Database Management System LINTER®

Version 5.9

PHP

Relational Expert Systems

RELEX®

Table of Contents

Introduction

This document lists the available PHP commands and describes their use.

Commands

Linter_Open_Connect

Establishes a connection between PHP and a Linter database on the DBserver with the given username and password.

Syntax

int Linter_Open_Connect(username, password, DBserver, mode);

Parameter	Description
username	The name of the user allowed to connect to the data base server.
password	The password for the user.
DBserver	Linter server name.
mode	Can be zero or an OR-combination of transaction modes and supported code pages.

Returned value

If the function succeeds, the returned value is the ID of the new connection. Otherwise, the returned value is negative. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

All string parameters are case-sensitive. If username or password exceed 66 characters, they will be truncated. If DBserver exceeds 8 characters, it will be truncated.

Linter_Close_Connect

Closes a connection identified by ConnectID.

Syntax

int Linter_Close_Connect(ConnectID);

Parameter	Description
ConnectID	ID for the connection to be closed.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

If a connection is the parent for one or more cursors, they will be closed too. For more information about connections and cursors, see document “LinterAPI”.

Linter_Open_Cursor

Opens a cursor between PHP and the Linter database.

Syntax

int LinterOpenCursor(ConnectID);

Parameter	Description
ConnectID	ID for the connect on which the cursor will be opened.

Returned value

If the function succeeds, the returned value is the ID of the new cursor. Otherwise, the returned value is negative. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

The argument ConnectID specifies the parent connection. For more information about connections and cursors, see document “LinterAPI”.

Linter_Close_Cursor

Closes a cursor identified by CursorID.

Syntax

int Linter_Close_Cursor(CursorID);

Parameter	Description
CursorID	ID for the cursor to be closed.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Linter_Get_Cursor_Opt

Gets the value of the option identified by option_name for the cursor identified by CursorID.

Syntax

string | int Linter_Get_Cursor_Opt(CursorID, option_name);

Parameter	Description
CursorI D	ID for the cursor on which the request will be sent.
option_name	Is described in the table below.

Option name	Type	Get/Set	Description
CO_NAME	string	+/+	Name of cursor.

Option name	Type	Get/Set	Description
CO_DT_FORMAT	string	+/+	Date and time format.
CO_COL_COU NT	integer	+/-	Number of columns in answer.
CO_ROW_COU NT	integer	+/-	Number of rows in answer.
CO_ERR_ROW	integer	+/-	Error line number.
CO_ERR_POS	integer	+/-	Error position in line.
CO_TRANS_MODE	integer	+/-	Transaction mode: • TM_OPTIMISTIC; • TM_EXCLUSIVE; • TM_AUTOCOMMIT.
CO_CUR_ROW	integer	+/-	Current row in cursor.
CO_CUR_ROWID	integer	+/-	ROWID of current row in cursor.
CO_CONNECT_ID	integer	+/-	Identifier of the parent connection or error code LPE_NOT_CURSOR.
CO_NODE_NAME	string	+/+	Name of a Linter server.
CO_LAST_ROWID	integer	+/-	ROWID of current row in cursor.
CO_ASYNC_MODE	integer	-/+	Switch to an asochronous mode.
CO_SYNC_MODE	integer	-/+	Switch to synchronous mode.

Returned value

If the function succeeds, the cursor option value is returned. Otherwise, the returned value is negative. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Linter_Set_Cursor_Opt

Sets the option identified by Option name to the value for the cursor identified by CursorID.

Syntax

```
int Linter_Set_Cursor_Opt(CursorID, Optionname, value);
```

Parameter	Description
CursorI D	ID for the cursor on which the request will be sent.
option name	Is described in the function “Linter_Get_Cursor_Opt”.
Value	Depends from the value of the Option name, described in the function “Linter_Get_Cursor_Opt”.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Linter_Exec_Direct

Executes the query using the cursor CursorID.

Syntax

int Linter_Exec_Direct(CursorID, query);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
query	The SQL command to be sent to the Linter server

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

The query itself must be terminated with a semi-colon (;).

Stored procedure can be created and accomplished by Linter_Exec_Direct command.

Linter_Prepare

Prepares the query for the cursor CursorID.

Syntax

int Linter_Prepare(CursorID, query);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
query	The SQL command to be sent to the Linter server

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Linter_Execute

Execute a prepared query for the cursor CursorID specifying values to be passed to the query.

Syntax

int Linter_Execute(CursorID, params);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
params	Params is an array containing values to be passed to the parameters in the prepared query.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

The number of items in the array must be equal to or greater than the number of parameters in prepared query. If the prepared query has no parameters, params can be omitted.

Linter_Fetch

Fetches the row specified by direction from the Linter cursor CursorID.

Syntax

int Linter_Fetch(CursorID, direction[, number]);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
direction	Is described in the table below.
number	Number of the row which will be fetch.

Direction	Description
FETCH_FIRST	Fetch the first row.
FETCH_LAST	Fetch the last row.
FETCH_NEXT	Fetch the next row.
FETCH_PREV	Fetch the previous row.
FETCH_ABSNUM	Fetch by absolute row number.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

If the direction is FETCH_ABSNUM, a target row number must be specified. Row numbering starts from 1.

Li nter_Get_Data_Row

Gets data row from cursor identified by CursorID.

Syntax

array | int Linter_Get_Data_Row(CursorID[,nif]);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
nif	NULL flag. If nif =0 then NULL flag is not set, otherwise NULL flag is set.

Returned value

If the function succeeds, the return value is an array of data. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

Return array is indexed by the element number in the array.

Index numbering starts from 0.

Stored procedure can be used in Get_Data_Row command. In this case, the first element of the returned array is the return value of the stored procedure. If the returned value of the stored procedure is a cursor, then this cursor can be used further in any functions.

Li nter_Get_Data_Array

Gets data row from cursor identified by CursorID.

Syntax

array | int Linter_Get_Data_Array(CursorID);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.

Returned value

If the function succeeds, the return value is an array of data. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

Return array is indexed by the column names of the row. Column name as an index of the array is case sensitive.

Example

```
SELECT ID, NAME, FROM TAB1;
ans = Lnter_Get_Data_Array(CursorID);
ans[TD'] - is the first element of the returned array;
ans['NAME'] - is the second element of the returned array
```

Linter_Get_Col_Prop

Gets properties of the column identified by column from the cursor identified by CursorID.

Syntax

object | int Linter_Get_Col_Prop(CursorID, column);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
column	Name of the column, whose properties will be returned.

Object Properties	Type	Description
Name	string	Column name
table	string	Table name
user	string	Owner of table
type	integer	Column type
type_name	string	Column type name
length	integer	Column length
precision	integer	Column precision
scale	integer	Column scale
is_null	integer	NULL value indicator

Returned value

Column name, Table name, Owner of table, Column type, Column type name, Column length, Column precision, Column scale, NULL value indicator

If the function succeeds, the return value is the object property value. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

Column numbering starts with 1.

Linter_Blob_Append

Appends size bytes from value to the BLOB content of the current row of the cursor identified by CursorID.

Syntax

```
int Linter_Blob_Append(CursorID, value, size);
```

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
value	String data type value.
size	Amount of bytes to be appended.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

This function can be used only for the table with one BLOB column.

Linter_Blob_Append_Ex

Appends size bytes from value to the BLOB content of the current row of the cursor identified by CursorID and sets the type of the Blob column.

Syntax

int Linter_Blob_Append_Ex(CursorID, value, size, type);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
value	String data type value.
size	Amount of bytes to be appended.
type	Data type to be set for BLOB column

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Linter_Blob_Add

Appends size bytes from value to the specified by num BLOB content of the current row of the cursor identified by CursorID.

Syntax

int Linter_Blob_ADD(CursorID, value, size, num);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
value	String data type value.
size	Amount of bytes to be appended.
num	Number of the BLOB column in the current row of the CursorID.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

BLOB-column numbering in the row of the cursor starts from 1.

Linter_Blob_Add_Ex

Appends size bytes from value to the specified by num BLOB content of the current row of the cursor identified by CursorID and sets the type for that BLOB column.

Syntax

int Linter_Blob_ADD_Ex(CursorID, value, size, num, type);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
Value	String data type value.
Size	Amount of bytes to be appended.
Num	Number of the BLOB column in the current row of the CursorID.
Type	Data type to be set for BLOB column.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

BLOB-column numbering in the row of the cursor starts from 1.

Linter_Blob_Clear

Clears, deletes, the entire BLOB value of the current row of the cursor identified by CursorID.

Syntax

```
int Linter_Blob_Clear(CursorID);
```

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

This function can be used only for the table with one BLOB column.

Linter_Blob_Purge

Clears, deletes, the specified by num entire BLOB value of the current row of the cursor identified by CursorID.

Syntax

```
int Linter_Blob_Purge(CursorID, num);
```

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
Num	Number of the BLOB column in the current row of the CursorID.

Returned value

On success, the returned value is LPE_SUCCESS. Otherwise, the returned value is negative. If LPE_LINTER_ERROR is the returned value, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

BLOB -column numbering in the row of the cursor starts from 1

Linter_Blob_Get_Data

Returns BLOB data from the cursor identified by CursorID from an offset of from and with a length of size.

Syntax

string | int Linter_Blob_Get_Data(CursorID, from, size);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
from	The beginning position in the BLOB column to from get data.
size	Amount to be read in bytes.

Returned value

If the function succeeds, the return value is a segment of BLOB data. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

This function can be used only for the table with one BLOB column.

Linter_Blob_Fetch

Returns specified by num BLOB data from the cursor identified by CursorID from an offset of from and with a length of size.

Syntax

string | int Linter_Blob_Fetch(CursorID, from, size, num);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
from	The beginning position in the BLOB column to from get data.
size	Amount to be read in bytes.
num	Number of the BLOB column in the current row of the CursorD.

Returned value

If the function succeeds, the return value is a segment of BLOB data. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

BLOB -column numbering in the row of the cursor starts from 1.

Linter_Blob_Get_Size

Gets the size of BLOB data from the current row of the cursor identified by CursorID.

Syntax

int Linter_Blob_Get_Size(CursorID[,num);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
Num	Number of the BLOB column in the current row of the CursorID.

Returned value

If the function succeeds, the return value is the size of BLOB data. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

If the table contains more than one BLOB column num must be specified. BLOB-column numbering in the row of the cursor starts from 1.

Linter_Last_Error

Gets the last Linter or system error code, according to type, from the cursor identified by CursorID.

Syntax

int Linter_Last_Error(CursorID, type);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.
Type	Type of the returned error is described in the table below.

Type	Value	Description
LINTER_ERROR	0	Linter’s error
SYSTEM_ERROR	1	System’s error

Returned value

The returned value is the last Linter error code or system error.

Linter_Error_Msg

Gets the text description of the last Linter error code for the cursor identified by CursorID.

Syntax

string | int Linter_Error_Msg(CursorID);

Parameter	Description
CursorID	ID for the cursor on which the request will be sent.

Returned value

If the function succeeds, the return value is a text message. Otherwise, the returned value is a negative integer. If the returned value is LPE_LINTER_ERROR, it is a Linter error. To get additional error information, use Linter_Last_Error. Another error code indicates an error in the module.

Remarks

The table SYSTEM.ERRORS must exist in the database.

Module Internal Errors

Constant name	Value	Description
LPE_SUCCESS	0	Successful completion.
LPE_LINTER_ERROR	-1	Linter SQL Server error has occurred.
LPE_NO_MEMORY	-2	Not enough memory.
LPE_I NVALID_PARAM	-3	Invalid parameter has been passed to function.
LPE_INVALID_CONNECT	-4	Invalid connection identifier.
LPE_I NVALID_CURSOR	-5	Invalid cursor identifier.
LPE_INVALID_COLNUM	-6	Invalid column number.
LPE_INVALID_PARAM_NUM	-7	Invalid parameter number.
LPE_I NVALID_PARAM_NAM E	-8	Invalid parameter name.
LPE_INVALID_STATE	-9	Invalid cursor state.
LPE_INVALID_DIRECT	-10	Invalid fetch direction.
LPE_UNKNOWN_OPTION	-11	Unknown cursor option.
LPE_BAD_DT_FORMAT	-12	Bad date/time format.
LPE_NO_BLOB_COLUMN	-13	No BLOB column in cursor.
LPE_NOT_CURSOR	-14	Not a cursor. May be a connection.
LPE_WRONG_PARAM_COU NT	-15	Wrong number of parameters has been passed to a function.
LPE_NOT_IMPLEMENTED	-999	Feature not implemented.