal Hex Code', 'SCHEMA', N'dbo', 'TABLE', N'ColorMaster', 'COLUMN', N'Color'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'ColorMaster', 'COLUMN', N'Row\_Id'  GO |

|  |
| --- |
| [dbo].[Company] |

MS\_Description

This table stores the Companies received from the XERO API Contact Method.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 563 |
| Created | 1:37:46 PM Thursday, 19 June 2014 |
| Last Modified | 1:37:46 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique ID | int | 4 | False | 1 - 1 |  |
|  | Org\_Id FK to Organisation.Row\_Id | int | 4 | True |  |  |
|  | CompanyCode Company Code Used for Calendar Event Titles. | varchar(50) | 50 | True |  |  |
|  | XeroName XERO Item Name Syncronised from the Service API call. | varchar(max) | max | True |  |  |
|  | ScrappedName This is the company name received in the Order Email. They are sometimes different to the Name listed in XERO. | varchar(max) | max | True |  |  |
|  | CreateEvent Create Google Events for this company - yes/no | bit | 1 | True |  |  |
|  | CreateInvoice Create Invoices in XERO for this Company - yes no. | bit | 1 | True |  |  |
|  | Created Date created from XERO Sync. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Company | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Company]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Org\_Id] [int] NULL,  [CompanyCode] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [XeroName] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [ScrappedName] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [CreateEvent] [bit] NULL,  [CreateInvoice] [bit] NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_Company\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Company] ADD CONSTRAINT [PK\_Company] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Companies received from the XERO API Contact Method.', 'SCHEMA', N'dbo', 'TABLE', N'Company', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Company Code Used for Calendar Event Titles. ', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'CompanyCode'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date created from XERO Sync.', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Create Google Events for this company - yes/no', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'CreateEvent'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Create Invoices in XERO for this Company - yes no.', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'CreateInvoice'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Organisation.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'Org\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique ID', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This is the company name received in the Order Email. They are sometimes different to the Name listed in XERO. ', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'ScrappedName'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'XERO Item Name Syncronised from the Service API call. ', 'SCHEMA', N'dbo', 'TABLE', N'Company', 'COLUMN', N'XeroName'  GO |

Used By

[[dbo].[Contacts]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Contacts)

[[dbo].[Property]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Property)

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

[[dbo].[SyncCompany]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/SyncCompany)

|  |
| --- |
| [dbo].[CompanyMail] |

MS\_Description

This table stores the Order Emails that have been received against this company.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Row Count (~) | 219 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | Company\_Id FK to Company.Row\_Id | int | 4 | True |  |  |
|  | Mail\_Id FK to Mail.Row\_Id | int | 4 | True |  |  |
|  | Created Date created. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Company\_Mail | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[CompanyMail]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Company\_Id] [int] NULL,  [Mail\_Id] [int] NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_CompanyMail\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[CompanyMail] ADD CONSTRAINT [PK\_Company\_Mail] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Order Emails that have been received against this company.', 'SCHEMA', N'dbo', 'TABLE', N'CompanyMail', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Company.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'CompanyMail', 'COLUMN', N'Company\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date created. ', 'SCHEMA', N'dbo', 'TABLE', N'CompanyMail', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Mail.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'CompanyMail', 'COLUMN', N'Mail\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'CompanyMail', 'COLUMN', N'Row\_Id'  GO |

Used By

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[CompanyOrders] |

MS\_Description

This table stores the relationship between the Company and Order Record.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Row Count (~) | 243 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | CompanyId FK to Company.Row\_Id | int | 4 | True |  |  |
|  | OrderId FK to Order.Row\_ID | int | 4 | True |  |  |
|  | Created Date Created. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_CompanyOrders | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[CompanyOrders]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [CompanyId] [int] NULL,  [OrderId] [int] NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_CompanyOrders\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[CompanyOrders] ADD CONSTRAINT [PK\_CompanyOrders] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the relationship between the Company and Order Record. ', 'SCHEMA', N'dbo', 'TABLE', N'CompanyOrders', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Company.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'CompanyOrders', 'COLUMN', N'CompanyId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date Created. ', 'SCHEMA', N'dbo', 'TABLE', N'CompanyOrders', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Order.Row\_ID', 'SCHEMA', N'dbo', 'TABLE', N'CompanyOrders', 'COLUMN', N'OrderId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'CompanyOrders', 'COLUMN', N'Row\_Id'  GO |

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[Contacts] |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 110 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:37:46 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id | int | 4 | False | 1 - 1 |  |
|  | CompanyId | int | 4 | True |  |  |
|  | Name | varchar(max) | max | True |  |  |
|  | Value | varchar(max) | max | True |  |  |
|  | ContactType | varchar(50) | 50 | True |  |  |
|  | Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Contacts | Row\_Id | True |

Foreign Keys

|  |  |  |
| --- | --- | --- |
| Name | No Check | Columns |
| FK\_Contacts\_Company | True | CompanyId->[[dbo].[Company].[Row\_Id]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company) |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Contacts]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [CompanyId] [int] NULL,  [Name] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Value] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [ContactType] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_Contacts\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Contacts] ADD CONSTRAINT [PK\_Contacts] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  ALTER TABLE [dbo].[Contacts] WITH NOCHECK ADD CONSTRAINT [FK\_Contacts\_Company] FOREIGN KEY ([CompanyId]) REFERENCES [dbo].[Company] ([Row\_Id])  GO  ALTER TABLE [dbo].[Contacts] NOCHECK CONSTRAINT [FK\_Contacts\_Company]  GO |

Uses

[[dbo].[Company]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company)

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[Events] |

MS\_Description

This table stores the Events that have been sychronised from each of the Google Calendars.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 4161 |
| Created | 3:46:46 PM Friday, 13 June 2014 |
| Last Modified | 5:08:41 PM Tuesday, 17 June 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id | int | 4 | False | 1 - 1 |
|  | CalendarId | int | 4 | True |  |
|  | EventId | varchar(500) | 500 | True |  |
|  | RecurringEventId | varchar(500) | 500 | True |  |
|  | ColorId | varchar(50) | 50 | True |  |
|  | Attendees | varchar(50) | 50 | True |  |
|  | AttendeesOmitted | bit | 1 | True |  |
|  | AnyoneCanAddSelf | bit | 1 | True |  |
|  | Locked | bit | 1 | True |  |
|  | Title | varchar(max) | max | True |  |
|  | Location | varchar(500) | 500 | True |  |
|  | Description | varchar(max) | max | True |  |
|  | Kind | varchar(50) | 50 | True |  |
|  | StartDate | datetime | 8 | True |  |
|  | EndDate | datetime | 8 | True |  |
|  | MovedOn | datetime | 8 | True |  |
|  | Status | varchar(50) | 50 | True |  |
|  | Recurrence | varchar(50) | 50 | True |  |
|  | Organizer | varchar(50) | 50 | True |  |
|  | Creator | varchar(50) | 50 | True |  |
|  | Sequence | int | 4 | True |  |
|  | Transperancy | varchar(50) | 50 | True |  |
|  | Updated | datetime | 8 | True |  |
|  | Visibility | varchar(50) | 50 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_OtherEvents | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Events]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [CalendarId] [int] NULL,  [EventId] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [RecurringEventId] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [ColorId] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Attendees] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [AttendeesOmitted] [bit] NULL,  [AnyoneCanAddSelf] [bit] NULL,  [Locked] [bit] NULL,  [Title] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Location] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Description] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Kind] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [StartDate] [datetime] NULL,  [EndDate] [datetime] NULL,  [MovedOn] [datetime] NULL,  [Status] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Recurrence] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Organizer] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Creator] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Sequence] [int] NULL,  [Transperancy] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Updated] [datetime] NULL,  [Visibility] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Events] ADD CONSTRAINT [PK\_OtherEvents] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Events that have been sychronised from each of the Google Calendars. ', 'SCHEMA', N'dbo', 'TABLE', N'Events', NULL, NULL  GO |

Used By

[[dbo].[InsertOtherEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertOtherEvents)

|  |
| --- |
| [dbo].[EventTracking] |

MS\_Description

This table tracks movements from one Google Calendar to another shared google Calendar, the date it was moved on, the new start date and end time etc.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 290 |
| Created | 10:55:10 AM Tuesday, 3 June 2014 |
| Last Modified | 10:55:10 AM Tuesday, 3 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | CalendarId FK to Calendar.Row\_Id | int | 4 | True |  |  |
|  | OrderId FK to Order.Row\_Id | varchar(max) | max | True |  |  |
|  | Title Title of the Event | varchar(max) | max | True |  |  |
|  | MovedOn Date it was Moved. | datetime | 8 | True |  |  |
|  | StartDate Start Date / Time of Event | datetime | 8 | True |  |  |
|  | EndDate End Date / Time of the Event | datetime | 8 | True |  |  |
|  | Status Status of the Event | varchar(500) | 500 | True |  |  |
|  | Created Created Date | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_EventTracking | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[EventTracking]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [CalendarId] [int] NULL,  [OrderId] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Title] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [MovedOn] [datetime] NULL,  [StartDate] [datetime] NULL,  [EndDate] [datetime] NULL,  [Status] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_\_EventTrac\_\_Creat\_\_282DF8C2] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[EventTracking] ADD CONSTRAINT [PK\_EventTracking] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table tracks movements from one Google Calendar to another shared google Calendar, the date it was moved on, the new start date and end time etc. ', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Calendar.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'CalendarId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created Date', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'End Date / Time of the Event', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'EndDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date it was Moved. ', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'MovedOn'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Order.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'OrderId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Start Date / Time of Event', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'StartDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Status of the Event', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'Status'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Title of the Event', 'SCHEMA', N'dbo', 'TABLE', N'EventTracking', 'COLUMN', N'Title'  GO |

Used By

[[dbo].[InsertEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertEvents)

|  |
| --- |
| [dbo].[Mail] |

MS\_Description

This table stores the Order emails that have been received.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 219 |
| Created | 1:40:55 PM Thursday, 19 June 2014 |
| Last Modified | 1:40:55 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_ID Unique Id | int | 4 | False | 1 - 1 |  |
|  | Org\_Id FK to Organisation.Row\_Id | int | 4 | True |  |  |
|  | FileName File Name of the Email | varchar(max) | max | True |  |  |
|  | SentDate Date it was sent | varchar(max) | max | True |  |  |
|  | Subject Subject of the Email | varchar(max) | max | True |  |  |
|  | Created Created date. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Mail | Row\_ID | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Mail]  (  [Row\_ID] [int] NOT NULL IDENTITY(1, 1),  [Org\_Id] [int] NULL,  [FileName] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [SentDate] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Subject] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_Mail\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Mail] ADD CONSTRAINT [PK\_Mail] PRIMARY KEY CLUSTERED ([Row\_ID]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Order emails that have been received. ', 'SCHEMA', N'dbo', 'TABLE', N'Mail', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created date. ', 'SCHEMA', N'dbo', 'TABLE', N'Mail', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'File Name of the Email', 'SCHEMA', N'dbo', 'TABLE', N'Mail', 'COLUMN', N'FileName'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Organisation.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'Mail', 'COLUMN', N'Org\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'Mail', 'COLUMN', N'Row\_ID'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date it was sent', 'SCHEMA', N'dbo', 'TABLE', N'Mail', 'COLUMN', N'SentDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Subject of the Email', 'SCHEMA', N'dbo', 'TABLE', N'Mail', 'COLUMN', N'Subject'  GO |

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[InsertPropertyInvoiceNo]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertPropertyInvoiceNo)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[OrderContacts] |

MS\_Description

This table stores the Contacts that were specified in the Order Email. These are people that are relating to a specific order.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Row Count (~) | 255 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | OrderId FK to Order.Row\_Id | int | 4 | True |  |  |
|  | ContactId FK to Contact.Row\_Id | int | 4 | True |  |  |
|  | Created Date Created. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_OrderContacts | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[OrderContacts]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [OrderId] [int] NULL,  [ContactId] [int] NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_OrderContacts\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[OrderContacts] ADD CONSTRAINT [PK\_OrderContacts] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Contacts that were specified in the Order Email. These are people that are relating to a specific order. ', 'SCHEMA', N'dbo', 'TABLE', N'OrderContacts', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Contact.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderContacts', 'COLUMN', N'ContactId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date Created. ', 'SCHEMA', N'dbo', 'TABLE', N'OrderContacts', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Order.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderContacts', 'COLUMN', N'OrderId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderContacts', 'COLUMN', N'Row\_Id'  GO |

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[OrderEvents] |

MS\_Description

This table stores the Events that were created resulting from an Order.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 243 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 5:08:17 PM Tuesday, 17 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | Order\_Id FK to Order.Row\_Id | int | 4 | True |  |  |
|  | TimeZone Timezone of the Calendar | varchar(max) | max | True |  |  |
|  | EventCreated Flag of if event was created | bit | 1 | True |  | ((0)) |
|  | Created Created Date. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_OrderItemEvents | Row\_Id | True |

Foreign Keys

|  |  |  |
| --- | --- | --- |
| Name | No Check | Columns |
| FK\_OrderItemEvents\_Orders | True | Order\_Id->[[dbo].[Orders].[Row\_Id]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders) |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[OrderEvents]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Order\_Id] [int] NULL,  [TimeZone] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [EventCreated] [bit] NULL CONSTRAINT [DF\_OrderEvents\_EventCreated] DEFAULT ((0)),  [Created] [datetime] NULL CONSTRAINT [DF\_OrderEvents\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[OrderEvents] ADD CONSTRAINT [PK\_OrderItemEvents] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  ALTER TABLE [dbo].[OrderEvents] WITH NOCHECK ADD CONSTRAINT [FK\_OrderItemEvents\_Orders] FOREIGN KEY ([Order\_Id]) REFERENCES [dbo].[Orders] ([Row\_Id])  GO  ALTER TABLE [dbo].[OrderEvents] NOCHECK CONSTRAINT [FK\_OrderItemEvents\_Orders]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Events that were created resulting from an Order. ', 'SCHEMA', N'dbo', 'TABLE', N'OrderEvents', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created Date. ', 'SCHEMA', N'dbo', 'TABLE', N'OrderEvents', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Flag of if event was created', 'SCHEMA', N'dbo', 'TABLE', N'OrderEvents', 'COLUMN', N'EventCreated'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Order.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderEvents', 'COLUMN', N'Order\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderEvents', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Timezone of the Calendar', 'SCHEMA', N'dbo', 'TABLE', N'OrderEvents', 'COLUMN', N'TimeZone'  GO |

Uses

[[dbo].[Orders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders)

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

[[dbo].[UpdateOrderEvent]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/UpdateOrderEvent)

|  |
| --- |
| [dbo].[OrderItems] |

MS\_Description

This table stores the Order Items that have been sourced from the Order Email.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 669 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | Order\_Id Order Id | int | 4 | True |  |  |
|  | Xero\_Id Xero ID | int | 4 | True |  |  |
|  | Name Name | varchar(max) | max | True |  |  |
|  | ClientPrice Price | varchar(max) | max | True |  |  |
|  | CostPrice Cost Price | varchar(50) | 50 | True |  |  |
|  | Comments Comments | varchar(max) | max | True |  |  |
|  | Created Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_OrderItems | Row\_Id | True |

Foreign Keys

|  |  |  |
| --- | --- | --- |
| Name | No Check | Columns |
| FK\_OrderItems\_Order | True | Order\_Id->[[dbo].[Orders].[Row\_Id]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders) |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[OrderItems]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Order\_Id] [int] NULL,  [Xero\_Id] [int] NULL,  [Name] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [ClientPrice] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [CostPrice] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Comments] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_OrderItems\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[OrderItems] ADD CONSTRAINT [PK\_OrderItems] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  ALTER TABLE [dbo].[OrderItems] WITH NOCHECK ADD CONSTRAINT [FK\_OrderItems\_Order] FOREIGN KEY ([Order\_Id]) REFERENCES [dbo].[Orders] ([Row\_Id])  GO  ALTER TABLE [dbo].[OrderItems] NOCHECK CONSTRAINT [FK\_OrderItems\_Order]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Order Items that have been sourced from the Order Email. ', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Price ', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'ClientPrice'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Comments', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'Comments'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Cost Price', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'CostPrice'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Name ', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'Name'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Order Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'Order\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Xero ID', 'SCHEMA', N'dbo', 'TABLE', N'OrderItems', 'COLUMN', N'Xero\_Id'  GO |

Uses

[[dbo].[Orders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders)

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[Orders] |

MS\_Description

This table Stores the Orders that have been received.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 243 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | OrderType\_Id FK to OrderType.Row\_ID | int | 4 | True |  |  |
|  | Property\_Id FK to Property.Row\_Id | int | 4 | True |  |  |
|  | Mail\_Id FK to Mail.Row\_Id | int | 4 | True |  |  |
|  | OrderId FK to Order.Row\_ID | varchar(max) | max | True |  |  |
|  | RequiredDate Required date of the Service | datetime | 8 | True |  |  |
|  | Description Description of the Order | varchar(max) | max | True |  |  |
|  | Created Created. | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Order | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Orders]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [OrderType\_Id] [int] NULL,  [Property\_Id] [int] NULL,  [Mail\_Id] [int] NULL,  [OrderId] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [RequiredDate] [datetime] NULL,  [Description] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_Order\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Orders] ADD CONSTRAINT [PK\_Order] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table Stores the Orders that have been received. ', 'SCHEMA', N'dbo', 'TABLE', N'Orders', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created. ', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Description of the Order', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'Description'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Mail.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'Mail\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Order.Row\_ID', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'OrderId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to OrderType.Row\_ID', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'OrderType\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Property.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'Property\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Required date of the Service', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'RequiredDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'Orders', 'COLUMN', N'Row\_Id'  GO |

Used By

[[dbo].[OrderEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderEvents)

[[dbo].[OrderItems]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderItems)

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[OrderType] |

MS\_Description

This Procedure Stores the Type of Orders used by the system.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 0 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 1:12:58 AM Tuesday, 13 May 2014 |

Columns

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls |
|  | Row\_Id | int | 4 | False |
|  | Type | varchar(50) | 50 | True |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Order\_Type | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[OrderType]  (  [Row\_Id] [int] NOT NULL,  [Type] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[OrderType] ADD CONSTRAINT [PK\_Order\_Type] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This Procedure Stores the Type of Orders used by the system. ', 'SCHEMA', N'dbo', 'TABLE', N'OrderType', NULL, NULL  GO |

|  |
| --- |
| [dbo].[Organisation] |

MS\_Description

This table will store the internal organisations using this software service. The solution will be multi-tennant so this will store companies using it.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 0 |
| Created | 1:35:31 PM Thursday, 19 June 2014 |
| Last Modified | 1:36:57 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls |
|  | Row\_Id Uniquie Id | int | 4 | False |
|  | Name Company Name | varchar(100) | 100 | True |
|  | IsActive Is Active | bit | 1 | True |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Organisation | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Organisation]  (  [Row\_Id] [int] NOT NULL,  [Name] [varchar] (100) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [IsActive] [bit] NULL  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[Organisation] ADD CONSTRAINT [PK\_Organisation] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table will store the internal organisations using this software service. The solution will be multi-tennant so this will store companies using it.', 'SCHEMA', N'dbo', 'TABLE', N'Organisation', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Is Active', 'SCHEMA', N'dbo', 'TABLE', N'Organisation', 'COLUMN', N'IsActive'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Company Name', 'SCHEMA', N'dbo', 'TABLE', N'Organisation', 'COLUMN', N'Name'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Uniquie Id', 'SCHEMA', N'dbo', 'TABLE', N'Organisation', 'COLUMN', N'Row\_Id'  GO |

|  |
| --- |
| [dbo].[ProductGroups] |

MS\_Description

This table will store the Product Groups.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 0 |
| Created | 11:56:26 AM Wednesday, 28 May 2014 |
| Last Modified | 11:57:15 AM Wednesday, 28 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id | int | 4 | False | 1 - 1 |  |
|  | ProductId | int | 4 | True |  |  |
|  | Name | varchar(50) | 50 | True |  |  |
|  | Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_ProductGroups | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[ProductGroups]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [ProductId] [int] NULL,  [Name] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_ProductGroups\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[ProductGroups] ADD CONSTRAINT [PK\_ProductGroups] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table will store the Product Groups. ', 'SCHEMA', N'dbo', 'TABLE', N'ProductGroups', NULL, NULL  GO |

|  |
| --- |
| [dbo].[Products] |

MS\_Description

This table stores the Products that are synchronised from XERO Items method.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Heap | True |
| Row Count (~) | 321 |
| Created | 2:11:38 PM Saturday, 24 May 2014 |
| Last Modified | 2:11:38 PM Saturday, 24 May 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |
|  | CompanyId FK to Company.Row\_Id | varchar(max) | max | True |  |
|  | XeroCode XERO Code | varchar(100) | 100 | True |  |
|  | ProductDescription Product Description received in the Order Emails | varchar(500) | 500 | True |  |
|  | XeroItemDescription Item Description in XERO | varchar(500) | 500 | True |  |
|  | SalesUnitPrice Sales Price from XERO | decimal(18,2) | 9 | True |  |
|  | SalesAccountCode Sales Account Code from XERO | varchar(50) | 50 | True |  |
|  | SalesTaxType Sales Tax Type from XERO. | varchar(100) | 100 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Products | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Products]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [CompanyId] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [XeroCode] [varchar] (100) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [ProductDescription] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [XeroItemDescription] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [SalesUnitPrice] [decimal] (18, 2) NULL,  [SalesAccountCode] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [SalesTaxType] [varchar] (100) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Products] ADD CONSTRAINT [PK\_Products] PRIMARY KEY NONCLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Products that are synchronised from XERO Items method. ', 'SCHEMA', N'dbo', 'TABLE', N'Products', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Company.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'CompanyId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Product Description received in the Order Emails', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'ProductDescription'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Sales Account Code from XERO', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'SalesAccountCode'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Sales Tax Type from XERO.', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'SalesTaxType'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Sales Price from XERO', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'SalesUnitPrice'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'XERO Code', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'XeroCode'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Item Description in XERO', 'SCHEMA', N'dbo', 'TABLE', N'Products', 'COLUMN', N'XeroItemDescription'  GO |

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[GetProductSchedules]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetProductSchedules)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

[[dbo].[SyncProducts]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/SyncProducts)

|  |
| --- |
| [dbo].[ProductSchedules] |

MS\_Description

This table stores the Product Configuration Settings.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 125 |
| Created | 12:14:09 AM Saturday, 17 May 2014 |
| Last Modified | 12:14:10 AM Saturday, 17 May 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id | int | 4 | False | 1 - 1 |
|  | XeroCode | varchar(max) | max | True |  |
|  | Value | varchar(100) | 100 | True |  |
|  | Title | varchar(500) | 500 | True |  |
|  | ColorId | int | 4 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_ProductSchedules | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[ProductSchedules]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [XeroCode] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Value] [varchar] (100) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Title] [varchar] (500) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [ColorId] [int] NULL  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[ProductSchedules] ADD CONSTRAINT [PK\_ProductSchedules] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Product Configuration Settings. ', 'SCHEMA', N'dbo', 'TABLE', N'ProductSchedules', NULL, NULL  GO |

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[GetProductSchedules]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetProductSchedules)

|  |
| --- |
| [dbo].[ProductUser] |

MS\_Description

This Table is Not used.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 0 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 11:55:43 AM Wednesday, 28 May 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id | int | 4 | False | 1 - 1 |
|  | User\_Id | int | 4 | True |  |
|  | Product\_Id | varchar(50) | 50 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Product\_User | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[ProductUser]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [User\_Id] [int] NULL,  [Product\_Id] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[ProductUser] ADD CONSTRAINT [PK\_Product\_User] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This Table is Not used. ', 'SCHEMA', N'dbo', 'TABLE', N'ProductUser', NULL, NULL  GO |

|  |
| --- |
| [dbo].[Property] |

MS\_Description

This table stores the properties received from Order Emails

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 219 |
| Created | 2:58:05 PM Wednesday, 28 May 2014 |
| Last Modified | 1:37:46 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | Company\_Id FK to Company.Row\_ID | int | 4 | True |  |  |
|  | Property\_Id FK to Property.Row\_Id | varchar(max) | max | True |  |  |
|  | Name Property Address | varchar(max) | max | True |  |  |
|  | Latitude Property Lat | varchar(50) | 50 | True |  |  |
|  | Longitude Property Long | varchar(50) | 50 | True |  |  |
|  | Created Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_Property | Row\_Id | True |

Foreign Keys

|  |  |  |
| --- | --- | --- |
| Name | No Check | Columns |
| FK\_Property\_Company | True | Company\_Id->[[dbo].[Company].[Row\_Id]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company) |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[Property]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Company\_Id] [int] NULL,  [Property\_Id] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Name] [varchar] (max) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Latitude] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Longitude] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_Property\_Created] DEFAULT (getdate())  ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]  GO  ALTER TABLE [dbo].[Property] ADD CONSTRAINT [PK\_Property] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  ALTER TABLE [dbo].[Property] WITH NOCHECK ADD CONSTRAINT [FK\_Property\_Company] FOREIGN KEY ([Company\_Id]) REFERENCES [dbo].[Company] ([Row\_Id])  GO  ALTER TABLE [dbo].[Property] NOCHECK CONSTRAINT [FK\_Property\_Company]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the properties received from Order Emails', 'SCHEMA', N'dbo', 'TABLE', N'Property', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Company.Row\_ID', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Company\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Property Lat', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Latitude'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Property Long', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Longitude'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Property Address', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Name'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Property.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Property\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'Property', 'COLUMN', N'Row\_Id'  GO |

Uses

[[dbo].[Company]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company)

Used By

[[dbo].[GetDataforCalenderEntry]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry)

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[PropertyInvoices] |

MS\_Description

This table stores the Invoices that have been created in XERO against Property records resulting from Order Emials.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 211 |
| Created | 1:12:58 AM Tuesday, 13 May 2014 |
| Last Modified | 12:23:17 AM Friday, 23 May 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | PropertyId FK to Property.Row\_Id | int | 4 | True |  |  |
|  | MailId FK to Mail.Row\_Id | int | 4 | True |  |  |
|  | InvoiceNo Invoice Number in XERO | varchar(50) | 50 | True |  |  |
|  | Created Created | datetime | 8 | True |  | (getdate()) |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_ProductInvoices | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[PropertyInvoices]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [PropertyId] [int] NULL,  [MailId] [int] NULL,  [InvoiceNo] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_ProductInvoices\_Created] DEFAULT (getdate())  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[PropertyInvoices] ADD CONSTRAINT [PK\_ProductInvoices] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table stores the Invoices that have been created in XERO against Property records resulting from Order Emials. ', 'SCHEMA', N'dbo', 'TABLE', N'PropertyInvoices', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created', 'SCHEMA', N'dbo', 'TABLE', N'PropertyInvoices', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Invoice Number in XERO', 'SCHEMA', N'dbo', 'TABLE', N'PropertyInvoices', 'COLUMN', N'InvoiceNo'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Mail.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'PropertyInvoices', 'COLUMN', N'MailId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Property.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'PropertyInvoices', 'COLUMN', N'PropertyId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'PropertyInvoices', 'COLUMN', N'Row\_Id'  GO |

Used By

[[dbo].[InsertPropertyInvoiceNo]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertPropertyInvoiceNo)

|  |
| --- |
| [dbo].[User] |

MS\_Description

This table will contact the Users of the system.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 0 |
| Created | 2:54:54 PM Thursday, 19 June 2014 |
| Last Modified | 2:54:54 PM Thursday, 19 June 2014 |

Columns

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity | Default |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |  |
|  | OrgId FK to Organisation.Row\_id | int | 4 | True |  |  |
|  | UserType FK to UserType.Row\_Id | int | 4 | True |  |  |
|  | FirstName First Name | varchar(50) | 50 | True |  |  |
|  | LastName Last Name | varchar(50) | 50 | True |  |  |
|  | EmailAddress Email address/Login | varchar(50) | 50 | True |  |  |
|  | Password Password | varchar(50) | 50 | True |  |  |
|  | IsActive Is Active User Flag | bit | 1 | True |  |  |
|  | Created Created | datetime | 8 | True |  | (getdate()) |
|  | Updated Updated | datetime | 8 | True |  |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_User | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[User]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [OrgId] [int] NULL,  [UserType] [int] NULL,  [FirstName] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [LastName] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [EmailAddress] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [Password] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL,  [IsActive] [bit] NULL,  [Created] [datetime] NULL CONSTRAINT [DF\_User\_Created] DEFAULT (getdate()),  [Updated] [datetime] NULL  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[User] ADD CONSTRAINT [PK\_User] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table will contact the Users of the system. ', 'SCHEMA', N'dbo', 'TABLE', N'User', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Created ', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'Created'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Email address/Login', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'EmailAddress'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'First Name ', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'FirstName'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Is Active User Flag', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'IsActive'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Last Name', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'LastName'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to Organisation.Row\_id', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'OrgId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Password', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'Password'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'Row\_Id'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Updated', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'Updated'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'FK to UserType.Row\_Id', 'SCHEMA', N'dbo', 'TABLE', N'User', 'COLUMN', N'UserType'  GO |

|  |
| --- |
| [dbo].[UserTypes] |

MS\_Description

This table will contain the different types of Users configured within the system.

Properties

|  |  |
| --- | --- |
| Property | Value |
| Collation | SQL\_Latin1\_General\_CP1\_CI\_AS |
| Row Count (~) | 0 |
| Created | 11:32:27 AM Wednesday, 28 May 2014 |
| Last Modified | 11:32:27 AM Wednesday, 28 May 2014 |

Columns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Name | Data Type | Max Length (Bytes) | Allow Nulls | Identity |
|  | Row\_Id Unique Id | int | 4 | False | 1 - 1 |
|  | Name User Type | varchar(50) | 50 | True |  |

Indexes

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Name | Columns | Unique |
|  | PK\_UserTypes | Row\_Id | True |

SQL Script

|  |
| --- |
| CREATE TABLE [dbo].[UserTypes]  (  [Row\_Id] [int] NOT NULL IDENTITY(1, 1),  [Name] [varchar] (50) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NULL  ) ON [PRIMARY]  GO  ALTER TABLE [dbo].[UserTypes] ADD CONSTRAINT [PK\_UserTypes] PRIMARY KEY CLUSTERED ([Row\_Id]) ON [PRIMARY]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This table will contain the different types of Users configured within the system. ', 'SCHEMA', N'dbo', 'TABLE', N'UserTypes', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'User Type ', 'SCHEMA', N'dbo', 'TABLE', N'UserTypes', 'COLUMN', N'Name'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Id', 'SCHEMA', N'dbo', 'TABLE', N'UserTypes', 'COLUMN', N'Row\_Id'  GO |

|  |
| --- |
| Stored Procedures |

Objects

|  |
| --- |
| Name |
| [dbo.GetDataforCalenderEntry](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetDataforCalenderEntry) This stored procedure will return the order information from the database in preparation for sending this to Google Calendar. |
| [dbo.GetProductSchedules](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/GetProductSchedules) This Prodcedure returns the Product Configuration for the selected Product XERO Code. It would return optionally: the Title of the Calendar Entry to create, the standard duration of the Event that is to be created, The Colour of the Google Calendar Event to be Created or an email address to forward the order to instead of creating the google calendar event. |
| [dbo.InsertCalendars](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertCalendars) The Procedure will insert the unique google calendar email address and store it in the calendar db if it does not exist. |
| [dbo.InsertCancelledOrder](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertCancelledOrder) The Procedure inserts Cancelled Orders into the CancelledOrders table If a cancelled order has been received via email. |
| [dbo.InsertEvents](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertEvents) This Procedure Stores the Events that are found for all of the Google Calendars stored in the Calendar table. |
| [dbo.InsertOtherEvents](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertOtherEvents) This procedure stores all events sourced from Google Calendars for all the calendars in the calendar table, into the Events table. This data is syncronised with Google Calendar for all the shared calendars and will be available for display on a custom calendar. |
| [dbo.InsertPropertyInvoiceNo](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/InsertPropertyInvoiceNo) |
| [dbo.Processmail](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail) This procedure is used to read and parse the html file received by the Windows Service which is stored on the filesystem and create all the related fields in normalised database tables. |
| [dbo.SyncCompany](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/SyncCompany) This procedure receives XML from the XERO Contact API method and stores these values in the Company table. |
| [dbo.SyncProducts](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/SyncProducts) This procedure receives XML from the XERO Items API method and stores these values in the Product table. |
| [dbo.UpdateOrderEvent](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/UpdateOrderEvent) This procedure inserts into OrderEvents when a Google Calendar event is created against an order. |

|  |
| --- |
| [dbo].[GetDataforCalenderEntry] |

MS\_Description

This stored procedure will return the order information from the database in preparation for sending this to Google Calendar.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @OrderId Order Id received via the external system. | varchar(50) | 50 |

Permissions

|  |  |  |
| --- | --- | --- |
| Type | Action | Owning Principal |
| Grant | Execute | public |

SQL Script

|  |
| --- |
| --[GetDataforCalenderEntry] '10986249-10'  CREATE PROC [dbo].[GetDataforCalenderEntry]  @OrderId varchar(50)  AS  select p.Company\_Id, OE.Order\_Id ,O.OrderId, C.CompanyCode Company, C.CompanyCode + ':' + P.Name + ':' + ISNULL(Ps.Title,PR.ProductDescription)Title,PR.ProductDescription,  case when  CHARINDEX('phone PROPERTY', OI.Name)!=0 then  replace(OI.Name,left(OI.Name, CHARINDEX('phone PROPERTY', OI.Name)-1) + 'phone PROPERTY','') else OI.Name end Comments  ,Dateadd(HH,10,M.SentDate) StartDate,  PS.Value,  P.Name + ',Australia' Location,  Convert(date,RequiredDate)RequiredDate,  PS.ColorId,  Xero\_Id into #temp  from Orders O  inner join OrderEvents OE on O.Row\_Id=OE.Order\_Id  inner join OrderItems OI on OI.Order\_Id=O.Row\_Id  inner join Property P on P.Row\_Id=O.Property\_Id  left join Products PR on PR.Row\_Id=OI.Xero\_Id  inner join CompanyOrders CO on CO.OrderId=O.Row\_Id  inner join Company C on C.Row\_Id=CO.CompanyId  left join ProductSchedules PS on Ps.XeroCode=PR.XeroCode and pr.ProductDescription=OI.Name  inner join Mail M on M.Row\_ID=O.Mail\_Id  --inner join Contacts Con on Con.CompanyId=C.Row\_Id  --inner join ProductUser PU on PU.Product\_Id=PR.Row\_Id  --inner join [User] U on U.Row\_Id=PU.User\_Id  where OE.EventCreated=0 and O.OrderId=@OrderId and  len(replace(OI.Name,'Adjustment: Agent',''))>0 and OI.name not like 'Facing%' and OI.name not like 'Total%'  and C.CreateEvent=1  order by OE.Order\_Id,Xero\_Id desc  --select \* from #temp  Select T.OrderId,Company,Title,StartDate,EndDate, Location,Comments,REPLACE(REPLACE(C.Name + ' on ' + Value, CHAR(13),''), CHAR(10),'') SalesContact,RequiredDate,ColorId,ProductDescription From  (  Select Order\_Id ORD,CONVERT(date, RequiredDate)RequiredDate, OrderId,Company,Title,  DATEADD(MINUTE,0,StartDate)StartDate,DATEADD(MINUTE,Convert(int,Value),StartDate) EndDate,  Location,T.Comments ,ColorId,ProductDescription from #temp T1  Outer Apply(Select Top 1 Comments from #temp T2 where T1.OrderId=T2.OrderId and T2.ProductDescription is null order by t2.Comments desc )T  )T inner join OrderContacts OC on OC.OrderId=T.ORD  inner join Contacts C on C.Row\_Id=OC.ContactId where title is not null and C.ContactType='Sales' and EndDate is not null    /\*  ;with cte1  as  (  select OrderId,Order\_Id,Company,  STUFF((SELECT ': ' + CAST(Title AS VARCHAR(max)) [text()]  FROM #temp  WHERE Order\_Id =T1.Order\_Id and ProductDescription is not null  FOR XML PATH(''), TYPE)  .value('.','NVARCHAR(MAX)'),1,2,' ') Description,StartDate,Location  from #temp T1 where ProductDescription is not null  group by OrderId,Order\_Id,Company,StartDate,Location  )  ,cte2 as (  select Order\_Id,STUFF((SELECT ': ' + CAST(Title AS VARCHAR(max)) [text()]  FROM #temp  WHERE Order\_Id =T2.Order\_Id and ProductDescription is null  FOR XML PATH(''), TYPE)  .value('.','NVARCHAR(MAX)'),1,2,' ') Comments from #temp T2 where ProductDescription is null  group by Order\_Id  )  Select \* From cte1 T1  left join cte2 T2 on T1.Order\_Id=T2.Order\_Id  --inner join OrderContacts OC on OC.OrderId=T1.Order\_Id  --inner join Contacts C on C.Row\_Id=OC.ContactId  \*/  GO  GRANT EXECUTE ON [dbo].[GetDataforCalenderEntry] TO [public]  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This stored procedure will return the order information from the database in preparation for sending this to Google Calendar.', 'SCHEMA', N'dbo', 'PROCEDURE', N'GetDataforCalenderEntry', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Order Id received via the external system.', 'SCHEMA', N'dbo', 'PROCEDURE', N'GetDataforCalenderEntry', 'PARAMETER', N'@OrderId'  GO |

Uses

[[dbo].[Company]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company)

[[dbo].[CompanyOrders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CompanyOrders)

[[dbo].[Contacts]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Contacts)

[[dbo].[Mail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Mail)

[[dbo].[OrderContacts]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderContacts)

[[dbo].[OrderEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderEvents)

[[dbo].[OrderItems]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderItems)

[[dbo].[Orders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders)

[[dbo].[Products]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Products)

[[dbo].[ProductSchedules]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/ProductSchedules)

[[dbo].[Property]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Property)

|  |
| --- |
| [dbo].[GetProductSchedules] |

MS\_Description

This Prodcedure returns the Product Configuration for the selected Product XERO Code. It would return optionally: the Title of the Calendar Entry to create, the standard duration of the Event that is to be created, The Colour of the Google Calendar Event to be Created or an email address to forward the order to instead of creating the google calendar event.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @XeroID XERO Code stored in Product table. | int | 4 |

SQL Script

|  |
| --- |
| CREATE PROCEDURE [dbo].[GetProductSchedules]  @XeroID int  AS  SELECT PE.Value from Products P inner join ProductSchedules PE on P.XeroCode=PE.XeroCode  WHERE P.Row\_Id=@XeroID  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This Prodcedure returns the Product Configuration for the selected Product XERO Code. It would return optionally: the Title of the Calendar Entry to create, the standard duration of the Event that is to be created, The Colour of the Google Calendar Event to be Created or an email address to forward the order to instead of creating the google calendar event. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'GetProductSchedules', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'XERO Code stored in Product table.', 'SCHEMA', N'dbo', 'PROCEDURE', N'GetProductSchedules', 'PARAMETER', N'@XeroID'  GO |

Uses

[[dbo].[Products]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Products)

[[dbo].[ProductSchedules]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/ProductSchedules)

|  |
| --- |
| [dbo].[InsertCalendars] |

MS\_Description

The Procedure will insert the unique google calendar email address and store it in the calendar db if it does not exist.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @Calendar Google Calendar Email Address | varchar(50) | 50 |

SQL Script

|  |
| --- |
| CREATE PROC [dbo].[InsertCalendars]  @Calendar varchar(50)  AS  IF NOT EXISTS(SELECT 1 FROM Calendars where Name=@Calendar)  INSERT INTO Calendars(Name)(SELECT @Calendar)  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'The Procedure will insert the unique google calendar email address and store it in the calendar db if it does not exist. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCalendars', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Google Calendar Email Address', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCalendars', 'PARAMETER', N'@Calendar'  GO |

Uses

[[dbo].[Calendars]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Calendars)

|  |
| --- |
| [dbo].[InsertCancelledOrder] |

MS\_Description

The Procedure inserts Cancelled Orders into the CancelledOrders table If a cancelled order has been received via email.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @Service Product/Service to be Cancelled | varchar(max) | max |
| @Property Property Address | varchar(max) | max |
| @RequiredDate The Date the Service was Required prior to Cancelling | varchar(max) | max |
| @FileName The Filename of the Cancelled order (email-html) stored on the filesystem. | varchar(max) | max |

SQL Script

|  |
| --- |
| CREATE Procedure [dbo].[InsertCancelledOrder]  @Service varchar(max),  @Property varchar(max),  @RequiredDate varchar(max),  @FileName varchar(max)  as  Insert into [dbo].[CancelledOrders]  Select @Service,@Property,@RequiredDate,@FileName  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'The Procedure inserts Cancelled Orders into the CancelledOrders table If a cancelled order has been received via email. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCancelledOrder', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'The Filename of the Cancelled order (email-html) stored on the filesystem. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCancelledOrder', 'PARAMETER', N'@FileName'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Property Address', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCancelledOrder', 'PARAMETER', N'@Property'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'The Date the Service was Required prior to Cancelling', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCancelledOrder', 'PARAMETER', N'@RequiredDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Product/Service to be Cancelled', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertCancelledOrder', 'PARAMETER', N'@Service'  GO |

Uses

[[dbo].[CancelledOrders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CancelledOrders)

|  |
| --- |
| [dbo].[InsertEvents] |

MS\_Description

This Procedure Stores the Events that are found for all of the Google Calendars stored in the Calendar table.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @Calendar Google Calendar email address | varchar(200) | 200 |
| @OrderId Order Id that the Event was created for. | varchar(max) | max |
| @Title Title of the Event created. | varchar(max) | max |
| @StartDate Start Date/Time of the Event | datetime | 8 |
| @EndDate End Date/Time of the Event | datetime | 8 |
| @MovedOn Date the Event was Moved between Calendars. | datetime | 8 |
| @Status Status of the Event according to Google. | varchar(50) | 50 |

SQL Script

|  |
| --- |
| CREATE PROC [dbo].[InsertEvents]  @Calendar varchar(200),  @OrderId varchar(max),  @Title varchar(max),  @StartDate datetime,  @EndDate datetime,  @MovedOn datetime,  @Status varchar(50)  AS  IF NOT EXISTS(SELECT 1 FROM Calendars where Name=@Calendar)  INSERT INTO Calendars(Name)(SELECT @Calendar)  DECLARE @CalendarId int=(Select Row\_Id from Calendars where Name=@Calendar)  IF NOT EXISTS(Select 1 from EventTracking where CalendarId=@CalendarId and OrderId=@OrderId and Title=@Title)  Insert into EventTracking(CalendarId,OrderId,Title,StartDate,EndDate,MovedOn,Status)  Select @CalendarId,@OrderId,@Title,@StartDate,@EndDate,DATEADD(HH,10,@MovedOn),@Status  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This Procedure Stores the Events that are found for all of the Google Calendars stored in the Calendar table. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Google Calendar email address', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@Calendar'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'End Date/Time of the Event', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@EndDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Date the Event was Moved between Calendars.', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@MovedOn'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Order Id that the Event was created for.', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@OrderId'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Start Date/Time of the Event', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@StartDate'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Status of the Event according to Google.', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@Status'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Title of the Event created.', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertEvents', 'PARAMETER', N'@Title'  GO |

Uses

[[dbo].[Calendars]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Calendars)

[[dbo].[EventTracking]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/EventTracking)

|  |
| --- |
| [dbo].[InsertOtherEvents] |

MS\_Description

This procedure stores all events sourced from Google Calendars for all the calendars in the calendar table, into the Events table. This data is syncronised with Google Calendar for all the shared calendars and will be available for display on a custom calendar.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @Calendar | varchar(50) | 50 |
| @EventId | varchar(500) | 500 |
| @RecurringEventId | varchar(500) | 500 |
| @ColorId | varchar(50) | 50 |
| @Attendees | varchar(50) | 50 |
| @AttendeesOmitted | bit | 1 |
| @AnyoneCanAddSelf | bit | 1 |
| @Locked | bit | 1 |
| @Title | varchar(max) | max |
| @Location | varchar(500) | 500 |
| @Description | varchar(max) | max |
| @Kind | varchar(50) | 50 |
| @StartDate | datetime | 8 |
| @EndDate | datetime | 8 |
| @MovedOn | datetime | 8 |
| @Status | varchar(50) | 50 |
| @Recurrence | varchar(50) | 50 |
| @Organizer | varchar(50) | 50 |
| @Creator | varchar(50) | 50 |
| @Sequence | int | 4 |
| @Transperancy | varchar(50) | 50 |
| @Updated | datetime | 8 |
| @Visibility | varchar(50) | 50 |

SQL Script

|  |
| --- |
| CREATE PROC [dbo].[InsertOtherEvents]  @Calendar VARCHAR(50),  @EventId VARCHAR(500),  @RecurringEventId VARCHAR(500),  @ColorId VARCHAR(50),  @Attendees VARCHAR(50),  @AttendeesOmitted BIT,  @AnyoneCanAddSelf BIT,  @Locked BIT,  @Title VARCHAR(max),  @Location VARCHAR(500),  @Description VARCHAR(max),  @Kind VARCHAR(50),  @StartDate DATETIME,  @EndDate DATETIME,  @MovedOn DATETIME,  @Status VARCHAR(50),  @Recurrence VARCHAR(50),  @Organizer VARCHAR(50),  @Creator VARCHAR(50),  @Sequence INT,  @Transperancy VARCHAR(50),  @Updated DATETIME,  @Visibility VARCHAR(50)  AS  DECLARE @CalendarId INT=(SELECT row\_id FROM calendars WHERE name = @Calendar)  IF NOT EXISTS(SELECT 1  FROM events  WHERE eventid = @EventId)  INSERT INTO events  (calendarid,  eventid,  recurringeventid,  colorid,  attendees,  attendeesomitted,  anyonecanaddself,  locked,  title,  location,  description,  kind,  startdate,  enddate,  movedon,  status,  recurrence,  organizer,  creator,  sequence,  transperancy,  updated,  visibility)  SELECT @CalendarId,  @EventId,  @RecurringEventID,  @ColorId,  @Attendees,  @AttendeesOmitted,  @AnyoneCanAddSelf,  @Locked,  @Title,  @Location,  @Description,  @Kind,  @StartDate,  @EndDate,  @MovedOn,  @Status,  @Recurrence,  @Organizer,  @Creator,  @Sequence,  @Transperancy,  @Updated,  @Visibility  IF EXISTS(SELECT 1 FROM events  WHERE eventid = @EventId  )  UPDATE [dbo].[events]  SET [calendarid] = @CalendarId,  [eventid] = @EventId,  [recurringeventid] = @RecurringEventId,  [colorid] = @ColorId,  [attendees] = @Attendees,  [attendeesomitted] = @AttendeesOmitted,  [anyonecanaddself] = @AnyoneCanAddSelf,  [locked] = @Locked,  [title] = @Title,  [location] = @Location,  [description] = @Description,  [kind] = @Kind,  [startdate] = @StartDate,  [enddate] = @EndDate,  [movedon] = @MovedOn,  [status] = @Status,  [recurrence] = @Recurrence,  [organizer] = @Organizer,  [creator] = @Creator,  [sequence] = @Sequence,  [transperancy] = @Transperancy,  [updated] = @Updated,  [visibility] = @Visibility  WHERE eventid = @EventId  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This procedure stores all events sourced from Google Calendars for all the calendars in the calendar table, into the Events table. This data is syncronised with Google Calendar for all the shared calendars and will be available for display on a custom calendar. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertOtherEvents', NULL, NULL  GO |

Uses

[[dbo].[Calendars]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Calendars)

[[dbo].[Events]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Events)

|  |
| --- |
| [dbo].[InsertPropertyInvoiceNo] |

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @PropertyId Unique Property Id | int | 4 |
| @InvoiceNo Invoice ID returned from XERO integration | varchar(50) | 50 |

SQL Script

|  |
| --- |
| CREATE PROCEDURE [dbo].[InsertPropertyInvoiceNo]  @PropertyId int,  @InvoiceNo varchar(50)  AS  Declare @MailId int=(Select max(Row\_Id) from Mail )  INSERT INTO PropertyInvoices(PropertyId,MailId,InvoiceNo)  SELECT @PropertyId,@MailId,@InvoiceNo  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Invoice ID returned from XERO integration', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertPropertyInvoiceNo', 'PARAMETER', N'@InvoiceNo'  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Unique Property Id', 'SCHEMA', N'dbo', 'PROCEDURE', N'InsertPropertyInvoiceNo', 'PARAMETER', N'@PropertyId'  GO |

Uses

[[dbo].[Mail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Mail)

[[dbo].[PropertyInvoices]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/PropertyInvoices)

|  |
| --- |
| [dbo].[Processmail] |

MS\_Description

This procedure is used to read and parse the html file received by the Windows Service which is stored on the filesystem and create all the related fields in normalised database tables.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @X XML File | xml | max |

SQL Script

|  |
| --- |
| CREATE PROCEDURE [dbo].[Processmail] @X XML  AS    UPDATE Company SET ScrappedName= REPLACE(REPLACE(ScrappedName, CHAR(13),''), CHAR(10),'')  DECLARE @COMPANYNAME VARCHAR(500)='',@COMPANYID INT,@CreateInvoice INT  SELECT @COMPANYNAME= x.i.value('CompanyName[1]', 'varchar(MAX)')FROM @x.nodes('/Root')x(i)  SELECT @CreateInvoice=CreateInvoice FROM Company WHERE ScrappedName=REPLACE(REPLACE(@COMPANYNAME, CHAR(13),''), CHAR(10),'')  IF @CreateInvoice=0 RETURN  DECLARE @cnt INT=1  SELECT x.i.value('.', 'varchar(MAX)')val1,  ROW\_NUMBER()  OVER(ORDER BY (SELECT 1))rid  INTO #temp  FROM @x.nodes('/Root/OrderItemDetails/n')x(i)  CREATE TABLE #temp1  (  rid INT,  value VARCHAR(MAX),  cost1 VARCHAR(MAX),  cost2 VARCHAR(MAX),  comments VARCHAR(MAX)  )  WHILE @cnt < (SELECT COUNT(0) FROM #temp)  BEGIN  INSERT INTO #temp1  SELECT SUBSTRING(value, 1, 1)RID,  SUBSTRING(value, 3, Len(value)),  cost1,  cost2,  ''  FROM (SELECT \*  FROM dbo.Split('|', (SELECT TOP 1 val1  FROM #temp  WHERE rid = @cnt)))t  SELECT @cnt = @cnt + 1  END      UPDATE #temp1  SET comments = (SELECT x.i.value('PropertyDetails[1]', 'varchar(MAX)') + ' ' + x.i.value('PropertyComments[1]', 'varchar(MAX)')  FROM @x.nodes('/Root')x(i))  WHERE value = 'PROPERTY READY'    BEGIN TRAN  INSERT INTO mail  (  FileName,  sentdate,  subject)  SELECT  x.i.value('FileName[1]', 'varchar(MAX)'),  ltrim(rtrim(x.i.value('SentDate[1]', 'varchar(MAX)'))),  ltrim(rtrim(x.i.value('Subject[1]', 'varchar(MAX)')))    FROM @x.nodes('/Root')x(i)  DECLARE @MAILID INT=(SELECT MAX(Row\_ID) FROM Mail)  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  /\*  IF NOT EXISTS(SELECT 1 FROM Company WHERE ScrappedName=@COMPANYNAME)  BEGIN  SELECT @COMPANYNAME=ScrappedName FROM Company WHERE ScrappedName =@COMPANYNAME  --INSERT INTO Company(Name)  --SELECT @COMPANYNAME  END  \*/  SELECT @COMPANYID=ROW\_ID FROM Company WHERE ScrappedName=REPLACE(REPLACE(@COMPANYNAME, CHAR(13),''), CHAR(10),'')  INSERT INTO CompanyMail  (Company\_Id,  Mail\_Id)  SELECT @COMPANYID,(SELECT MAX(row\_id)  FROM mail)    IF (@@ERROR <> 0) GOTO ERR\_HANDLER  INSERT INTO property  (company\_id,  property\_id,  name  )  SELECT (SELECT @COMPANYID),  ltrim(rtrim(x.i.value('PropertyId[1]', 'varchar(MAX)'))),  ltrim(rtrim(x.i.value('PropertyName[1]', 'varchar(MAX)')))  FROM @x.nodes('/Root')x(i)    UPDATE Property SET Name= REPLACE(REPLACE(Name, CHAR(13),''), CHAR(10),'')  UPDATE Property SET Property\_Id= REPLACE(REPLACE(Property\_Id, CHAR(13),''), CHAR(10),'')    IF (@@ERROR <> 0) GOTO ERR\_HANDLER  DECLARE @OrdersOutput TABLE  (  ORDERID INT  );  ;WITH cte  AS ((SELECT x.i.value('.', 'varchar(MAX)')OrderId,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/OrderItemId/Id')x(i))),  cte1  AS ((SELECT x.i.value('.', 'varchar(MAX)')RequiredDate,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/RequiredDate')x(i))),  cte2  AS ((SELECT x.i.value('.', 'varchar(MAX)')Descr,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/Description')x(i)))    INSERT INTO orders  (ordertype\_id,  property\_id,  Mail\_Id,  orderid,  requireddate,  [Description])  OUTPUT INSERTED.Row\_Id INTO @OrdersOutput  SELECT (SELECT 1),  (SELECT MAX(row\_id)  FROM property),  @MAILID,  cte.orderid,  convert(datetime,cte1.requireddate,103),  ltrim(rtrim(cte2.Descr))  FROM cte  INNER JOIN cte1  ON cte.rid = cte1.rid  LEFT JOIN cte2  ON cte.rid = cte2.rid      IF (@@ERROR <> 0) GOTO ERR\_HANDLER  --UPDATE O SET O.Product\_Id=P.Row\_Id FROM Orders O INNER JOIN Products P ON P.ProductDescription=O.Description  --INSERT INTO PRODUCTS(ProductId)  --SELECT ROW\_ID FROM Orders WHERE [Description]  --IN  --(  --SELECT x.i.value('.', 'varchar(MAX)')Descr FROM @x.nodes('/Root/Description')x(i)  --)  --IF (@@ERROR <> 0) GOTO ERR\_HANDLER  INSERT INTO OrderEvents(Order\_Id)  SELECT \* FROM @OrdersOutput  INSERT INTO CompanyOrders(CompanyId, OrderId)  SELECT @COMPANYID,ORDERID FROM @OrdersOutput  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  ;WITH cte  AS ((SELECT x.i.value('.', 'varchar(MAX)')SalesContact,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/SalesContact')x(i))),  cte1  AS ((SELECT x.i.value('.', 'varchar(MAX)')SalesContactNumber,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1)) RID  FROM @x.nodes('/Root/SalesContactNumber')x(i)))    INSERT INTO contacts  (CompanyId,name,  value,  ContactType)  SELECT  @COMPANYID,  ltrim(rtrim(cte.salescontact)),  ltrim(rtrim(cte1.salescontactnumber)),  'Sales'  FROM cte  INNER JOIN cte1  ON cte.rid = cte1.rid WHERE ltrim(rtrim(cte.salescontact)) NOT IN(SELECT Name FROM Contacts WHERE ContactType='Sales' ) ;  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  ;WITH cte  AS ((SELECT x.i.value('.', 'varchar(MAX)')AdminContact,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/AdminContact')x(i))),  cte1  AS ((SELECT x.i.value('.', 'varchar(MAX)')AdminContactEmail,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1)) RID  FROM @x.nodes('/Root/AdminContactEmail')x(i)))    INSERT INTO contacts  (CompanyId,name,  value,  ContactType)  SELECT  @COMPANYID,ltrim(rtrim(cte.AdminContact)),  ltrim(rtrim(cte1.AdminContactEmail)),  'Admin'  FROM cte  INNER JOIN cte1  ON cte.rid = cte1.rid WHERE ltrim(rtrim(cte.AdminContact)) NOT IN(SELECT Name FROM Contacts WHERE ContactType='Admin' ) ;    IF (@@ERROR <> 0) GOTO ERR\_HANDLER  UPDATE #temp1 SET value=REPLACE(REPLACE(value, CHAR(13),''), CHAR(10),'')  UPDATE Orders SET OrderId=REPLACE(OrderId, ' ','')  -- (SELECT TOP 1 isnull(Row\_Id,-1) FROM xeroitems X INNER JOIN #temp1 ON T1.value=X.Name),  DECLARE @CompanyIds varchar(max)=''  SELECT @CompanyIds=@CompanyIds + ',' + CompanyId FROM Products P INNER JOIN #temp1 F ON P.ProductDescription=F.value  SELECT COL CompanyId INTO #CompanyProducts FROM dbo.QFN\_SPLIT(@CompanyIds,',')  DELETE FROM #CompanyProducts WHERE (CompanyId !=@CompanyId OR CompanyId=0)  DECLARE @CompanyIdExist BIT=0  IF EXISTS(SELECT CompanyId FROM #CompanyProducts WHERE CompanyId=@CompanyId)  SELECT @CompanyIdExist=1  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  SELECT O.row\_id,  NULL Xero\_Id,  value,  cost1,  cost2,  comments  INTO #final  FROM #temp1 T1  INNER JOIN (SELECT x.i.value('.', 'varchar(MAX)')OrderId,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/OrderItemId/Id')x(i))T  ON T.rid = T1.rid  INNER JOIN orders O  ON O.OrderId = T.orderid    UPDATE F SET F.Xero\_Id=P.Row\_Id FROM #final F INNER JOIN Products P ON P.ProductDescription=F.value  INNER JOIN Orders O ON O.Row\_Id=F.Row\_Id WHERE P.CompanyId=case when @CompanyIdExist=1 then @COMPANYID else P.CompanyId end  UPDATE #final SET Xero\_Id=(SELECT TOP 1 Row\_Id FROM Products WHERE XeroCode='RESTRAV')WHERE value like 'Travel%'  UPDATE #final SET Xero\_Id=(SELECT TOP 1 Row\_Id FROM Products WHERE XeroCode='MWARCH')WHERE value like 'Archive Search and Retrieval%'  UPDATE #final SET Xero\_Id=(SELECT TOP 1 Row\_Id FROM Products WHERE XeroCode='RESSC')WHERE value like 'Additional Image%'  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  INSERT INTO orderitems  (order\_id,  xero\_id,  name,  clientprice,  costprice,  comments)  SELECT Row\_Id,Xero\_Id,value,cost1,cost2,comments FROM #final  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  ;WITH cte  AS ((SELECT x.i.value('.', 'varchar(MAX)')OrderId,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/OrderItemId/Id')x(i)))  ,cte1  AS ((SELECT x.i.value('.', 'varchar(MAX)')SalesContact,  ROW\_NUMBER()OVER(ORDER BY (SELECT 1))RID  FROM @x.nodes('/Root/SalesContact')x(i)))  INSERT INTO OrderContacts(OrderId,ContactId)  SELECT O.Row\_Id,C.Row\_Id FROM cte CROSS APPLY(SELECT TOP 1 SalesContact FROM cte1 WHERE cte.RID>=cte1.RID)T  INNER JOIN Orders O ON O.OrderId=cte.OrderId  INNER JOIN Contacts C ON ltrim(rtrim(C.Name))=ltrim(rtrim(T.SalesContact))  SELECT DISTINCT O.OrderId,PR.Name Property,C.XeroName,C.CreateInvoice,O.Property\_Id FROM #final F INNER JOIN Products P  ON P.ProductDescription=F.value  INNER JOIN Orders O ON O.Row\_Id=F.Row\_Id  INNER JOIN CompanyOrders CO ON CO.OrderId=O.Row\_Id  INNER JOIN Company C ON C.Row\_Id=CO.CompanyId  INNER JOIN Property PR ON PR.Row\_Id=O.Property\_Id  WHERE F.value NOT LIKE 'Total%'AND C.Row\_Id=@COMPANYID  IF @CompanyIdExist=1  SELECT P.\*,F.\* ,O.OrderId,P.ProductDescription,O.Property\_Id FROM #final F INNER JOIN Products P ON P.ProductDescription=F.value  INNER JOIN Orders O ON O.Row\_Id=F.Row\_Id INNER JOIN Property PR ON PR.Row\_Id=O.Property\_Id  WHERE F.value NOT LIKE 'Total%' AND p.CompanyId!=0  ELSE  SELECT P.\*,F.\* ,O.OrderId,P.ProductDescription,O.Property\_Id FROM #final F INNER JOIN Products P ON P.ProductDescription=F.value  INNER JOIN Orders O ON O.Row\_Id=F.Row\_Id INNER JOIN Property PR ON PR.Row\_Id=O.Property\_Id  WHERE F.value NOT LIKE 'Total%' AND p.CompanyId=0  IF (@@ERROR <> 0) GOTO ERR\_HANDLER  DROP TABLE #temp  DROP TABLE #temp1  COMMIT TRAN  RETURN 0  ERR\_HANDLER:  ROLLBACK TRAN  RETURN 1  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This procedure is used to read and parse the html file received by the Windows Service which is stored on the filesystem and create all the related fields in normalised database tables.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Processmail', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'XML File', 'SCHEMA', N'dbo', 'PROCEDURE', N'Processmail', 'PARAMETER', N'@X'  GO |

Uses

[[dbo].[Company]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company)

[[dbo].[CompanyMail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CompanyMail)

[[dbo].[CompanyOrders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/CompanyOrders)

[[dbo].[Contacts]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Contacts)

[[dbo].[Mail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Mail)

[[dbo].[OrderContacts]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderContacts)

[[dbo].[OrderEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderEvents)

[[dbo].[OrderItems]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderItems)

[[dbo].[Orders]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Orders)

[[dbo].[Products]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Products)

[[dbo].[Property]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Property)

[[dbo].[QFN\_SPLIT]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Functions/Table-valued_Functions/QFN_SPLIT)

[[dbo].[split]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Functions/Table-valued_Functions/split)

|  |
| --- |
| [dbo].[SyncCompany] |

MS\_Description

This procedure receives XML from the XERO Contact API method and stores these values in the Company table.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @X XML | xml | max |

SQL Script

|  |
| --- |
| CREATE PROC [dbo].[SyncCompany]  @X xml  AS  SELECT x.i.value('@Name', 'varchar(MAX)')Name    INTO #Source  FROM @x.nodes('/ContactList/Items')x(i)  INSERT INTO Company(XeroName)  SELECT Name From #Source WHERE NOT EXISTS(SELECT 1 FROM Company WHERE XeroName=Name)  Delete from Company where Not exists(Select 1 from #Source where Name=XeroName)  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This procedure receives XML from the XERO Contact API method and stores these values in the Company table.', 'SCHEMA', N'dbo', 'PROCEDURE', N'SyncCompany', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'XML', 'SCHEMA', N'dbo', 'PROCEDURE', N'SyncCompany', 'PARAMETER', N'@X'  GO |

Uses

[[dbo].[Company]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Company)

|  |
| --- |
| [dbo].[SyncProducts] |

MS\_Description

This procedure receives XML from the XERO Items API method and stores these values in the Product table.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @X XML | xml | max |

SQL Script

|  |
| --- |
| CREATE PROC [dbo].[SyncProducts]  @X xml  AS  UPDATE Products set XeroCode= REPLACE(REPLACE(XeroCode, CHAR(13),''), CHAR(10),'')  SELECT x.i.value('@Code', 'varchar(MAX)')XeroCode,  x.i.value('.', 'varchar(MAX)')XeroItemDescription,  x.i.value('@SalesUnitPrice', 'varchar(MAX)')SalesUnitPrice,  x.i.value('@SalesAccountCode', 'varchar(MAX)')SalesAccountCode,  x.i.value('@SalesTaxType', 'varchar(MAX)')SalesTaxType    INTO #Source  FROM @x.nodes('/XeroItems/Items')x(i)  UPDATE #Source SET SalesTaxType=CASE SalesTaxType WHEN 'OUTPUT' THEN 'GST on Income'  WHEN 'EXEMPTOUTPUT' THEN 'GST Free Income'  END  ;MERGE Products AS X  USING #Source AS S  ON (X.XeroCode = S.XeroCode)  WHEN NOT MATCHED  THEN INSERT(XeroCode, XeroItemDescription,SalesUnitPrice,SalesAccountCode,SalesTaxType)  VALUES(S.XeroCode, S.XeroItemDescription,S.SalesUnitPrice,S.SalesAccountCode,S.SalesTaxType)  WHEN MATCHED  THEN UPDATE SET X.XeroItemDescription = S.XeroItemDescription  ,X.SalesUnitPrice=S.SalesUnitPrice,X.SalesAccountCode=S.SalesAccountCode,X.SalesTaxType=S.SalesTaxType;  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This procedure receives XML from the XERO Items API method and stores these values in the Product table.', 'SCHEMA', N'dbo', 'PROCEDURE', N'SyncProducts', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'XML', 'SCHEMA', N'dbo', 'PROCEDURE', N'SyncProducts', 'PARAMETER', N'@X'  GO |

Uses

[[dbo].[Products]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/Products)

|  |
| --- |
| [dbo].[UpdateOrderEvent] |

MS\_Description

This procedure inserts into OrderEvents when a Google Calendar event is created against an order.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @OrderId Order Id | int | 4 |

SQL Script

|  |
| --- |
| CREATE PROC [dbo].[UpdateOrderEvent]  @OrderId int  as  update OrderEvents set EventCreated=1 where Order\_Id=@OrderId  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This procedure inserts into OrderEvents when a Google Calendar event is created against an order. ', 'SCHEMA', N'dbo', 'PROCEDURE', N'UpdateOrderEvent', NULL, NULL  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'Order Id ', 'SCHEMA', N'dbo', 'PROCEDURE', N'UpdateOrderEvent', 'PARAMETER', N'@OrderId'  GO |

Uses

[[dbo].[OrderEvents]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Tables/OrderEvents)

|  |
| --- |
| Table-valued Functions |

Objects

|  |
| --- |
| Name |
| [dbo.QFN\_SPLIT](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Functions/Table-valued_Functions/QFN_SPLIT) This function spilts a commar separated result set. |
| [dbo.split](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Functions/Table-valued_Functions/split) |

|  |
| --- |
| [dbo].[QFN\_SPLIT] |

MS\_Description

This function spilts a commar separated result set.

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @INPUTSTR | varchar(max) | max |
| @SEPRATOR | varchar(10) | 10 |

SQL Script

|  |
| --- |
| CREATE FUNCTION [dbo].[QFN\_SPLIT]  (  @INPUTSTR VARCHAR(MAX),  @SEPRATOR VARCHAR(10)  )  RETURNS @TEMP TABLE (COL VARCHAR(MAX),ID INT)  AS    BEGIN  DECLARE @I INT ,@ID INT  SET @ID = 1  SET @INPUTSTR = RTRIM(LTRIM(@INPUTSTR))  SET @I = CHARINDEX(@SEPRATOR, @INPUTSTR)    WHILE @I >= 1  BEGIN  INSERT @TEMP VALUES(LEFT(@INPUTSTR, @I - 1),@ID)  SET @INPUTSTR = SUBSTRING(@INPUTSTR, @I + 1, LEN(@INPUTSTR) - @I)  SET @I = CHARINDEX(@SEPRATOR, @INPUTSTR)  SET @ID = @ID + 1  END  IF @INPUTSTR <> '\'  INSERT @TEMP VALUES(@INPUTSTR,@ID)    RETURN  END  GO  EXEC sp\_addextendedproperty N'MS\_Description', N'This function spilts a commar separated result set. ', 'SCHEMA', N'dbo', 'FUNCTION', N'QFN\_SPLIT', NULL, NULL  GO |

Used By

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| [dbo].[split] |

Properties

|  |  |
| --- | --- |
| Property | Value |
| ANSI Nulls On | True |
| Quoted Identifier On | True |

Parameters

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Max Length (Bytes) |
| @del | char | 1 |
| @str | varchar(max) | max |

SQL Script

|  |
| --- |
| CREATE FUNCTION [dbo].[split]  (  @del char(1),  @str varchar(max)  )  RETURNS @tResult TABLE  (  VALUE VARCHAR(MAX),  COST1 VARCHAR(MAX),  COST2 VARCHAR(MAX)  )  AS  BEGIN  declare @xml xml  set @xml = N'<root><r>' + replace(@str,@del,'</r><r>') + '</r></root>'  insert into @tResult(VALUE,COST1,COST2)  select  r.value('r[1]','varchar(max)') ,  r.value('r[2]','varchar(max)'),  r.value('r[3]','varchar(max)')  from @xml.nodes('/root') as records(r)  RETURN  END  GO |

Used By

[[dbo].[Processmail]](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Programmability/Stored_Procedures/Processmail)

|  |
| --- |
| Database Roles |

Objects

|  |
| --- |
| Name |
| [db\_accessadmin](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_accessadmin) |
| [db\_backupoperator](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_backupoperator) |
| [db\_datareader](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_datareader) |
| [db\_datawriter](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_datawriter) |
| [db\_ddladmin](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_ddladmin) |
| [db\_denydatareader](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_denydatareader) |
| [db\_denydatawriter](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_denydatawriter) |
| [db\_owner](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_owner) |
| [db\_securityadmin](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/db_securityadmin) |
| [public](#AMAZONA-HLGH6D4/User_databases/CampaignTrack_Dev/Security/Roles/Database_Roles/public) |

|  |
| --- |
| db\_accessadmin |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_backupoperator |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_datareader |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_datawriter |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_ddladmin |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_denydatareader |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_denydatawriter |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_owner |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| db\_securityadmin |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |

|  |
| --- |
| public |

Properties

|  |  |
| --- | --- |
| Property | Value |
| Owner | dbo |