

DBMS_XMLSTORE

DBMS_XMLSTORE provides the ability to store XML data in relational tables.



Note:

The PL/SQL package DBMS_XMLSTORE is deprecated in Oracle Database 23ai.

DBMS_XMLSTORE is a non-standard Oracle-proprietary package that enables you to store and manipulate XML data in Oracle Database. This package is deprecated, and can be desupported in a future release. Oracle recommends that you use regular SQL DML and with standard XQuery and SQL/XML to store and manage XML data. Using standard functionality provides future-proof way to store and manipulate XML data.

This chapter contains the following sections:

- [Security Model](#)
- [Types](#)
- [Summary of DBMS_XMLSTORE Subprograms](#)



See Also:

Oracle XML DB Developer's Guide

DBMS_XMLSTORE Security Model

Owned by XDB, the DBMS_XMLSTORE package must be created by SYS or XDB. The EXECUTE privilege is granted to PUBLIC. Subprograms in this package are executed using the privileges of the current user.

Types

The DBMS_XMLSTORE subprograms use the ctxType Type.

Table 241-1 *Types of DBMS_XMLSTORE*

Type	Description
ctxType	The type of the query context handle. This is the return type of NEWCONTEXT .

Summary of DBMS_XMLSTORE Subprograms

This table lists the DBMS_XMLSTORE subprograms and briefly describes them.

Table 241-2 DBMS_XMLSTORE Package Subprograms

Method	Description
CLEARKEYCOLUMNLIST	Clears the key column list.
CLEARUPDATECOLUMNLIST	Clears the update column list.
CLOSECONTEXT	It closes/deallocates a particular save context.
DELETEXML	Deletes records specified by data from the XML document, from the table specified at the context creation time.
INSERTXML	Inserts the XML document into the table specified at the context creation time.
NEWCONTEXT	Creates a save context, and returns the context handle.
SETKEYCOLUMN	This method adds a column to the key column list.
SETROWTAG	Names the tag used in the XML document., to enclose the XML elements corresponding to the database.
SETUPDATECOLUMN	Adds a column to the "update column list".
UPDATERXML	Updates the table given the XML document.

CLEARKEYCOLUMNLIST

This procedure clears the key column list.

Syntax

```
PROCEDURE clearKeyColumnList(  
    ctxHdl IN ctxType);
```

Table 241-3 CLEARKEYCOLUMNLIST Procedure Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.

CLEARUPDATECOLUMNLIST

This procedure clears the update column list.

Syntax

```
PROCEDURE clearUpdateColumnList(  
    ctxHdl IN ctxType);
```

Table 241-4 CLEARUPDATECOLUMNLIST Procedure Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.

CLOSECONTEXT

This procedure closes/deallocates a particular save context.

Syntax

```
PROCEDURE closeContext(ctxHdl IN ctxType);
```

Table 241-5 CLOSECONTEXT Procedure Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.

DELETEXML

DELETEXML deletes records specified by data from the XML document from the table specified at the context creation time, and returns the number of rows deleted.

Syntax

The following syntax uses a `VARCHAR2` type for the `xDoc` parameter.

```
FUNCTION deleteXML(  
    ctxHdl IN ctxPType,  
    xDoc IN VARCHAR2)  
RETURN NUMBER;
```

The following syntax uses a `CLOB` type for the `xDoc` parameter.

```
FUNCTION deleteXML(  
    ctxHdl IN ctxType,  
    xDoc IN CLOB)  
RETURN NUMBER;
```

The following syntax uses an `XMLType` type for the `xDoc` parameter.

```
FUNCTION deleteXML(  
    ctxHdl IN ctxType,  
    xDoc IN XMLType)  
RETURN NUMBER;
```

Parameters

Table 241-6 DELETEXML Function Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.
xDoc	(IN)	String containing the XML document.

INSERTXML

Inserts the XML document into the table specified at the context creation time, and returns the number of rows inserted.

Note that if a user passes an XML file for `insertXML` to `DBMS_XMLSTORE` that contains extra elements (elements that do not match any columns in the table), Oracle tries to insert into those columns unless `SETUPDATECOLUMN` is used. The use of `setUpdateColumn` is optional only if the elements in the XML file match up to the columns in the table.

Syntax

```
FUNCTION insertXML(  
    ctxHdl IN ctxType,  
    xDoc IN VARCHAR2)  
RETURN NUMBER;
```

```
FUNCTION insertXML(  
    ctxHdl IN ctxType,  
    xDoc IN CLOB)  
RETURN NUMBER;
```

```
FUNCTION insertXML(  
    ctxHdl IN ctxType,  
    xDoc IN XMLType)  
RETURN NUMBER;
```

Parameters

Table 241-7 INSERTXML Function Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.
xDoc	(IN)	String containing the XML document.

NEWCONTEXT

`NEWCONTEXT` creates a save context and returns the context handle.

Syntax

```
FUNCTION newContext(  
    targetTable IN VARCHAR2)  
RETURN ctxType;
```

Table 241-8 NEWCONTEXT Function Parameters

Parameter	IN / OUT	Description
targetTable	(IN)	The target table into which to load the XML document.

SETKEYCOLUMN

This method adds a column to the "key column list".

The value for the column cannot be `NULL`. In case of update or delete, the columns in the key column list make up the `WHERE` clause of the statement. The key columns list must be specified before updates can complete; this is optional for delete operations

Syntax

```
PROCEDURE setKeyColumn(  
    ctxHdl IN ctxType,  
    colName IN VARCHAR2);
```

Table 241-9 SETKEYCOLUMN Procedure Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.
colName	(IN)	Column to be added to the key column list; cannot be <code>NULL</code> .

SETROWTAG

This procedure names the tag used in the XML document, to enclose the XML elements corresponding to database records.

Syntax

```
PROCEDURE setRowTag(  
    ctxHdl IN ctxType,  
    tag IN VARCHAR2);
```

Table 241-10 SETROWTAG Procedure Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.
tag	(IN)	Tag name.

SETUPDATECOLUMN

`SETUPDATECOLUMN` adds a column to the update column list.

In case of insert, the default is to insert values to all the columns in the table. In case of updates, the default is to only update the columns corresponding to the tags present in the `ROW` element of the XML document. When the update column list is specified, the columns making up this list alone will get updated or inserted into.

Note that if a user passes an XML file for `INSERTXML` to `DBMS_XMLSTORE` which contains extra elements (ones that do not match up to any columns in the table), Oracle will try to insert into those columns unless `setUpdateColumn` is used. The use of `setUpdateColumn` is optional only if the elements in the XML file match up to the columns in the table.

Syntax

```
PROCEDURE setUpdateColumn(  
    ctxHdl IN ctxType,  
    colName IN VARCHAR2);
```

Table 241-11 SETUPDATECOLUMN Procedure Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.
colName	(IN)	Column to be added to the update column list.

UPDATEXML

Updates the table specified at the context creation time with data from the XML document, and returns the number of rows updated.

The options are described in the following table.

Syntax

The following syntax passes the `xDoc` parameter as a `VARCHAR2`.

```
FUNCTION updateXML(  
    ctxHdl IN ctxType,  
    xDoc IN VARCHAR2)  
RETURN NUMBER;
```

The following syntax passes the `xDoc` parameter as a `CLOB`.

```
FUNCTION updateXML(  
    ctxHdl IN ctxType,  
    xDoc IN CLOB)  
RETURN NUMBER;
```

The following syntax passes the `xDoc` parameter as a `XMLType`.

```
FUNCTION updateXML(  
    ctxHdl IN ctxType,  
    xDoc IN XMLType)  
RETURN NUMBER;
```

Parameters

Table 241-12 UPDATEXML Function Parameters

Parameter	IN / OUT	Description
ctxHdl	(IN)	Context handle.
xDoc	(IN)	String containing the XML document.