|  |
| --- |
| **Documentation()**  **OnRun()**  **Sync(SyncType : 'Sales,Item,Vehicle';"Start Date" : Date;"End Date" : Date)**  **DealerInfo.RESET;**  **DealerInfo.SETRANGE(Active, TRUE);**  **DealerInfo.SETFILTER("company Database", '<>%1', '');**  **//DealerInfo.SETFILTER("company Database",'%1', 'DEVTRADERS');**  **IF DealerInfo.FINDSET THEN REPEAT**  **CASE SyncType OF**  **SyncType::Item:**  **BEGIN**  **Item.RESET;**  **Item.SETRANGE("No.",'AUDI R8');**  **IF Item.FINDSET THEN REPEAT**  **ConnectSQL(DealerInfo);**  **ReadItem(DealerInfo, Item);**  **IF SQLDataReader.HasRows THEN BEGIN**  **DisconnectSQL;**  **UpdateItem(DealerInfo, Item);**  **END ELSE BEGIN**  **DisconnectSQL;**  **InsertItem(DealerInfo, Item);**  **END;**  **DisconnectSQL;**  **UNTIL Item.NEXT = 0;**  **END;**  **SyncType::Sales:**  **BEGIN**  **Item.RESET;**  **IF Item.FINDSET THEN REPEAT**  **SalesPrice.RESET;**  **SalesPrice.SETRANGE("Starting Date", "Start Date","End Date");**  **IF SalesPrice.FINDSET THEN**  **REPEAT**  **ConnectSQL(DealerInfo);**  **ReadSalesPrice(DealerInfo, SalesPrice);**  **IF SQLDataReader.HasRows THEN BEGIN**  **DisconnectSQL;**  **UpdateSalesPrice(DealerInfo, SalesPrice);**  **END ELSE BEGIN**  **DisconnectSQL;**  **InsertSalesPrice(DealerInfo, SalesPrice);**  **END;**  **DisconnectSQL;**  **UNTIL SalesPrice.NEXT =0;**  **UNTIL Item.NEXT = 0;**  **END;**  **SyncType::Vehicle:**  **BEGIN**  **SerialNoInfo.RESET;**  **SerialNoInfo.SETRANGE("Serial No.",'PRABESH1234567899');**  **IF SerialNoInfo.FINDSET THEN REPEAT**  **ConnectSQL(DealerInfo);**  **ReadVehicle(DealerInfo, SerialNoInfo);**  **IF SQLDataReader.HasRows THEN BEGIN**  **DisconnectSQL;**  **UpdateVehicle(DealerInfo, SerialNoInfo);**  **END ELSE BEGIN**  **DisconnectSQL;**  **InsertVehicle(DealerInfo, SerialNoInfo);**  **END;**  **DisconnectSQL;**  **UNTIL SerialNoInfo.NEXT = 0;**  **END**  **END;**  **UNTIL DealerInfo.NEXT = 0;**  **MESSAGE('Item sales price sync done.');**  **LOCAL UpdateSalesPrice(DealerInfo : Record "Dealer Information";SalesPrice : Record "Sales Price")**  **CLEAR(commandtext);**  **ConnectSQL(DealerInfo);**  **CASE SalesPrice."Sales Type" OF**  **SalesPrice."Sales Type"::Customer:**  **SalesTypeVar := 0;**  **SalesPrice."Sales Type"::"Customer Price Group":**  **SalesTypeVar := 1;**  **SalesPrice."Sales Type"::"All Customers":**  **SalesTypeVar := 2;**  **SalesPrice."Sales Type"::Campaign:**  **SalesTypeVar := 3;**  **END;**  **IF SalesPrice."Ending Date" <> 0D THEN**  **EndDate := FORMAT(SalesPrice."Ending Date")**  **ELSE**  **EndDate := FORMAT('1753-01-01 00:00:00.000');**  **commandtext := UpdateTxt + '[' + DealerInfo."Company Name" + '$Sales Price' + ']' + SpaceTxt + SetTxt +**  **'[Unit Price] = @UnitPrice,' +**  **'[Item No\_] = @itemNo,' +**  **'[Sales Type] = @SalesType,' +**  **'[Sales Code] = @SalesCode,' +**  **'[Starting Date] = @StartingDate,' +**  **'[Ending Date] = @EndingDate,' +**  **'[Variant Code] = @VariantCode,' +**  **'[Minimum Quantity] = @MinimumQuantity,' +**  **'[Unit of Measure Code] = @UnitOfMeasureCode,' +**  **'[Currency Code] = @CurrencyCode' +**  **SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+**  **SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+**  **SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;**    **{ commandtext := UpdateTxt + '[' + 'DEVMOTORSPVTLTD$Sales Price' + ']' + SpaceTxt + SetTxt +**  **'[Unit Price] = @UnitPrice,' +**  **'[Item No\_] = @itemNo,' +**  **'[Sales Type] = @SalesType,' +**  **'[Sales Code] = @SalesCode,' +**  **'[Starting Date] = @StartingDate,' +**  **'[Ending Date] = @EndingDate,' +**  **'[Variant Code] = @VariantCode,' +**  **'[Minimum Quantity] = @MinimumQuantity,' +**  **'[Unit of Measure Code] = @UnitOfMeasureCode,' +**  **'[Currency Code] = @CurrencyCode' +**  **SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+**  **SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+**  **SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;**  **}**    **SetupSQLCommand;**  **SQLCommand.Parameters.AddWithValue('@UnitPrice',SalesPrice."Unit Price");**  **SQLCommand.Parameters.AddWithValue('@itemNo',FORMAT(SalesPrice."Item No."));**  **SQLCommand.Parameters.AddWithValue('@SalesType',SalesTypeVar);**  **SQLCommand.Parameters.AddWithValue('@SalesCode',FORMAT(SalesPrice."Sales Code"));**  **SQLCommand.Parameters.AddWithValue('@StartingDate',FORMAT(SalesPrice."Starting Date"));**  **SQLCommand.Parameters.AddWithValue('@EndingDate',EndDate);**  **SQLCommand.Parameters.AddWithValue('@VariantCode',FORMAT(SalesPrice."Variant Code"));**  **SQLCommand.Parameters.AddWithValue('@MinimumQuantity',SalesPrice."Minimum Quantity");**  **SQLCommand.Parameters.AddWithValue('@UnitOfMeasureCode',FORMAT(SalesPrice."Unit of Measure Code"));**  **SQLCommand.Parameters.AddWithValue('@CurrencyCode',FORMAT(SalesPrice."Currency Code"));**  **SQLCommand.ExecuteNonQuery;**  **LOCAL InsertSalesPrice(DBINfo : Record "Dealer Information";SalesPrice : Record "Sales Price")**  **CLEAR(commandtext);**  **ConnectSQL(DBINfo);**  **CASE SalesPrice."Sales Type" OF**  **SalesPrice."Sales Type"::Customer:**  **SalesTypeVar := 0;**  **SalesPrice."Sales Type"::"Customer Price Group":**  **SalesTypeVar := 1;**  **SalesPrice."Sales Type"::"All Customers":**  **SalesTypeVar := 2;**  **SalesPrice."Sales Type"::Campaign:**  **SalesTypeVar := 3;**  **END;**  **IF SalesPrice."Ending Date" <> 0D THEN**  **EndDate := FORMAT(SalesPrice."Ending Date")**  **ELSE**  **EndDate := FORMAT('1753-01-01 00:00:00.000');**  **commandtext := InsertTxt + '['+ DBINfo."Company Name" +'$Sales Price'+ ']' + SpaceTxt + '([Unit Price], [Item No\_],[Sales Type],[Sales Code], [Starting Date], '+**  **'[Ending Date],[Variant Code], [Minimum Quantity], [Unit of Measure Code], [Currency Code],[Price Includes VAT],[Allow Invoice Disc\_],' +**  **'[VAT Bus\_ Posting Gr\_ (Price)],[Allow Line Disc\_])'+ ValuesTxt + '(@UnitPrice, @itemNo, @SalesType, @SalesCode,' +**  **'@StartingDate, @EndingDate, @VariantCode, @MinimumQuantity, @UnitOfMeasureCode, @CurrencyCode, @PriceInclVat, @AllowInvDisc, @VatBusPos, @AllowLineDisc)';**  **{commandtext := InsertTxt + '[' + 'DEVMOTORSPVTLTD$Sales Price' +']' + SpaceTxt + '([Unit Price], [Item No\_],[Sales Type],[Sales Code], [Starting Date], '+**  **'[Ending Date],[Variant Code], [Minimum Quantity], [Unit of Measure Code], [Currency Code],[Price Includes VAT],[Allow Invoice Disc\_],' +**  **'[VAT Bus\_ Posting Gr\_ (Price)],[Allow Line Disc\_])'+ ValuesTxt + '(@UnitPrice, @itemNo, @SalesType, @SalesCode,' +**  **'@StartingDate, @EndingDate, @VariantCode, @MinimumQuantity, @UnitOfMeasureCode, @CurrencyCode, @PriceInclVat, @AllowInvDisc, @VatBusPos, @AllowLineDisc)';**    **}**    **SetupSQLCommand;**  **SQLCommand.Parameters.AddWithValue('@UnitPrice',SalesPrice."Unit Price");**  **SQLCommand.Parameters.AddWithValue('@itemNo',FORMAT(SalesPrice."Item No."));**  **SQLCommand.Parameters.AddWithValue('@SalesType',SalesTypeVar);**  **SQLCommand.Parameters.AddWithValue('@SalesCode',FORMAT(SalesPrice."Sales Code"));**  **SQLCommand.Parameters.AddWithValue('@StartingDate',FORMAT(SalesPrice."Starting Date"));**  **SQLCommand.Parameters.AddWithValue('@EndingDate',EndDate);**  **SQLCommand.Parameters.AddWithValue('@VariantCode',FORMAT(SalesPrice."Variant Code"));**  **SQLCommand.Parameters.AddWithValue('@MinimumQuantity',SalesPrice."Minimum Quantity");**  **SQLCommand.Parameters.AddWithValue('@UnitOfMeasureCode',FORMAT(SalesPrice."Unit of Measure Code"));**  **SQLCommand.Parameters.AddWithValue('@CurrencyCode',FORMAT(SalesPrice."Currency Code"));**  **SQLCommand.Parameters.AddWithValue('@PriceInclVat', SalesPrice."Price Includes VAT");**  **SQLCommand.Parameters.AddWithValue('@AllowInvDisc', SalesPrice."Allow Invoice Disc.");**  **SQLCommand.Parameters.AddWithValue('@AllowLineDisc', SalesPrice."Allow Line Disc.");**  **SQLCommand.Parameters.AddWithValue('@VatBusPos', FORMAT(SalesPrice."VAT Bus. Posting Gr. (Price)"));**  **SQLCommand.ExecuteNonQuery;**  **LOCAL ReadSalesPrice(DealerInfo : Record "Dealer Information";SalesPrice : Record "Sales Price")**  **ClearSQLCommand;**  **CASE SalesPrice."Sales Type" OF**  **SalesPrice."Sales Type"::Customer:**  **SalesTypeVar := 0;**  **SalesPrice."Sales Type"::"Customer Price Group":**  **SalesTypeVar := 1;**  **SalesPrice."Sales Type"::"All Customers":**  **SalesTypeVar := 2;**  **SalesPrice."Sales Type"::Campaign:**  **SalesTypeVar := 3;**  **END;**  **commandtext := ReadCommandTxt + '[' + DealerInfo."Company Name" + '$Sales Price' + ']' +**  **SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+**  **SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+**  **SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;**  **{commandtext := ReadCommandTxt + '[' + 'DEVMOTORSPVTLTD$Sales Price' + ']' +**  **SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+**  **SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+**  **SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;**  **}**  **SetupSQLCommand;**  **SQLDataReader := SQLCommand.ExecuteReader;**  **SQLDataReader.Read;**  **LOCAL ConnectSQL(DealerInfo : Record "Dealer Information")**  **SQLConnectionMgt.SetupSQLConnection(SQLConnection, DealerInfo."company Database");**  **LOCAL DisconnectSQL()**  **SQLConnectionMgt.CloseSQLConnection(SQLConnection);**  **LOCAL ClearSQLCommand()**  **CLEAR(commandtext);**  **LOCAL SetupSQLCommand()**  **SQLConnectionMgt.SetupSQLCommand(SQLConnection,SQLCommand,commandtext,SQLCommandType::Text);**  **LOCAL HasSalesPrice() : Boolean**  **LOCAL UpdateItem(DealerInfo : Record "Dealer Information";Item : Record Item)**  **CLEAR(commandtext);**  **ConnectSQL(DealerInfo);**  **CASE Item."Item Type" OF**  **Item."Item Type"::" ":**  **ItemTypeVar := 0;**  **Item."Item Type"::Item:**  **ItemTypeVar := 1;**  **Item."Item Type"::"Model Version":**  **ItemTypeVar := 2;**  **END;**  **CASE Item.Type OF**  **Item.Type::Inventory:**  **TypeVar :=0;**  **Item.Type::Service:**  **TypeVar:= 1;**  **END;**  **IF Item."Next Counting Start Date" <> 0D THEN**  **NextCountingStartDate := FORMAT(Item."Next Counting Start Date")**  **ELSE**  **NextCountingStartDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Date Filter" <> 0D THEN**  **DateFilter := FORMAT(Item."Date Filter")**  **ELSE**  **DateFilter := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Unit Cost Calc. Date" <> 0D THEN**  **LastUnitCostCalcDate := FORMAT(Item."Last Unit Cost Calc. Date")**  **ELSE**  **LastUnitCostCalcDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Counting Period Update" <> 0D THEN**  **LastCountingPeriodUpdate := FORMAT(Item."Last Counting Period Update")**  **ELSE**  **LastCountingPeriodUpdate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Scratch Start Date" <> 0D THEN**  **ScratchStart := FORMAT(Item."Scratch Start Date")**  **ELSE**  **ScratchStart := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Scratch End Date" <> 0D THEN**  **ScratchEnd := FORMAT(Item."Scratch Start Date")**  **ELSE**  **ScratchEnd := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Next Counting End Date" <> 0D THEN**  **NextCountingEndDate := FORMAT(Item."Next Counting Start Date")**  **ELSE**  **NextCountingEndDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Phys. Invt. Date" <> 0D THEN**  **LastPhyDate := FORMAT(Item."Last Phys. Invt. Date")**  **ELSE**  **LastPhyDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Time Modified" <> 0T THEN**  **LastTimeMod := FORMAT(Item."Last Time Modified")**  **ELSE**  **LastTimeMod := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Date Modified" <> 0D THEN**  **LastDateModified := FORMAT(Item."Last Date Modified")**  **ELSE**  **LastDateModified := FORMAT('1753-01-01 00:00:00.000');**  **commandtext := UpdateTxt + '[' + DealerInfo."Company Name" + '$Item' + ']' + SpaceTxt + SetTxt +**  **' [No\_] = @No,' +**  **' [No\_ 2] = @No2,' +**  **' [Description] = @Description,' +**  **' [Search Description] = @SearchDescription,' +**  **' [Description 2] = @Description2,' +**  **' [Base Unit of Measure] = @BaseUnitofMeasure,' +**  **' [Price Unit Conversion] = @PriceUnitConversion,' +**  **' [Type] = @Type,' +**  **' [Inventory Posting Group] = @InventoryPostingGroup,' +**  **' [Shelf No\_] = @ShelfNo,' +**  **' [Item Disc\_ Group] = @ItemDiscGroup,' +**  **' [Allow Invoice Disc\_] = @AllowInvoiceDisc,' +**  **' [Statistics Group] = @StatisticsGroup,' +**  **' [Commission Group] = @CommissionGroup,' +**  **' [Unit Price] = @UnitPrice,' +**  **' [Price\_Profit Calculation] = @PriceProfitCalculation,' +**  **' [Profit \_] = @Profit,' +**  **' [Costing Method] = @CostingMethod,' +**  **' [Unit Cost] = @UnitCost,' +**  **' [Standard Cost] = @StandardCost,' +**  **' [Last Direct Cost] = @LastDirectCost,' +**  **' [Indirect Cost \_] = @IndirectCost,' +**  **' [Cost is Adjusted] = @CostisAdjusted,' +**  **' [Allow Online Adjustment] = @AllowOnlineAdjustment,' +**  **' [Vendor No\_] = @VendorNo,' +**  **' [Vendor Item No\_] = @VendorItemNo,' +**  **' [Lead Time Calculation] = @LeadTimeCalculation,' +**  **' [Reorder Point] = @ReorderPoint,' +**  **' [Maximum Inventory] = @MaximumInventory,' +**  **' [Reorder Quantity] = @ReorderQuantity,' +**  **' [Alternative Item No\_] = @AlternativeItemNo,' +**  **' [Unit List Price] = @UnitListPrice,' +**  **' [Duty Due \_] = @DutyDue,' +**  **' [Duty Code] = @DutyCode,' +**  **' [Gross Weight] = @GrossWeight,' +**  **' [Net Weight] = @NetWeight,' +**  **' [Units per Parcel] = @UnitsperParcel,' +**  **' [Unit Volume] = @UnitVolume,' +**  **' [Durability] = @Durability,' +**  **' [Freight Type] = @FreightType,' +**  **' [Tariff No\_] = @TariffNo,' +**  **' [Duty Unit Conversion] = @DutyUnitConversion,' +**  **' [Country\_Region Purchased Code] = @CountryRegionPurchasedCode,' +**  **' [Budget Quantity] = @BudgetQuantity,' +**  **' [Budgeted Amount] = @BudgetedAmount,' +**  **' [Budget Profit] = @BudgetProfit,' +**  **' [Blocked] = @Blocked,' +**  **' [Last Date Modified] = @LastDateModified,' +**  **' [Last Time Modified] = @LastTimeModified,' +**  **' [Price Includes VAT] = @PriceIncludesVAT,' +**  **' [VAT Bus\_ Posting Gr\_ (Price)] = @VATBusPostingGrPrice,' +**  **' [Gen\_ Prod\_ Posting Group] = @GenProdPostingGroup,' +**  **' [Picture] = @Picture,' +**  **' [Country\_Region of Origin Code] = @CountryRegionofOriginCode,' +**  **' [Automatic Ext\_ Texts] = @AutomaticExtTexts,' +**  **' [No\_ Series] = @NoSeries,' +**  **' [Tax Group Code] = @TaxGroupCode,' +**  **' [VAT Prod\_ Posting Group] = @VATProdPostingGroup,' +**  **' [Reserve] = @Reserve,' +**  **' [Global Dimension 1 Code] = @GlobalDimension1Code,' +**  **' [Global Dimension 2 Code] = @GlobalDimension2Code,' +**  **' [Stockout Warning] = @StockoutWarning,' +**  **' [Prevent Negative Inventory] = @PreventNegativeInventory,' +**  **' [Application Wksh\_ User ID] = @ApplicationWkshUserID,' +**  **' [Assembly Policy] = @AssemblyPolicy,' +**  **' [GTIN] = @GTIN,' +**  **' [Default Deferral Template Code] = @DefaultDeferralTemplateCode,' +**  **' [Low-Level Code] = @LowLevelCode,' +**  **' [Lot Size] = @LotSize,' +**  **' [Serial Nos\_] = @SerialNos,' +**  **' [Last Unit Cost Calc\_ Date] = @LastUnitCostCalcDate,' +**  **' [Rolled-up Material Cost] = @RolledupMaterialCost,' +**  **' [Rolled-up Capacity Cost] = @RolledupCapacityCost,' +**  **' [Scrap \_] = @Scrap,' +**  **' [Inventory Value Zero] = @InventoryValueZero,' +**  **' [Discrete Order Quantity] = @DiscreteOrderQuantity,' +**  **' [Minimum Order Quantity] = @MinimumOrderQuantity,' +**  **' [Maximum Order Quantity] = @MaximumOrderQuantity,' +**  **' [Safety Stock Quantity] = @SafetyStockQuantity,' +**  **' [Order Multiple] = @OrderMultiple,' +**  **' [Safety Lead Time] = @SafetyLeadTime,' +**  **' [Flushing Method] = @FlushingMethod,' +**  **' [Replenishment System] = @ReplenishmentSystem,' +**  **' [Rounding Precision] = @RoundingPrecision,' +**  **' [Sales Unit of Measure] = @SalesUnitofMeasure,' +**  **' [Purch\_ Unit of Measure] = @PurchUnitofMeasure,' +**  **' [Time Bucket] = @TimeBucket,' +**  **' [Reordering Policy] = @ReorderingPolicy,' +**  **' [Include Inventory] = @IncludeInventory,' +**  **' [Manufacturing Policy] = @ManufacturingPolicy,' +**  **' [Rescheduling Period] = @ReschedulingPeriod,' +**  **' [Lot Accumulation Period] = @LotAccumulationPeriod,' +**  **' [Dampener Period] = @DampenerPeriod,' +**  **' [Dampener Quantity] = @DampenerQuantity,' +**  **' [Overflow Level] = @OverflowLevel,' +**  **' [Manufacturer Code] = @ManufacturerCode,' +**  **' [Item Category Code] = @ItemCategoryCode,' +**  **' [Created From Nonstock Item] = @CreatedFromNonstockItem,' +**  **' [Product Group Code] = @ProductGroupCode,' +**  **' [Service Item Group] = @ServiceItemGroup,' +**  **' [Item Tracking Code] = @ItemTrackingCode,' +**  **' [Lot Nos\_] = @LotNos,' +**  **' [Expiration Calculation] = @ExpirationCalculation,' +**  **' [Warehouse Class Code] = @WarehouseClassCode,' +**  **' [Special Equipment Code] = @SpecialEquipmentCode,' +**  **' [Put-away Template Code] = @PutawayTemplateCode,' +**  **' [Put-away Unit of Measure Code] = @PutawayUnitofMeasureCode,' +**  **' [Phys Invt Counting Period Code] = @PhysInvtCountingPeriodCode,' +**  **' [Last Counting Period Update] = @LastCountingPeriodUpdate,' +**  **' [Use Cross-Docking] = @UseCrossDocking,' +**  **' [Next Counting Start Date] = @NextCountingStartDate,' +**  **' [Next Counting End Date] = @NextCountingEndDate,' +**  **' [Item Type] = @ItemType,' +**  **' [Bhairawal Self No\_] = @BhairawalSelfNo,' +**  **' [ML Capacity] = @MLCapacity,' +**  **' [1st Service Claim Amount] = @1stServiceClaimAmount,' +**  **' [2nd Service Claim Amount] = @2ndServiceClaimAmount,' +**  **' [3rd Service Claim Amount] = @3rdServiceClaimAmount,' +**  **' [4th Service Claim Amount] = @4thServiceClaimAmount,' +**  **' [Additional Service Amount] = @AdditionalServiceAmount,' +**  **' [PDI Claim Amount] = @PDIClaimAmount,' +**  **' [CRM Model Version] = @CRMModelVersion,' +**  **' [Edit Not Allowed] = @EditNotAllowed,' +**  **' [Item Category] = @ItemCategory,' +**  **' [Supplier Code] = @SupplierCode,' +**  **' [FMS Code] = @FMSCode,' +**  **' [ABC Code] = @ABCCode,' +**  **' [Report Model Target] = @ReportModelTarget,' +**  **' [Routing No\_] = @RoutingNo,' +**  **' [Production BOM No\_] = @ProductionBOMNo,' +**  **' [Single-Level Material Cost] = @SingleLevelMaterialCost,' +**  **' [Single-Level Capacity Cost] = @SingleLevelCapacityCost,' +**  **' [Single-Level Subcontrd\_ Cost] = @SingleLevelSubcontrdCost,' +**  **' [Single-Level Cap\_ Ovhd Cost] = @SingleLevelCapOvhdCost,' +**  **' [Single-Level Mfg\_ Ovhd Cost] = @SingleLevelMfgOvhdCost,' +**  **' [Overhead Rate] = @OverheadRate,' +**  **' [Rolled-up Subcontracted Cost] = @RolledupSubcontractedCost,' +**  **' [Rolled-up Mfg\_ Ovhd Cost] = @RolledupMfgOvhdCost,' +**  **' [Rolled-up Cap\_ Overhead Cost] = @RolledupCapOverheadCost,' +**  **' [Order Tracking Policy] = @OrderTrackingPolicy,' +**  **' [Critical] = @Critical,' +**  **' [Common Item No\_] = @CommonItemNo,' +**  **' [Free Service Rate] = @FreeServiceRate,' +**  **' [Paid Service Rate] = @PaidServiceRate,' +**  **' [Scratch Start Date] = @ScratchStartDate,' +**  **' [Scratch End Date] = @ScratchEndDate,' +**  **' [Running Model Version] = @RunningModelversion' +**  **SpaceTxt + WhereTxt + '[No\_]' + ' = ''' + FORMAT(Item."No.") +'''';**    **SetupSQLCommand;**  **SQLCommand.Parameters.AddWithValue('@No',Item."No.");**  **SQLCommand.Parameters.AddWithValue('@No2',Item."No. 2");**  **SQLCommand.Parameters.AddWithValue('@Description',Item.Description);**  **SQLCommand.Parameters.AddWithValue('@SearchDescription',Item."Search Description");**  **SQLCommand.Parameters.AddWithValue('@Description2',Item."Description 2");**  **SQLCommand.Parameters.AddWithValue('@BaseUnitofMeasure',Item."Base Unit of Measure");**  **SQLCommand.Parameters.AddWithValue('@PriceUnitConversion',Item."Price Unit Conversion");**  **SQLCommand.Parameters.AddWithValue('@Type', TypeVar);**  **SQLCommand.Parameters.AddWithValue('@InventoryPostingGroup',Item."Inventory Posting Group");**  **SQLCommand.Parameters.AddWithValue('@ShelfNo',Item."Shelf No.");**  **SQLCommand.Parameters.AddWithValue('@ItemDiscGroup',Item."Item Disc. Group");**  **SQLCommand.Parameters.AddWithValue('@AllowInvoiceDisc',Item."Allow Invoice Disc.");**  **SQLCommand.Parameters.AddWithValue('@StatisticsGroup',Item."Statistics Group");**  **SQLCommand.Parameters.AddWithValue('@CommissionGroup',Item."Commission Group");**  **SQLCommand.Parameters.AddWithValue('@UnitPrice',Item."Unit Price");**  **SQLCommand.Parameters.AddWithValue('@PriceProfitCalculation',Item."Price/Profit Calculation");**  **SQLCommand.Parameters.AddWithValue('@Profit',Item."Profit %");**  **SQLCommand.Parameters.AddWithValue('@CostingMethod',Item."Costing Method");**  **SQLCommand.Parameters.AddWithValue('@UnitCost',Item."Unit Cost");**  **SQLCommand.Parameters.AddWithValue('@StandardCost',Item."Standard Cost");**  **SQLCommand.Parameters.AddWithValue('@LastDirectCost',Item."Last Direct Cost");**  **SQLCommand.Parameters.AddWithValue('@IndirectCost',Item."Indirect Cost %");**  **SQLCommand.Parameters.AddWithValue('@CostisAdjusted',Item."Cost is Adjusted");**  **SQLCommand.Parameters.AddWithValue('@AllowOnlineAdjustment',Item."Allow Online Adjustment");**  **SQLCommand.Parameters.AddWithValue('@VendorNo',Item."Vendor No.");**  **SQLCommand.Parameters.AddWithValue('@VendorItemNo',Item."Vendor Item No.");**  **SQLCommand.Parameters.AddWithValue('@LeadTimeCalculation',FORMAT(Item."Lead Time Calculation"));**  **SQLCommand.Parameters.AddWithValue('@ReorderPoint',Item."Reorder Point");**  **SQLCommand.Parameters.AddWithValue('@MaximumInventory',Item."Maximum Inventory");**  **SQLCommand.Parameters.AddWithValue('@ReorderQuantity',Item."Reorder Quantity");**  **SQLCommand.Parameters.AddWithValue('@AlternativeItemNo',Item."Alternative Item No.");**  **SQLCommand.Parameters.AddWithValue('@UnitListPrice',Item."Unit List Price");**  **SQLCommand.Parameters.AddWithValue('@DutyDue',Item."Duty Due %");**  **SQLCommand.Parameters.AddWithValue('@DutyCode',Item."Duty Code");**  **SQLCommand.Parameters.AddWithValue('@GrossWeight',Item."Gross Weight");**  **SQLCommand.Parameters.AddWithValue('@NetWeight',Item."Net Weight");**  **SQLCommand.Parameters.AddWithValue('@UnitsperParcel',Item."Units per Parcel");**  **SQLCommand.Parameters.AddWithValue('@UnitVolume',Item."Unit Volume");**  **SQLCommand.Parameters.AddWithValue('@Durability',Item.Durability);**  **SQLCommand.Parameters.AddWithValue('@FreightType',Item."Freight Type");**  **SQLCommand.Parameters.AddWithValue('@TariffNo',Item."Tariff No.");**  **SQLCommand.Parameters.AddWithValue('@DutyUnitConversion',Item."Duty Unit Conversion");**  **SQLCommand.Parameters.AddWithValue('@CountryRegionPurchasedCode',Item."Country/Region Purchased Code");**  **SQLCommand.Parameters.AddWithValue('@BudgetQuantity',Item."Budget Quantity");**  **SQLCommand.Parameters.AddWithValue('@BudgetedAmount',Item."Budgeted Amount");**  **SQLCommand.Parameters.AddWithValue('@BudgetProfit',Item."Budget Profit");**  **SQLCommand.Parameters.AddWithValue('@Comment',Item.Comment);**  **SQLCommand.Parameters.AddWithValue('@Blocked',Item.Blocked);**  **SQLCommand.Parameters.AddWithValue('@CostisPostedtoGL',Item."Cost is Posted to G/L");**  **SQLCommand.Parameters.AddWithValue('@LastDateModified',LastDateModified);**  **SQLCommand.Parameters.AddWithValue('@LastTimeModified',LastTimeMod);**  **SQLCommand.Parameters.AddWithValue('@DateFilter',DateFilter);**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension1Filter',Item."Global Dimension 1 Filter");**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension2Filter',Item."Global Dimension 2 Filter");**  **SQLCommand.Parameters.AddWithValue('@LocationFilter',Item."Location Filter");**  **SQLCommand.Parameters.AddWithValue('@NetInvoicedQty',Item."Net Invoiced Qty.");**  **SQLCommand.Parameters.AddWithValue('@NetChange',Item."Net Change");**  **SQLCommand.Parameters.AddWithValue('@Purchases',Item."Purchases (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@Sales',Item."Sales (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PositiveAdjmt',Item."Positive Adjmt. (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@NegativeAdjmt',Item."Negative Adjmt. (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PurchasesLCY',Item."Purchases (LCY)");**  **SQLCommand.Parameters.AddWithValue('@SalesLCY',Item."Sales (LCY)");**  **SQLCommand.Parameters.AddWithValue('@PositiveAdjmtLCY',Item."Positive Adjmt. (LCY)");**  **SQLCommand.Parameters.AddWithValue('@NegativeAdjmtLCY',Item."Negative Adjmt. (LCY)");**  **SQLCommand.Parameters.AddWithValue('@COGSLCY',Item."COGS (LCY)");**  **SQLCommand.Parameters.AddWithValue('@QtyonPurchOrder',Item."Qty. on Purch. Order");**  **SQLCommand.Parameters.AddWithValue('@QtyonSalesOrder',Item."Qty. on Sales Order");**  **SQLCommand.Parameters.AddWithValue('@PriceIncludesVAT',Item."Price Includes VAT");**  **SQLCommand.Parameters.AddWithValue('@DropShipmentFilter',Item."Drop Shipment Filter");**  **SQLCommand.Parameters.AddWithValue('@VATBusPostingGrPrice',Item."VAT Bus. Posting Gr. (Price)");**  **SQLCommand.Parameters.AddWithValue('@GenProdPostingGroup',Item."Gen. Prod. Posting Group");**  **SQLCommand.Parameters.AddWithValue('@Picture',FORMAT(Item.Picture));**  **SQLCommand.Parameters.AddWithValue('@Transferred',Item."Transferred (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@TransferredLCY',Item."Transferred (LCY)");**  **SQLCommand.Parameters.AddWithValue('@CountryRegionofOriginCode',Item."Country/Region of Origin Code");**  **SQLCommand.Parameters.AddWithValue('@AutomaticExtTexts',Item."Automatic Ext. Texts");**  **SQLCommand.Parameters.AddWithValue('@NoSeries',Item."No. Series");**  **SQLCommand.Parameters.AddWithValue('@TaxGroupCode',Item."Tax Group Code");**  **SQLCommand.Parameters.AddWithValue('@VATProdPostingGroup',Item."VAT Prod. Posting Group");**  **SQLCommand.Parameters.AddWithValue('@Reserve',Item.Reserve);**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonInventory',Item."Reserved Qty. on Inventory");**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonPurchOrders',Item."Reserved Qty. on Purch. Orders");**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonSalesOrders',Item."Reserved Qty. on Sales Orders");**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension1Code',Item."Global Dimension 1 Code");**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension2Code',Item."Global Dimension 2 Code");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonOutboundTransfer',Item."Res. Qty. on Outbound Transfer");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonInboundTransfer',Item."Res. Qty. on Inbound Transfer");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonSalesReturns',Item."Res. Qty. on Sales Returns");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonPurchReturns',Item."Res. Qty. on Purch. Returns");**  **SQLCommand.Parameters.AddWithValue('@StockoutWarning',Item."Stockout Warning");**  **SQLCommand.Parameters.AddWithValue('@PreventNegativeInventory',Item."Prevent Negative Inventory");**  **SQLCommand.Parameters.AddWithValue('@CostofOpenProductionOrders',Item."Cost of Open Production Orders");**  **SQLCommand.Parameters.AddWithValue('@ApplicationWkshUserID',Item."Application Wksh. User ID");**  **SQLCommand.Parameters.AddWithValue('@AssemblyPolicy',Item."Assembly Policy");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonAssemblyOrder',Item."Res. Qty. on Assembly Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonAsmComp',Item."Res. Qty. on Asm. Comp.");**  **SQLCommand.Parameters.AddWithValue('@QtyonAssemblyOrder',Item."Qty. on Assembly Order");**  **SQLCommand.Parameters.AddWithValue('@QtyonAsmComponent',Item."Qty. on Asm. Component");**  **SQLCommand.Parameters.AddWithValue('@QtyonJobOrder',Item."Qty. on Job Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonJobOrder',Item."Res. Qty. on Job Order");**  **SQLCommand.Parameters.AddWithValue('@GTIN',Item.GTIN);**  **SQLCommand.Parameters.AddWithValue('@DefaultDeferralTemplateCode',Item."Default Deferral Template Code");**  **SQLCommand.Parameters.AddWithValue('@LowLevelCode',Item."Low-Level Code");**  **SQLCommand.Parameters.AddWithValue('@LotSize',Item."Lot Size");**  **SQLCommand.Parameters.AddWithValue('@SerialNos',Item."Serial Nos.");**  **SQLCommand.Parameters.AddWithValue('@LastUnitCostCalcDate',LastUnitCostCalcDate);**  **SQLCommand.Parameters.AddWithValue('@RolledupMaterialCost',Item."Rolled-up Material Cost");**  **SQLCommand.Parameters.AddWithValue('@RolledupCapacityCost',Item."Rolled-up Capacity Cost");**  **SQLCommand.Parameters.AddWithValue('@Scrap',Item."Scrap %");**  **SQLCommand.Parameters.AddWithValue('@InventoryValueZero',Item."Inventory Value Zero");**  **SQLCommand.Parameters.AddWithValue('@DiscreteOrderQuantity',Item."Discrete Order Quantity");**  **SQLCommand.Parameters.AddWithValue('@MinimumOrderQuantity',Item."Minimum Order Quantity");**  **SQLCommand.Parameters.AddWithValue('@MaximumOrderQuantity',Item."Maximum Order Quantity");**  **SQLCommand.Parameters.AddWithValue('@SafetyStockQuantity',Item."Safety Stock Quantity");**  **SQLCommand.Parameters.AddWithValue('@OrderMultiple',Item."Order Multiple");**  **SQLCommand.Parameters.AddWithValue('@SafetyLeadTime',FORMAT(Item."Safety Lead Time"));**  **SQLCommand.Parameters.AddWithValue('@FlushingMethod',Item."Flushing Method");**  **SQLCommand.Parameters.AddWithValue('@ReplenishmentSystem',Item."Replenishment System");**  **SQLCommand.Parameters.AddWithValue('@ScheduledReceipt',Item."Scheduled Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@ScheduledNeed',Item."Scheduled Need (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@RoundingPrecision',Item."Rounding Precision");**  **SQLCommand.Parameters.AddWithValue('@BinFilter',Item."Bin Filter");**  **SQLCommand.Parameters.AddWithValue('@VariantFilter',Item."Variant Filter");**  **SQLCommand.Parameters.AddWithValue('@SalesUnitofMeasure',Item."Sales Unit of Measure");**  **SQLCommand.Parameters.AddWithValue('@PurchUnitofMeasure',Item."Purch. Unit of Measure");**  **SQLCommand.Parameters.AddWithValue('@TimeBucket',FORMAT(Item."Time Bucket"));**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonProdOrder',Item."Reserved Qty. on Prod. Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonProdOrderComp',Item."Res. Qty. on Prod. Order Comp.");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonReqLine',Item."Res. Qty. on Req. Line");**  **SQLCommand.Parameters.AddWithValue('@ReorderingPolicy',Item."Reordering Policy");**  **SQLCommand.Parameters.AddWithValue('@IncludeInventory',Item."Include Inventory");**  **SQLCommand.Parameters.AddWithValue('@ManufacturingPolicy',Item."Manufacturing Policy");**  **SQLCommand.Parameters.AddWithValue('@ReschedulingPeriod',FORMAT(Item."Rescheduling Period"));**  **SQLCommand.Parameters.AddWithValue('@LotAccumulationPeriod',FORMAT(Item."Lot Accumulation Period"));**  **SQLCommand.Parameters.AddWithValue('@DampenerPeriod',FORMAT(Item."Dampener Period"));**  **SQLCommand.Parameters.AddWithValue('@DampenerQuantity',Item."Dampener Quantity");**  **SQLCommand.Parameters.AddWithValue('@OverflowLevel',Item."Overflow Level");**  **SQLCommand.Parameters.AddWithValue('@PlanningTransferShipQty',Item."Planning Transfer Ship. (Qty).");**  **SQLCommand.Parameters.AddWithValue('@PlanningWorksheet',Item."Planning Worksheet (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@StockkeepingUnitExists',Item."Stockkeeping Unit Exists");**  **SQLCommand.Parameters.AddWithValue('@ManufacturerCode',Item."Manufacturer Code");**  **SQLCommand.Parameters.AddWithValue('@ItemCategoryCode',Item."Item Category Code");**  **SQLCommand.Parameters.AddWithValue('@CreatedFromNonstockItem',Item."Created From Nonstock Item");**  **SQLCommand.Parameters.AddWithValue('@ProductGroupCode',Item."Product Group Code");**  **SQLCommand.Parameters.AddWithValue('@SubstitutesExist',Item."Substitutes Exist");**  **SQLCommand.Parameters.AddWithValue('@QtyinTransit',Item."Qty. in Transit");**  **SQLCommand.Parameters.AddWithValue('@TransOrdReceipt',Item."Trans. Ord. Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@TransOrdShipment',Item."Trans. Ord. Shipment (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@QtyAssignedtoship',Item."Qty. Assigned to ship");**  **SQLCommand.Parameters.AddWithValue('@QtyPicked',Item."Qty. Picked");**  **SQLCommand.Parameters.AddWithValue('@ServiceItemGroup',Item."Service Item Group");**  **SQLCommand.Parameters.AddWithValue('@QtyonServiceOrder',Item."Qty. on Service Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonServiceOrders',Item."Res. Qty. on Service Orders");**  **SQLCommand.Parameters.AddWithValue('@ItemTrackingCode',Item."Item Tracking Code");**  **SQLCommand.Parameters.AddWithValue('@LotNos',Item."Lot Nos.");**  **SQLCommand.Parameters.AddWithValue('@ExpirationCalculation', FORMAT(Item."Expiration Calculation"));**  **SQLCommand.Parameters.AddWithValue('@LotNoFilter',Item."Lot No. Filter");**  **SQLCommand.Parameters.AddWithValue('@SerialNoFilter',Item."Serial No. Filter");**  **SQLCommand.Parameters.AddWithValue('@QtyonPurchReturn',Item."Qty. on Purch. Return");**  **SQLCommand.Parameters.AddWithValue('@QtyonSalesReturn',Item."Qty. on Sales Return");**  **SQLCommand.Parameters.AddWithValue('@NoofSubstitutes',Item."No. of Substitutes");**  **SQLCommand.Parameters.AddWithValue('@WarehouseClassCode',Item."Warehouse Class Code");**  **SQLCommand.Parameters.AddWithValue('@SpecialEquipmentCode',Item."Special Equipment Code");**  **SQLCommand.Parameters.AddWithValue('@PutawayTemplateCode',Item."Put-away Template Code");**  **SQLCommand.Parameters.AddWithValue('@PutawayUnitofMeasureCode',Item."Put-away Unit of Measure Code");**  **SQLCommand.Parameters.AddWithValue('@PhysInvtCountingPeriodCode',Item."Phys Invt Counting Period Code");**  **SQLCommand.Parameters.AddWithValue('@LastCountingPeriodUpdate',LastCountingPeriodUpdate);**  **SQLCommand.Parameters.AddWithValue('@LastPhysInvtDate',LastPhyDate);**  **SQLCommand.Parameters.AddWithValue('@UseCrossDocking',Item."Use Cross-Docking");**  **SQLCommand.Parameters.AddWithValue('@NextCountingStartDate',NextCountingStartDate);**  **SQLCommand.Parameters.AddWithValue('@NextCountingEndDate',NextCountingEndDate);**  **SQLCommand.Parameters.AddWithValue('@IdentifierCode',Item."Identifier Code");**  **SQLCommand.Parameters.AddWithValue('@ItemType',ItemTypeVar);**  **SQLCommand.Parameters.AddWithValue('@BhairawalSelfNo',Item."Bhairawal Self No.");**  **SQLCommand.Parameters.AddWithValue('@QtyonPurchInvoice',Item."Qty. on Purch. Invoice");**  **SQLCommand.Parameters.AddWithValue('@QtyonSalesInvoice',Item."Qty. on Sales Invoice");**  **SQLCommand.Parameters.AddWithValue('@MLCapacity',Item."ML Capacity");**  **SQLCommand.Parameters.AddWithValue('@1stServiceClaimAmount',Item."1st Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@2ndServiceClaimAmount',Item."2nd Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@3rdServiceClaimAmount',Item."3rd Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@4thServiceClaimAmount',Item."4th Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@AdditionalServiceAmount',Item."Additional Service Amount");**  **SQLCommand.Parameters.AddWithValue('@PDIClaimAmount',Item."PDI Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@CRMModelVersion',Item."CRM Model Version");**  **SQLCommand.Parameters.AddWithValue('@EditNotAllowed',Item."Edit Not Allowed");**  **SQLCommand.Parameters.AddWithValue('@ItemCategory',Item."Item Category");**  **SQLCommand.Parameters.AddWithValue('@SupplierCode',Item."Supplier Code");**  **SQLCommand.Parameters.AddWithValue('@FMSCode',Item."FMS Code");**  **SQLCommand.Parameters.AddWithValue('@ABCCode',Item."ABC Code");**  **SQLCommand.Parameters.AddWithValue('@ReportModelTarget',Item."Report Model Target");**  **SQLCommand.Parameters.AddWithValue('@RoutingNo',Item."Routing No.");**  **SQLCommand.Parameters.AddWithValue('@ProductionBOMNo',Item."Production BOM No.");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelMaterialCost',Item."Single-Level Material Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelCapacityCost',Item."Single-Level Capacity Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelSubcontrdCost',Item."Single-Level Subcontrd. Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelCapOvhdCost',Item."Single-Level Cap. Ovhd Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelMfgOvhdCost',Item."Single-Level Mfg. Ovhd Cost");**  **SQLCommand.Parameters.AddWithValue('@OverheadRate',Item."Overhead Rate");**  **SQLCommand.Parameters.AddWithValue('@RolledupSubcontractedCost',Item."Rolled-up Subcontracted Cost");**  **SQLCommand.Parameters.AddWithValue('@RolledupMfgOvhdCost',Item."Rolled-up Mfg. Ovhd Cost");**  **SQLCommand.Parameters.AddWithValue('@RolledupCapOverheadCost',Item."Rolled-up Cap. Overhead Cost");**  **SQLCommand.Parameters.AddWithValue('@PlanningIssues',Item."Planning Issues (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlanningReceipt',Item."Planning Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlannedOrderReceipt',Item."Planned Order Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@FPOrderReceipt',Item."FP Order Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@RelOrderReceipt',Item."Rel. Order Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlanningRelease',Item."Planning Release (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlannedOrderRelease',Item."Planned Order Release (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PurchReqReceipt',Item."Purch. Req. Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PurchReqRelease',Item."Purch. Req. Release (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@OrderTrackingPolicy',Item."Order Tracking Policy");**  **SQLCommand.Parameters.AddWithValue('@ProdForecastQuantityBase',Item."Prod. Forecast Quantity (Base)");**  **SQLCommand.Parameters.AddWithValue('@ProductionForecastName',Item."Production Forecast Name");**  **SQLCommand.Parameters.AddWithValue('@ComponentForecast',Item."Component Forecast");**  **SQLCommand.Parameters.AddWithValue('@QtyonProdOrder',Item."Qty. on Prod. Order");**  **SQLCommand.Parameters.AddWithValue('@QtyonComponentLines',Item."Qty. on Component Lines");**  **SQLCommand.Parameters.AddWithValue('@Critical',Item.Critical);**  **SQLCommand.Parameters.AddWithValue('@CommonItemNo',Item."Common Item No.");**  **SQLCommand.Parameters.AddWithValue('@FreeServiceRate',Item."Free Service Rate");**  **SQLCommand.Parameters.AddWithValue('@PaidServiceRate',Item."Paid Service Rate");**  **SQLCommand.Parameters.AddWithValue('@ScratchStartDate',ScratchStart);**  **SQLCommand.Parameters.AddWithValue('@ScratchEndDate',ScratchEnd);**  **SQLCommand.Parameters.AddWithValue('@RunningModelVersion',Item."Running Model Version");**  **SQLCommand.ExecuteNonQuery;**  **LOCAL InsertItem(DBINfo : Record "Dealer Information";Item : Record Item)**  **CLEAR(commandtext);**  **ConnectSQL(DBINfo);**  **CASE Item."Item Type" OF**  **Item."Item Type"::" ":**  **ItemTypeVar := 0;**  **Item."Item Type"::Item:**  **ItemTypeVar := 1;**  **Item."Item Type"::"Model Version":**  **ItemTypeVar := 2;**  **END;**  **CASE Item.Type OF**  **Item.Type::Inventory:**  **TypeVar :=0;**  **Item.Type::Service:**  **TypeVar:= 1;**  **END;**  **IF Item."Next Counting Start Date" <> 0D THEN**  **NextCountingStartDate := FORMAT(Item."Next Counting Start Date")**  **ELSE**  **NextCountingStartDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Date Filter" <> 0D THEN**  **DateFilter := FORMAT(Item."Date Filter")**  **ELSE**  **DateFilter := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Unit Cost Calc. Date" <> 0D THEN**  **LastUnitCostCalcDate := FORMAT(Item."Last Unit Cost Calc. Date")**  **ELSE**  **LastUnitCostCalcDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Counting Period Update" <> 0D THEN**  **LastCountingPeriodUpdate := FORMAT(Item."Last Counting Period Update")**  **ELSE**  **LastCountingPeriodUpdate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Scratch Start Date" <> 0D THEN**  **ScratchStart := FORMAT(Item."Scratch Start Date")**  **ELSE**  **ScratchStart := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Scratch End Date" <> 0D THEN**  **ScratchEnd := FORMAT(Item."Scratch Start Date")**  **ELSE**  **ScratchEnd := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Next Counting End Date" <> 0D THEN**  **NextCountingEndDate := FORMAT(Item."Next Counting Start Date")**  **ELSE**  **NextCountingEndDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Phys. Invt. Date" <> 0D THEN**  **LastPhyDate := FORMAT(Item."Last Phys. Invt. Date")**  **ELSE**  **LastPhyDate := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Time Modified" <> 0T THEN**  **LastTimeMod := FORMAT(Item."Last Time Modified")**  **ELSE**  **LastTimeMod := FORMAT('1753-01-01 00:00:00.000');**  **IF Item."Last Date Modified" <> 0D THEN**  **LastDateModified := FORMAT(Item."Last Date Modified")**  **ELSE**  **LastDateModified := FORMAT('1753-01-01 00:00:00.000');**  **commandtext :=InsertTxt + '['+ DBINfo."Company Name" +'$Item'+ ']' + SpaceTxt + '([No\_], [No\_ 2], [Description],' +**  **'[Search Description], [Description 2], [Base Unit of Measure], [Price Unit Conversion], [Type], [Inventory Posting Group],'+**  **' [Shelf No\_], [Item Disc\_ Group], [Allow Invoice Disc\_], [Statistics Group], [Commission Group], [Unit Price],'+**  **' [Price\_Profit Calculation], [Profit \_], [Costing Method], [Unit Cost], [Standard Cost], [Last Direct Cost],'+**  **' [Indirect Cost \_], [Cost is Adjusted], [Allow Online Adjustment], [Vendor No\_], [Vendor Item No\_], [Lead Time Calculation],'+**  **' [Reorder Point], [Maximum Inventory], [Reorder Quantity], [Alternative Item No\_], [Unit List Price], [Duty Due \_],'+**  **' [Duty Code], [Gross Weight], [Net Weight], [Units per Parcel], [Unit Volume], [Durability], [Freight Type], [Tariff No\_],'+**  **' [Duty Unit Conversion], [Country\_Region Purchased Code], [Budget Quantity], [Budgeted Amount], [Budget Profit], [Blocked],'+**  **' [Last Date Modified], [Last Time Modified], [Price Includes VAT], [VAT Bus\_ Posting Gr\_ (Price)],'+**  **' [Gen\_ Prod\_ Posting Group], [Picture], [Country\_Region of Origin Code], [Automatic Ext\_ Texts], [No\_ Series], '+**  **'[Tax Group Code], [VAT Prod\_ Posting Group], [Reserve], [Global Dimension 1 Code], [Global Dimension 2 Code], '+**  **'[Stockout Warning], [Prevent Negative Inventory], [Application Wksh\_ User ID], [Assembly Policy], [GTIN], '+**  **'[Default Deferral Template Code], [Low-Level Code], [Lot Size], [Serial Nos\_], [Last Unit Cost Calc\_ Date], '+**  **'[Rolled-up Material Cost], [Rolled-up Capacity Cost], [Scrap \_], [Inventory Value Zero], [Discrete Order Quantity],'+**  **' [Minimum Order Quantity], [Maximum Order Quantity], [Safety Stock Quantity], [Order Multiple], [Safety Lead Time], '+**  **'[Flushing Method], [Replenishment System], [Rounding Precision], [Sales Unit of Measure], [Purch\_ Unit of Measure], '+**  **'[Time Bucket], [Reordering Policy], [Include Inventory], [Manufacturing Policy], [Rescheduling Period],'+**  **' [Lot Accumulation Period], [Dampener Period], [Dampener Quantity], [Overflow Level], [Manufacturer Code], '+**  **'[Item Category Code], [Created From Nonstock Item], [Product Group Code], [Service Item Group], [Item Tracking Code], '+**  **'[Lot Nos\_], [Expiration Calculation], [Warehouse Class Code], [Special Equipment Code], [Put-away Template Code],'+**  **' [Put-away Unit of Measure Code], [Phys Invt Counting Period Code], [Last Counting Period Update], [Use Cross-Docking],'+**  **' [Next Counting Start Date], [Next Counting End Date], [Item Type], [Bhairawal Self No\_], [ML Capacity],'+**  **'[1st Service Claim Amount], [2nd Service Claim Amount], [3rd Service Claim Amount], [4th Service Claim Amount],'+**  **' [Additional Service Amount], [PDI Claim Amount], [CRM Model Version], [Edit Not Allowed], [Item Category],'+**  **' [Supplier Code], [FMS Code], [ABC Code],[Report Model Target], [Routing No\_], [Production BOM No\_],'+**  **' [Single-Level Material Cost], [Single-Level Capacity Cost], [Single-Level Subcontrd\_ Cost], [Single-Level Cap\_ Ovhd Cost],'+**  **' [Single-Level Mfg\_ Ovhd Cost], [Overhead Rate], [Rolled-up Subcontracted Cost], [Rolled-up Mfg\_ Ovhd Cost],'+**  **' [Rolled-up Cap\_ Overhead Cost], [Order Tracking Policy], [Critical], [Common Item No\_], [Free Service Rate], '+**  **'[Paid Service Rate], [Scratch Start Date], [Scratch End Date],[Running Model Version])' + ValuesTxt +**  **'(@No,@No2,@Description,@SearchDescription,@Description2,@BaseUnitofMeasure,@PriceUnitConversion,@Type,@InventoryPostingGroup,'+**  **'@ShelfNo,@ItemDiscGroup,@AllowInvoiceDisc,@StatisticsGroup,@CommissionGroup,@UnitPrice,@PriceProfitCalculation,@Profit,@CostingMethod,'+**  **'@UnitCost,@StandardCost,@LastDirectCost,@IndirectCost,@CostisAdjusted,@AllowOnlineAdjustment,@VendorNo,@VendorItemNo,@LeadTimeCalculation,'+**  **'@ReorderPoint,@MaximumInventory,@ReorderQuantity,@AlternativeItemNo,@UnitListPrice,@DutyDue,@DutyCode,@GrossWeight,@NetWeight,@UnitsperParcel,'+**  **'@UnitVolume,@Durability,@FreightType,@TariffNo,@DutyUnitConversion,@CountryRegionPurchasedCode,@BudgetQuantity,@BudgetedAmount,'+**  **'@BudgetProfit,@Blocked,@LastDateModified,@LastTimeModified,@PriceIncludesVAT,@VATBusPostingGrPrice,@GenProdPostingGroup,'+**  **'@Picture,@CountryRegionofOriginCode,@AutomaticExtTexts,@NoSeries,@TaxGroupCode,@VATProdPostingGroup,@Reserve,@GlobalDimension1Code,'+**  **'@GlobalDimension2Code,@StockoutWarning,@PreventNegativeInventory,@ApplicationWkshUserID,@AssemblyPolicy,@GTIN,'+**  **'@DefaultDeferralTemplateCode,@LowLevelCode,@LotSize,@SerialNos,@LastUnitCostCalcDate,@RolledupMaterialCost,@RolledupCapacityCost,'+**  **'@Scrap,@InventoryValueZero,@DiscreteOrderQuantity,@MinimumOrderQuantity,@MaximumOrderQuantity,@SafetyStockQuantity,@OrderMultiple,'+**  **'@SafetyLeadTime,@FlushingMethod,@ReplenishmentSystem,@RoundingPrecision,@SalesUnitofMeasure,@PurchUnitofMeasure,@TimeBucket,'+**  **'@ReorderingPolicy,@IncludeInventory,@ManufacturingPolicy,@ReschedulingPeriod,@LotAccumulationPeriod,@DampenerPeriod,@DampenerQuantity,'+**  **'@OverflowLevel,@ManufacturerCode,@ItemCategoryCode,@CreatedFromNonstockItem,@ProductGroupCode,@ServiceItemGroup,@ItemTrackingCode,'+**  **'@LotNos,@ExpirationCalculation,@WarehouseClassCode,@SpecialEquipmentCode,@PutawayTemplateCode,@PutawayUnitofMeasureCode,'+**  **'@PhysInvtCountingPeriodCode,@LastCountingPeriodUpdate,@UseCrossDocking,@NextCountingStartDate,@NextCountingEndDate,@ItemType,'+**  **'@BhairawalSelfNo,@MLCapacity,@1stServiceClaimAmount,@2ndServiceClaimAmount,@3rdServiceClaimAmount,@4thServiceClaimAmount,'+**  **'@AdditionalServiceAmount,@PDIClaimAmount,@CRMModelVersion,@EditNotAllowed,@ItemCategory,@SupplierCode,@FMSCode,@ABCCode,@ReportModelTarget,'+**  **'@RoutingNo,@ProductionBOMNo,@SingleLevelMaterialCost,@SingleLevelCapacityCost,@SingleLevelSubcontrdCost,'+**  **'@SingleLevelCapOvhdCost,@SingleLevelMfgOvhdCost,@OverheadRate,@RolledupSubcontractedCost,@RolledupMfgOvhdCost,'+**  **'@RolledupCapOverheadCost,@OrderTrackingPolicy,@Critical,@CommonItemNo,@FreeServiceRate,@PaidServiceRate,@ScratchStartDate,@ScratchEndDate,@RunningModelVersion)';**    **SetupSQLCommand;**  **SQLCommand.Parameters.AddWithValue('@No',Item."No.");**  **SQLCommand.Parameters.AddWithValue('@No2',Item."No. 2");**  **SQLCommand.Parameters.AddWithValue('@Description',Item.Description);**  **SQLCommand.Parameters.AddWithValue('@SearchDescription',Item."Search Description");**  **SQLCommand.Parameters.AddWithValue('@Description2',Item."Description 2");**  **SQLCommand.Parameters.AddWithValue('@BaseUnitofMeasure',Item."Base Unit of Measure");**  **SQLCommand.Parameters.AddWithValue('@PriceUnitConversion',Item."Price Unit Conversion");**  **SQLCommand.Parameters.AddWithValue('@Type', TypeVar);**  **SQLCommand.Parameters.AddWithValue('@InventoryPostingGroup',Item."Inventory Posting Group");**  **SQLCommand.Parameters.AddWithValue('@ShelfNo',Item."Shelf No.");**  **SQLCommand.Parameters.AddWithValue('@ItemDiscGroup',Item."Item Disc. Group");**  **SQLCommand.Parameters.AddWithValue('@AllowInvoiceDisc',Item."Allow Invoice Disc.");**  **SQLCommand.Parameters.AddWithValue('@StatisticsGroup',Item."Statistics Group");**  **SQLCommand.Parameters.AddWithValue('@CommissionGroup',Item."Commission Group");**  **SQLCommand.Parameters.AddWithValue('@UnitPrice',Item."Unit Price");**  **SQLCommand.Parameters.AddWithValue('@PriceProfitCalculation',Item."Price/Profit Calculation");**  **SQLCommand.Parameters.AddWithValue('@Profit',Item."Profit %");**  **SQLCommand.Parameters.AddWithValue('@CostingMethod',Item."Costing Method");**  **SQLCommand.Parameters.AddWithValue('@UnitCost',Item."Unit Cost");**  **SQLCommand.Parameters.AddWithValue('@StandardCost',Item."Standard Cost");**  **SQLCommand.Parameters.AddWithValue('@LastDirectCost',Item."Last Direct Cost");**  **SQLCommand.Parameters.AddWithValue('@IndirectCost',Item."Indirect Cost %");**  **SQLCommand.Parameters.AddWithValue('@CostisAdjusted',Item."Cost is Adjusted");**  **SQLCommand.Parameters.AddWithValue('@AllowOnlineAdjustment',Item."Allow Online Adjustment");**  **SQLCommand.Parameters.AddWithValue('@VendorNo',Item."Vendor No.");**  **SQLCommand.Parameters.AddWithValue('@VendorItemNo',Item."Vendor Item No.");**  **SQLCommand.Parameters.AddWithValue('@LeadTimeCalculation',Item."Lead Time Calculation");**  **SQLCommand.Parameters.AddWithValue('@ReorderPoint',Item."Reorder Point");**  **SQLCommand.Parameters.AddWithValue('@MaximumInventory',Item."Maximum Inventory");**  **SQLCommand.Parameters.AddWithValue('@ReorderQuantity',Item."Reorder Quantity");**  **SQLCommand.Parameters.AddWithValue('@AlternativeItemNo',Item."Alternative Item No.");**  **SQLCommand.Parameters.AddWithValue('@UnitListPrice',Item."Unit List Price");**  **SQLCommand.Parameters.AddWithValue('@DutyDue',Item."Duty Due %");**  **SQLCommand.Parameters.AddWithValue('@DutyCode',Item."Duty Code");**  **SQLCommand.Parameters.AddWithValue('@GrossWeight',Item."Gross Weight");**  **SQLCommand.Parameters.AddWithValue('@NetWeight',Item."Net Weight");**  **SQLCommand.Parameters.AddWithValue('@UnitsperParcel',Item."Units per Parcel");**  **SQLCommand.Parameters.AddWithValue('@UnitVolume',Item."Unit Volume");**  **SQLCommand.Parameters.AddWithValue('@Durability',Item.Durability);**  **SQLCommand.Parameters.AddWithValue('@FreightType',Item."Freight Type");**  **SQLCommand.Parameters.AddWithValue('@TariffNo',Item."Tariff No.");**  **SQLCommand.Parameters.AddWithValue('@DutyUnitConversion',Item."Duty Unit Conversion");**  **SQLCommand.Parameters.AddWithValue('@CountryRegionPurchasedCode',Item."Country/Region Purchased Code");**  **SQLCommand.Parameters.AddWithValue('@BudgetQuantity',Item."Budget Quantity");**  **SQLCommand.Parameters.AddWithValue('@BudgetedAmount',Item."Budgeted Amount");**  **SQLCommand.Parameters.AddWithValue('@BudgetProfit',Item."Budget Profit");**  **SQLCommand.Parameters.AddWithValue('@Comment',Item.Comment);**  **SQLCommand.Parameters.AddWithValue('@Blocked',Item.Blocked);**  **SQLCommand.Parameters.AddWithValue('@CostisPostedtoGL',Item."Cost is Posted to G/L");**  **SQLCommand.Parameters.AddWithValue('@LastDateModified',LastDateModified);**  **SQLCommand.Parameters.AddWithValue('@LastTimeModified',LastTimeMod);**  **SQLCommand.Parameters.AddWithValue('@DateFilter',DateFilter);**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension1Filter',Item."Global Dimension 1 Filter");**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension2Filter',Item."Global Dimension 2 Filter");**  **SQLCommand.Parameters.AddWithValue('@LocationFilter',Item."Location Filter");**  **SQLCommand.Parameters.AddWithValue('@Inventory',Item.Inventory);**  **SQLCommand.Parameters.AddWithValue('@NetInvoicedQty',Item."Net Invoiced Qty.");**  **SQLCommand.Parameters.AddWithValue('@NetChange',Item."Net Change");**  **SQLCommand.Parameters.AddWithValue('@Purchases',Item."Purchases (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@Sales',Item."Sales (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PositiveAdjmt',Item."Positive Adjmt. (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@NegativeAdjmt',Item."Negative Adjmt. (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PurchasesLCY',Item."Purchases (LCY)");**  **SQLCommand.Parameters.AddWithValue('@SalesLCY',Item."Sales (LCY)");**  **SQLCommand.Parameters.AddWithValue('@PositiveAdjmtLCY',Item."Positive Adjmt. (LCY)");**  **SQLCommand.Parameters.AddWithValue('@NegativeAdjmtLCY',Item."Negative Adjmt. (LCY)");**  **SQLCommand.Parameters.AddWithValue('@COGSLCY',Item."COGS (LCY)");**  **SQLCommand.Parameters.AddWithValue('@QtyonPurchOrder',Item."Qty. on Purch. Order");**  **SQLCommand.Parameters.AddWithValue('@QtyonSalesOrder',Item."Qty. on Sales Order");**  **SQLCommand.Parameters.AddWithValue('@PriceIncludesVAT',Item."Price Includes VAT");**  **SQLCommand.Parameters.AddWithValue('@DropShipmentFilter',Item."Drop Shipment Filter");**  **SQLCommand.Parameters.AddWithValue('@VATBusPostingGrPrice',Item."VAT Bus. Posting Gr. (Price)");**  **SQLCommand.Parameters.AddWithValue('@GenProdPostingGroup',Item."Gen. Prod. Posting Group");**  **SQLCommand.Parameters.AddWithValue('@Picture',FORMAT(Item.Picture));**  **SQLCommand.Parameters.AddWithValue('@Transferred',Item."Transferred (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@TransferredLCY',Item."Transferred (LCY)");**  **SQLCommand.Parameters.AddWithValue('@CountryRegionofOriginCode',Item."Country/Region of Origin Code");**  **SQLCommand.Parameters.AddWithValue('@AutomaticExtTexts',Item."Automatic Ext. Texts");**  **SQLCommand.Parameters.AddWithValue('@NoSeries',Item."No. Series");**  **SQLCommand.Parameters.AddWithValue('@TaxGroupCode',Item."Tax Group Code");**  **SQLCommand.Parameters.AddWithValue('@VATProdPostingGroup',Item."VAT Prod. Posting Group");**  **SQLCommand.Parameters.AddWithValue('@Reserve',Item.Reserve);**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonInventory',Item."Reserved Qty. on Inventory");**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonPurchOrders',Item."Reserved Qty. on Purch. Orders");**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonSalesOrders',Item."Reserved Qty. on Sales Orders");**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension1Code',Item."Global Dimension 1 Code");**  **SQLCommand.Parameters.AddWithValue('@GlobalDimension2Code',Item."Global Dimension 2 Code");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonOutboundTransfer',Item."Res. Qty. on Outbound Transfer");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonInboundTransfer',Item."Res. Qty. on Inbound Transfer");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonSalesReturns',Item."Res. Qty. on Sales Returns");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonPurchReturns',Item."Res. Qty. on Purch. Returns");**  **SQLCommand.Parameters.AddWithValue('@StockoutWarning',Item."Stockout Warning");**  **SQLCommand.Parameters.AddWithValue('@PreventNegativeInventory',Item."Prevent Negative Inventory");**  **SQLCommand.Parameters.AddWithValue('@CostofOpenProductionOrders',Item."Cost of Open Production Orders");**  **SQLCommand.Parameters.AddWithValue('@ApplicationWkshUserID',Item."Application Wksh. User ID");**  **SQLCommand.Parameters.AddWithValue('@AssemblyPolicy',Item."Assembly Policy");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonAssemblyOrder',Item."Res. Qty. on Assembly Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonAsmComp',Item."Res. Qty. on Asm. Comp.");**  **SQLCommand.Parameters.AddWithValue('@QtyonAssemblyOrder',Item."Qty. on Assembly Order");**  **SQLCommand.Parameters.AddWithValue('@QtyonAsmComponent',Item."Qty. on Asm. Component");**  **SQLCommand.Parameters.AddWithValue('@QtyonJobOrder',Item."Qty. on Job Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonJobOrder',Item."Res. Qty. on Job Order");**  **SQLCommand.Parameters.AddWithValue('@GTIN',Item.GTIN);**  **SQLCommand.Parameters.AddWithValue('@DefaultDeferralTemplateCode',Item."Default Deferral Template Code");**  **SQLCommand.Parameters.AddWithValue('@LowLevelCode',Item."Low-Level Code");**  **SQLCommand.Parameters.AddWithValue('@LotSize',Item."Lot Size");**  **SQLCommand.Parameters.AddWithValue('@SerialNos',Item."Serial Nos.");**  **SQLCommand.Parameters.AddWithValue('@LastUnitCostCalcDate',LastUnitCostCalcDate);**  **SQLCommand.Parameters.AddWithValue('@RolledupMaterialCost',Item."Rolled-up Material Cost");**  **SQLCommand.Parameters.AddWithValue('@RolledupCapacityCost',Item."Rolled-up Capacity Cost");**  **SQLCommand.Parameters.AddWithValue('@Scrap',Item."Scrap %");**  **SQLCommand.Parameters.AddWithValue('@InventoryValueZero',Item."Inventory Value Zero");**  **SQLCommand.Parameters.AddWithValue('@DiscreteOrderQuantity',Item."Discrete Order Quantity");**  **SQLCommand.Parameters.AddWithValue('@MinimumOrderQuantity',Item."Minimum Order Quantity");**  **SQLCommand.Parameters.AddWithValue('@MaximumOrderQuantity',Item."Maximum Order Quantity");**  **SQLCommand.Parameters.AddWithValue('@SafetyStockQuantity',Item."Safety Stock Quantity");**  **SQLCommand.Parameters.AddWithValue('@OrderMultiple',Item."Order Multiple");**  **SQLCommand.Parameters.AddWithValue('@SafetyLeadTime',FORMAT(Item."Safety Lead Time"));**  **SQLCommand.Parameters.AddWithValue('@FlushingMethod',Item."Flushing Method");**  **SQLCommand.Parameters.AddWithValue('@ReplenishmentSystem',Item."Replenishment System");**  **SQLCommand.Parameters.AddWithValue('@ScheduledReceipt',Item."Scheduled Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@ScheduledNeed',Item."Scheduled Need (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@RoundingPrecision',Item."Rounding Precision");**  **SQLCommand.Parameters.AddWithValue('@BinFilter',Item."Bin Filter");**  **SQLCommand.Parameters.AddWithValue('@VariantFilter',Item."Variant Filter");**  **SQLCommand.Parameters.AddWithValue('@SalesUnitofMeasure',Item."Sales Unit of Measure");**  **SQLCommand.Parameters.AddWithValue('@PurchUnitofMeasure',Item."Purch. Unit of Measure");**  **SQLCommand.Parameters.AddWithValue('@TimeBucket',FORMAT(Item."Time Bucket"));**  **SQLCommand.Parameters.AddWithValue('@ReservedQtyonProdOrder',Item."Reserved Qty. on Prod. Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonProdOrderComp',Item."Res. Qty. on Prod. Order Comp.");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonReqLine',Item."Res. Qty. on Req. Line");**  **SQLCommand.Parameters.AddWithValue('@ReorderingPolicy',Item."Reordering Policy");**  **SQLCommand.Parameters.AddWithValue('@IncludeInventory',Item."Include Inventory");**  **SQLCommand.Parameters.AddWithValue('@ManufacturingPolicy',Item."Manufacturing Policy");**  **SQLCommand.Parameters.AddWithValue('@ReschedulingPeriod',FORMAT(Item."Rescheduling Period"));**  **SQLCommand.Parameters.AddWithValue('@LotAccumulationPeriod',FORMAT(Item."Lot Accumulation Period"));**  **SQLCommand.Parameters.AddWithValue('@DampenerPeriod',FORMAT(Item."Dampener Period"));**  **SQLCommand.Parameters.AddWithValue('@DampenerQuantity',Item."Dampener Quantity");**  **SQLCommand.Parameters.AddWithValue('@OverflowLevel',Item."Overflow Level");**  **SQLCommand.Parameters.AddWithValue('@PlanningTransferShipQty',Item."Planning Transfer Ship. (Qty).");**  **SQLCommand.Parameters.AddWithValue('@PlanningWorksheet',Item."Planning Worksheet (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@StockkeepingUnitExists',Item."Stockkeeping Unit Exists");**  **SQLCommand.Parameters.AddWithValue('@ManufacturerCode',Item."Manufacturer Code");**  **SQLCommand.Parameters.AddWithValue('@ItemCategoryCode',Item."Item Category Code");**  **SQLCommand.Parameters.AddWithValue('@CreatedFromNonstockItem',Item."Created From Nonstock Item");**  **SQLCommand.Parameters.AddWithValue('@ProductGroupCode',Item."Product Group Code");**  **SQLCommand.Parameters.AddWithValue('@SubstitutesExist',Item."Substitutes Exist");**  **SQLCommand.Parameters.AddWithValue('@QtyinTransit',Item."Qty. in Transit");**  **SQLCommand.Parameters.AddWithValue('@TransOrdReceipt',Item."Trans. Ord. Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@TransOrdShipment',Item."Trans. Ord. Shipment (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@QtyAssignedtoship',Item."Qty. Assigned to ship");**  **SQLCommand.Parameters.AddWithValue('@QtyPicked',Item."Qty. Picked");**  **SQLCommand.Parameters.AddWithValue('@ServiceItemGroup',Item."Service Item Group");**  **SQLCommand.Parameters.AddWithValue('@QtyonServiceOrder',Item."Qty. on Service Order");**  **SQLCommand.Parameters.AddWithValue('@ResQtyonServiceOrders',Item."Res. Qty. on Service Orders");**  **SQLCommand.Parameters.AddWithValue('@ItemTrackingCode',Item."Item Tracking Code");**  **SQLCommand.Parameters.AddWithValue('@LotNos',Item."Lot Nos.");**  **SQLCommand.Parameters.AddWithValue('@ExpirationCalculation', FORMAT(Item."Expiration Calculation"));**  **SQLCommand.Parameters.AddWithValue('@LotNoFilter',Item."Lot No. Filter");**  **SQLCommand.Parameters.AddWithValue('@SerialNoFilter',Item."Serial No. Filter");**  **SQLCommand.Parameters.AddWithValue('@QtyonPurchReturn',Item."Qty. on Purch. Return");**  **SQLCommand.Parameters.AddWithValue('@QtyonSalesReturn',Item."Qty. on Sales Return");**  **SQLCommand.Parameters.AddWithValue('@NoofSubstitutes',Item."No. of Substitutes");**  **SQLCommand.Parameters.AddWithValue('@WarehouseClassCode',Item."Warehouse Class Code");**  **SQLCommand.Parameters.AddWithValue('@SpecialEquipmentCode',Item."Special Equipment Code");**  **SQLCommand.Parameters.AddWithValue('@PutawayTemplateCode',Item."Put-away Template Code");**  **SQLCommand.Parameters.AddWithValue('@PutawayUnitofMeasureCode',Item."Put-away Unit of Measure Code");**  **SQLCommand.Parameters.AddWithValue('@PhysInvtCountingPeriodCode',Item."Phys Invt Counting Period Code");**  **SQLCommand.Parameters.AddWithValue('@LastCountingPeriodUpdate',LastCountingPeriodUpdate);**  **SQLCommand.Parameters.AddWithValue('@LastPhysInvtDate',LastPhyDate);**  **SQLCommand.Parameters.AddWithValue('@UseCrossDocking',Item."Use Cross-Docking");**  **SQLCommand.Parameters.AddWithValue('@NextCountingStartDate',NextCountingStartDate);**  **SQLCommand.Parameters.AddWithValue('@NextCountingEndDate',NextCountingEndDate);**  **SQLCommand.Parameters.AddWithValue('@IdentifierCode',Item."Identifier Code");**  **SQLCommand.Parameters.AddWithValue('@ItemType',ItemTypeVar);**  **SQLCommand.Parameters.AddWithValue('@BhairawalSelfNo',Item."Bhairawal Self No.");**  **SQLCommand.Parameters.AddWithValue('@QtyonPurchInvoice',Item."Qty. on Purch. Invoice");**  **SQLCommand.Parameters.AddWithValue('@QtyonSalesInvoice',Item."Qty. on Sales Invoice");**  **SQLCommand.Parameters.AddWithValue('@MLCapacity',Item."ML Capacity");**  **SQLCommand.Parameters.AddWithValue('@1stServiceClaimAmount',Item."1st Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@2ndServiceClaimAmount',Item."2nd Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@3rdServiceClaimAmount',Item."3rd Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@4thServiceClaimAmount',Item."4th Service Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@AdditionalServiceAmount',Item."Additional Service Amount");**  **SQLCommand.Parameters.AddWithValue('@PDIClaimAmount',Item."PDI Claim Amount");**  **SQLCommand.Parameters.AddWithValue('@CRMModelVersion',Item."CRM Model Version");**  **SQLCommand.Parameters.AddWithValue('@EditNotAllowed',Item."Edit Not Allowed");**  **SQLCommand.Parameters.AddWithValue('@ItemCategory',Item."Item Category");**  **SQLCommand.Parameters.AddWithValue('@SupplierCode',Item."Supplier Code");**  **SQLCommand.Parameters.AddWithValue('@FMSCode',Item."FMS Code");**  **SQLCommand.Parameters.AddWithValue('@ABCCode',Item."ABC Code");**  **SQLCommand.Parameters.AddWithValue('@ReportModelTarget',Item."Report Model Target");**  **SQLCommand.Parameters.AddWithValue('@RoutingNo',Item."Routing No.");**  **SQLCommand.Parameters.AddWithValue('@ProductionBOMNo',Item."Production BOM No.");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelMaterialCost',Item."Single-Level Material Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelCapacityCost',Item."Single-Level Capacity Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelSubcontrdCost',Item."Single-Level Subcontrd. Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelCapOvhdCost',Item."Single-Level Cap. Ovhd Cost");**  **SQLCommand.Parameters.AddWithValue('@SingleLevelMfgOvhdCost',Item."Single-Level Mfg. Ovhd Cost");**  **SQLCommand.Parameters.AddWithValue('@OverheadRate',Item."Overhead Rate");**  **SQLCommand.Parameters.AddWithValue('@RolledupSubcontractedCost',Item."Rolled-up Subcontracted Cost");**  **SQLCommand.Parameters.AddWithValue('@RolledupMfgOvhdCost',Item."Rolled-up Mfg. Ovhd Cost");**  **SQLCommand.Parameters.AddWithValue('@RolledupCapOverheadCost',Item."Rolled-up Cap. Overhead Cost");**  **SQLCommand.Parameters.AddWithValue('@PlanningIssues',Item."Planning Issues (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlanningReceipt',Item."Planning Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlannedOrderReceipt',Item."Planned Order Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@FPOrderReceipt',Item."FP Order Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@RelOrderReceipt',Item."Rel. Order Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlanningRelease',Item."Planning Release (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PlannedOrderRelease',Item."Planned Order Release (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PurchReqReceipt',Item."Purch. Req. Receipt (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@PurchReqRelease',Item."Purch. Req. Release (Qty.)");**  **SQLCommand.Parameters.AddWithValue('@OrderTrackingPolicy',Item."Order Tracking Policy");**  **SQLCommand.Parameters.AddWithValue('@ProdForecastQuantityBase',Item."Prod. Forecast Quantity (Base)");**  **SQLCommand.Parameters.AddWithValue('@ProductionForecastName',Item."Production Forecast Name");**  **SQLCommand.Parameters.AddWithValue('@ComponentForecast',Item."Component Forecast");**  **SQLCommand.Parameters.AddWithValue('@QtyonProdOrder',Item."Qty. on Prod. Order");**  **SQLCommand.Parameters.AddWithValue('@QtyonComponentLines',Item."Qty. on Component Lines");**  **SQLCommand.Parameters.AddWithValue('@Critical',Item.Critical);**  **SQLCommand.Parameters.AddWithValue('@CommonItemNo',Item."Common Item No.");**  **SQLCommand.Parameters.AddWithValue('@FreeServiceRate',Item."Free Service Rate");**  **SQLCommand.Parameters.AddWithValue('@PaidServiceRate',Item."Paid Service Rate");**  **SQLCommand.Parameters.AddWithValue('@ScratchStartDate',ScratchStart);**  **SQLCommand.Parameters.AddWithValue('@ScratchEndDate',ScratchEnd);**  **SQLCommand.Parameters.AddWithValue('@RunningModelVersion',Item."Running Model Version");**  **SQLCommand.ExecuteNonQuery;**  **LOCAL ReadItem(DealerInfo : Record "Dealer Information";Item : Record Item)**  **ClearSQLCommand;**  **commandtext := ReadCommandTxt + '[' + DealerInfo."Company Name" + '$Item' + ']' +**  **SpaceTxt + WhereTxt + '[No\_]' + ' = ''' + FORMAT(Item."No.") +'''';**  **SetupSQLCommand;**  **SQLDataReader := SQLCommand.ExecuteReader;**  **SQLDataReader.Read;**  **LOCAL UpdateVehicle(DealerInfo : Record "Dealer Information";SerialNoInfo : Record "Serial No. Information")**  **CLEAR(commandtext);**  **ConnectSQL(DealerInfo);**  **CASE SerialNoInfo.Status OF**  **SerialNoInfo.Status::" ":**  **StatusVar := 0;**  **SerialNoInfo.Status::New:**  **StatusVar := 1;**  **SerialNoInfo.Status::Used:**  **StatusVar := 2;**  **END;**  **CASE SerialNoInfo."Scheme Option" OF**  **SerialNoInfo."Scheme Option"::" ":**  **SchemeOptionVar := 0;**  **SerialNoInfo."Scheme Option"::"5 Year AMC":**  **SchemeOptionVar := 1;**  **SerialNoInfo."Scheme Option"::"Scratch Coupon":**  **SchemeOptionVar := 2;**  **SerialNoInfo."Scheme Option"::"Two Year Free Service":**  **SchemeOptionVar := 3;**  **END;**  **IF SerialNoInfo."Date Filter" <> 0D THEN**  **DateFilter := FORMAT(SerialNoInfo."Date Filter")**  **ELSE**  **DateFilter := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Last Date Modified" <> 0D THEN**  **LastDateModified := FORMAT(SerialNoInfo."Last Date Modified")**  **ELSE**  **LastDateModified := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Creation Date" <> 0D THEN**  **CreationDate := FORMAT(SerialNoInfo."Creation Date")**  **ELSE**  **CreationDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Sales Date" <> 0D THEN**  **SalesDate := FORMAT(SerialNoInfo."Sales Date")**  **ELSE**  **SalesDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Scheme Expiry Date" <> 0D THEN**  **SchemeExpiryDate := FORMAT(SerialNoInfo."Scheme Expiry Date")**  **ELSE**  **SchemeExpiryDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Amc Serial Modified Date" <> 0D THEN**  **AmcSerialModifiedDate := FORMAT(SerialNoInfo."Amc Serial Modified Date")**  **ELSE**  **AmcSerialModifiedDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Scheme Start Date" <> 0D THEN**  **SchemeStartDate := FORMAT(SerialNoInfo."Scheme Start Date")**  **ELSE**  **SchemeStartDate := FORMAT('1753-01-01 00:00:00.000');**  **commandtext := UpdateTxt + '[' + DealerInfo."Company Name" + '$Serial No\_ Information' + ']' + SpaceTxt + SetTxt +**  **'[Item No\_] = @ItemNo,' +**  **'[Variant Code] = @VariantCode,' +**  **'[Serial No\_] = @SerialNo,' +**  **'[Description] = @Description,' +**  **'[Blocked] = @Blocked,' +**  **'[VIN] = @VIN,' +**  **'[Make Code] = @MakeCode,' +**  **'[Model Code] = @ModelCode,' +**  **'[Model Commercial Name] = @ModelCommercialName,' +**  **'[Production Year] = @ProductionYear,' +**  **'[Registration No\_] = @RegistrationNo,' +**  **'[Status] = @Status,' +**  **'[Last Date Modified] = @LastDateModified,' +**  **'[Creation Date] = @CreationDate,' +**  **'[Kilometrage] = @Kilometrage,' +**  **'[Sales Date] = @SalesDate,' +**  **'[Hour] = @Hour,' +**  **'[Body Color Code] = @BodyColorCode,' +**  **'[Syakar No\_] = @SyakarNo,' +**  **'[Engine No\_] = @EngineNo,' +**  **'[No\_ Series] = @NoSeries,' +**  **'[Gear Box No\_] = @GearBoxNo,' +**  **'[F\_I\_P No\_] = @FIPNo,' +**  **'[Front Axle No\_] = @FrontAxleNo,' +**  **'[Rear Axle No\_] = @RearAxleNo,' +**  **'[Battery Make] = @BatteryMake,' +**  **'[Battery No\_] = @BatteryNo,' +**  **'[Principal Serial No\_] = @PrincipalSerialNo,' +**  **'[VC No\_] = @VCNo,' +**  **'[Purchased] = @Purchased,' +**  **'[AMC Serial No\_] = @AMCSerialNo,' +**  **'[Expired] = @Expired,' +**  **'[Scheme] = @Scheme,' +**  **'[Scheme Option] = @SchemeOption,' +**  **'[Scheme Expiry Date] = @SchemeExpiryDate,' +**  **'[Old Vehicle Registration No\_] = @OldVehicleRegistrationNo,' +**  **'[Purchase Order No\_] = @PurchaseOrderNo,' +**  **'[Amc Serial Modified Date] = @AmcSerialModifiedDate,' +**  **'[Scheme Start Date] = @SchemeStartDate' +**  **SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SerialNoInfo."Item No.") +''''+ SpaceTxt+ AndText+**  **SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SerialNoInfo."Variant Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Serial No\_]' + ' = ''' + FORMAT(SerialNoInfo."Serial No.") +'''';**    **SetupSQLCommand;**  **SQLCommand.Parameters.AddWithValue('@ItemNo', SerialNoInfo."Item No.");**  **SQLCommand.Parameters.AddWithValue('@VariantCode', SerialNoInfo."Variant Code");**  **SQLCommand.Parameters.AddWithValue('@SerialNo', SerialNoInfo."Serial No.");**  **SQLCommand.Parameters.AddWithValue('@Description', SerialNoInfo.Description);**  **SQLCommand.Parameters.AddWithValue('@Blocked', SerialNoInfo.Blocked);**  **SQLCommand.Parameters.AddWithValue('@VIN', FORMAT(SerialNoInfo.VIN));**  **SQLCommand.Parameters.AddWithValue('@MakeCode', SerialNoInfo."Make Code");**  **SQLCommand.Parameters.AddWithValue('@ModelCode', SerialNoInfo."Model Code");**  **SQLCommand.Parameters.AddWithValue('@ModelCommercialName', SerialNoInfo."Model Commercial Name");**  **SQLCommand.Parameters.AddWithValue('@ProductionYear', SerialNoInfo."Production Year");**  **SQLCommand.Parameters.AddWithValue('@RegistrationNo', SerialNoInfo."Registration No.");**  **SQLCommand.Parameters.AddWithValue('@Status', StatusVar);**  **SQLCommand.Parameters.AddWithValue('@LastDateModified', LastDateModified);**  **SQLCommand.Parameters.AddWithValue('@CreationDate', CreationDate);**  **SQLCommand.Parameters.AddWithValue('@Kilometrage', SerialNoInfo.Kilometrage);**  **SQLCommand.Parameters.AddWithValue('@SalesDate', SalesDate);**  **SQLCommand.Parameters.AddWithValue('@Hour', SerialNoInfo.Hour);**  **SQLCommand.Parameters.AddWithValue('@BodyColorCode', SerialNoInfo."Body Color Code");**  **SQLCommand.Parameters.AddWithValue('@SyakarNo', SerialNoInfo."Syakar No.");**  **SQLCommand.Parameters.AddWithValue('@EngineNo', SerialNoInfo."Engine No.");**  **SQLCommand.Parameters.AddWithValue('@NoSeries', SerialNoInfo."No. Series");**  **SQLCommand.Parameters.AddWithValue('@GearBoxNo', SerialNoInfo."Gear Box No.");**  **SQLCommand.Parameters.AddWithValue('@FIPNo', SerialNoInfo."F.I.P No.");**  **SQLCommand.Parameters.AddWithValue('@FrontAxleNo', SerialNoInfo."Front Axle No.");**  **SQLCommand.Parameters.AddWithValue('@RearAxleNo', SerialNoInfo."Rear Axle No.");**  **SQLCommand.Parameters.AddWithValue('@BatteryMake', SerialNoInfo."Battery Make");**  **SQLCommand.Parameters.AddWithValue('@BatteryNo', SerialNoInfo."Battery No.");**  **SQLCommand.Parameters.AddWithValue('@PrincipalSerialNo', SerialNoInfo."Principal Serial No.");**  **SQLCommand.Parameters.AddWithValue('@VCNo', SerialNoInfo."VC No.");**  **SQLCommand.Parameters.AddWithValue('@Purchased', SerialNoInfo.Purchased);**  **SQLCommand.Parameters.AddWithValue('@AMCSerialNo', SerialNoInfo."AMC Serial No.");**  **SQLCommand.Parameters.AddWithValue('@Expired', SerialNoInfo.Expired);**  **SQLCommand.Parameters.AddWithValue('@Scheme', SerialNoInfo.Scheme);**  **SQLCommand.Parameters.AddWithValue('@SchemeOption', SchemeOptionVar);**  **SQLCommand.Parameters.AddWithValue('@SchemeExpiryDate', SchemeExpiryDate);**  **SQLCommand.Parameters.AddWithValue('@OldVehicleRegistrationNo', SerialNoInfo."Old Vehicle Registration No.");**  **SQLCommand.Parameters.AddWithValue('@PurchaseOrderNo', SerialNoInfo."Purchase Order No.");**  **SQLCommand.Parameters.AddWithValue('@AmcSerialModifiedDate', AmcSerialModifiedDate);**  **SQLCommand.Parameters.AddWithValue('@SchemeStartDate', FORMAT(SerialNoInfo."Scheme Start Date"));**  **SQLCommand.ExecuteNonQuery;**  **LOCAL InsertVehicle(DBINfo : Record "Dealer Information";SerialNoInfo : Record "Serial No. Information")**  **CLEAR(commandtext);**  **ConnectSQL(DBINfo);**  **CASE SerialNoInfo.Status OF**  **SerialNoInfo.Status::" ":**  **StatusVar := 0;**  **SerialNoInfo.Status::New:**  **StatusVar := 1;**  **SerialNoInfo.Status::Used:**  **StatusVar := 2;**  **END;**  **CASE SerialNoInfo."Scheme Option" OF**  **SerialNoInfo."Scheme Option"::" ":**  **SchemeOptionVar := 0;**  **SerialNoInfo."Scheme Option"::"5 Year AMC":**  **SchemeOptionVar := 1;**  **SerialNoInfo."Scheme Option"::"Scratch Coupon":**  **SchemeOptionVar := 2;**  **SerialNoInfo."Scheme Option"::"Two Year Free Service":**  **SchemeOptionVar := 3;**  **END;**  **IF SerialNoInfo."Date Filter" <> 0D THEN**  **DateFilter := FORMAT(SerialNoInfo."Date Filter")**  **ELSE**  **DateFilter := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Last Date Modified" <> 0D THEN**  **LastDateModified := FORMAT(SerialNoInfo."Last Date Modified")**  **ELSE**  **LastDateModified := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Creation Date" <> 0D THEN**  **CreationDate := FORMAT(SerialNoInfo."Creation Date")**  **ELSE**  **CreationDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Sales Date" <> 0D THEN**  **SalesDate := FORMAT(SerialNoInfo."Sales Date")**  **ELSE**  **SalesDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Scheme Expiry Date" <> 0D THEN**  **SchemeExpiryDate := FORMAT(SerialNoInfo."Scheme Expiry Date")**  **ELSE**  **SchemeExpiryDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Amc Serial Modified Date" <> 0D THEN**  **AmcSerialModifiedDate := FORMAT(SerialNoInfo."Amc Serial Modified Date")**  **ELSE**  **AmcSerialModifiedDate := FORMAT('1753-01-01 00:00:00.000');**  **IF SerialNoInfo."Scheme Start Date" <> 0D THEN**  **SchemeStartDate := FORMAT(SerialNoInfo."Scheme Start Date")**  **ELSE**  **SchemeStartDate := FORMAT('1753-01-01 00:00:00.000');**  **commandtext :=InsertTxt + '['+ DBINfo."Company Name" +'$Serial No\_ Information'+ ']' + SpaceTxt + '([Item No\_], [Variant Code], [Serial No\_],'+**  **'[Description], [Blocked], [VIN], '+**  **'[Make Code], [Model Code], [Model Commercial Name], [Production Year], [Registration No\_], [Status],'+**  **'[Last Date Modified], [Creation Date], [Kilometrage], [Sales Date], [Hour], [Body Color Code], [Syakar No\_], [Engine No\_], [No\_ Series],'+**  **'[Gear Box No\_], [F\_I\_P No\_], [Front Axle No\_], [Rear Axle No\_], [Battery Make], [Battery No\_], [Principal Serial No\_], [VC No\_], [Purchased],'+**  **'[AMC Serial No\_], [Expired], [Scheme], [Scheme Option], [Scheme Expiry Date], [Old Vehicle Registration No\_], [Purchase Order No\_],'+**  **'[Amc Serial Modified Date], [Scheme Start Date])' + ValuesTxt +**  **'(@ItemNo, @VariantCode, @SerialNo, @Description, @Blocked, @VIN, @MakeCode, @ModelCode, @ModelCommercialName, @ProductionYear,'+**  **'@RegistrationNo, @Status, @LastDateModified, @CreationDate, @Kilometrage, @SalesDate, @Hour, @BodyColorCode, @SyakarNo, @EngineNo,'+**  **'@NoSeries, @GearBoxNo, @FIPNo, @FrontAxleNo, @RearAxleNo, @BatteryMake, @BatteryNo, @PrincipalSerialNo, @VCNo, @Purchased, @AMCSerialNo, '+**  **'@Expired, @Scheme, @SchemeOption, @SchemeExpiryDate, @OldVehicleRegistrationNo, @PurchaseOrderNo, @AmcSerialModifiedDate, @SchemeStartDate)';**    **SetupSQLCommand;**  **SQLCommand.Parameters.AddWithValue('@ItemNo', SerialNoInfo."Item No.");**  **SQLCommand.Parameters.AddWithValue('@VariantCode', SerialNoInfo."Variant Code");**  **SQLCommand.Parameters.AddWithValue('@SerialNo', SerialNoInfo."Serial No.");**  **SQLCommand.Parameters.AddWithValue('@Description', SerialNoInfo.Description);**  **SQLCommand.Parameters.AddWithValue('@Blocked', SerialNoInfo.Blocked);**  **SQLCommand.Parameters.AddWithValue('@VIN', FORMAT(SerialNoInfo.VIN));**  **SQLCommand.Parameters.AddWithValue('@MakeCode', SerialNoInfo."Make Code");**  **SQLCommand.Parameters.AddWithValue('@ModelCode', SerialNoInfo."Model Code");**  **SQLCommand.Parameters.AddWithValue('@ModelCommercialName', SerialNoInfo."Model Commercial Name");**  **SQLCommand.Parameters.AddWithValue('@ProductionYear', SerialNoInfo."Production Year");**  **SQLCommand.Parameters.AddWithValue('@RegistrationNo', SerialNoInfo."Registration No.");**  **SQLCommand.Parameters.AddWithValue('@Status', StatusVar);**  **SQLCommand.Parameters.AddWithValue('@LastDateModified', LastDateModified);**  **SQLCommand.Parameters.AddWithValue('@CreationDate', CreationDate);**  **SQLCommand.Parameters.AddWithValue('@Kilometrage', SerialNoInfo.Kilometrage);**  **SQLCommand.Parameters.AddWithValue('@SalesDate', SalesDate);**  **SQLCommand.Parameters.AddWithValue('@Hour', SerialNoInfo.Hour);**  **SQLCommand.Parameters.AddWithValue('@BodyColorCode', SerialNoInfo."Body Color Code");**  **SQLCommand.Parameters.AddWithValue('@SyakarNo', SerialNoInfo."Syakar No.");**  **SQLCommand.Parameters.AddWithValue('@EngineNo', SerialNoInfo."Engine No.");**  **SQLCommand.Parameters.AddWithValue('@NoSeries', SerialNoInfo."No. Series");**  **SQLCommand.Parameters.AddWithValue('@GearBoxNo', SerialNoInfo."Gear Box No.");**  **SQLCommand.Parameters.AddWithValue('@FIPNo', SerialNoInfo."F.I.P No.");**  **SQLCommand.Parameters.AddWithValue('@FrontAxleNo', SerialNoInfo."Front Axle No.");**  **SQLCommand.Parameters.AddWithValue('@RearAxleNo', SerialNoInfo."Rear Axle No.");**  **SQLCommand.Parameters.AddWithValue('@BatteryMake', SerialNoInfo."Battery Make");**  **SQLCommand.Parameters.AddWithValue('@BatteryNo', SerialNoInfo."Battery No.");**  **SQLCommand.Parameters.AddWithValue('@PrincipalSerialNo', SerialNoInfo."Principal Serial No.");**  **SQLCommand.Parameters.AddWithValue('@VCNo', SerialNoInfo."VC No.");**  **SQLCommand.Parameters.AddWithValue('@Purchased', SerialNoInfo.Purchased);**  **SQLCommand.Parameters.AddWithValue('@AMCSerialNo', SerialNoInfo."AMC Serial No.");**  **SQLCommand.Parameters.AddWithValue('@Expired', SerialNoInfo.Expired);**  **SQLCommand.Parameters.AddWithValue('@Scheme', SerialNoInfo.Scheme);**  **SQLCommand.Parameters.AddWithValue('@SchemeOption', SchemeOptionVar);**  **SQLCommand.Parameters.AddWithValue('@SchemeExpiryDate', SchemeExpiryDate);**  **SQLCommand.Parameters.AddWithValue('@OldVehicleRegistrationNo', SerialNoInfo."Old Vehicle Registration No.");**  **SQLCommand.Parameters.AddWithValue('@PurchaseOrderNo', SerialNoInfo."Purchase Order No.");**  **SQLCommand.Parameters.AddWithValue('@AmcSerialModifiedDate', AmcSerialModifiedDate);**  **SQLCommand.Parameters.AddWithValue('@SchemeStartDate', FORMAT(SerialNoInfo."Scheme Start Date"));**  **SQLCommand.ExecuteNonQuery;**  **LOCAL ReadVehicle(DealerInfo : Record "Dealer Information";SerialNoInfo : Record "Serial No. Information")**  **ClearSQLCommand;**  **commandtext := ReadCommandTxt + '[' + DealerInfo."Company Name" + '$Serial No\_ Information' + ']' +**  **SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SerialNoInfo."Item No.") +''''+ SpaceTxt+ AndText+**  **SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SerialNoInfo."Variant Code") +'''' + SpaceTxt + AndText+**  **SpaceTxt + '[Serial No\_]' + ' = ''' + FORMAT**(SerialNoInfo."Serial No.") +'''';  SetupSQLCommand;  SQLDataReader := SQLCommand.ExecuteReader;  SQLDataReader.Read; |