

Limnor Studio – References

Part – III

Database Components

Contents

1	EasyDataSet.....	8
1.1	Properties	8
1.1.1	DatabaseConnection - ConnectionItem.....	8
1.1.2	DataBindingSource - BindingSource.....	8
1.1.3	DataName - String	8
1.1.4	DataTable - DataTable	8
1.1.5	Description - String.....	8
1.1.6	Fields - FieldList	8
1.1.7	ReadOnly - Boolean	9
1.1.8	HasData - Boolean	9
1.1.9	IsConnectionReady - Boolean	9
1.1.10	LastError - String.....	9
1.1.11	Name - String	9
1.1.12	Parameters - DbParameterList.....	9
1.1.13	QueryOnStart - Boolean	9
1.1.14	ReadOnly - Boolean	9
1.1.15	RowCount - Int32	9
1.1.16	RowIndex - Int32	10
1.1.17	ShowErrorMessage - Boolean.....	10
1.1.18	SQL - SQLStatement.....	10
1.2	Methods	10
1.2.1	AddNewRecord	10
1.2.2	BeginEditCurrentRecord	10
1.2.3	CancelEditCurrentRecord.....	10
1.2.4	CreateDataBinding(String bindToPropertyName,String sourceFieldName) Binding	10

1.2.5	DeleteCurrentRecord.....	10
1.2.6	EndEditCurrentRecord	10
1.2.7	FetchData.....	11
1.2.8	GetColumnSum(String fieldName) Double.....	11
1.2.9	GetQueryInfo() String	11
1.2.10	MoveFirst.....	11
1.2.11	MoveLast.....	11
1.2.12	MoveNext	11
1.2.13	MovePrevious	11
1.2.14	Query	11
1.2.15	QueryWithParameterValues(Object[] values)	11
1.2.16	SelectConnection(Form dialogParent)	11
1.2.17	SelectQuery(Form dialogParent).....	12
1.2.18	SetConnectionString(String connectionString)	12
1.2.19	SetCredential(String user,String userPassword,String databasePassword)	12
1.2.20	SetFieldValue(String fieldName,Object value).....	12
1.2.21	SetParameterValue(String parameterName,Object value)	12
1.2.22	Update() Boolean	12
1.3	Events	12
1.3.1	AfterQuery - EventHandler	12
1.3.2	AfterUpdate - EventHandler	12
1.3.3	BeforePrepareCommands - EventHandler	13
1.3.4	BeforeQuery - EventHandler.....	13
1.3.5	BeforeUpdate - EventHandler.....	13
1.3.6	DataFilled - EventHandler	13
1.3.7	LastErrorMessageChanged - EventHandler	13
1.3.8	SqlChanged - EventHandler	13
2	EasyGrid	13
2.1	Properties	13
2.1.1	AutoSave - Boolean	14
2.1.2	DatabaseConnection - ConnectionItem.....	14
2.1.3	DataName - String	14

2.1.4	FieldEditors - FieldEditorList	14
2.1.5	Fields - FieldList	14
2.1.6	ReadOnly - Boolean	14
2.1.7	HasData - Boolean	14
2.1.8	IsValidCurrentRow - Boolean.....	14
2.1.9	IsConnectionReady - Boolean	14
2.1.10	LogFile - String.....	15
2.1.11	Parameters - DbParameterList.....	15
2.1.12	QueryOnStart - Boolean	15
2.1.13	ShowErrorMessage - Boolean.....	15
2.1.14	SQL - SQLStatement.....	15
2.2	Methods	15
2.2.1	GetColumnSum(String fieldName) Double	15
2.2.2	Query	15
2.2.3	QueryWithParameterValues(Object[] values)	15
2.2.4	SetCredential(String user,String userPassword,String databasePassword)	15
2.2.5	SetCredentialByUI(Form caller) Boolean	16
2.2.6	SetFieldValue(String fieldName,Object value).....	16
2.2.7	SetKeepColumnSettings(Boolean keepColumnSettings).....	16
2.2.8	SetParameterValue(String parameterName,Object value)	16
2.2.9	UpdateData() Boolean	16
2.3	Events	16
2.3.1	AfterUpdate - EventHandler	16
2.3.2	BeforeUpdate - EventHandler.....	16
2.3.3	DataFilled - EventHandler	16
2.3.4	ExecutionError - EventHandler.....	16
2.3.5	Lookup - EventHandler	17
3	EasyGridDetail	17
3.1	Properties	17
3.1.1	DetailsKeyColumns - String[].....	17
3.1.2	HasData - Boolean	17
3.1.3	Master - IMasterSource	17

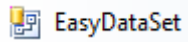
3.1.4	MasterKeyColumns - String[]	17
3.2	Methods	17
3.2.1	Query	17
4	EasyQuery	18
4.1	Properties	18
4.1.1	CommittedRowCount - Int32	18
4.1.2	ConnectionString - String	18
4.1.3	DatabaseConnection - ConnectionItem.....	18
4.1.4	DataStorage - DataSet	18
4.1.5	DataTable - DataTable	18
4.1.6	DeleteSql - String.....	18
4.1.7	Description - String.....	18
4.1.8	Fields - FieldList.....	18
4.1.9	ForReadOnly - Boolean	19
4.1.10	HasData - Boolean	19
4.1.11	InsertSql - String	19
4.1.12	IsConnectionReady - Boolean	19
4.1.13	IsJet - Boolean	19
4.1.14	IsOdbc - Boolean	19
4.1.15	IsOleDb - Boolean.....	19
4.1.16	IsOracle - Boolean	19
4.1.17	IsSQLSever - Boolean	19
4.1.18	LastError - String.....	20
4.1.19	LogFile - String.....	20
4.1.20	Name - String	20
4.1.21	Parameters - ParameterList.....	20
4.1.22	QueryOnStart - Boolean	20
4.1.23	ReadOnly - Boolean	20
4.1.24	RowCount - Int32	20
4.1.25	RowIndex - Int32	20
4.1.26	ShowErrorMessage - Boolean.....	20
4.1.27	SQL - SQLStatement.....	21

4.1.28	TableToUpdate - String	21
4.1.29	UpdateSql - String.....	21
4.2	Methods	21
4.2.1	AddNewRecord	21
4.2.2	BeginEditCurrentRecord	21
4.2.3	CancelEditCurrentRecord.....	21
4.2.4	CreateDataBinding(String bindToPropertyName,String sourceFieldName) Binding	21
4.2.5	DeleteCurrentRecord.....	21
4.2.6	EndEditCurrentRecord	21
4.2.7	FetchData	22
4.2.8	GetColumnSum(String fieldName) Double	22
4.2.9	GetQueryInfo() String	22
4.2.10	LogMessage(Exception e,String message,Object[] values).....	22
4.2.11	LogMessage(String message,Object[] values).....	22
4.2.12	MoveFirst	22
4.2.13	MoveLast.....	22
4.2.14	MoveNext	22
4.2.15	MovePrevious	22
4.2.16	Query	22
4.2.17	QueryWithParameterValues(Object[] values)	23
4.2.18	SelectConnection(Form dialogParent)	23
4.2.19	SelectQuery(Form dialogParent).....	23
4.2.20	SetConnectionString(String connectionString)	23
4.2.21	SetCredential(String user,String userPassword,String databasePassword)	23
4.2.22	SetCredentialByUI(Form caller) Boolean	23
4.2.23	SetFieldValue(String fieldName,Object value).....	23
4.2.24	SetLogFilePath(String logFileName)	23
4.2.25	SetParameterValue(String parameterName,Object value)	23
4.2.26	Update() Boolean	23
4.3	Events	24
4.3.1	AfterQuery - EventHandler	24
4.3.2	AfterUpdate - EventHandler	24

4.3.3	BeforePrepareCommands - EventHandler	24
4.3.4	BeforeQuery - EventHandler.....	24
4.3.5	BeforeUpdate - EventHandler.....	24
4.3.6	DataFilled - EventHandler	24
4.3.7	LastErrorMessageChanged - EventHandler	24
4.3.8	SqlChanged - EventHandler	24
5	EasyTransactor	25
5.1	Properties	25
5.1.1	DatabaseCommands - EasyUpdater[].....	25
5.1.2	ErrorMessage - String	25
5.1.3	ShowErrorMessage - Boolean	25
5.2	Methods	25
5.2.1	ExecuteActionsInTransaction(IsolationLevel isolationLevel).....	25
5.3	Events	25
5.3.1	ExecuteInOneTransaction - ExecuteInOneTransactionHandler.....	26
5.3.2	ExecutionError - EventHandler.....	26
6	EasyUpdater	26
6.1	Properties	26
6.1.1	AffectedRowCount - Int32	26
6.1.2	ConnectionString - String.....	26
6.1.3	DatabaseConnection - ConnectionItem.....	26
6.1.4	ErrorMessage - String	26
6.1.5	ExecutionCommand - SQLNoneQuery.....	26
6.1.6	HasError - Boolean	27
6.1.7	IsConnectionReady - Boolean	27
6.1.8	IsOdbc - Boolean	27
6.1.9	IsOleDb - Boolean.....	27
6.1.10	IsOracle - Boolean	27
6.1.11	IsSQLSever - Boolean	27
6.1.12	Parameters - DbParameterList.....	27
6.1.13	ShowErrorMessage - Boolean.....	27
6.2	Methods	27

6.2.1	Execute	28
6.2.2	ExecuteUseDataTableForParameterValuesByColumnName(DataTable parameterValues) 28	
6.2.3	ExecuteUseDataTableForParameterValuesByColumnPosition(DataTable parameterValues) 28	
6.2.4	ExecuteWithParameterValues(Object[] values).....	28
6.2.5	SetParameterValue(String parameterName,Object value)	28
6.2.6	SetParameterValueToNull(String parameterName)	28
6.3	Events	28
6.3.1	ExecuteError - EventHandler.....	28
6.3.2	ExecuteFinish - EventHandler	29

1 EasyDataSet



It combines EasyQuery and DataSet into one component. It provides a Query Builder to let you build SQL query. It can also be used to provide master-records in a master-detail relationship.

It is derived from DataSet in the Microsoft .Net Framework. For reference of DataSet, see <http://msdn.microsoft.com/en-us/library/bwy42y0e.aspx>

1.1 Properties

1.1.1 DatabaseConnection - ConnectionItem

Connection to the database

1.1.2 DataBindingSource - BindingSource

The data binding source to provide data held in this DataSet to other controls

1.1.3 DataName - String

Name of the DataTable holding the data for this instance

1.1.4 DataTable - DataTable

The DataTable for holding the records for this instance

1.1.5 Description - String

Description for this query

1.1.6 Fields - FieldList

Fields in this data table. It represents the current row at runtime.

1.1.7 ForReadOnly - Boolean

Indicate whether the changing of data is desired

1.1.8 HasData - Boolean

The number of records currently hold in the DataTable

1.1.9 IsConnectionReady - Boolean

Accessing this property will test connect to the database. If the connection is made then this property is True; otherwise it is False.

1.1.10 LastError - String

Error message from the last failed operation

1.1.11 Name - String

The name of the component

1.1.12 Parameters - DbParameterList

Parameters for filtering the data.

1.1.13 QueryOnStart - Boolean

Indicates whether to make database query when this object is created.

1.1.14 ReadOnly - Boolean

Indicate whether the data is changeable

1.1.15 RowCount - Int32

The number of records currently hold in the DataTable

1.1.16 RowIndex - Int32

When retrieving values by Fields property, this property indicates the row number. 0 indicates the first row; 1 indicates the second row, and so on.

1.1.17 ShowErrorMessage - Boolean

Controls whether error message is displayed when an error occurs. Error messages are logged in a text file. The file path is indicated by LogFile property.

1.1.18 SQL - SQLStatement

SQL statement for querying database

1.2 Methods

1.2.1 AddNewRecord

Add a new record

1.2.2 BeginEditCurrentRecord

Begin editing the current record

1.2.3 CancelEditCurrentRecord

Cancel editing the current record

1.2.4 CreateDataBinding(String bindToPropertyName,String sourceFieldName) Binding

Create a data binding object. It is to be added to DataBindings property via DataBindings.Add(obj) method, where obj is created by this method.

1.2.5 DeleteCurrentRecord

Delete the current record

1.2.6 EndEditCurrentRecord

End editing the current record

1.2.7 FetchData

Query the database for data. It may be called after setting parameter values.

1.2.8 GetColumnSum(String fieldName) Double

Calculate summary of the column specified by fieldName.

1.2.9 GetQueryInfo() String

Get information about this query

1.2.10 MoveFirst

Set the first record of the first table in DataStorage as the current record

1.2.11 MoveLast

Set the last record of the first table in DataStorage as the current record

1.2.12 MoveNext

Set the next record of the first table in DataStorage as the current record

1.2.13 MovePrevious

Set the previous record of the first table in DataStorage as the current record

1.2.14 Query

Compose the query commands and query the database for data

1.2.15 QueryWithParameterValues(Object[] values)

Execute database query with the parameter values

1.2.16 SelectConnection(Form dialogParent)

Use a dialogue box to select database connection type and other parameters.

1.2.17 SelectQuery(Form dialogParent)

Use a dialogue box to make database query.

1.2.18 SetConnectionString(String connectionString)

Set connection string for accessing the database. It should be called at the event of BeforePrepareCommands and before event BeforeQuery.

1.2.19 SetCredential(String user,String userPassword,String databasePassword)

Set credential for accessing the database. It should be called at the event of BeforePrepareCommands and before event BeforeQuery.

1.2.20 SetFieldValue(String fieldName,Object value)

Set field value for the current record

1.2.21 SetParameterValue(String parameterName,Object value)

Set query parameter value.

1.2.22 Update() Boolean

Save changed data back to the database

1.3 Events

1.3.1 AfterQuery - EventHandler

This event occurs after querying the database to get the data.

1.3.2 AfterUpdate - EventHandler

This event occurs after executing Update action.

1.3.3 BeforePrepareCommands - EventHandler

This event occurs before creating database accessing commands for loading the data. SetCredential, SetConnectionString, or SelectConnection can be executed at this moment to give desired database accessing parameters.

1.3.4 BeforeQuery - EventHandler

This event occurs after the database accessing commands are created and before querying the database to get the data.

1.3.5 BeforeUpdate - EventHandler

This event occurs before executing Update action.

1.3.6 DataFilled - EventHandler

Occurs when data are retrieved from the database

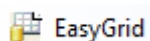
1.3.7 LastErrorMessageChanged - EventHandler

Occurs when the LastError property is changed

1.3.8 SqlChanged - EventHandler

Occurs when setting the SQL property succeeded

2 EasyGrid



This control combines EasyQuery, DataSet and DataGridView into one component

It is derived from DataGridView in the Microsoft .Net Framework. For the reference of DataGridView see <http://msdn.microsoft.com/en-us/library/system.windows.forms.datagridview.aspx>

2.1 Properties

2.1.1 AutoSave - Boolean

True: the changes will be automatically saved back to the database when the form is closed; False: the changes will be saved to the database only by UpdateData action

2.1.2 DatabaseConnection - ConnectionItem

Connection to the database

2.1.3 DataName - String

Name of the DataTable holding the data for this instance

2.1.4 FieldEditors - FieldEditorList

Field editors for editing cells at runtime by the user.

2.1.5 Fields - FieldList

Fields in this data table. It represents the current row at runtime.

2.1.6 ForReadOnly - Boolean

Indicate whether the changing of data is desired

2.1.7 HasData - Boolean

Gets a value indicating whether RowCount is greater than 0

2.1.8 HasValidCurrentRow - Boolean

Gets a value indicating whether current row is valid to be used.

2.1.9 IsConnectionReady - Boolean

Accessing this property will test connect to the database. If the connection is made then this property is True; otherwise it is False.

2.1.10 LogFile - String

Indicates the file path for logging errors

2.1.11 Parameters - DbParameterList

Parameters for filtering the data.

2.1.12 QueryOnStart - Boolean

Indicates whether to make database query when this object is created.

2.1.13 ShowErrorMessage - Boolean

Controls whether error message is displayed when an error occurs. Error messages are logged in a text file. The file path is indicated by LogFile property.

2.1.14 SQL - SQLStatement

SQL statement for querying database

2.2 Methods

2.2.1 GetColumnSum(String fieldName) Double

Calculate summary of the column specified by fieldName.

2.2.2 Query

Query the database to get data

2.2.3 QueryWithParameterValues(Object[] values)

Execute database query with the parameter values

2.2.4 SetCredential(String user,String userPassword,String databasePassword)

Set credential for accessing the database.

2.2.5 SetCredentialByUI(Form caller) Boolean

Launch a dialogue box to set credential for accessing the database.

2.2.6 SetFieldValue(String fieldName,Object value)

Set field value for the current record

2.2.7 SetKeepColumnSettings(Boolean keepColumnSettings)

Setting SQL property usually will remove all column settings. Calling this method with True for keepColumnSettings prevents column settings being removed when SQL property is changed.

2.2.8 SetParameterValue(String parameterName,Object value)

Set query parameter value.

2.2.9 UpdateData() Boolean

Save changed data to the database

2.3 Events

2.3.1 AfterUpdate - EventHandler

This event occurs after executing Update action.

2.3.2 BeforeUpdate - EventHandler

This event occurs before executing Update action.

2.3.3 DataFilled - EventHandler

Occurs when data are retrieved from the database

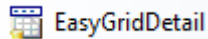
2.3.4 ExecutionError - EventHandler

Occurs when the Query method fails. The ErrorMessage property gives information about why the execution failed.

2.3.5 Lookup - EventHandler

Occurs when the user select an item from a dropdown list for editing a field.

3 EasyGridDetail



This is a data grid for showing detail records in a master-detail relationship. The master records may be provided by an EasyGrid or an EasyDataSet

It is derived from EasyGrid

3.1 Properties

3.1.1 DetailsKeyColumns - String[]

Key fields of the details data table for setting the master-details relationship

3.1.2 HasData - Boolean

Gets a value indicating whether RowCount is greater than 0

3.1.3 Master - IMasterSource

This is the component holding the master records

3.1.4 MasterKeyColumns - String[]

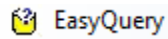
Key fields of the master data table for setting the master-details relationship

3.2 Methods

3.2.1 Query

Query the database to get data

4 EasyQuery



It defines database query to provide data to a DataSet. It provides a Query Builder to let you build SQL query.

4.1 Properties

4.1.1 CommittedRowCount - Int32

The number of records currently held in memory, which are also in the database

4.1.2 ConnectionString - String

Connection string used for connecting to the database

4.1.3 DatabaseConnection - ConnectionItem

Connection to the database

4.1.4 DataStorage - DataSet

This is a DataSet to store the query results

4.1.5 DataTable - DataTable

The DataTable for holding the records for this instance

4.1.6 DeleteSql - String

SQL statement for deleting a row from the database

4.1.7 Description - String

Description for this query

4.1.8 Fields - FieldList

Fields in this data table. It represents the current row at runtime.

4.1.9 ForReadOnly - Boolean

Indicate whether the changing of data is desired

4.1.10 HasData - Boolean

Gets a value indicating whether RowCount is greater than 0

4.1.11 InsertSql - String

SQL statement for inserting a new row to the database

4.1.12 IsConnectionReady - Boolean

Accessing this property will test connect to the database. If the connection is made then this property is True; otherwise it is False.

4.1.13 IsJet - Boolean

Indicates the database connection is through a Jet database driver

4.1.14 IsOdbc - Boolean

Indicates the database connection is through an OdbcConnection for ODBC drives such as MySQL, PostgreSQL, etc.

4.1.15 IsOleDb - Boolean

Indicates the database connection is through an OleDbConnection for OleDb drives such as Microsoft Access, Excel, etc.

4.1.16 IsOracle - Boolean

Indicates the database connection is through an OracleConnection for Oracle database

4.1.17 IsSQLSever - Boolean

Indicates the database connection is through a SqlConnection for Microsoft SQL Server

4.1.18 LastError - String

Error message from the last failed operation

4.1.19 LogFile - String

Indicates the file path for logging errors

4.1.20 Name - String

The name of the component

4.1.21 Parameters - ParameterList

Parameters for filtering the data.

4.1.22 QueryOnStart - Boolean

Indicates whether to make database query when this object is created.

4.1.23 ReadOnly - Boolean

Indicate whether the data is changeable

4.1.24 RowCount - Int32

The number of records currently hold in the DataTable

4.1.25 RowIndex - Int32

When retrieving values by Fields property, this property indicates the row number. 0 indicates the first row; 1 indicates the second row, and so on.

4.1.26 ShowErrorMessage - Boolean

Controls whether error message is displayed when an error occurs. Error messages are logged in a text file. The file path is indicated by LogFile property.

4.1.27 SQL - SQLStatement

SQL statement for querying database

4.1.28 TableToUpdate - String

The table that can be updated

4.1.29 UpdateSql - String

SQL statement for updating a row in the database

4.2 Methods

4.2.1 AddNewRecord

Add a new record

4.2.2 BeginEditCurrentRecord

Begin editing the current record

4.2.3 CancelEditCurrentRecord

Cancel editing the current record

4.2.4 CreateDataBinding(String bindToPropertyName,String sourceFieldName) Binding

Create a data binding object. It is to be added to DataBindings property via DataBindings.Add(obj) method, where obj is created by this method.

4.2.5 DeleteCurrentRecord

Delete the current record

4.2.6 EndEditCurrentRecord

End editing the current record

4.2.7 FetchData

Query the database for data. It may be called after setting parameter values.

4.2.8 GetColumnSum(String fieldName) Double

Calculate summary of the column specified by fieldName.

4.2.9 GetQueryInfo() String

Get information about this query

4.2.10 LogMessage(Exception e,String message,Object[] values)

log message into the log file

4.2.11 LogMessage(String message,Object[] values)

log message into the log file

4.2.12 MoveFirst

Set the first record of the first table in DataStorage as the current record

4.2.13 MoveLast

Set the last record of the first table in DataStorage as the current record

4.2.14 MoveNext

Set the next record of the first table in DataStorage as the current record

4.2.15 MovePrevious

Set the previous record of the first table in DataStorage as the current record

4.2.16 Query

Compose the query commands and query the database for data

4.2.17 QueryWithParameterValues(Object[] values)

Execute database query with the parameter values

4.2.18 SelectConnection(Form dialogParent)

Use a dialogue box to select database connection type and other parameters.

4.2.19 SelectQuery(Form dialogParent)

Use a dialogue box to make database query.

4.2.20 SetConnectionString(String connectionString)

Set connection string for accessing the database. It should be called at the event of BeforePrepareCommands and before event BeforeQuery.

4.2.21 SetCredential(String user,String userPassword,String databasePassword)

Set credential for accessing the database.

4.2.22 SetCredentialByUI(Form caller) Boolean

Launch a dialogue box to set credential for accessing the database.

4.2.23 SetFieldValue(String fieldName,Object value)

Set field value for the current record

4.2.24 SetLogFilePath(String logFileName)

Set the full file path for the log file.

4.2.25 SetParameterValue(String parameterName,Object value)

Set query parameter value.

4.2.26 Update() Boolean

Save changed data back to the database

4.3 Events

4.3.1 AfterQuery - EventHandler

This event occurs after querying the database to get the data.

4.3.2 AfterUpdate - EventHandler

This event occurs after executing Update action.

4.3.3 BeforePrepareCommands - EventHandler

This event occurs before creating database accessing commands for loading the data. SetCredential, SetConnectionString, or SelectConnection can be executed at this moment to give desired database accessing parameters.

4.3.4 BeforeQuery - EventHandler

This event occurs after the database accessing commands are created and before querying the database to get the data.

4.3.5 BeforeUpdate - EventHandler

This event occurs before executing Update action.

4.3.6 DataFilled - EventHandler

Occurs when data are retrieved from the database

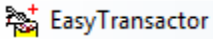
4.3.7 LastErrorMessageChanged - EventHandler

Occurs when the LastError property is changed

4.3.8 SqlChanged - EventHandler

Occurs when setting the SQL property succeeded

5 EasyTransactor



This component executes database updating commands in a transaction. It can be used for adding new records, modifying existing records, or deleting records.

5.1 Properties

5.1.1 DatabaseCommands - EasyUpdater[]

Gets an array containing EasyUpdater components to be included in the database transaction when ExecuteActionsInTransaction is executed. The database connection for the first EasyUpdater will be used for all other EasyUpdater components.

5.1.2 ErrorMessage - String

Gets the error message for the last Execute action if it failed

5.1.3 ShowErrorMessage - Boolean

Controls whether error message is displayed when event ExecutionError occurs.

5.2 Methods

5.2.1 ExecuteActionsInTransaction(IsolationLevel isolationLevel)

Executing an action created from this method causes event ExecuteInOneTransaction to occur and thus all actions assigned to event ExecuteInOneTransaction are executed. All Execute actions belonging to components listed in DatabaseCommands are executed in one single database transaction. The transaction isolation level is specified by parameter isolationLevel.

5.3 Events

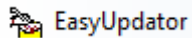
5.3.1 ExecuteInOneTransaction - ExecuteInOneTransactionHandler

Occurs when an ExecuteActionsInTransaction action is executed. All Execute actions belonging to DatabaseCommands and assigned to this event will be executed in one single database transaction. Do not assign an ExecuteActionsInTransaction action to this event because it will cause a deadlock.

5.3.2 ExecutionError - EventHandler

Occurs when the Execute method fails. The ErrorMessage property gives information about why the execution failed.

6 EasyUpdator



This component executes database updating commands. It can be used for adding new records, modifying existing records, or deleting records.

6.1 Properties

6.1.1 AffectedRowCount - Int32

Gets a value indicating the number of affected rows of the last Execute action

6.1.2 ConnectionString - String

Connection string used for connecting to the database

6.1.3 DatabaseConnection - ConnectionItem

Connection to the database

6.1.4 ErrorMessage - String

Gets the error message for the last Execute action if it failed

6.1.5 ExecutionCommand - SQLNoneQuery

Database command to update the database

6.1.6 HasError - Boolean

Gets a value indicating whether the last Execute action failed

6.1.7 IsConnectionReady - Boolean

Accessing this property will test connect to the database. If the connection is made then this property is True; otherwise it is False.

6.1.8 IsOdbc - Boolean

Indicates the database connection is through an OdbcConnection for ODBC drives such as MySQL, PostgreSQL, etc.

6.1.9 IsOleDb - Boolean

Indicates the database connection is through an OleDbConnection for OleDb drives such as Microsoft Access, Excel, etc.

6.1.10 IsOracle - Boolean

Indicates the database connection is through an OracleConnection for Oracle database

6.1.11 IsSQLSever - Boolean

Indicates the database connection is through a SqlConnection for Microsoft SQL Server

6.1.12 Parameters - DbParameterList

Parameters for filtering the data.

6.1.13 ShowErrorMessage - Boolean

Controls whether error message is displayed when event ExecuteError occurs.

6.2 Methods

6.2.1 Execute

Execute database updating command. If more than one Execute action is assigned to event `ExecuteInOneTransaction` then those Execute actions will be executed in one single database transaction.

6.2.2 `ExecuteUseDataTableForParameterValuesByColumnName(DataTable parameterValues)`

Execute the database updating command using values from each row of a `DataTable` for parameter values. The values of the columns are mapped to the parameters by column names. White spaces, '(', ')', '\', '/', tab, carriage return and new-line are ignored in column names when doing matching. For example, column Price is mapped to parameter `@Price`, column Quantity is mapped to parameter `@Quantity`, etc. For example, Order Qty matches `@OrderQty`. It executes as many times as the number of rows in the `DataTable`.

6.2.3 `ExecuteUseDataTableForParameterValuesByColumnPosition(DataTable parameterValues)`

Execute the database updating command using values from each row of a `DataTable` for parameter values. The value of the first column is for the first parameter, the second column is for the second parameter, and so on. It executes as many times as the number of rows in the `DataTable`.

6.2.4 `ExecuteWithParameterValues(Object[] values)`

Execute the database updating command with all the parameter values

6.2.5 `SetParameterValue(String parameterName, Object value)`

Set query parameter value.

6.2.6 `SetParameterValueToNull(String parameterName)`

Set the query parameter value to `DBNull`.

6.3 Events

6.3.1 `ExecuteError - EventHandler`

Occurs when the Execute method fails. The `ErrorMessage` property gives information about why the execution failed.

6.3.2 ExecuteFinish - EventHandler

Occurs when the Execute method completes successfully.