



Infor VISUAL API Toolkit Shop Floor Class Library Reference

Important Notices

The material contained in this publication (including any supplementary information) constitutes and contains confidential and proprietary information of Infor.

By gaining access to the attached, you acknowledge and agree that the material (including any modification, translation or adaptation of the material) and all copyright, trade secrets and all other right, title and interest therein, are the sole property of Infor and that you shall not gain right, title or interest in the material (including any modification, translation or adaptation of the material) by virtue of your review thereof other than the non-exclusive right to use the material solely in connection with and the furtherance of your license and use of software made available to your company from Infor pursuant to a separate agreement, the terms of which separate agreement shall govern your use of this material and all supplemental related materials ("Purpose").

In addition, by accessing the enclosed material, you acknowledge and agree that you are required to maintain such material in strict confidence and that your use of such material is limited to the Purpose described above. Although Infor has taken due care to ensure that the material included in this publication is accurate and complete, Infor cannot warrant that the information contained in this publication is complete, does not contain typographical or other errors, or will meet your specific requirements. As such, Infor does not assume and hereby disclaims all liability, consequential or otherwise, for any loss or damage to any person or entity which is caused by or relates to errors or omissions in this publication (including any supplementary information), whether such errors or omissions result from negligence, accident or any other cause.

Without limitation, U.S. export control laws and other applicable export and import laws govern your use of this material and you will neither export or re-export, directly or indirectly, this material nor any related materials or supplemental information in violation of such laws, or use such materials for any purpose prohibited by such laws.

Trademark Acknowledgements

The word and design marks set forth herein are trademarks and/or registered trademarks of Infor and/or related affiliates and subsidiaries. All rights reserved. All other company, product, trade or service names referenced may be registered trademarks or trademarks of their respective owners.

Publication Information

Release: Infor VISUAL API Toolkit

Publication date: August 13, 2024

About this guide

This guide describes the objects available in the Infor VISUAL API Toolkit Shop Floor class library.

Intended audience

The intended audience of this guide is developers who are using the API Toolkit to extend the VISUAL solution.

Contacting Support

If you have questions about Infor products, go to the Infor Customer Portal at <https://customerportal.infor.com/csmcore/>

If we update this document after the product release, we will post the new version on this Web site. We recommend that you check this Web site periodically for updated documentation.

If you have comments about Infor documentation, contact <https://docs.infor.com/en-us>.

Supported languages

These languages are supported for use with the toolkit:

- Visual Basic
- C#

While it is possible to use any .NET-aware programming language with the toolkit, other languages are not officially supported.

Support information











The API Toolkit will be updated regularly as more class members are added to each assembly, schema changes are made, and any reported issues are resolved. Infor Support cannot assist you with developing customized code using the API Toolkit. For assistance with customizations, contact Infor Consulting Services or your channel partner.

The functionality provided within the API Toolkit will not be extended beyond the standard functionality experienced in the VISUAL application itself. Enhancement requests with compelling business cases detailing how suggested alternatives are not viable will be evaluated and considered.

Infor is not responsible for data incorrectly entered to the database through the use of the API Toolkit. Customers must establish a full test environment to ensure that data created by APIs functions in the same manner as data created through the VISUAL interface.

Lsa.Vmfg.ShopFloor Namespace

Classes

| | Class | Description |
|---|-------------------------------------|---|
|  | ChangeWOStatus | Transaction to change the status of Work Orders. Caller has the option to "cascade" the change to all objects subordinate to the key provided by setting EXPLODE to true. |
|  | CopyWorkOrder | Transaction to copy Work Orders. |
|  | CostCategory | Maintain Cost Categories. |
|  | CostGroup | Maintain Cost Groups. |
|  | DeleteLaborTicket | Transaction for Deleting a Labor Ticket. Note: Posted Labor Tickets cannot be deleted. |
|  | EditLaborTicket | Transaction for editing Labor Tickets. |
|  | GetWorkOrderSummary | Service to obtain a single data table containing summary information for a Work Order. |
|  | LaborTicket | Transaction for creating a Labor Ticket. Three types of transactions are supported: Setup, Run, and Indirect. |
|  | ShopResource | Maintain Shop Resources. |
|  | WorkOrder | Maintain Work Orders. |

ChangeWOStatus Class

Transaction to change the status of Work Orders. Caller has the option to "cascade" the change to all objects subordinate to the key provided by setting EXPLODE to true.

Inheritance Hierarchy

[System.Object](#)

[System.MarshalByRefObject](#)

[System.ComponentModel.Component](#)

BusinessObject

BusinessTransaction

Lsa.Vmfg.ShopFloor.ChangeWOStatus

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
[SerializableAttribute]  
public class ChangeWOStatus : BusinessTransaction
```

VB




```
<SerializableAttribute>  
Public Class ChangeWOStatus  
    Inherits BusinessTransaction
```

The **ChangeWOStatus** type exposes the following members.

Constructors

| | Name | Description |
|---|--|-------------|
|  | ChangeWOStatus() | Constructor |
|  | ChangeWOStatus(String) | Constructor |

Methods

| | Name | Description |
|---|-----------------------------|---|
|  | NewInputRow | Inserts a new row into the CHANGE_WO_STATUS transaction data table. See ChangeWorkOrderStatus . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

Transaction

| | Name | Data set returned by Prepare | Description |
|--|---------------------------------------|------------------------------|--|
| | ChangeWorkOrderStatus | CHANGE_WO_STATUS | This transaction allows for the change of an existing work order's status. The status change may be just applied to the key supplied, or may be applied to the specified key and all of its subordinates in the structure. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus Constructor

Overload List

| | Name | Description |
|---|--|-------------|
|  | ChangeWOStatus() | Constructor |
|  | ChangeWOStatus(String) | Constructor |

See Also

[ChangeWOStatus Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus Constructor

Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public ChangeWOStatus()
```

VB

```
Public Sub New
```

See Also

[ChangeWOStatus Class](#)

[ChangeWOStatus Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus Constructor (String)

Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public ChangeWOStatus(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[ChangeWOStatus Class](#)




[ChangeWOStatus Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus.ChangeWOStatus Methods

The [ChangeWOStatus](#) type exposes the following members.

Methods

| | Name | Description |
|---|-----------------------------|--|
|  | NewInputRow | Inserts a new row into the CHANGE_WO_STATUS transaction data table. |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

See Also

[ChangeWOStatus Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus.NewInputRow Method

Inserts a new row into the CHANGE_WO_STATUS transaction data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewInputRow()
```

VB

```
Public Overridable Function NewInputRow As DataRow
```

Return Value

Type: **DataRow**

See Also

[ChangeWOStatus Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus.Prepare Method

Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Prepare()
```

VB

```
Public Overridable Sub Prepare
```

See Also

[ChangeWOStatus Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWOStatus.Save Method

Saves the transaction(s).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[ChangeWOStatus Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ChangeWorkOrderStatus

This transaction allows for the change of an existing work order's status. The status change may be just applied to the key supplied, or may be applied to the specified key and all of its subordinates in the structure.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

DataSet name returned from Prepare: CHANGE_WO_STATUS

Primary Key: ENTRY_NO

| Column Name | Type | Description |
|--------------------|---------|---|
| ENTRY_NO | Integer | Uniquely numbers each transaction being provided to the set. |
| WORKORDER_TYPE | String | Work Order Type of the object being changed. |
| WORKORDER_BASE_ID | String | Work Order Base ID of the object being changed. |
| WORKORDER_LOT_ID | String | Work Order Lot ID of the object being changed. |
| WORKORDER_SPLIT_ID | String | Work Order Split ID of the object being changed. |
| WORKORDER_SUB_ID | String | Work Order Sub ID (leg/detail) of the object being changed. |
| OPERATION_SEQ_NO | Integer | Operation sequence number of the work order being changed. Only applicable if the object being changed is an Operation or Requirement. |
| PIECE_NO | Integer | Piece number of the work order being changed. Only applicable if the object being changed is a Requirement. |
| NEW_STATUS | String | The value for the new status. Valid values are U,F,R,C, or X. |
| EXPLODE | Boolean | Boolean flag (true or false) to signify whether the status change is propagated to all children of the specified work order key. Default value is true. |
| SITE_ID | String | Site ID of the parts that are to be updated after the status change. Required. |

See Also

[ChangeWOStatus Class](#)

[ChangeWOStatus.NewInputRow](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder Class

Transaction to copy Work Orders.

Inheritance Hierarchy

[System.Object](#)
[System.MarshalByRefObject](#)
[System.ComponentModel.Component](#)
BusinessObject
BusinessTransaction
 Lsa.Vmfg.ShopFloor.CopyWorkOrder

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#



```
[SerializableAttribute]
public class CopyWorkOrder : BusinessTransaction
```

VB




```
<SerializableAttribute>
Public Class CopyWorkOrder
    Inherits BusinessTransaction
```

The **CopyWorkOrder** type exposes the following members.

Constructors

| | Name | Description |
|---|---------------------------------------|----------------------------------|
|  | CopyWorkOrder() | Business Transaction Constructor |
|  | CopyWorkOrder(String) | Business Transaction Constructor |

Methods

| | Name | Description |
|---|-----------------------------|--|
|  | NewInputRow | Inserts a new row into the COPY_WORK_ORDER transaction data table. See CopyWorkOrder . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s) |

Transaction


| | Name | Data set returned by Prepare | Description |
|--|-------------------------------|------------------------------|--|
| | CopyWorkOrder | COPY_WORK_ORDER | This transaction will copy an existing Work Order structure to a new Work Order. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder Constructor

Overload List

| | Name | Description |
|---|---------------------------------------|----------------------------------|
|  | CopyWorkOrder() | Business Transaction Constructor |
|  | CopyWorkOrder(String) | Business Transaction Constructor |

See Also

[CopyWorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder Constructor

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public CopyWorkOrder()
```

VB

```
Public Sub New
```

See Also

[CopyWorkOrder Class](#)

[CopyWorkOrder Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder Constructor (String)

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public CopyWorkOrder(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[CopyWorkOrder Class](#)




[CopyWorkOrder Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder.CopyWorkOrder Methods

The [CopyWorkOrder](#) type exposes the following members.

Methods

| | Name | Description |
|---|-----------------------------|--|
|  | NewInputRow | Inserts a new row into the COPY_WORK_ORDER transaction data table. |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s) |

See Also

[CopyWorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder.NewInputRow Method

Inserts a new row into the COPY_WORK_ORDER transaction data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewInputRow()
```

VB

```
Public Overridable Function NewInputRow As DataRow
```

Return Value

Type: **DataRow**

See Also

[CopyWorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder.Prepare Method

Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Prepare()
```

VB

```
Public Overridable Sub Prepare
```

See Also

[CopyWorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder.Save Method

Saves the transaction(s)

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[CopyWorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CopyWorkOrder

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

DataSet name returned from Prepare: COPY_WORK_ORDER

Primary Key: ENTRY_NO

| Column Name | Type | Description |
|-------------------|---------|---|
| ENTRY_NO | Integer | Unique identifier for this transaction entry. |
| SOURCE_TYPE | String | Work order type of the source document. Valid values are "W", "Q", and "M". |
| SOURCE_BASE_ID | String | The Base ID of the source document. |
| SOURCE_LOT_ID | String | The Lot ID of the source document. |
| SOURCE_SPLIT_ID | String | The Split ID of the source document. |
| TARGET_TYPE | String | Work order type of the target document. Valid values are "W", "Q", and "M". |
| TARGET_BASE_ID | String | The Base ID of the target document. |
| TARGET_LOT_ID | String | The Lot ID of the target document. |
| TARGET_SPLIT_ID | String | The Split ID of the target document. |
| TARGET_STATUS | String | The status for the target document. Default value is "U" (unreleased). |
| DESIRED_QTY | Decimal | The desired quantity for the resultant work order. Default value is 1. |
| DESIRED_RLS_DATE | Date | The desired release date for the target work order. Default value is the current date. |
| WANT_DATE | Date | The want date for the target work order. Default value is the current date. |
| HARD_RELEASE_DATE | String | Setting to determine if target work order will have a hard release date. Valid values are "Y" or "N". Default value is "N". |
| FORWARD_SCHEDULE | String | Setting to determine if target work order will be forward scheduled. Valid values are "Y" or "N". Default value is "N". |
| DRAWING_ID | String | The ID of the drawing that depicts the part. |
| DRAWING_REV_NO | String | The revision ID of the drawing that depicts the part. |

| Column Name | Type | Description |
|----------------|--------|---|
| WAREHOUSE_ID | String | ID of the warehouse that receives the finished goods from the work order. |
| PRODUCT_CODE | String | Product code associated with the finished good. |
| COMMODITY_CODE | String | Commodity code associated with the finished good. |

See Also

[CopyWorkOrder Class](#)

[CopyWorkOrder.NewInputRow](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory Class

Maintain Cost Categories.

Inheritance Hierarchy

[System.Object](#)

[System.MarshalByRefObject](#)

[System.ComponentModel.Component](#)

BusinessObject

BusinessDocument

Lsa.Vmfg.ShopFloor.CostCategory

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public class CostCategory : BusinessDocument
```

VB









```
Public Class CostCategory  
    Inherits BusinessDocument
```

The **CostCategory** type exposes the following members.

Constructors

| | Name | Description |
|---|--------------------------------------|-------------|
|  | CostCategory() | Constructor |
|  | CostCategory(String) | Constructor |

Methods

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Cost Categories based on search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Cost Categories based on search criteria, row count limited by maxRecords. |
|  | Exists | Determine if a specific Cost Category exists. |
|  | Find | Retrieve a specific Cost Category. |
|  | Load() | Load all Cost Categories. |
|  | Load(String) | Load a specific Cost Category. |
|  | NewCostCategoryRow | Add a new COST_CATEGORY Row. |
|  | Save | Save all previously loaded Cost Categories to the database. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory Constructor

Overload List

| | Name | Description |
|---|--------------------------------------|-------------|
|  | CostCategory() | Constructor |
|  | CostCategory(String) | Constructor |

See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory Constructor

Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public CostCategory()
```

VB

```
Public Sub New
```

See Also

[CostCategory Class](#)

[CostCategory Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory Constructor (String)

Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public CostCategory(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[CostCategory Class](#)









[CostCategory Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.CostCategory Methods

The [CostCategory](#) type exposes the following members.

Methods

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Cost Categories based on search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Cost Categories based on search criteria, row count limited by maxRecords. |
|  | Exists | Determine if a specific Cost Category exists. |
|  | Find | Retrieve a specific Cost Category. |
|  | Load() | Load all Cost Categories. |
|  | Load(String) | Load a specific Cost Category. |
|  | NewCostCategoryRow | Add a new COST_CATEGORY Row. |
|  | Save | Save all previously loaded Cost Categories to the database. |



See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Browse Method

Overload List

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Cost Categories based on search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Cost Categories based on search criteria, row count limited by maxRecords. |

See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Browse Method (String, String, String)

Retrieve Cost Categories based on search criteria.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

Return Value

Type: **DataSet**

See Also

[CostCategory Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Browse Method (String, String, String, Int32, Int32)

Retrieve Cost Categories based on search criteria, row count limited by maxRecords.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns,  
    int startRecord,  
    int maxRecords  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String,  
    startRecord As Integer,  
    maxRecords As Integer  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

startRecord

Type: [System.Int32](#)

maxRecords

Type: [System.Int32](#)

Return Value

Type: **DataSet**

See Also

[CostCategory Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Exists Method

Determine if a specific Cost Category exists.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual bool Exists(  
    string costCategoryID  
)
```

VB

```
Public Overridable Function Exists (  
    costCategoryID As String  
) As Boolean
```

Parameters

costCategoryID

Type: [System.String](#)

Return Value

Type: [Boolean](#)

See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Find Method

Retrieve a specific Cost Category.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Find(  
    string costCategoryID  
)
```

VB

```
Public Overridable Sub Find (  
    costCategoryID As String  
)
```

Parameters

costCategoryID

Type: [System.String](#)



See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Load Method

Overload List

| | Name | Description |
|---|------------------------------|--------------------------------|
|  | Load() | Load all Cost Categories. |
|  | Load(String) | Load a specific Cost Category. |

See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Load Method

Load all Cost Categories.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load()
```

VB

```
Public Overridable Sub Load
```

See Also

[CostCategory Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Load Method (String)

Load a specific Cost Category.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load(  
    string costCategoryID  
)
```

VB

```
Public Overridable Sub Load (  
    costCategoryID As String  
)
```

Parameters

costCategoryID

Type: [System.String](#)

See Also

[CostCategory Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.NewCostCategoryRow Method

Add a new COST_CATEGORY Row.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewCostCategoryRow(  
    string costCategoryID  
)
```

VB

```
Public Overridable Function NewCostCategoryRow (  
    costCategoryID As String  
) As DataRow
```

Parameters

costCategoryID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostCategory.Save Method

Save all previously loaded Cost Categories to the database.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[CostCategory Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup Class

Maintain Cost Groups.

Inheritance Hierarchy

[System.Object](#)
[System.MarshalByRefObject](#)
[System.ComponentModel.Component](#)
BusinessObject
BusinessDocument
Lsa.Vmfg.ShopFloor.CostGroup

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public class CostGroup : BusinessDocument
```

VB









```
Public Class CostGroup  
    Inherits BusinessDocument
```

The **CostGroup** type exposes the following members.

Constructors

| | Name | Description |
|---|-----------------------------------|-------------|
|  | CostGroup() | Constructor |
|  | CostGroup(String) | Constructor |

Methods

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Cost Groups based on a search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Cost Groups based on a search criteria, row count limited by maxRecords. |
|  | Exists | Determine if a specific Cost Group ID exists. |
|  | Find | Find a specific Cost Group ID. |
|  | Load() | Load all Cost Group IDs |
|  | Load(String) | Load a specific Cost Group ID |
|  | NewCostGroupRow | Add a new COST_GROUP Row. |
|  | Save | Save all previously loaded Cost Groups to the database. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup Constructor

Overload List

| | Name | Description |
|---|-----------------------------------|-------------|
|  | CostGroup() | Constructor |
|  | CostGroup(String) | Constructor |

See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup Constructor

Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public CostGroup()
```

VB

```
Public Sub New
```

See Also

[CostGroup Class](#)

[CostGroup Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup Constructor (String)

Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public CostGroup(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[CostGroup Class](#)









[CostGroup Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.CostGroup Methods

The [CostGroup](#) type exposes the following members.

Methods

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Cost Groups based on a search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Cost Groups based on a search criteria, row count limited by maxRecords. |
|  | Exists | Determine if a specific Cost Group ID exists. |
|  | Find | Find a specific Cost Group ID. |
|  | Load() | Load all Cost Group IDs |
|  | Load(String) | Load a specific Cost Group ID |
|  | NewCostGroupRow | Add a new COST_GROUP Row. |
|  | Save | Save all previously loaded Cost Groups to the database. |



See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Browse Method

Overload List

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Cost Groups based on a search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Cost Groups based on a search criteria, row count limited by maxRecords. |

See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Browse Method (String, String, String)

Retrieve Cost Groups based on a search criteria.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

Return Value

Type: **DataSet**

See Also

[CostGroup Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Browse Method (String, String, String, Int32, Int32)

Retrieve Cost Groups based on a search criteria, row count limited by maxRecords.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns,  
    int startRecord,  
    int maxRecords  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String,  
    startRecord As Integer,  
    maxRecords As Integer  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

startRecord

Type: [System.Int32](#)

maxRecords

Type: [System.Int32](#)

Return Value

Type: **DataSet**

See Also

[CostGroup Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Exists Method

Determine if a specific Cost Group ID exists.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual bool Exists(  
    string costGroupID  
)
```

VB

```
Public Overridable Function Exists (  
    costGroupID As String  
) As Boolean
```

Parameters

costGroupID

Type: [System.String](#)

Return Value

Type: [Boolean](#)

See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Find Method

Find a specific Cost Group ID.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Find(  
    string costGroupID  
)
```

VB

```
Public Overridable Sub Find (  
    costGroupID As String  
)
```

Parameters

costGroupID

Type: [System.String](#)



See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Load Method

Overload List

| | Name | Description |
|---|------------------------------|-------------------------------|
|  | Load() | Load all Cost Group IDs |
|  | Load(String) | Load a specific Cost Group ID |

See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Load Method

Load all Cost Group IDs

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load()
```

VB

```
Public Overridable Sub Load
```

See Also

[CostGroup Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Load Method (String)

Load a specific Cost Group ID

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load(  
    string costGroupID  
)
```

VB

```
Public Overridable Sub Load (  
    costGroupID As String  
)
```

Parameters

costGroupID

Type: [System.String](#)

See Also

[CostGroup Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.NewCostGroupRow Method

Add a new COST_GROUP Row.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewCostGroupRow(  
    string costGroupID  
)
```

VB

```
Public Overridable Function NewCostGroupRow (  
    costGroupID As String  
) As DataRow
```

Parameters

costGroupID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

CostGroup.Save Method

Save all previously loaded Cost Groups to the database.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[CostGroup Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket Class

Transaction for Deleting a Labor Ticket. Note: Posted Labor Tickets cannot be deleted.

Inheritance Hierarchy

[System.Object](#)

[System.MarshalByRefObject](#)

[System.ComponentModel.Component](#)

BusinessObject

BusinessTransaction

Lsa.Vmfg.ShopFloor.DeleteLaborTicket

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#



```
[SerializableAttribute]  
public class DeleteLaborTicket : BusinessTransaction
```

VB




```
<SerializableAttribute>  
Public Class DeleteLaborTicket  
    Inherits BusinessTransaction
```

The **DeleteLaborTicket** type exposes the following members.

Constructors

| | Name | Description |
|---|---|----------------------------------|
|  | DeleteLaborTicket() | Business Transaction Constructor |
|  | DeleteLaborTicket(String) | Business Transaction Constructor |

Methods

| | Name | Description |
|---|-----------------------------------|--|
|  | NewDeleteLaborRow | Inserts a new row into the DELETE_LABOR transaction data table. See DeleteLabor . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

Transaction



| | Name | Data set returned by Prepare | Description |
|--|-----------------------------|------------------------------|---|
| | DeleteLabor | DELETE_LABOR | This transaction deletes a Labor Ticket. An option is available to re-open the operation associated with the deleted ticket or leave it closed. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket Constructor

Overload List

| | Name | Description |
|---|---|----------------------------------|
|  | DeleteLaborTicket() | Business Transaction Constructor |
|  | DeleteLaborTicket(String) | Business Transaction Constructor |

See Also

[DeleteLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket Constructor

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public DeleteLaborTicket()
```

VB

```
Public Sub New
```

See Also

[DeleteLaborTicket Class](#)

[DeleteLaborTicket Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket Constructor (String)

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public DeleteLaborTicket(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[DeleteLaborTicket Class](#)




[DeleteLaborTicket Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket.DeleteLaborTicket Methods

The [DeleteLaborTicket](#) type exposes the following members.

Methods

| | Name | Description |
|---|-----------------------------------|--|
|  | NewDeleteLaborRow | Inserts a new row into the DELETE_LABOR transaction data table. See DeleteLabor . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

See Also

[DeleteLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket.NewDeleteLaborRow Method

Inserts a new row into the DELETE_LABOR transaction data table.

See [DeleteLabor](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewDeleteLaborRow(  
    int transactionID  
)
```

VB

```
Public Overridable Function NewDeleteLaborRow (  
    transactionID As Integer  
) As DataRow
```

Parameters

transactionID

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[DeleteLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket.Prepare Method

Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Prepare()
```

VB

```
Public Overridable Sub Prepare
```

See Also

[DeleteLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLaborTicket.Save Method

Saves the transaction(s).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[DeleteLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

DeleteLabor

This transaction deletes a Labor Ticket. An option is available to re-open the operation associated with the deleted ticket or leave it closed.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

DataSet name returned from Prepare: DELETE_LABOR

Primary Key: TRANSACTION_ID

| Column Name | Type | Description |
|------------------|---------|---|
| TRANSACTION_ID | Integer | Unique integer value. Required. |
| REOPEN_OPERATION | Boolean | Determines whether or not to reopen the associated operation of the deleted labor ticket. Valid values are "true" or "false". |

See Also

[DeleteLaborTicket Class](#)

[DeleteLaborTicket.NewDeleteLaborRow](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket Class

Transaction for editing Labor Tickets.

Inheritance Hierarchy

[System.Object](#)

[System.MarshalByRefObject](#)

[System.ComponentModel.Component](#)

BusinessObject

BusinessTransaction

Lsa.Vmfg.ShopFloor.EditLaborTicket

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#



```
[SerializableAttribute]  
public class EditLaborTicket : BusinessTransaction
```

VB





```
<SerializableAttribute>  
Public Class EditLaborTicket  
    Inherits BusinessTransaction
```

The **EditLaborTicket** type exposes the following members.

Constructors

| | Name | Description |
|---|---|----------------------------------|
|  | EditLaborTicket() | Business Transaction Constructor |
|  | EditLaborTicket(String) | Business Transaction Constructor |

Methods

| | Name | Description |
|---|---------------------------------|--|
|  | NewEditLaborRow | Inserts a new row into the EDIT_LABOR transaction data table. See EditLabor . |
|  | NewTraceRow | Inserts a new row into the TRACE transaction data table. See Edit Labor . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

Transaction



| | Name | Data set returned by Prepare | Description |
|--|---------------------------|------------------------------|--|
| | EditLabor | EDIT_LABOR | This transaction edits a Labor Ticket. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket Constructor

Overload List

| | Name | Description |
|---|---|----------------------------------|
|  | EditLaborTicket() | Business Transaction Constructor |
|  | EditLaborTicket(String) | Business Transaction Constructor |

See Also

[EditLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket Constructor

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public EditLaborTicket()
```

VB

```
Public Sub New
```

See Also

[EditLaborTicket Class](#)

[EditLaborTicket Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket Constructor (String)

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public EditLaborTicket(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[EditLaborTicket Class](#)





[EditLaborTicket Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket.EditLaborTicket Methods

The [EditLaborTicket](#) type exposes the following members.

Methods

| | Name | Description |
|---|---------------------------------|--|
|  | NewEditLaborRow | Inserts a new row into the EDIT_LABOR transaction data table. See EditLabor . |
|  | NewTraceRow | Inserts a new row into the TRACE transaction data table. See EditLabor . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

See Also

[EditLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket.NewEditLaborRow Method

Inserts a new row into the EDIT_LABOR transaction data table.

See [EditLabor](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewEditLaborRow(  
    int transactionID  
)
```

VB

```
Public Overridable Function NewEditLaborRow (  
    transactionID As Integer  
) As DataRow
```

Parameters

transactionID

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[EditLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket.NewTraceRow Method

Inserts a new row into the TRACE transaction data table.

See [EditLabor](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewTraceRow(  
    int transactionID,  
    string traceID  
)
```

VB

```
Public Overridable Function NewTraceRow (  
    transactionID As Integer,  
    traceID As String  
) As DataRow
```

Parameters

transactionID

Type: [System.Int32](#)

traceID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[EditLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket.Prepare Method

Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Prepare()
```

VB

```
Public Overridable Sub Prepare
```

See Also

[EditLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

EditLaborTicket.Save Method

Saves the transaction(s).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[EditLaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

Edit Labor

This transaction edits a Labor Ticket.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

DataSet name returned from Prepare: EDIT_LABOR

Primary Key: ENTRY_NO

| Column Name | Type | Description |
|------------------|-----------|--|
| TRANSACTION_ID | Integer | Unique integer value. Required. |
| TRANSACTION_DATE | String | Date the Transaction occurred on. |
| DEPARTMENT_ID | String | Department ID work was performed in. Defaults to value specified in the employee table. |
| EARNING_CODE_ID | String | Earning code of employee. Defaults to value specified in the employee table. |
| CLOCK_IN_TIME | Date/Time | Clock in time of day. Required for all transactions. |
| CLOCK_OUT_TIME | Date/Time | Clock out time of day. Required for all transactions. |
| HOURS_WORKED | Decimal | Hours worked. This is the time span represented by the difference between clock out and clock in minus the time in break hours. Required for all transactions. |
| BREAK_HOURS | Decimal | Hours on break. This value is used to calculate the total Hours Worked. Optional. |
| DESCRIPTION | String | Description of the labor transaction. Optional. |
| GOOD_QTY | Decimal | Quantity successfully produced for this ticket. |
| BAD_QTY | Decimal | Quantity deviated (scrap) for this ticket. Defaults to 0. Not applicable for indirect transactions. |
| HOURLY_COST | Decimal | Hourly Cost. Override of cost per hour for setup. Defaults to employee base pay rate if not provided. |

| Column Name | Type | Description |
|-------------------|-----------|--|
| UNIT_COST | String | Unit Cost. Used to override the per unit cost. Optional. Defaults to 0 for direct and indirect transactions. |
| INDIRECT_ID | String | Indirect ID. Applicable only for indirect labor. Required if creating an indirect labor transaction. Must match an existing indirect ID. |
| MULTIPLIER_1 | Decimal | Used for overtime purposes. Specify a value greater than 1 for overtime, otherwise specify 1. Defaults to 1. |
| MULTIPLIER_2 | Decimal | Used for overtime purposes. Specify a value greater than 1 for overtime, otherwise specify 1. Defaults to 1. |
| REOPEN_OPERATION | Boolean | Boolean indicating that the operation has been reopened. |
| RUN_COMPLETE | Boolean | Boolean indicating that the run phase of the operation is now complete. |
| SETUP_COMPLETED | Boolean | Boolean indicating that the setup phase of the operation is now complete. |
| DEVIATION_ID | String | Indicates the reason for any bad quantities |
| UNADJ_CLOCK_IN | Date/Time | The date and time the employee actually clocked in. |
| UNADJ_CLOCK_OUT | Date/Time | The date and time the employee actually clocked out. |
| GL_ACCT_ID | String | G/L Account ID. Applicable only for indirect labor. Defaults to Indirect ID's G/L Account if not provided. Must match an existing Account ID. |
| INDIRECT_CODE | String | Indirect Code that defines the type of indirect labor. Used when creating an indirect labor transaction. Optional. Defaults to indirect code of Indirect ID if not provided. |
| POSTING_CANDIDATE | Boolean | Indicates that the labor ticket can be posted. |
| IN_PROCESS_TICKET | Boolean | Indicates that the labor ticket is in process. |

| Column Name | Type | Description |
|------------------------------|---------|--|
| TRACE_REQUIRED | Boolean | Indicates that trace information must be specified. |
| PART_ID | String | The Part ID of the item. Not applicable for Operations. |
| START_IN_PROCESS_TICKET | Boolean | Used to indicate a clocked-in employee who will clock out later to complete an operation. The clock in and clock out times must be the same, and Hours Worked must be blank (not zero). |
| UPDATE_HRS_WRKED_WITH_BREAKS | Boolean | If TRUE, the computed HOURS_WORKED of the labor ticket will be reduced by the unpaid break hours from the input row's BREAK_HOURS value. The default is FALSE |
| PRORATE_TYPE | String | Proration Type. Required for prorated labor transactions. |
| SHIFT_DATE | Date | Date at start of shift. May disagree with transaction date. Defaults to transaction date. |
| BREAK_HOURS_UNPRORATED | Decimal | This is a decimal value containing the total unpaid, un-prorated break hours for the labor ticket. This value only applies when UPDATE_HRS_WRKED_WITH_BREAKS is TRUE and the program determines that the labor ticket requires proration. The computed prorated hours will be reduced by this amount. |
| SITE_ID | String | Site ID for this labor ticket transaction. Required. |
| ENTITY_ID | String | Entity ID for this labor ticket transaction. Required. |

Sub-Table Name: TRACE:

Primary Key: ENTRY_NO, TRACE_ID

Trace information may be required, depending on the trace profile of the part being transacted. The Trace sub-table is never applicable for SETUP and INDIRECT transaction types.

| Column Name | Type | Description |
|--------------------|---------|---|
| TRANSACTION_ID | Integer | Determines which labor transaction this row of trace information belongs to. |
| TRACE_ID | String | Trace ID. Lot or serial number for the parts being reported. If the part's trace profile supports auto numbering, and you wish to have the Trace Ids auto numbered, you must set the TRACE_ID values to the format "<n>" where n is a unique integer. |
| ALPHA_PROPERTY_1 | String | Alphanumeric property. May be required, depending on the trace profile. This is true for all ALPHA_PROPERTY and NUMERIC_PROPERTY fields. |
| ALPHA_PROPERTY_2 | String | Alphanumeric property. May be required. |
| ALPHA_PROPERTY_3 | String | Alphanumeric property. May be required. |
| ALPHA_PROPERTY_4 | String | Alphanumeric property. May be required. |
| ALPHA_PROPERTY_5 | String | Alphanumeric property. May be required. |
| NUMERIC_PROPERTY_1 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_2 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_3 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_4 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_5 | Decimal | Numeric property. May be required. |
| COMMENTS | String | Optional user comments on specific lot or serial number. |
| EXPIRATION_DATE | Date | Expiration date. Determines shelf life of lot. Optional. |
| QTY | Decimal | Quantity of transaction associated directly with this trace ID. |
| UNAVAILABLE_QTY | Decimal | |

See Also

[EditLaborTicket Class](#)

[EditLaborTicket.NewEditLaborRow](#)

[EditLaborTicket.NewTraceRow](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary Class

Service to obtain a single data table containing summary information for a Work Order.

Inheritance Hierarchy

[System.Object](#)
[System.MarshalByRefObject](#)
[System.ComponentModel.Component](#)
BusinessObject
BusinessService
Lsa.Vmfg.ShopFloor.GetWorkOrderSummary

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#



```
[SerializableAttribute]  
public class GetWorkOrderSummary : BusinessService
```

VB




```
<SerializableAttribute>  
Public Class GetWorkOrderSummary  
    Inherits BusinessService
```

The **GetWorkOrderSummary** type exposes the following members.

Constructors

| | Name | Description |
|---|---|--|
|  | GetWorkOrderSummary() | Service to populate a DataTable with Work Order Summary information. |
|  | GetWorkOrderSummary(String) | Constructor. |

Methods

| | Name | Description |
|---|-----------------------------|--|
|  | Execute | Executes the service |
|  | NewInputRow | Add new input request row for the service. |
|  | Prepare | Prepares the service |

Data Tables



| | Table Type | Table Name |
|--|-------------------|------------------------|
| | Header Table | GET_WORK_ORDER_SUMMARY |
| | Results Sub-table | WORK_ORDER_SUMMARY |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary Constructor

Overload List

| | Name | Description |
|---|---|--|
|  | GetWorkOrderSummary() | Service to populate a DataTable with Work Order Summary information. |
|  | GetWorkOrderSummary(String) | Constructor. |

See Also

[GetWorkOrderSummary Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary Constructor

Service to populate a DataTable with Work Order Summary information.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public GetWorkOrderSummary()
```

VB

```
Public Sub New
```

See Also

[GetWorkOrderSummary Class](#)

[GetWorkOrderSummary Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary Constructor (String)

Constructor.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public GetWorkOrderSummary(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[GetWorkOrderSummary Class](#)




[GetWorkOrderSummary Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary.GetWorkOrderSummary Methods

The [GetWorkOrderSummary](#) type exposes the following members.

Methods

| | Name | Description |
|---|-----------------------------|--|
|  | Execute | Executes the service |
|  | NewInputRow | Add new input request row for the service. |
|  | Prepare | Prepares the service |

See Also

[GetWorkOrderSummary Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary.Execute Method

Executes the service

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Execute()
```

VB

```
Public Overridable Sub Execute
```

See Also

[GetWorkOrderSummary Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary.NewInputRow Method

Add new input request row for the service.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewInputRow(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID  
)
```

VB

```
Public Overridable Function NewInputRow (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String  
) As DataRow
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[GetWorkOrderSummary Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary.Prepare Method

Prepares the service

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Prepare()
```

VB

```
Public Overridable Sub Prepare
```

See Also

[GetWorkOrderSummary Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

GetWorkOrderSummary Data Tables

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

DataSet Name returned from Prepare: WORK_ORDER_SUMMARY

Header Table

Table Name: GET_WORK_ORDER_SUMMARY

Primary Key: TYPE, BASE_ID, LOT_ID, SPLIT_ID

| Column Name | Type | Description |
|-------------|--------|--|
| TYPE | String | Work Order Type. "W", "M", or "Q". Required. |
| BASE_ID | String | Work Order Base ID. Required. |
| LOT_ID | String | Work Order Lot ID. Required. |
| SPLIT_ID | String | Work Order Split ID. Required. |

Results Sub-table

Table Name: WORK_ORDER_SUMMARY

| Column Name | Type | Description |
|--------------|---------|---|
| DETAIL | String | A summary of the key data for the work order item. The value and format varies based on the type of row. For example, for an operation the format would be BASE_ID / LOT ID / SEQUENCE_NUMBER. Each Detail item is padded with spaces. The number of spaces is determined by that item's position in the work order hierarchy. |
| RECORD_TYPE | String | Either "Header", "Leg" "Operation" or "Material". |
| PCT_COMPLETE | Decimal | The percentage completed for the item. |
| CLOSE_DATE | Date | The CLOSED_DATE of the item. |

| Column Name | Type | Description |
|------------------|---------|--|
| PART_ID | String | The Part ID of the item. Not applicable for Operations. |
| PART_DESC | String | The Part Description of the item. Not applicable for Operations. |
| RESOURCE_ID | String | The Resource ID of the item. Only applies to Operations. |
| RESOURCE_DESC | String | The Resource Description of the item. Only applies to Operations. |
| TYPE | String | The type of the work order. "W", "M", or "Q". Part of the primary key of the current row. |
| BASE_ID | String | The work order base ID. Part of the primary key of the current row. |
| LOT_ID | String | The work order lot ID. Part of the primary key of the current row. |
| SPLIT_ID | String | The work order split ID. Part of the primary key of the current row. |
| SUB_ID | String | The work order sub ID. Part of the primary key of the current row. |
| OPERATION_SEQ_NO | Integer | The operation sequence number. Part of the primary key of the current row. Only applies to Operations and Materials. |
| PIECE_NO | Integer | The piece number. Part of the primary key of the current row. Only applies to Materials. |
| SUBOR_WO_SUB_ID | String | The SUB_ID of the part Work Order header record. Only applies to Leg Materials. |
| ROW_NO | Integer | A unique integer value for the row. |

See Also

[GetWorkOrderSummary Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket Class

Transaction for creating a Labor Ticket. Three types of transactions are supported: Setup, Run, and Indirect.

Inheritance Hierarchy

[System.Object](#)
[System.MarshalByRefObject](#)
[System.ComponentModel.Component](#)
BusinessObject
BusinessTransaction
Lsa.Vmfg.ShopFloor.LaborTicket

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#



```
[SerializableAttribute]  
public class LaborTicket : BusinessTransaction
```

VB







```
<SerializableAttribute>  
Public Class LaborTicket  
    Inherits BusinessTransaction
```

The **LaborTicket** type exposes the following members.

Constructors

| | Name | Description |
|---|-------------------------------------|----------------------------------|
|  | LaborTicket() | Business Transaction Constructor |
|  | LaborTicket(String) | Business Transaction Constructor |

Methods

| Name | Description |
|---|---|
|  NewIndirectLaborRow | Inserts a new row into the LABOR transaction data table for indirect transactions. See LaborTicket . |
|  NewRunLaborRow | Inserts a new row into the LABOR transaction data table for run transactions. See LaborTicket . |
|  NewSetupLaborRow | Inserts a new row into the LABOR transaction data table for setup transactions. See LaborTicket . |
|  NewTraceRow | Inserts a new row into the TRACE transaction data table for run transactions. See LaborTicket . |
|  Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  Save | Saves the transaction(s). |

Transaction



| Name | Data set returned by Prepare | Description |
|-----------------------------|------------------------------|--|
| LaborTicket | LABOR | This transaction creates a Labor Ticket. Three types of labor transactions are supported (Run, Setup, and Indirect). The type of transaction created depends on the value provided for the TRANSACTION_TYPE field. Valid TRANSACTION_TYPE values are RUN, SETUP, and INDIRECT. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket Constructor

Overload List

| | Name | Description |
|---|-------------------------------------|----------------------------------|
|  | LaborTicket() | Business Transaction Constructor |
|  | LaborTicket(String) | Business Transaction Constructor |

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket Constructor

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public LaborTicket()
```

VB

```
Public Sub New
```

See Also

[LaborTicket Class](#)

[LaborTicket Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket Constructor (String)

Business Transaction Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public LaborTicket(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[LaborTicket Class](#)







[LaborTicket Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.LaborTicket Methods

The [LaborTicket](#) type exposes the following members.

Methods

| | Name | Description |
|---|-------------------------------------|---|
|  | NewIndirectLaborRow | Inserts a new row into the LABOR transaction data table for indirect transactions. See LaborTicket . |
|  | NewRunLaborRow | Inserts a new row into the LABOR transaction data table for run transactions. See LaborTicket . |
|  | NewSetupLaborRow | Inserts a new row into the LABOR transaction data table for setup transactions. See LaborTicket . |
|  | NewTraceRow | Inserts a new row into the TRACE transaction data table for run transactions. See LaborTicket . |
|  | Prepare | Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction. |
|  | Save | Saves the transaction(s). |

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.NewIndirectLaborRow Method

Inserts a new row into the LABOR transaction data table for indirect transactions.

See [LaborTicket](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewIndirectLaborRow(  
    int entryNo  
)
```

VB

```
Public Overridable Function NewIndirectLaborRow (  
    entryNo As Integer  
) As DataRow
```

Parameters

entryNo

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.NewRunLaborRow Method

Inserts a new row into the LABOR transaction data table for run transactions.

See [LaborTicket](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewRunLaborRow(  
    int entryNo  
)
```

VB

```
Public Overridable Function NewRunLaborRow (  
    entryNo As Integer  
) As DataRow
```

Parameters

entryNo

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.NewSetupLaborRow Method

Inserts a new row into the LABOR transaction data table for setup transactions.

See [LaborTicket](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewSetupLaborRow(  
    int entryNo  
)
```

VB

```
Public Overridable Function NewSetupLaborRow (  
    entryNo As Integer  
) As DataRow
```

Parameters

entryNo

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.NewTraceRow Method

Inserts a new row into the TRACE transaction data table for run transactions.

See [LaborTicket](#).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewTraceRow(  
    int entryNo,  
    string traceID  
)
```

VB

```
Public Overridable Function NewTraceRow (  
    entryNo As Integer,  
    traceID As String  
) As DataRow
```

Parameters

entryNo

Type: [System.Int32](#)

traceID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.Prepare Method

Creates an empty dataset for the transaction. You must populate the dataset prior to saving the transaction.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Prepare()
```

VB

```
Public Overridable Sub Prepare
```

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket.Save Method

Saves the transaction(s).

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[LaborTicket Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

LaborTicket

This transaction creates a Labor Ticket. Three types of labor transactions are supported (Run, Setup, and Indirect). The type of transaction created depends on the value provided for the TRANSACTION_TYPE field. Valid TRANSACTION_TYPE values are RUN, SETUP, and INDIRECT.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

DataSet name returned from Prepare: LABOR

Primary Key: ENTRY_NO

| Column Name | Type | Description |
|------------------|-----------|---|
| ENTRY_NO | Integer | Unique integer value. Required. |
| TRANSACTION_TYPE | String | Defines which type of labor transaction to perform. Valid values are RUN, SETUP, and INDIRECT. |
| BASE_ID | String | Work order base ID. Not applicable for indirect transactions. |
| LOT_ID | String | Work order lot ID. Not applicable for indirect transactions. |
| SPLIT_ID | String | Work order split ID. Not applicable for indirect transactions. |
| SUB_ID | String | Work order sub ID (leg/detail). Not applicable for indirect transactions. |
| SEQ_NO | Integer | Operation sequence number. Not applicable for indirect transactions. |
| EMPLOYEE_ID | String | Employee ID. Required for all transactions. |
| CLOCK_IN | Date/Time | Clock in time of day. Required for all transactions. |
| CLOCK_OUT | Date/Time | Clock out time of day. Required for all transactions. |
| DEVIATED_QTY | Decimal | Quantity deviated (scrap) for this ticket. Defaults to 0. Not applicable for indirect transactions. |

| Column Name | Type | Description |
|---------------|---------|--|
| HOURS_WORKED | Decimal | Hours worked. This is the time span represented by the difference between clock out and clock in minus the time in break hours. Required for all transactions. |
| HOURS_BREAK | Decimal | Hours on break. This value is used to calculate the total Hours Worked. Optional. |
| RESOURCE_ID | String | Resource ID of resource where actual work was performed. Optional. Defaults to resource of the operation being reported. Not applicable for indirect transactions. |
| DEPARTMENT_ID | String | Department ID work was performed in. Defaults to value specified in the employee table. |
| EARNING_CODE | String | Earning code of employee. Defaults to value specified in the employee table. |
| MULTIPLIER_1 | Decimal | Used for overtime purposes. Specify a value greater than 1 for overtime, otherwise specify 1. Defaults to 1. |
| MULTIPLIER_2 | Decimal | Used for proration purposes. Specify a value less than 1 for proration, otherwise specify 1. Defaults to 1. |
| SHIFT_DATE | Date | Date at start of shift. May disagree with transaction date. Defaults to transaction date. |
| BREAK_HOURS | Decimal | Hours on break during this transaction. |
| UNIT_COST | String | Unit Cost. Used to override the per unit cost. Optional. Defaults to 0 for direct and indirect transactions. |

| Column Name | Type | Description |
|-------------------------|---------|---|
| SETUP_COMPLETE | Boolean | Boolean indicating that the setup phase of the operation is now complete. |
| INDIRECT_ID | String | Indirect ID. Applicable only for indirect labor. Required if creating an indirect labor transaction. Must match an existing indirect ID. |
| INDIRECT_CODE | String | Indirect Code that defines the type of indirect labor. Used when creating an indirect labor transaction. Optional. Defaults to indirect code of Indirect ID if not provided. |
| USER_ID | String | The User ID of the person performing the transaction. Defaults to SYSADM. |
| GL_ACCT_ID | String | G/L Account ID. Applicable only for indirect labor. Defaults to Indirect ID's G/L Account if not provided. Must match an existing Account ID. |
| HOURLY_COST | Decimal | Hourly Cost. Override of cost per hour for setup. Defaults to employee base pay rate if not provided. |
| DESCRIPTION | String | Description of the labor transaction. Optional. |
| START_IN_PROCESS_TICKET | Boolean | Used to indicate a clocked-in employee who will clock out later to complete an operation. The clock in and clock out times must be the same, and Hours Worked must be blank (not zero). |
| PRORATE_ID | String | Proration ID. Required for prorated labor transactions. |
| PRORATE_TYPE | String | Proration Type. Required for prorated labor transactions. |

| Column Name | Type | Description |
|-----------------------|---------|---|
| AUTO_RPT_BACKFLUSH_LT | Boolean | Used to indicate an Auto-Report Backflushed labor transaction. Optional, but may be required if associated Shop Resource requires Auto-Reporting. |
| ALT_EMP_BASE_PAY_RATE | Decimal | Employee Base Pay Rate override. Used to override the employee's standard hourly rate. Optional. Defaults to employee's base pay rate if not provided or 0. |
| SITE_ID | String | Site ID for this labor ticket transaction. Required. |
| ENTITY_ID | String | Entity ID for this labor ticket transaction. Required. |

Sub-Table Name: TRACE:

Primary Key: ENTRY_NO, TRACE_ID

Trace information may be required, depending on the trace profile of the part being transacted. The Trace sub-table is never applicable for SETUP and INDIRECT transaction types.

| Column Name | Type | Description |
|--------------------|---------|---|
| ENTRY_NO | Integer | Determines which labor transaction this row of trace information belongs to. |
| TRACE_ID | String | Trace ID. Lot or serial number for the parts being reported. If the part's trace profile supports auto numbering, and you wish to have the Trace Ids auto numbered, you must set the TRACE_ID values to the format "<n>" where n is a unique integer. |
| ALPHA_PROPERTY_1 | String | Alphanumeric property. May be required, depending on the trace profile. This is true for all ALPHA_PROPERTY and NUMERIC_PROPERTY fields. |
| ALPHA_PROPERTY_2 | String | Alphanumeric property. May be required. |
| ALPHA_PROPERTY_3 | String | Alphanumeric property. May be required. |
| ALPHA_PROPERTY_4 | String | Alphanumeric property. May be required. |
| ALPHA_PROPERTY_5 | String | Alphanumeric property. May be required. |
| NUMERIC_PROPERTY_1 | Decimal | Numeric property. May be required. |

| Column Name | Type | Description |
|--------------------|---------|---|
| NUMERIC_PROPERTY_2 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_3 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_4 | Decimal | Numeric property. May be required. |
| NUMERIC_PROPERTY_5 | Decimal | Numeric property. May be required. |
| COMMENTS | String | Optional user comments on specific lot or serial number. |
| EXPIRATION_DATE | Date | Expiration date. Determines shelf life of lot. Optional. |
| QTY | Decimal | Quantity of transaction associated directly with this trace ID. |
| UNAVAILABLE_QTY | Decimal | |

See Also

[LaborTicket Class](#)

[LaborTicket.NewIndirectLaborRow](#)

[LaborTicket.NewRunLaborRow](#)

[LaborTicket.NewSetupLaborRow](#)

[LaborTicket.NewTraceRow](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource Class

Maintain Shop Resources.

Inheritance Hierarchy

[System.Object](#)
[System.MarshalByRefObject](#)
[System.ComponentModel.Component](#)
BusinessObject
BusinessDocument
 Lsa.Vmfg.ShopFloor.ShopResource

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
[SerializableAttribute]
public class ShopResource : BusinessDocument
```

VB











```
<SerializableAttribute>
Public Class ShopResource
    Inherits BusinessDocument
```

The **ShopResource** type exposes the following members.

Constructors

| | Name | Description |
|---|--------------------------------------|-------------------------------|
|  | ShopResource() | Business Documnet Constructor |
|  | ShopResource(String) | Business Document Constructor |

Methods



| | Name | Description |
|---|--|--|
|  | Browse(String, String, String) | Retrieve Shop Resources based on search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Shop Resources based on search criteria, limited by record count. |
|  | Exists | Determine if a Shop Resource exists. |
|  | Find | Retrieve a specific Shop Resource. Only the top-level table (SHOP_RESOURCE) is returned. |
|  | Load() | Load all Shop Resources. |
|  | Load(String) | Load a specific Shop Resource. |
|  | Load(Stream, String) | Load from stream and rename using new key. |
|  | NewShopResourceRow | Add a new row to the SHOP_RESOURCE table. |
|  | NewShopResourceSiteRow | Add a new row to the SHOP_RESOURCE_SITE table. |
|  | Save | Save all previously loaded Shop Resources to the database. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource Constructor

Overload List

| | Name | Description |
|---|--------------------------------------|-------------------------------|
|  | ShopResource() | Business Documnet Constructor |
|  | ShopResource(String) | Business Document Constructor |

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource Constructor

Business Documnet Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public ShopResource()
```

VB

```
Public Sub New
```

See Also

[ShopResource Class](#)

[ShopResource Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource Constructor (String)

Business Document Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public ShopResource(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[ShopResource Class](#)











[ShopResource Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.ShopResource Methods

The [ShopResource](#) type exposes the following members.

Methods

| | Name | Description |
|---|--|--|
|  | Browse(String, String, String) | Retrieve Shop Resources based on search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Shop Resources based on search criteria, limited by record count. |
|  | Exists | Determine if a Shop Resource exists. |
|  | Find | Retrieve a specific Shop Resource. Only the top-level table (SHOP_RESOURCE) is returned. |
|  | Load() | Load all Shop Resources. |
|  | Load(String) | Load a specific Shop Resource. |
|  | Load(Stream, String) | Load from stream and rename using new key. |
|  | NewShopResourceRow | Add a new row to the SHOP_RESOURCE table. |
|  | NewShopResourceSiteRow | Add a new row to the SHOP_RESOURCE_SITE table. |
|  | Save | Save all previously loaded Shop Resources to the database. |

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Browse Method

Overload List

| | Name | Description |
|---|--|--|
|  | Browse(String, String, String) | Retrieve Shop Resources based on search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Shop Resources based on search criteria, limited by record count. |

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Browse Method (String, String, String)

Retrieve Shop Resources based on search criteria.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

Return Value

Type: **DataSet**

See Also

[ShopResource Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Browse Method (String, String, String, Int32, Int32)

Retrieve Shop Resources based on search criteria, limited by record count.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns,  
    int startRecord,  
    int maxRecords  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String,  
    startRecord As Integer,  
    maxRecords As Integer  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

startRecord

Type: [System.Int32](#)

maxRecords

Type: [System.Int32](#)

Return Value

Type: **DataSet**

See Also

[ShopResource Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Exists Method

Determine if a Shop Resource exists.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual bool Exists(  
    string shopResourceID  
)
```

VB

```
Public Overridable Function Exists (  
    shopResourceID As String  
) As Boolean
```

Parameters

shopResourceID

Type: [System.String](#)

Return Value

Type: [Boolean](#)

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Find Method

Retrieve a specific Shop Resource. Only the top-level table (SHOP_RESOURCE) is returned.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Find(  
    string shopResourceID  
)
```

VB

```
Public Overridable Sub Find (  
    shopResourceID As String  
)
```

Parameters

shopResourceID

Type: [System.String](#)




See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Load Method

Overload List

| | Name | Description |
|---|--------------------------------------|--|
|  | Load() | Load all Shop Resources. |
|  | Load(String) | Load a specific Shop Resource. |
|  | Load(Stream, String) | Load from stream and rename using new key. |

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Load Method

Load all Shop Resources.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load()
```

VB

```
Public Overridable Sub Load
```

See Also

[ShopResource Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Load Method (String)

Load a specific Shop Resource.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load(  
    string shopResourceID  
)
```

VB

```
Public Overridable Sub Load (  
    shopResourceID As String  
)
```

Parameters

shopResourceID

Type: [System.String](#)

See Also

[ShopResource Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Load Method (Stream, String)

Load from stream and rename using new key.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load(  
    Stream stream,  
    string shopResourceID  
)
```

VB

```
Public Overridable Sub Load (  
    stream As Stream,  
    shopResourceID As String  
)
```

Parameters

stream

Type: [System.IO.Stream](#)

shopResourceID

Type: [System.String](#)

See Also

[ShopResource Class](#)

[Load Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.NewShopResourceRow Method

Add a new row to the SHOP_RESOURCE table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public DataRow NewShopResourceRow(  
    string shopResourceID  
)
```

VB

```
Public Function NewShopResourceRow (  
    shopResourceID As String  
) As DataRow
```

Parameters

shopResourceID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.NewShopResourceSiteRow Method

Add a new row to the SHOP_RESOURCE_SITE table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public DataRow NewShopResourceSiteRow(  
    string shopResourceID,  
    string siteID  
)
```

VB

```
Public Function NewShopResourceSiteRow (  
    shopResourceID As String,  
    siteID As String  
) As DataRow
```

Parameters

shopResourceID

Type: [System.String](#)

siteID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

ShopResource.Save Method

Save all previously loaded Shop Resources to the database.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[ShopResource Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder Class

Maintain Work Orders.

Inheritance Hierarchy

[System.Object](#)
[System.MarshalByRefObject](#)
[System.ComponentModel.Component](#)
BusinessObject
BusinessDocument
 Lsa.Vmfg.ShopFloor.WorkOrder

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#



```
[SerializableAttribute]
public class WorkOrder : BusinessDocument
```

VB






```
<SerializableAttribute>
Public Class WorkOrder
    Inherits BusinessDocument
```







The **WorkOrder** type exposes the following members.

Constructors

| | Name | Description |
|---|-----------------------------------|-------------------------------|
|  | WorkOrder() | Business Document Constructor |
|  | WorkOrder(String) | Business Document Constructor |

Methods

| Name | Description |
|---|--|
|  Browse(String, String, String) | Retrieve Work Orders based on a search criteria. |
|  Browse(String, String, String, Int32, Int32) | Retrieve Work Orders based on a search criteria, row count limited by maxRecords. |
|  Exists | Determines if a specific Work Order exists. |
|  Find | Retrieves a specific Work Order. Only the top-level data table (WORK_ORDER) is returned. |
|  Load | Load a specific Work Order. |
|  NewCoProductRow | Inserts a new row into the CO_PRODUCT data table. |
|  NewOperationBinaryRow | Inserts a new row into the OPERATION_BINARY data table. Only binary type "D" (long text) is supported. |
|  NewOperationResourceRow | Inserts a new row into the OPERATION_RESOURCE data table. |
|  NewOperationRow(String, String, String, String, String) | Inserts a new row into the OPERATION data table, automatically assigning the next available sequence number. |
|  NewOperationRow(String, String, String, String, String, Int32) | Inserts a new row into the OPERATION data table. |
|  NewOperServiceCostRow | Inserts a new row into the OPER_SERVICE_COST data table. |
|  NewRequirementBinaryRow | Inserts a new row into the REQUIREMENT_BINARY data table. Only binary type "D" (long text) is supported. |
|  NewRequirementCostRow | Inserts a new row into the REQUIREMENT_COST data table. |
|  NewRequirementRow(String, String, String, String, String, Int32) | Inserts a new row into the REQUIREMENT data table, assigning the next available piece number. |
|  NewRequirementRow(String, String, String, String, String, Int32, Int32) | Inserts a new row into the REQUIREMENT data table. |
|  NewWorkOrderBinaryRow | Inserts a new row into the WORK_ORDER_BINARY data table. Only binary type "D" (long text) is supported. |



| | |
|--|---|
|  NewWorkOrderLegRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table, assigning the next available SUB_ID. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow(). |
|  NewWorkOrderLegRow(String, String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow(). |
|  NewWorkOrderMilestoneRow | Inserts a new row into the WORKORD_MILESTONE data table. |
|  NewWorkOrderRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkOrderLegRow(). |
|  NewWorkOrderRow(String, String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkOrderLegRow(). |
|  Save | Saves all previously loaded Work Orders to the database. |

See Also

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder Constructor

Overload List

| | Name | Description |
|---|-----------------------------------|-------------------------------|
|  | WorkOrder() | Business Document Constructor |
|  | WorkOrder(String) | Business Document Constructor |

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder Constructor

Business Document Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public WorkOrder()
```

VB

```
Public Sub New
```

See Also

[WorkOrder Class](#)

[WorkOrder Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder Constructor (String)

Business Document Constructor

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public WorkOrder(  
    string databaseInstanceName  
)
```

VB

```
Public Sub New (  
    databaseInstanceName As String  
)
```

Parameters

databaseInstanceName

Type: [System.String](#)

See Also

[WorkOrder Class](#)















[WorkOrder Overload](#)









[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.WorkOrder Methods

The [WorkOrder](#) type exposes the following members.

Methods

| Name | Description |
|--|--|
|  Browse(String, String, String) | Retrieve Work Orders based on a search criteria. |
|  Browse(String, String, String, Int32, Int32) | Retrieve Work Orders based on a search criteria, row count limited by maxRecords. |
|  Exists | Determines if a specific Work Order exists. |
|  Find | Retrieves a specific Work Order. Only the top-level data table (WORK_ORDER) is returned. |
|  Load | Load a specific Work Order. |
|  NewCoProductRow | Inserts a new row into the CO_PRODUCT data table. |
|  NewOperationBinaryRow | Inserts a new row into the OPERATION_BINARY data table. Only binary type "D" (long text) is supported. |
|  NewOperationResourceRow | Inserts a new row into the OPERATION_RESOURCE data table. |
|  NewOperationRow(String, String, String, String, String) | Inserts a new row into the OPERATION data table, automatically assigning the next available sequence number. |
|  NewOperationRow(String, String, String, String, String, Int32) | Inserts a new row into the OPERATION data table. |
|  NewOperServiceCostRow | Inserts a new row into the OPER_SERVICE_COST data table. |
|  NewRequirementBinaryRow | Inserts a new row into the REQUIREMENT_BINARY data table. Only binary type "D" (long text) is supported. |
|  NewRequirementCostRow | Inserts a new row into the REQUIREMENT_COST data table. |
|  NewRequirementRow(String, String, String, String, String, Int32) | Inserts a new row into the REQUIREMENT data table, assigning the next available piece number. |

| | |
|---|---|
|  NewRequirementRow(String, String, String, String, String, Int32, Int32) | Inserts a new row into the REQUIREMENT data table. |
|  NewWorkOrderBinaryRow | Inserts a new row into the WORK_ORDER_BINARY data table. Only binary type "D" (long text) is supported. |
|  NewWorkOrderLegRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table, assigning the next available SUB_ID. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow(). |
|  NewWorkOrderLegRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow(). |
|  NewWorkOrderMilestoneRow | Inserts a new row into the WORKORD_MILESTONE data table. |
|  NewWorkOrderRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkOrderLegRow(). |
|  NewWorkOrderRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkOrderLegRow(). |
|  Save | Saves all previously loaded Work Orders to the database. |



See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Browse Method

Overload List

| | Name | Description |
|---|--|---|
|  | Browse(String, String, String) | Retrieve Work Orders based on a search criteria. |
|  | Browse(String, String, String, Int32, Int32) | Retrieve Work Orders based on a search criteria, row count limited by maxRecords. |

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Browse Method (String, String, String)

Retrieve Work Orders based on a search criteria.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

Return Value

Type: **DataSet**

See Also

[WorkOrder Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Browse Method (String, String, String, Int32, Int32)

Retrieve Work Orders based on a search criteria, row count limited by maxRecords.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataSet Browse(  
    string columnNames,  
    string searchCondition,  
    string sortColumns,  
    int startRecord,  
    int maxRecords  
)
```

VB

```
Public Overridable Function Browse (  
    columnNames As String,  
    searchCondition As String,  
    sortColumns As String,  
    startRecord As Integer,  
    maxRecords As Integer  
) As DataSet
```

Parameters

columnNames

Type: [System.String](#)

searchCondition

Type: [System.String](#)

sortColumns

Type: [System.String](#)

startRecord

Type: [System.Int32](#)

maxRecords

Type: [System.Int32](#)

Return Value

Type: **DataSet**

See Also

[WorkOrder Class](#)

[Browse Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Exists Method

Determines if a specific Work Order exists.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual bool Exists(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID,  
    string subID  
)
```

VB

```
Public Overridable Function Exists (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String,  
    subID As String  
) As Boolean
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

subID

Type: [System.String](#)

Return Value

Type: [Boolean](#)

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Find Method

Retrieves a specific Work Order. Only the top-level data table (WORK_ORDER) is returned.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Find(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID  
)
```

VB

```
Public Overridable Sub Find (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String  
)
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Load Method

Load a specific Work Order.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Load(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID  
)
```

VB

```
Public Overridable Sub Load (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String  
)
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewCoProductRow Method

Inserts a new row into the CO_PRODUCT data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewCoProductRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    string partID  
)
```

VB

```
Public Overridable Function NewCoProductRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    partID As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

partID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewOperationBinaryRow Method

Inserts a new row into the OPERATION_BINARY data table. Only binary type "D" (long text) is supported.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewOperationBinaryRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int sequenceNo,  
    string binaryType  
)
```

VB

```
Public Overridable Function NewOperationBinaryRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    sequenceNo As Integer,  
    binaryType As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

sequenceNo

Type: [System.Int32](#)

binaryType

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewOperationResourceRow Method

Inserts a new row into the OPERATION_RESOURCE data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewOperationResourceRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int sequenceNo,  
    string resourceID  
)
```

VB

```
Public Overridable Function NewOperationResourceRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    sequenceNo As Integer,  
    resourceID As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

sequenceNo

Type: [System.Int32](#)

resourceID

Type: [System.String](#)

Return Value

Type: **DataRow**



See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewOperationRow Method

Overload List

| | Name | Description |
|---|--|--|
|  | NewOperationRow(String, String, String, String, String) | Inserts a new row into the OPERATION data table, automatically assigning the next available sequence number. |
|  | NewOperationRow(String, String, String, String, String, Int32) | Inserts a new row into the OPERATION data table. |

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewOperationRow Method (String, String, String, String, String)

Inserts a new row into the OPERATION data table, automatically assigning the next available sequence number.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewOperationRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID  
)
```

VB

```
Public Overridable Function NewOperationRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

WorkOrder.NewOperationRow Method (String, String, String, String, String)

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewOperationRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewOperationRow Method (String, String, String, String, Int32)

Inserts a new row into the OPERATION data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewOperationRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int sequenceNo  
)
```

VB

```
Public Overridable Function NewOperationRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    sequenceNo As Integer  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

sequenceNo

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewOperationRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewOperServiceCostRow Method

Inserts a new row into the OPER_SERVICE_COST data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewOperServiceCostRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int sequenceNo,  
    decimal qty  
)
```

VB

```
Public Overridable Function NewOperServiceCostRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    sequenceNo As Integer,  
    qty As Decimal  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

sequenceNo

Type: [System.Int32](#)

qty

Type: [System.Decimal](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewRequirementBinaryRow Method

Inserts a new row into the REQUIREMENT_BINARY data table. Only binary type "D" (long text) is supported.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewRequirementBinaryRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int operationSeqNo,  
    int pieceNo,  
    string binaryType  
)
```

VB

```
Public Overridable Function NewRequirementBinaryRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    operationSeqNo As Integer,  
    pieceNo As Integer,  
    binaryType As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

operationSeqNo

Type: [System.Int32](#)

pieceNo

Type: [System.Int32](#)

binaryType

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewRequirementCostRow Method

Inserts a new row into the REQUIREMENT_COST data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewRequirementCostRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int operationSeqNo,  
    int pieceNo,  
    decimal qty  
)
```

VB

```
Public Overridable Function NewRequirementCostRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    operationSeqNo As Integer,  
    pieceNo As Integer,  
    qty As Decimal  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

operationSeqNo

Type: [System.Int32](#)

pieceNo

Type: [System.Int32](#)

qty

Type: [System.Decimal](#)

Return Value

Type: **DataRow**



See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewRequirementRow Method

Overload List

| | Name | Description |
|---|---|---|
|  | NewRequirementRow(String, String, String, String, String, Int32) | Inserts a new row into the REQUIREMENT data table, assigning the next available piece number. |
|  | NewRequirementRow(String, String, String, String, String, Int32, Int32) | Inserts a new row into the REQUIREMENT data table. |

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewRequirementRow Method (String, String, String, String, String, Int32)

Inserts a new row into the REQUIREMENT data table, assigning the next available piece number.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewRequirementRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int operationSeqNo  
)
```

VB

```
Public Overridable Function NewRequirementRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    operationSeqNo As Integer  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

operationSeqNo

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewRequirementRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewRequirementRow Method (String, String, String, String, Int32, Int32)

Inserts a new row into the REQUIREMENT data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewRequirementRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    int operationSeqNo,  
    int pieceNo  
)
```

VB

```
Public Overridable Function NewRequirementRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    operationSeqNo As Integer,  
    pieceNo As Integer  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

operationSeqNo

Type: [System.Int32](#)

pieceNo

Type: [System.Int32](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewRequirementRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderBinaryRow Method

Inserts a new row into the WORK_ORDER_BINARY data table. Only binary type "D" (long text) is supported.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewWorkOrderBinaryRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    string binaryType  
)
```

VB

```
Public Overridable Function NewWorkOrderBinaryRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    binaryType As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

binaryType

Type: [System.String](#)

Return Value

Type: **DataRow**



See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderLegRow Method

Overload List

| | Name | Description |
|---|--|---|
|  | NewWorkOrderLegRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table, assigning the next available SUB_ID. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow(). |
|  | NewWorkOrderLegRow(String, String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow(). |

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderLegRow Method (String, String, String, String)

Inserts a new row into the WORK_ORDER table, assigning the next available SUB_ID. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow().

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewWorkOrderLegRow(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID  
)
```

VB

```
Public Overridable Function NewWorkOrderLegRow (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String  
) As DataRow
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewWorkOrderLegRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderLegRow Method (String, String, String, String, String)

Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order Leg row (SUB_ID > "0"). To insert a Work Order Header row, use NewWorkOrderRow().

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewWorkOrderLegRow(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID,  
    string subID  
)
```

VB

```
Public Overridable Function NewWorkOrderLegRow (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String,  
    subID As String  
) As DataRow
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

subID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewWorkOrderLegRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderMilestoneRow Method

Inserts a new row into the WORKORD_MILESTONE data table.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewWorkOrderMilestoneRow(  
    string workorderType,  
    string workorderBaseID,  
    string workorderLotID,  
    string workorderSplitID,  
    string workorderSubID,  
    string milestoneID  
)
```

VB

```
Public Overridable Function NewWorkOrderMilestoneRow (  
    workorderType As String,  
    workorderBaseID As String,  
    workorderLotID As String,  
    workorderSplitID As String,  
    workorderSubID As String,  
    milestoneID As String  
) As DataRow
```

Parameters

workorderType

Type: [System.String](#)

workorderBaseID

Type: [System.String](#)

workorderLotID

Type: [System.String](#)

workorderSplitID

Type: [System.String](#)

workorderSubID

Type: [System.String](#)

milestoneID

Type: [System.String](#)

Return Value

Type: **DataRow**



See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderRow Method

Overload List

| | Name | Description |
|---|---|---|
|  | NewWorkOrderRow(String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkORderLegRow(). |
|  | NewWorkOrderRow(String, String, String, String, String) | Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkORderLegRow(). |

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderRow Method (String, String, String, String)

Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkOrderLegRow().

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewWorkOrderRow(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID  
)
```

VB

```
Public Overridable Function NewWorkOrderRow (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String  
) As DataRow
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewWorkOrderRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.NewWorkOrderRow Method (String, String, String, String, String)

Inserts a new row into the WORK_ORDER table. NOTE: This method only inserts a Work Order header row (SUB_ID = "0"). To insert a Work Order Leg row, use NewWorkORderLegRow().

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual DataRow NewWorkOrderRow(  
    string type,  
    string baseID,  
    string lotID,  
    string splitID,  
    string subID  
)
```

VB

```
Public Overridable Function NewWorkOrderRow (  
    type As String,  
    baseID As String,  
    lotID As String,  
    splitID As String,  
    subID As String  
) As DataRow
```

Parameters

type

Type: [System.String](#)

baseID

Type: [System.String](#)

lotID

Type: [System.String](#)

splitID

Type: [System.String](#)

subID

Type: [System.String](#)

Return Value

Type: **DataRow**

See Also

[WorkOrder Class](#)

[NewWorkOrderRow Overload](#)

[Lsa.Vmfg.ShopFloor Namespace](#)

WorkOrder.Save Method

Saves all previously loaded Work Orders to the database.

Namespace: [Lsa.Vmfg.ShopFloor](#)

Assembly: VmfgShopFloor (in VmfgShopFloor.dll) Version: 8.1.100.0 (8.1.100.0)

Syntax

C#

```
public virtual void Save()
```

VB

```
Public Overridable Sub Save
```

See Also

[WorkOrder Class](#)

[Lsa.Vmfg.ShopFloor Namespace](#)