

DBMS_XMLDOM

The `DBMS_XMLDOM` package is used to access `XMLType` objects, and implements the Document Object Model (DOM), an application programming interface for HTML and XML documents.

This chapter contains the following topics:

- [Overview](#)
- [Security Model](#)
- [Constants](#)
- [Types](#)
- [Exceptions](#)
- [Subprogram Groups](#)
- [Subprogram Groups](#)
- [Summary of DBMS_XMLDOM Subprograms](#)



See Also:

Oracle XML Developer's Kit Programmer's Guide

DBMS_XMLDOM Overview

The Document Object Model (DOM) is an application programming interface (API) for HTML and XML documents. It defines the logical structure of documents, and the manner in which they are accessed and manipulated.

In the DOM specification, the term "document" is used in the broad sense. XML is being increasingly used to represent many different kinds of information that may be stored in diverse systems. This information has been traditionally be seen as "data"; nevertheless, XML presents this data as documents, and the `DBMS_XMLDOM` package allows you access to both schema-based and non schema-based documents.



Note:

Read-from and write-to files must be on the server file system.

With DOM, anything found in an HTML or XML document can be accessed, changed, deleted, or added using the Document Object Model, with a few exceptions. In particular, the DOM interfaces for the XML internal and external subsets have not yet been specified.

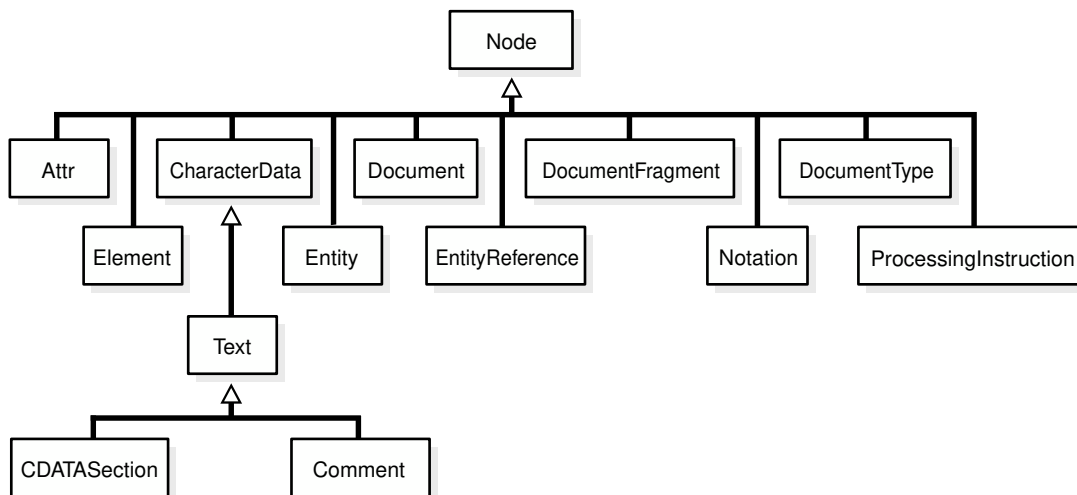
One important objective of the W3C DOM specification is to provide a standard programming interface that can be used in a wide variety of environments, programming languages, and

applications. Because the DOM standard is object-oriented while PL/SQL is essentially a procedural language, some changes had to be made:

- Various DOM interfaces such as `Node`, `Element`, and others have equivalent PL/SQL types `DOMNode`, `DOMElement`, respectively.
- Various `DOMException` codes such as `WRONG_DOCUMENT_ERR`, `HIERARCHY_REQUEST_ERR`, and others, have similarly named PL/SQL exceptions.
- Various DOM Node type codes such as `ELEMENT_NODE`, `ATTRIBUTE_NODE`, and others, have similarly named PL/SQL constants.
- Subprograms defined on a DOM type become functions or procedures that accept it as a parameter. For example, to perform [APPENDCHILD Function](#) on a `DOMNode` `n`, the [APPENDCHILD Function](#) PL/SQL function is provided.
- To perform `setAttribute` on a `DOMElement` `elem` [SETATTRIBUTE Procedures](#), use PL/SQL procedure .

DOM defines an inheritance hierarchy. For example, `Document`, `Element`, and `Attr` are defined to be subtypes of `Node` (see [Figure 233-1](#)). Thus, a method defined in the `Node` interface should be available in these as well. Since such inheritance is not supported in PL/SQL, it is implemented through direct invocation of the `MAKENODE` function. Calling `MAKENODE` on various DOM types converts these types into a `DOMNode`. The appropriate functions or procedures that accept `DOMNodes` can then be called to operate on these types. If, subsequently, type specific functionality is desired, the `DOMNode` can be converted back into the original type by the `makeXXX` functions, where `DOMXXX` is the desired DOM type.

Figure 233-1 Inheritance Diagram for DOM Types



The implementation of this interface follows the REC-DOM-Level-1-19981001.

DBMS_XMLDOM Security Model

Owned by `XDB`, the `DBMS_XMLDOM` 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.

DBMS_XMLDOM Constants

The `DBMS_XMLDOM` package defines several constants that can be used for specifying parameter values.

These constants are listed in the following table.

Table 233-1 Defined Constants for DBMS_XMLDOM

| Constant | Type | Value | Description |
|--|--------------------------|-------|---|
| <code>ELEMENT_NODE</code> | <code>PLS_INTEGER</code> | 1 | The Node is an Element. |
| <code>ATTRIBUTE_NODE</code> | <code>PLS_INTEGER</code> | 2 | The Node is an Attribute. |
| <code>TEXT_NODE</code> | <code>PLS_INTEGER</code> | 3 | The Node is a Text node. |
| <code>CDATA_SECTION_NODE</code> | <code>PLS_INTEGER</code> | 4 | The Node is a CDATASection. |
| <code>ENTITY_REFERENCE_NODE</code> | <code>PLS_INTEGER</code> | 5 | The Node is an Entity Reference. |
| <code>ENTITY_NODE</code> | <code>PLS_INTEGER</code> | 6 | The Node is an Entity. |
| <code>PROCESSING_INSTRUCTION_NODE</code> | <code>PLS_INTEGER</code> | 7 | The Node is a Processing Instruction. |
| <code>COMMENT_NODE</code> | <code>PLS_INTEGER</code> | 8 | The Node is a Comment. |
| <code>DOCUMENT_NODE</code> | <code>PLS_INTEGER</code> | 9 | The Node is a Document. |
| <code>DOCUMENT_TYPE_NODE</code> | <code>PLS_INTEGER</code> | 10 | The Node is a Document Type Definition. |
| <code>DOCUMENT_FRAGMENT_NODE</code> | <code>PLS_INTEGER</code> | 11 | The Node is a Document fragment. |
| <code>NOTATION_NODE</code> | <code>PLS_INTEGER</code> | 12 | The Node is a Notation. |

DBMS_XMLDOM Types

This table lists and briefly describes the types for the `DBMS_XMLDOM.DOMTYPE` package.

Table 233-2 XDB_XMLDOM Types

| Type | Description |
|----------------------------------|--|
| <code>DOMATTR</code> | Implements the DOM Attribute interface. |
| <code>DOMCDATASECTION</code> | Implements the DOM CDATASection interface. |
| <code>DOMCHARACTERDATA</code> | Implements the DOM Character Data interface. |
| <code>DOMCOMMENT</code> | Implements the DOM Comment interface. |
| <code>DOMDOCUMENT</code> | Implements the DOM Document interface. |
| <code>DOMDOCUMENTFRAGMENT</code> | Implements the DOM DocumentFragment interface. |
| <code>DOMDOCUMENTTYPE</code> | Implements the DOM Document Type interface. |
| <code>DOMELEMENT</code> | Implements the DOM Element interface. |
| <code>DOMENTITY</code> | Implements the DOM Entity interface. |
| <code>DOMENTITYREFERENCE</code> | Implements the DOM EntityReference interface. |

Table 233-2 (Cont.) XDB_XMLDOM Types

| Type | Description |
|--------------------------|--|
| DOMIMPLEMENTATION | Implements the DOM Implementation interface. |
| DOMNAMEDNODEMAP | Implements the DOM Named Node Map interface. |
| DOMNODE | Implements the DOM Node interface. |
| DOMNODELIST | Implements the DOM NodeList interface. |
| DOMNOTATION | Implements the DOM Notation interface. |
| DOMPROCESSINGINSTRUCTION | Implements the DOM Processing instruction interface. |
| DOMTEXT | Implements the DOM Text interface. |

DBMS_XMLDOM Exceptions

DBMS_XMLDOM generates an exception when it encounters an issue.

This table lists the exceptions defined for DBMS_XMLDOM:

Table 233-3 Exceptions for DBMS_XMLDOM

| Exception | Description |
|-----------------------------|---|
| DOMSTRING_SIZE_ERR | If the specified range of text does not fit into a DOMString. |
| HIERARCHY_REQUEST_ERR | If any node is inserted somewhere it doesn't belong. |
| INDEX_SIZE_ERR | If index or size is negative, or greater than the allowed value. |
| INUSE_ATTRIBUTE_ERR | If an attempt is made to add an attribute that is already in use elsewhere. |
| INVALID_CHARACTER_ERR | If an invalid or illegal character is specified, such as in a name. See production 2 in the XML specification for the definition of a legal character, and production 5 for the definition of a legal name character. |
| NO_DATA_ALLOWED_ERROR | If data is specified for a node that does not support data. |
| NOT_FOUND_ERR | If an attempt is made to reference a node in a context where it does not exist. |
| NO_MODIFICATION_ALLOWED_ERR | If an attempt is made to modify an object where modifications are not allowed. |
| NOT_SUPPORTED_ERR | If the implementation does not support the requested type of object or operation. |
| WRONG_DOCUMENT_ERR | If a node is used in a different document than the one that created it (that doesn't support it). |

DBMS_XMLDOM Subprogram Groups

DBMS_XMLDOM subprograms are divided into groups according to W3C Interfaces.

- [DOMNode Subprograms](#)
- [DOMAttr Subprograms](#)
- [DOMCDataSection Subprograms](#)

- [DOMCharacterData Subprograms](#)
- [DOMComment Subprograms](#)
- [DOMDocument Subprograms](#)
- [DOMDocumentFragment Subprograms](#)
- [DOMDocumentType Subprograms](#)
- [DOMElement Subprograms](#)
- [DOMEntity Subprograms](#)
- [DOMEntityReference Subprograms](#)
- [DOMImplementation Subprograms](#)
- [DOMNamedNodeMap Subprograms](#)
- [DOMNodeList Subprograms](#)
- [DOMNotation Subprograms](#)
- [DOMProcessingInstruction Subprograms](#)
- [DOMText Subprograms](#)

DBMS_XMLDOM DOMNode Subprograms

This table lists and briefly describes the `DOMNode` subprograms of `DBMS_XMLDOM`.

Table 233-4 Summary of DOMNode Subprograms; DBMS_XMLDOM

| Subprogram | Description |
|--|--|
| ADOPTNODE Function | Adopts a node from another document |
| APPENDCHILD Function | Appends a new child to the node |
| CLONENODE Function | Clones the node |
| FREENODE Procedure | Frees all resources associated with the node |
| GETATTRIBUTES Function | Retrieves the attributes of the node |
| GETCHILDNODES Function | Retrieves the children of the node |
| GETEXPANDEDNAME Procedure and Functions | Retrieves the expanded name of the node |
| GETFIRSTCHILD Function | Retrieves the first child of the node |
| GETLASTCHILD Function | Retrieves the last child of the node |
| GETLOCALNAME Procedure and Functions | Retrieves the local part of the qualified name |
| GETNAMESPACE Procedure and Functions | Retrieves the node's namespace URI |
| GETNEXTSIBLING Function | Retrieves the next sibling of the node |
| GETNODENAME Function | Retrieves the Name of the Node |
| GETNODETYPE Function | Retrieves the Type of the node |
| GETNODEVALUE Function | Retrieves the Value of the Node |
| GETNODEVALUEASBINARYSTREAM Function & Procedure | Retrieves Node Value as binary stream |
| GETNODEVALUEASCHARACTERSTREAM Function & Procedure | Retrieves Node Value as character stream |
| GETOWNERDOCUMENT Function | Retrieves the owner document of the node |

Table 233-4 (Cont.) Summary of DOMNode Subprograms; DBMS_XMLDOM

| Subprogram | Description |
|--|--|
| GETPARENTNODE Function | Retrieves the parent of this node |
| GETPREFIX Function | Retrieves the namespace prefix |
| GETPREVIOUSIBLING Function | Retrieves the previous sibling of the node |
| GETSCHEMANODE Function | Retrieves the associated schema URI |
| HASATTRIBUTES Function | Tests if the node has attributes |
| HASCHILDNODES Function | Tests if the node has child nodes |
| IMPORTNODE Function | Imports a node from another document |
| INSERTBEFORE Function | Inserts a child before the reference child |
| ISNULL Functions | Tests if the node is <code>NULL</code> |
| MAKEATTR Function | Casts the node to an Attribute |
| MAKECDATASECTION Function | Casts the node to a CData Section |
| MAKECHARACTERDATA Function | Casts the node to Character Data |
| MAKECOMMENT Function | Casts the node to a Comment |
| MAKEDOCUMENT Function | Casts the node to a DOM Document |
| MAKEDOCUMENTFRAGMENT Function | Casts the node to a DOM Document Fragment |
| MAKEDOCUMENTTYPE Function | Casts the node to a DOM Document Type |
| MAKEELEMENT Function | Casts the node to a DOM Element |
| MAKEENTITY Function | Casts the node to a DOM Entity |
| MAKEENTITYREFERENCE Function | Casts the node to a DOM Entity Reference |
| MAKENOTATION Function | Casts the node to a DOM Notation |
| MAKEPROCESSINGINSTRUCTION Function | Casts the node to a DOM Processing Instruction |
| MAKETEXT Function | Casts the node to a DOM Text |
| REMOVECHILD Function | Removes a specified child from a node |
| REPLACECHILD Function | Replaces the old child with a new child |
| SETNODEVALUE Procedure | Sets the Value of the node |
| SETNODEVALUEASBINARYSTREAM Function & Procedure | Sets the Node Value as binary stream |
| SETNODEVALUEASCHARACTERSTREAM Function & Procedure | Sets the Node Value as a character stream |
| SETPREFIX Procedure | Sets the namespace prefix |
| USEBINARYSTREAM Function | Establishes that the stream is valid |
| WRITETOBUFFER Procedures | Writes the contents of the node to a buffer |
| WRITETOCLOB Procedures | Writes the contents of the node to a <code>CLOB</code> |
| WRITETOFILE Procedures | Writes the contents of the node to a file |

DBMS_XMLDOM DOMAttr Subprograms

This table lists the DOMAttr subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-5 Summary of DOMAttr Subprograms; DBMS_XMLDOM

| Method | Description |
|---|---|
| GETEXPANDEDNAME Procedure and Functions | Retrieves the expanded name of the attribute |
| GETLOCALNAME Procedure and Functions | Retrieves the local name of the attribute |
| GETNAME Functions | Retrieves the name of the attribute |
| GETNAMESPACE Procedure and Functions | Retrieves the NS URI of the attribute |
| GETOWNERELEMENT Function | Retrieves the Element node, parent of the attribute |
| GETQUALIFIEDNAME Functions | Retrieves the Qualified Name of the attribute |
| GETSPECIFIED Function | Tests if attribute was specified in the element |
| GETVALUE Function | Retrieves the value of the attribute |
| ISNULL Functions | Tests if the Attribute node is NULL |
| MAKENODE Functions | Casts the Attribute to a node |
| SETVALUE Procedure | Sets the value of the attribute |

DBMS_XMLDOM DOMCDataSection Subprograms

This table lists the DOMCdata subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-6 Summary of DOMCdata Subprograms; DBMS_XMLDOM

| Method | Description |
|------------------------------------|-----------------------------------|
| ISNULL Functions | Tests if the CDataSection is NULL |
| MAKENODE Functions | Casts the CDataSection to a node |

DBMS_XMLDOM DOMCharacterData Subprograms

This table lists the DOMCharacterData subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-7 Summary of DOMCharacterData Subprograms; DBMS_XMLDOM

| Method | Description |
|---------------------------------------|---|
| APPENDDATA Procedure | Appends the specified data to the node data |
| DELETEDATA Procedure | Deletes the data from the specified offSets |
| GETDATA Functions | Retrieves the data of the node |
| GETLENGTH Functions | Retrieves the length of the data |
| INSERTDATA Procedure | Inserts the data in the node at the specified offSets |
| ISNULL Functions | Tests if the CharacterData is NULL |
| MAKENODE Functions | Casts the CharacterData to a node |
| REPLACEDATA Procedure | Changes a range of characters in the node |

Table 233-7 (Cont.) Summary of DOMCharacterData Subprograms; DBMS_XMLDOM

| Method | Description |
|--|-------------------------------------|
| SETDATA Procedures | Sets the data to the node |
| SUBSTRINGDATA Function | Retrieves the substring of the data |

DBMS_XMLDOM DOMComment Subprograms

The table lists the DOMComment subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-8 Summary of DOMComment Subprograms; DBMS_XMLDOM

| Method | Description |
|------------------------------------|------------------------------|
| ISNULL Functions | Tests if the comment is NULL |
| MAKENODE Functions | Casts the Comment to a node |

DBMS_XMLDOM DOMDocument Subprograms

This table lists the DOMDocument subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-9 Summary of DOMDocument Subprograms; DBMS_XMLDOM

| Method | Description |
|--|--|
| CREATEATTRIBUTE Functions | Creates an Attribute |
| CREATECDATASECTION Function | Creates a <code>CDataSection</code> node |
| CREATECOMMENT Function | Creates a Comment node |
| CREATEDOCUMENT Function | Creates a new Document |
| CREATEDOCUMENTFRAGMENT Function | Creates a new Document Fragment |
| CREATEELEMENT Functions | Creates a new Element |
| CREATEENTITYREFERENCE Function | Creates an Entity reference |
| CREATEPROCESSINGINSTRUCTION Function | Creates a Processing Instruction |
| CREATETEXTNODE Function | Creates a Text node |
| FREEDOCFRAG Procedure | Frees the document fragment |
| FREEDOCUMENT Procedure | Frees the document |
| GETCHARSET Function | Retrieves the character set of the DOM document |
| GETDOCTYPE Function | Retrieves the DTD of the document |
| GETDOCUMENTELEMENT Function | Retrieves the root element of the document |
| GETELEMENTSBYTAGNAME Functions | Retrieves <ul style="list-style-type: none"> the elements in the DOMNODELIST by tag name elements in the subtree of a DOMNODELIST by tagname |
| GETIMPLEMENTATION Function | Retrieves the DOM implementation |

Table 233-9 (Cont.) Summary of DOMDocument Subprograms; DBMS_XMLDOM

| Method | Description |
|--|---|
| GETSTANDALONE Function | Retrieves the standalone property of the document |
| GETVERSION Function | Retrieves the version of the document |
| GETXMLTYPE Function | Retrieves the <code>XMLType</code> associated with the DOM Document |
| ISNULL Functions | Tests if the document is <code>NULL</code> |
| MAKENODE Functions | Casts the document to a node |
| NEWDOMDOCUMENT Functions | Creates a new document |
| SETCHARSET Procedure | Sets the character set of the DOM document |
| SETDOCTYPE Procedure | Sets the DTD of the document |
| SETSTANDALONE Procedure | Sets the standalone property of the document |
| SETVERSION Procedure | Sets the version of the document |
| WRITETOBUFFER Procedures | Writes the document to a buffer |
| WRITETOCLOB Procedures | Writes the document to a <code>CLOB</code> |
| WRITETOFILE Procedures | Writes the document to a file |

DBMS_XMLDOM DOMDocumentFragment Subprograms

This table lists the DOMDocumentFragment subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-10 Summary of DOMDocumentFragment Subprograms; DBMS_XMLDOM

| Method | Description |
|--|--|
| FREEDOCFRAG Procedure | Frees the specified document fragment |
| ISNULL Functions | Tests if the DocumentFragment is <code>NULL</code> |
| MAKENODE Functions | Casts the Document Fragment to a node |
| WRITETOBUFFER Procedures | Writes the contents of a document fragment into a buffer |

DBMS_XMLDOM DOMDocumentType Subprograms

This table lists the DOMDocumentType subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-11 Summary of DOMDocumentType Subprograms; DBMS_XMLDOM

| Method | Description |
|---------------------------------------|--|
| FINDENTITY Function | Finds the specified entity in the document type |
| FINDNOTATION Function | Finds the specified notation in the document type |
| GETENTITIES Function | Retrieves the nodemap of entities in the Document type |
| GETNAME Functions | Retrieves the name of the Document type |

Table 233-11 (Cont.) Summary of DOMDocumentType Subprograms; DBMS_XMLDOM

| Method | Description |
|---------------------------------------|---|
| GETNOTATIONS Function | Retrieves the nodemap of the notations in the Document type |
| GETPUBLICID Functions | Retrieves the public ID of the document type |
| GETSYSTEMID Functions | Retrieves the system ID of the document type |
| ISNULL Functions | Tests if the Document Type is <code>NULL</code> |
| MAKENODE Functions | Casts the document type to a node |

DBMS_XMLDOM DOMELEMENT Subprograms

This table lists the DOMELEMENT subprograms of DBMS_XMLDOM in alphabetical order and briefly describes them.

Table 233-12 Summary of DOMELEMENT Subprograms; DBMS_XMLDOM

| Method | Description |
|---|--|
| FREEELEMENT Procedure | Frees memory allocated to a DOMELEMENT handle |
| GETATTRIBUTE Functions | Retrieves the attribute node by name |
| GETATTRIBUTENODE Functions | Retrieves the attribute node by name |
| GETCHILDRENBYTAGNAME Functions | Retrieves children of the element by tag name |
| GETELEMENTSBYTAGNAME Functions | Retrieves <ul style="list-style-type: none"> the elements in the DOMNODELIST by tag name elements in the subtree of a DOMNODELIST by tagname |
| GETEXPANDEDNAME Procedure and Functions | Retrieves the expanded name of the element |
| GETLOCALNAME Procedure and Functions | Retrieves the local name of the element |
| GETNAMESPACE Procedure and Functions | Retrieves the NS URI of the element |
| GETQUALIFIEDNAME Functions | Retrieves the qualified name of the element |
| GETTAGNAME Function | Retrieves the Tag name of the element |
| HASATTRIBUTE Functions | Tests if an attribute exists |
| ISNULL Functions | Tests if the Element is <code>NULL</code> |
| MAKENODE Functions | Casts the Element to a node |
| NORMALIZE Procedure | Normalizes the text children of the element |
| REMOVEATTRIBUTE Procedures | Removes the attribute specified by the name |
| REMOVEATTRIBUTENODE Function | Removes the attribute node in the element |
| RESOLVENAMESPACEPREFIX Function | Resolve the prefix to a namespace URI |
| SETATTRIBUTE Procedures | Sets the attribute specified by the name |
| SETATTRIBUTENODE Functions | Sets the attribute node in the element |

DBMS_XMLDOM DOMENTITY Subprograms

This table lists and briefly describes the DOMENTITY subprograms of DBMS_XMLDOM.

Table 233-13 Summary of DOMEntity Subprograms; DBMS_XMLDOM

| Method | Description |
|--|---|
| GETNOTATIONNAME Function | Retrieves the notation name of the entity |
| GETPUBLICID Functions | Retrieves the public Id of the entity |
| GETSYSTEMID Functions | Retrieves the system Id of the entity |
| ISNULL Functions | Tests if the Entity is NULL |
| MAKENODE Functions | Casts the Entity to a node |

DBMS_XMLDOM DOMEntityReference Subprograms

This table lists and briefly describes the `DOMEntityReference` subprograms of `DBMS_XMLDOM`.

Table 233-14 Summary of DOMEntityReference Subprograms; DBMS_XMLDOM

| Method | Description |
|------------------------------------|--|
| ISNULL Functions | Tests if the <code>DOMEntityReference</code> is NULL |
| MAKENODE Functions | Casts the <code>DOMEntityReference</code> to NULL |

DBMS_XMLDOM DOMImplementation Subprograms

This table lists and briefly describes the `DOMImplementation` subprograms of `DBMS_XMLDOM`.

Table 233-15 Summary of DOMImplementation Subprograms; DBMS_XMLDOM

| Method | Description |
|-------------------------------------|--|
| ISNULL Functions | Tests if the <code>DOMImplementation</code> node is NULL |
| HASFEATURE Function | Tests if the <code>DOMImplementation</code> implements a feature |

DBMS_XMLDOM DOMNamedNodeMap Subprograms

This table lists and briefly describes the `DOMNamedNodeMap` subprograms of `DBMS_XMLDOM`.

Table 233-16 Summary of DOMNamedNodeMap Subprograms; DBMS_XMLDOM

| Method | Description |
|--|--|
| GETLENGTH Functions | Retrieves the number of items in the map |
| GETNAMEDITEM Function | Retrieves the item specified by the name |
| ISNULL Functions | Tests if the <code>NamedNodeMap</code> is NULL |
| ITEM Functions | Retrieves the item given the index in the map |
| REMOVENAMEDITEM Function | Removes the item specified by name |
| SETNAMEDITEM Function | Sets the item in the map specified by the name |

DBMS_XMLDOM DOMNodeList Subprograms

This table lists and briefly describes the `DOMNodeList` subprograms of `DBMS_XMLDOM`.

Table 233-17 Summary of DOMNodeList Subprograms; DBMS_XMLDOM

| Method | Description |
|--|---|
| FREENODELIST Procedure | Frees all resources associated with a nodelist |
| GETLENGTH Functions | Retrieves the number of items in the list |
| ISNULL Functions | Tests if the <code>NodeList</code> is <code>NULL</code> |
| ITEM Functions | Retrieves the item given the index in the list |

DBMS_XMLDOM DOMNotation Subprograms

This table lists and briefly describes the `DOMNotation` subprograms of `DBMS_XMLDOM`.

Table 233-18 Summary of DOMNotation Subprograms; DBMS_XMLDOM

| Method | Description |
|---------------------------------------|---|
| GETPUBLICID Functions | Retrieves the public Id of the notation |
| GETSYSTEMID Functions | Retrieves the system Id of the notation |
| ISNULL Functions | Tests if the <code>Notation</code> is <code>NULL</code> |
| MAKENODE Functions | Casts the notation to a node |

DBMS_XMLDOM DOMProcessingInstruction Subprograms

This table lists and briefly describes the `DOMProcessingInstruction` subprograms of `DBMS_XMLDOM`.

Table 233-19 Summary of DOMProcessingInstruction Subprograms; DBMS_XMLDOM

| Method | Description |
|------------------------------------|---|
| GETDATA Functions | Retrieves the data of the processing instruction |
| GETTARGET Function | Retrieves the target of the processing instruction |
| ISNULL Functions | Tests if the <code>Processing Instruction</code> is <code>NULL</code> |
| MAKENODE Functions | Casts the <code>Processing Instruction</code> to a node |
| SETDATA Procedures | Sets the data of the processing instruction |

DBMS_XMLDOM DOMText Subprograms

This table lists and briefly describes the `DOMText` subprograms of `DBMS_XMLDOM`.

Table 233-20 Summary of DOMText Subprograms; DBMS_XMLDOM

| Method | Description |
|------------------------------------|--|
| ISNULL Functions | Tests if the text is NULL |
| MAKENODE Functions | Casts the text to a node |
| SPLITTEXT Function | Splits the contents of the text node into 2 text nodes |

Summary of DBMS_XMLDOM Subprograms

This table lists the DBMS_XMLDOM subprograms and briefly describes them.

Table 233-21 Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|--|---|---|
| ADOPTNODE Function | Adopts a node from another document | DOMNode Subprograms |
| APPENDCHILD Function | Appends a new child to the node | DOMNode Subprograms |
| APPENDDATA Procedure | Appends the specified data to the node data | DOMCharacterData Subprograms |
| CLONENODE Function | Clones the node | DOMNode Subprograms |
| CREATEATTRIBUTE Functions | Creates an Attribute | DOMDocument Subprograms |
| CREATECDATASECTION Function | Creates a CDataSection node | DOMDocument Subprograms |
| CREATECOMMENT Function | Creates a Comment node | DOMDocument Subprograms |
| CREATEDOCUMENT Function | Creates a new Document | DOMDocument Subprograms |
| CREATEDOCUMENTFRAGMENT Function | Creates a new Document Fragment | DOMDocument Subprograms |
| CREATEELEMENT Functions | Creates a new Element | DOMDocument Subprograms |
| CREATEENTITYREFERENCE Function | Creates an Entity reference | DOMDocument Subprograms |
| CREATEPROCESSINGINSTRUCTION Function | Creates a Processing Instruction | DOMDocument Subprograms |
| CREATETEXTNODE Function | Creates a Text node | DOMDocument Subprograms |
| DELETEDATA Procedure | Deletes the data from the specified offSets | DOMCharacterData Subprograms |
| FINDENTITY Function | Finds the specified entity in the document type | DOMDocumentType Subprograms |
| FINDNOTATION Function | Finds the specified notation in the document type | DOMDocumentType Subprograms |
| FREEDOCFRAG Procedure | Frees the document fragment | DOMDocument Subprograms and DOMDocumentFragment Subprograms |
| FREEDOCUMENT Procedure | Frees the document | DOMDocument Subprograms |
| FREEELEMENT Procedure | Frees memory allocated to a DOMELEMENT handle | DOMELEMENT Subprograms |
| FREENODE Procedure | Frees all resources associated with the node | DOMNode Subprograms |

Table 233-21 (Cont.) Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|---|--|--|
| FREENODELIST Procedure | Frees all resources associated with a nodelist | DOMNodeList Subprograms |
| GETATTRIBUTE Functions | Retrieves the attribute node by name | DOMELEMENT Subprograms |
| GETATTRIBUTENODE Functions | Retrieves the attribute node by name | DOMELEMENT Subprograms |
| GETATTRIBUTES Function | Retrieves the attributes of the node | DOMNode Subprograms |
| GETCHARSET Function | Retrieves the character set of the DOM document | DOMDocument Subprograms |
| GETCHILDNODES Function | Retrieves the children of the node | DOMNode Subprograms |
| GETCHILDRENBYTAGNAME Functions | Retrieves children of the element by tag name | DOMCharacterData Subprograms |
| GETDATA Functions | Retrieves <ul style="list-style-type: none"> the data of the node the data of the processing instruction | <ul style="list-style-type: none"> DOMCharacterData Subprograms DOMProcessingInstruction Subprograms |
| GETDOCTYPE Function | Retrieves the DTD of the document | DOMDocument Subprograms |
| GETDOCUMENTELEMENT Function | Retrieves the root element of the document | DOMDocument Subprograms |
| GETELEMENTSBYTAGNAME Functions | Retrieves <ul style="list-style-type: none"> the elements in the DOMNodelist by tag name elements in the subtree of a DOMNodelist by tag name | <ul style="list-style-type: none"> DOMDocument Subprograms DOMELEMENT Subprograms |
| GETENTITIES Function | Retrieves the nodelist of entities in the Document type | DOMDocumentType Subprograms |
| GETEXPANDEDNAME Procedure and Functions | Retrieves <ul style="list-style-type: none"> the expanded name of the node the expanded name of the attribute the expanded name of the element | <ul style="list-style-type: none"> DOMNode Subprograms DOMAttr Subprograms DOMELEMENT Subprograms |
| GETFIRSTCHILD Function | Retrieves the first child of the node | DOMNode Subprograms |
| GETIMPLEMENTATION Function | Retrieves the DOM implementation | DOMDocument Subprograms |
| GETLASTCHILD Function | Retrieves the last child of the node | DOMNode Subprograms |
| GETLENGTH Functions | Retrieves <ul style="list-style-type: none"> the length of the data the number of items in the map the number of items in the list | <ul style="list-style-type: none"> DOMCharacterData Subprograms DOMNamedNodeMap Subprograms DOMNodeList Subprograms |
| GETLOCALNAME Procedure and Functions | Retrieves <ul style="list-style-type: none"> the local part of the qualified name the local name of the attribute the local name of the element | <ul style="list-style-type: none"> DOMNode Subprograms DOMAttr Subprograms DOMELEMENT Subprograms |

Table 233-21 (Cont.) Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|--|--|---|
| GETNAME Functions | Retrieves <ul style="list-style-type: none"> the name of the attribute the name of the Document type | <ul style="list-style-type: none"> DOMAttr Subprograms DOMDocumentType Subprograms |
| GETNAMEDITEM Function | Retrieves <ul style="list-style-type: none"> an item specified by name and namespace URI) | <ul style="list-style-type: none"> DOMNamedNodeMap Subprograms DOMNamedNodeMap Subprograms |
| GETNAMESPACE Procedure and Functions | Retrieves <ul style="list-style-type: none"> the node's namespace URI the NS URI of the attribute the NS URI of the element | <ul style="list-style-type: none"> DOMNode Subprograms DOMAttr Subprograms DOMElement Subprograms |
| GETNEXTSIBLING Function | Retrieves the next sibling of the node | DOMNode Subprograms |
| GETNODENAME Function | Retrieves the Name of the Node | DOMNode Subprograms |
| GETNODETYPE Function | Retrieves the Type of the node | DOMNode Subprograms |
| GETNODEVALUE Function | Retrieves the Value of the Node | DOMNode Subprograms |
| GETNODEVALUEASBINARYSTREAM Function & Procedure | Retrieves the Node Value as binary stream | DOMNode Subprograms |
| GETNODEVALUEASCHARACTERSTREAM Function & Procedure | Retrieves the Node Value as character stream | DOMNode Subprograms |
| GETNOTATIONNAME Function | Retrieves the notation name of the entity | DOMEntity Subprograms |
| GETNOTATIONS Function | Retrieves the nodemap of the notations in the Document type | DOMDocumentType Subprograms |
| GETTARGET Function | Retrieves the target of the processing instruction | DOMProcessingInstruction Subprograms |
| GETOWNERDOCUMENT Function | Retrieves the owner document of the node | DOMNode Subprograms |
| GETOWNERELEMENT Function | Retrieves the Element node, parent of the attribute | DOMAttr Subprograms |
| GETPARENTNODE Function | Retrieves the parent of this node | DOMNode Subprograms |
| GETPREFIX Function | Retrieves the namespace prefix) | DOMNode Subprograms |
| GETPREVIOUSIBLING Function | Retrieves the previous sibling of the node | DOMNode Subprograms |
| GETPUBLICID Functions | Retrieves <ul style="list-style-type: none"> the public ID of the document type the public Id of the entity the public Id of the notation | <ul style="list-style-type: none"> DOMDocumentType Subprograms DOMEntity Subprograms DOMNotation Subprograms |
| GETQUALIFIEDNAME Functions | Retrieves <ul style="list-style-type: none"> the Qualified Name of the attribute the qualified name of the element | <ul style="list-style-type: none"> DOMAttr Subprograms DOMElement Subprograms |
| GETSCHEMANODE Function | Retrieves the associated schema URI | DOMNode Subprograms |

Table 233-21 (Cont.) Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|--|--|---|
| GETSPECIFIED Function | Tests if attribute was specified in the element. | DOMAttr Subprograms |
| GETSTANDALONE Function | Retrieves the standalone property of the document | DOMDocument Subprograms |
| GETSYSTEMID Functions | Retrieves <ul style="list-style-type: none"> the system ID of the document type the system Id of the entity the system Id of the notation | <ul style="list-style-type: none"> DOMDocumentType Subprograms DOMEntity Subprograms DOMNotation Subprograms |
| GETTAGNAME Function | Retrieves the Tag name of the element | DOMELEMENT Subprograms |
| GETVALUE Function | Retrieves the value of the attribute | DOMAttr Subprograms |
| GETVERSION Function | Retrieves the version of the document | DOMDocument Subprograms) |
| GETXMLTYPE Function | Retrieves the <code>XMLType</code> associated with the DOM Document | DOMDocument Subprograms |
| HASATTRIBUTES Function | Tests if the node has attributes | DOMNode Subprograms |
| HASATTRIBUTE Functions | Tests if an attribute exists | DOMELEMENT Subprograms |
| HASCHILDNODES Function | Tests if the node has child nodes | DOMNode Subprograms |
| HASFEATURE Function | Tests if the <code>DOMImplementation</code> implements a feature | DOMImplementation Subprograms |
| IMPORTNODE Function | Imports a node from another document | DOMNode Subprograms |
| INSERTBEFORE Function | Inserts a child before the reference child | DOMNode Subprograms |
| INSERTDATA Procedure | Inserts the data in the node at the specified offSets | DOMCharacterData Subprograms |

Table 233-21 (Cont.) Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|-------------------------------|---|--|
| ISNULL Functions | Tests <ul style="list-style-type: none"> if the node is NULL if the Attribute node is NULL if the CDataSection is NULL if the CharacterData is NULL if the comment is NULL if the document is NULL if the DocumentFragment is NULL if the Document Type is NULL if the Element is NULL if the Entity is NULL if the DOMEntityReference is NULL if the DOMImplementation node is NULL if the NamedNodeMap is NULL if the NodeList is NULL if the Notation is NULL if the Processing Instruction is NULL if the text is NULL | <ul style="list-style-type: none"> DOMNode Subprograms DOMAttr Subprograms DOMCDataSection Subprograms DOMCharacterData Subprograms DOMComment Subprograms DOMDocument Subprograms DOMDocumentFragment Subprograms DOMDocumentType Subprograms DOMElement Subprograms DOMEntity Subprograms DOMEntityReference Subprograms DOMImplementation Subprograms DOMNamedNodeMap Subprograms DOMNodeList Subprograms DOMNotation Subprograms DOMProcessingInstruction Subprograms DOMText Subprograms |
| ITEM Functions | Retrieves <ul style="list-style-type: none"> the item given the index in the map the item given the index in the NodeList | <ul style="list-style-type: none"> DOMNamedNodeMap Subprograms DOMNodeList Subprograms |
| MAKEATTR Function | Casts the node to an Attribute | DOMNode Subprograms |
| MAKECDATASECTION Function | Casts the node to a CData Section | DOMNode Subprograms |
| MAKECHARACTERDATA Function | Casts the node to Character Data | DOMNode Subprograms |
| MAKECOMMENT Function | Casts the node to a Comment | DOMNode Subprograms |
| MAKEDOCUMENT Function | Casts the node to a DOM Document | DOMNode Subprograms |
| MAKEDOCUMENTFRAGMENT Function | Casts the node to a DOM Document Fragment | DOMNode Subprograms) |
| MAKEDOCUMENTTYPE Function | Casts the node to a DOM Document Type | DOMNode Subprograms |
| MAKEELEMENT Function | Casts the node to a DOM Element | DOMNode Subprograms |
| MAKEENTITY Function | Casts the node to a DOM Entity | DOMNode Subprograms |
| MAKEENTITYREFERENCE Function | Casts the node to a DOM Entity Reference | DOMNode Subprograms |

Table 233-21 (Cont.) Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|--|--|---|
| MAKENODE Functions | Casts <ul style="list-style-type: none"> the Attribute to a node the CDataSection to a node the CharacterData to a node the Comment to a node the document to a node the Document Fragment to a node the document type to a node the Element to a node the Entity to a node the DOMEntityReference to NULL the notation to a node the Processing Instruction to a node the text to a node | <ul style="list-style-type: none"> DOMAttr Subprograms DOMCDataSection Subprograms DOMCharacterData Subprograms DOMComment Subprograms DOMDocument Subprograms DOMDocumentFragment Subprograms DOMDocumentType Subprograms DOMEElement Subprograms DOMEntity Subprograms DOMEntityReference Subprograms DOMNotation Subprograms DOMProcessingInstruction Subprograms DOMText Subprograms |
| MAKENOTATION Function | Cast the node to a DOM Notation | DOMNode Subprograms |
| MAKEPROCESSINGINSTRUCTION Function | Cast the node to a DOM Processing Instruction | DOMNode Subprograms |
| MAKETEXT Function | Cast the node to a DOM Text | DOMNode Subprograms |
| NEWDOMDOCUMENT Functions | Creates a new document | DOMDocument Subprograms |
| NORMALIZE Procedure | Normalizes the text children of the element | DOMEElement Subprograms |
| REMOVEATTRIBUTE Procedures | Removes the attribute specified by the name | DOMEElement Subprograms |
| REMOVEATTRIBUTENODE Function | Removes the attribute node in the element | DOMEElement Subprograms |
| REMOVECHILD Function | Removes a specified child from a node | DOMNode Subprograms |
| REMOVENAMEDITEM Function | Removes the item specified by name | DOMNamedNodeMap Subprograms |
| REPLACECHILD Function | Replaces the old child with a new child | DOMNode Subprograms |
| REPLACEDATA Procedure | Changes a range of characters in the node | DOMCharacterData Subprograms |
| RESOLVENAMESPACEPREFIX Function | Resolve the prefix to a namespace URI | DOMEElement Subprograms |
| SETATTRIBUTE Procedures | Sets the attribute specified by the name | DOMEElement Subprograms |
| SETATTRIBUTENODE Functions | Sets the attribute node in the element | DOMEElement Subprograms |

Table 233-21 (Cont.) Summary of DBMS_XMLDOM Package Subprogram

| Subprogram | Description | Group |
|--|--|---|
| SETCHARSET Procedure | Sets the character set of the DOM document | DOMDocument Subprograms |
| SETDATA Procedures | Sets <ul style="list-style-type: none"> the data to the node the data of the processing instruction | <ul style="list-style-type: none"> DOMCharacterData Subprograms DOMProcessingInstruction Subprograms |
| SETDOCTYPE Procedure | Sets the DTD of the document. | DOMDocument Subprograms |
| SETNAMEDITEM Function | Sets the item in the map specified by the name | DOMNamedNodeMap Subprograms |
| SETNODEVALUE Procedure | Sets the Value of the node | DOMNode Subprograms |
| SETNODEVALUEASBINARYSTREAM Function & Procedure | Sets the Node Value as a binary stream | DOMNode Subprograms |
| SETNODEVALUEASCHARACTERSTREAM Function & Procedure | Sets the Node Value as a character stream | DOMNode Subprograms |
| SETPREFIX Procedure | Sets the namespace prefix | DOMNode Subprograms |
| SETSTANDALONE Procedure | Sets the standalone property of the document | DOMDocument Subprograms |
| SETVALUE Procedure | Sets the value of the attribute | DOMAttr Subprograms |
| SETVERSION Procedure | Sets the version of the document | DOMDocument Subprograms |
| SPLITTEXT Function | Splits the contents of the text node into 2 text nodes | DOMText Subprograms |
| SUBSTRINGDATA Function | Retrieves the substring of the data | DOMCharacterData Subprograms |
| USEBINARYSTREAM Function | Strabismus that the stream is valid for use | DOMNode Subprograms |
| WRITETOBUFFER Procedures | Writes <ul style="list-style-type: none"> the contents of the node to a buffer the document to a buffer the contents of a document fragment into a buffer | <ul style="list-style-type: none"> DOMNode Subprograms DOMDocument Subprograms DOMDocumentFragment Subprograms |
| WRITETOCLOB Procedures | Writes <ul style="list-style-type: none"> the contents of the node to a CLOB the document to a CLOB | <ul style="list-style-type: none"> DOMNode Subprograms DOMDocument Subprograms |
| WRITETOFILE Procedures | Writes <ul style="list-style-type: none"> the contents of the node to a file the document to a file | <ul style="list-style-type: none"> DOMNode Subprograms DOMDocument Subprograms |

ADOPTNODE Function

This function adopts a node from another document, and returns this new node.



See Also:

[DOMNode Subprograms](#) for other subprograms in this group

Syntax

```
DBMS_XMLDOM.ADOPTNODE (  
    doc             IN   DOMDocument,  
    importedNode    IN   DOMNode)  
RETURN DOMNODE;
```

Parameters

Table 233-22 ADOPTNODE Function Parameters

| Parameter | Description |
|--------------|------------------------------------|
| doc | Document that is adopting the node |
| importedNode | Node to adopt |

Usage Notes

Note that the [ADOPTNODE Function](#) removes the node from the source document while the [IMPORTNODE Function](#) clones the node in the source document.

APPENDCHILD Function

This function adds the node `newchild` to the end of the list of children of this node, and returns the newly added node. If the `newchild` is already in the tree, it is first removed.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.APPENDCHILD (  
    n             IN   DOMNode,  
    newchild      IN   DOMNode)  
RETURN DOMNODE;
```

Parameters

Table 233-23 APPENDCHILD Function Parameters

| Parameter | Description |
|-----------|--|
| n | DOMNode |
| newchild | The child to be appended to the list of children of node n |

APPENDDATA Procedure

This procedure appends the string to the end of the character data of the node. Upon success, data provides access to the concatenation of data and the specified string argument.



See Also:

[DOMCharacterData Subprograms](#)

Syntax

```
DBMS_XMLDOM.APPENDDATA (  
    cd      IN      DOMCHARACTERDATA,  
    arg     IN      VARCHAR2);
```

Parameters

Table 233-24 APPENDDATA Procedure Parameters

| Parameter | Description |
|-----------|---|
| cd | DOMCHARACTERDATA |
| arg | The data to append to the existing data |

CLONENODE Function

This function returns a duplicate of this node, and serves as a generic copy constructor for nodes. The duplicate node has no parent, its parent node is `NULL`.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.CLONENODE (  
    n      IN      DOMNODE,  
    deep   IN      BOOLEAN)  
RETURN DOMNODE;
```

Parameters

Table 233-25 CLONENODE Function Parameters

| Parameter | Description |
|-----------|---|
| n | DOMNODE |
| deep | Determines if children are to be cloned |

Usage Notes

- Cloning an Element copies all attributes and their values, including those generated by the XML processor to represent defaulted attributes, but this method does not copy any text it contains unless it is a deep clone, since the text is contained in a child Text node.
- Cloning an Attribute directly, as opposed to be cloned as part of an Element cloning operation, returns a specified attribute (specified is `TRUE`).
- Cloning any other type of node simply returns a copy of this node.

CREATEATTRIBUTE Functions

This function creates a `DOMATTR` node.



See Also:

[DOMDocument Subprograms](#)

Syntax

Creates a `DOMATTR` with the specified name:

```
DBMS_XMLDOM.CREATEATTRIBUTE (  
    doc      IN      DOMDOCUMENT,  
    name     IN      VARCHAR2)  
RETURN DOMATTR;
```

Creates a `DOMATTR` with the specified name and namespace URI:

```
DBMS_XMLDOM.CREATEATTRIBUTE (  
    doc      IN      DOMDOCUMENT,  
    qname    IN      VARCHAR2,  
    ns       IN      VARCHAR2)  
RETURN DOMATTR;
```

Parameters

Table 233-26 CREATEATTRIBUTE Function Parameters

| Parameter | Description |
|-----------|------------------------------|
| doc | DOMDOCUMENT |
| qname | New attribute qualified name |

Table 233-26 (Cont.) CREATEATTRIBUTE Function Parameters

| Parameter | Description |
|-----------|-------------|
| ns | Namespace |

CREATEDATASECTION Function

This function creates a `DOMCDATASECTION` node.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATEDATASECTION (  
    doc      IN      DOMDOCUMENT,  
    data     IN      VARCHAR2)  
RETURN DOMCDATASECTION;
```

Parameters

Table 233-27 CREATEDATASECTION Function Parameters

| Parameter | Description |
|-----------|--|
| doc | DOMDOCUMENT |
| data | Content of the <code>DOMCDATASECTION</code> node |

CREATECOMMENT Function

This function creates a `DOMCOMMENT` node.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATECOMMENT (  
    doc      IN      DOMDOCUMENT,  
    data     IN      VARCHAR2)  
RETURN DOMCOMMENT;
```

Parameters

Table 233-28 CREATECOMMENT Function Parameters

| Parameter | Description |
|-----------|--------------------------------|
| doc | DOMDOCUMENT |
| data | Content of the DOMComment node |

CREATEDOCUMENT Function

This function creates a DOMDOCUMENT with specified namespace URI, root element name, DTD.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATEDOCUMENT (  
    namespaceURI      IN      VARCHAR2,  
    qualifiedName      IN      VARCHAR2,  
    doctype            IN      DOMTYPE := NULL)  
RETURN DOMDOCUMENT;
```

Parameters

Table 233-29 CREATEDOCUMENT Function Parameters

| Parameter | Description |
|---------------|-------------------|
| namespaceURI | Namespace URI |
| qualifiedName | Root element name |
| doctype | Document type |

CREATEDOCUMENTFRAGMENT Function

This function creates a DOMDOCUMENTFRAGMENT.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATEDOCUMENTFRAGMENT (  
    doc      IN      DOMDOCUMENT)  
RETURN DOMDOCUMENTFRAGMENT;
```


Parameters

Table 233-30 CREATEDOCUMENTFRAGMENT Function Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDocument |

CREATEELEMENT Functions

This function creates a `DOMELEMENT`.



See Also:

[DOMDocument Subprograms](#)

Syntax

Creates a `DOMElement` with specified name:

```
DBMS_XMLDOM.CREATEELEMENT (  
    doc          IN      DOMDOCUMENT,  
    tagName      IN      VARCHAR2)  
RETURN DOMELEMENT;
```

Creates a `DOMElement` with specified name and namespace URI:

```
DBMS_XMLDOM.CREATEELEMENT (  
    doc          IN      DOMDOCUMENT,  
    tagName      IN      VARCHAR2,  
    ns           IN      VARCHAR2)  
RETURN DOMELEMENT;
```

Parameters

Table 233-31 CREATEELEMENT Function Parameters

| Parameter | Description |
|-----------|---|
| doc | DOMDOCUMENT |
| tagName | Tagname for new <code>DOMELEMENT</code> |
| ns | Namespace |

CREATEENTITYREFERENCE Function

This function creates a `DOMENTITYREFERENCE` node.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATEENTITYREFERENCE (  
    doc          IN      DOMDOCUMENT,  
    name         IN      VARCHAR2)  
RETURN DOMENTITYREFERENCE;
```

Parameters

Table 233-32 CREATEENTITYREFERENCE Function Parameters

| Parameter | Description |
|-----------|---------------------------|
| doc | DOMDOCUMENT |
| name | New entity reference name |

CREATEPROCESSINGINSTRUCTION Function

This function creates a `DOMPROCESSINGINSTRUCTION` node.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATEPROCESSINGINSTRUCTION (  
    doc          IN      DOMDocument,  
    target       IN      VARCHAR2,  
    data         IN      VARCHAR2)  
RETURN DOMPROCESSINGINSTRUCTION;
```

Parameters

Table 233-33 CREATEPROCESSINGINSTRUCTION Function Parameters

| Parameter | Description |
|-----------|--|
| doc | DOMDOCUMENT |
| target | Target of the new processing instruction |

Table 233-33 (Cont.) CREATEPROCESSINGINSTRUCTION Function Parameters

| Parameter | Description |
|-----------|--|
| data | Content data of the new processing instruction |

CREATETEXTNODE Function

This function creates a `DOMTEXT` node.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.CREATETEXTNODE(  
    doc      IN      DOMDocument,  
    data     IN      VARCHAR2)  
RETURN DOMTEXT;
```

Parameters

Table 233-34 CREATETEXTNODE Function Parameters

| Parameter | Description |
|-----------|--|
| doc | DOMDOCUMENT |
| data | Content of the <code>DOMText</code> node |

DELETEDATA Procedure

This procedure removes a range of characters from the node. Upon success, data and length reflect the change.



See Also:

[DOMCharacterData Subprograms](#)

Syntax

```
DBMS_XMLDOM.DELETEDATA(  
    cd      IN      DOMCHARACTERDATA,  
    offset  IN      NUMBER,  
    cnt     IN      NUMBER);
```

Parameters

Table 233-35 DELETEDATA PROCEDURE Parameters

| Parameter | Description |
|-----------|---|
| cd | DOMCHARACTERDATA |
| offset | The offset from which to delete the data |
| cnt | The number of characters (starting from offset) to delete |

FINDENTITY Function

This function finds an entity in the specified DTD, and returns that entity if found.



See Also:

[DOMDocumentType Subprograms](#)

Syntax

```
DBMS_XMLDOM.FINDENTITY (  
    dt      IN      DOMDOCUMENTTYPE,  
    name    IN      VARCHAR2,  
    par     IN      BOOLEAN)  
RETURN DUMENTITY;
```

Parameters

Table 233-36 FINDENTITY Function Parameters

| Parameter | Description |
|-----------|--|
| dt | The DTD |
| name | Entity to find |
| par | Flag to indicate type of entity; TRUE for parameter entity and FALSE for normal entity |

FINDNOTATION Function

This function finds the notation in the specified DTD, and returns it, if found.



See Also:

[DOMDocumentType Subprograms](#)

Syntax

```
DBMS_XMLDOM.FINDNOTATION(  
    dt          IN      DOMDocumentType,  
    name        IN      VARCHAR2)  
RETURN DOMNOTATION;
```

Parameters

Table 233-37 FINDNOTATION Function Parameters

| Parameter | Description |
|-----------|----------------------|
| dt | The DTD |
| name | The notation to find |

FREEDOCFRAG Procedure

This procedure frees the specified document fragment.



See Also:

[DOMDocument Subprograms](#) and [DOMDocumentFragment Subprograms](#)

Syntax

```
DBMS_XMLDOM.FREEDOCFRAG(  
    df          IN      DOMDOCUMENTFRAGMENT);
```

Parameters

Table 233-38 FREEDOCFRAG Procedure Parameters

| Parameter | Description |
|-----------|-----------------------|
| df | DOM document fragment |

FREEDOCUMENT Procedure

This procedure frees DOMDOCUMENT object.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.FREEDOCUMENT(  
    doc         IN      DOMDOCUMENT);
```

Parameters

Table 233-39 FREEDOCUMENT Procedure Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDOCUMENT |

FREEELEMENT Procedure

This procedure frees memory allocated to a DOMELEMENT handle.



See Also:

[DBMS_XMLDOM DOMELEMENT Subprograms](#)

Syntax

```
DBMS_XMLDOM.FREEELEMENT (
    elem IN DOMELEMENT);
```

Parameters

Table 233-40 FREEELEMENT Procedure Parameters

| Parameter | Description |
|-----------|--------------------|
| elem | Of type DOMELEMENT |

FREENODE Procedure

This procedure frees all resources associated with a DOMNODE.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.FREENODE (
    n      IN      DOMNODE);
```

Parameters

Table 233-41 FREENODE Procedure Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

FREENODELIST Procedure

This procedure frees all resources associated with a nodelist.



See Also:

[DBMS_XMLDOM DOMNodeList Subprograms](#)

Syntax

```
DBMS_XMLDOM.FREENODELIST(  
    nl IN DOMNodeList);
```

Parameters

Table 233-42 FREENODELIST Procedure Parameters

| Parameter | Description |
|-----------|---------------------|
| nl | Of type DOMNODELIST |

GETATTRIBUTE Functions

This function returns the value of an attribute of an DOMELEMENT by name.



See Also:

[DOMELEMENT Subprograms](#)

Syntax

Returns the value of a DOMELEMENT's attribute by name:

```
DBMS_XMLDOM.GETATTRIBUTE(  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2)  
RETURN VARCHAR2;
```

Returns the value of a DOMELEMENT's attribute by name and namespace URI:

```
DBMS_XMLDOM.GETATTRIBUTE(  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2,  
    ns        IN      VARCHAR2)  
RETURN VARCHAR2;
```

Parameters

Table 233-43 GETATTRIBUTE Function Parameters

| Parameter | Description |
|-----------|----------------|
| elem | The DOMELEMENT |
| name | Attribute name |
| ns | Namespace |

GETATTRIBUTENODE Functions

This function returns an attribute node from the DOMELEMENT by name. The function is overloaded. The specific forms of functionality are described along with the syntax declarations.



See Also:

[DOMELEMENT Subprograms](#)

Syntax

Returns an attribute node from the DOMELEMENT by name:

```
DBMS_XMLDOM.GETATTRIBUTENODE(  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2)  
RETURN DOMATTR;
```

Returns an attribute node from the DOMELEMENT by name and namespace URI:

```
DBMS_XMLDOM.GETATTRIBUTENODE(  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2,  
    ns        IN      VARCHAR2)  
RETURN DOMATTR;
```

Parameters

Table 233-44 GETATTRIBUTENODE Function Parameters

| Parameter | Description |
|-----------|---|
| elem | The DOMELEMENT |
| name | Attribute name; * matches any attribute |
| ns | Namespace |

GETATTRIBUTES Function

This function retrieves a `NAMEDNODEMAP` containing the attributes of this node (if it is an Element) or `NULL` otherwise.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETATTRIBUTES (  
    n          IN          DOMNode)  
RETURN DOMNAMEDNODEMAP;
```

Parameters

Table 233-45 GETATTRIBUTES Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNode |

GETCHARSET Function

This function retrieves the charset of the DOM document.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETCHARSET (  
    doc IN      DOMDocument)  
RETURN VARCHAR2;
```

Parameters

Table 233-46 GETCHARSET Function Parameters

| Parameter | Description |
|-----------|--------------|
| doc | DOM document |

Usage Notes

For a newly parsed document, we return the database charset. Once the `SETCHARSET` Procedure is called with a non-NULL value for `charset`, that `charset` is returned.

GETCHILDNODES Function

This function retrieves a `DOMNODELIST` that contains all children of this node. If there are no children, this is a `DOMNODELIST` containing no nodes.

 **See Also:**

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETCHILDNODES (  
    n            IN      DOMNode)  
RETURN DOMNodeList;
```


Parameters

Table 233-47 GETCHILDNODES Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETCHILDRENBYTAGNAME Functions

This function returns the children of the `DOMELEMENT`.

 **See Also:**

[DOMElement Subprograms](#)

Syntax

Returns children of the `DOMELEMENT` given the tag name:

```
DBMS_XMLDOM.GETCHILDRENBYTAGNAME (  
    elem        IN      DOMElement,  
    name        IN      VARCHAR2)  
RETURN DOMNODELIST;
```

Returns children of the `DOMELEMENT` given the tag name and namespace:

```
DBMS_XMLDOM.GETCHILDRENBYTAGNAME (  
    elem        IN      DOMElement,  
    name        IN      VARCHAR2,  
    ns          IN      VARCHAR2)  
RETURN DOMNODELIST;
```

Parameters

Table 233-48 GETCHILDRENBYTAGNAME Function Parameters

| Parameter | Description |
|-----------|-------------|
| elem | DOMELEMENT |
| name | Tag name |
| ns | Namespace |

GETDATA Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Gets the character data of the node that implements this interface (See Also: [DOMCharacterData Subprograms](#)):

```
DBMS_XMLDOM.GETDATA(  
    cd      IN      DOMCHARACTERDATA)  
RETURN VARCHAR2;
```

Returns the content data of the DOMProcessingInstruction (See Also: [DOMProcessingInstruction Subprograms](#)):

```
DBMS_XMLDOM.GETDATA(  
    pi      IN      DOMPROCESSINGINSTRUCTION)  
RETURN VARCHAR2;
```

Parameters

Table 233-49 GETDATA Function Parameters

| Parameter | Description |
|-----------|------------------------------|
| cd | DOMCHARACTERDATA |
| pi | The DOMPROCESSINGINSTRUCTION |

GETDOCTYPE Function

This function returns the DTD associated to the DOMDOCUMENT.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETDOCTYPE (  
    doc          IN      DOMDOCUMENT)  
RETURN DOMDOCUMENTTYPE;
```

Parameters

Table 233-50 GETDOCTYPE Function Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDOCUMENT |

GETDOCUMENTELEMENT Function

This function returns the root element of the `DOMDOCUMENT`.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETDOCUMENTELEMENT (  
    doc          IN      DOMDOCUMENT)  
RETURN DOMELEMENT;
```

Parameters

Table 233-51 GETDOCUMENTELEMENT Function Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDOCUMENT |

GETELEMENTSBYTAGNAME Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Returns a `DOMNODELIST` of all the elements with a specified tagname (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.GETELEMENTSBYTAGNAME (  
    doc          IN      DOMDOCUMENT,  
    tagname      IN      VARCHAR2)  
RETURN DOMNODELIST;
```

Returns the element children of the `DOMELEMENT` given the tag name (See Also: [DOMELEMENT Subprograms](#)):

```
DBMS_XMLDOM.GETELEMENTSBYTAGNAME (  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2)  
RETURN DOMNODELIST;
```

Returns the element children of the `DOMELEMENT` given the tag name and namespace (See Also: [DOMELEMENT Subprograms](#)):

```
DBMS_XMLDOM.GETELEMENTSBYTAGNAME (  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2,  
    ns        IN      VARCHAR2)  
RETURN DOMNODELIST;
```


Parameters

Table 233-52 GETELEMENTSBYTAGNAME Function Parameters

| Parameter | Description |
|-----------|---|
| doc | DOMDOCUMENT |
| tagname | Name of the tag to match on |
| elem | The DOMELEMENT |
| name | Tag name; using a wildcard(*) would match any tag |
| ns | Namespace |

GETENTITIES Function

This function retrieves a `DOMNAMEDNODEMAP` containing the general entities, both external and internal, declared in the DTD.

 **See Also:**
[DOMDocumentType Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETENTITIES (  
    dt      IN      DOMDocumentType)  
RETURN DOMNAMEDNODEMAP;
```

Parameters

Table 233-53 GETENTITIES Function Parameters

| Parameter | Description |
|-----------|-----------------|
| dt | DOMDOCUMENTTYPE |

GETEXPANDEDNAME Procedure and Functions

This subprogram is overloaded as a procedure and two functions. The specific forms of functionality are described along with the syntax declarations.

Syntax

Retrieves the expanded name of the `Node` if it is in an `Element` or `Attribute` type; otherwise, returns `NULL` (See Also: [DOMNode Subprograms](#))

```
DBMS_XMLDOM.GETEXPANDEDNAME (
    n          IN      DOMNODE
    data       OUT     VARCHAR);
```

Returns the expanded name of the `DOMAttr` (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.GETEXPANDEDNAME (
    a          IN      DOMAttr)
RETURN VARCHAR2;
```

Returns the expanded name of the `DOMElement` (See Also: [DOMElement Subprograms](#)):

```
DBMS_XMLDOM.GETEXPANDEDNAME (
    elem       IN      DOMELEMENT)
RETURN VARCHAR2;
```

Parameters

Table 233-54 GETEXPANDEDNAME Procedure and Function Parameters

| Parameter | Description |
|-----------|------------------------------------|
| n | DOMNODE |
| data | Returned expanded name of the Node |
| a | DOMATTR |
| elem | DOMELEMENT |

GETFIRSTCHILD Function

This function retrieves the first child of this node. If there is no such node, this returns `NULL`.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETFIRSTCHILD (
    n          IN      DOMNODE)
RETURN DOMNODE;
```

Parameters

Table 233-55 GETFIRSTCHILD Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETIMPLEMENTATION Function

This function returns the `DOMIMPLEMENTATION` object that handles this `DOMDOCUMENT`.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETIMPLEMENTATION(  
    doc      IN      DOMDOCUMENT)  
RETURN DOMIMPLEMENTATION;
```

Parameters

Table 233-56 GETIMPLEMENTATION Function Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDOCUMENT |

GETLASTCHILD Function

This function retrieves the last child of this node. If there is no such node, this returns `NULL`.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETLASTCHILD(  
    n      IN      DOMNODE)  
RETURN DOMNODE;
```

Parameters

Table 233-57 GETLASTCHILD Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETLENGTH Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Gets the number of characters in the data. This may have the value zero, because CharacterData nodes may be empty (See Also: [DOMCharacterData Subprograms](#)):

```
DBMS_XMLDOM.GETLENGTH (
    cd      IN      DOMCHARACTERDATA)
RETURN NUMBER;
```

Gets the number of nodes in this map. The range of valid child node indexes is 0 to length-1, inclusive (See Also: [DOMNamedNodeMap Subprograms](#)):

```
DBMS_XMLDOM.GETLENGTH (
    nnm      IN      DOMNAMEDNODEMAP)
RETURN NUMBER;
```

Gets the number of nodes in the list. The range of valid child node indexes is 0 to length-1, inclusive (See Also: [DOMNodeList Subprograms](#)):

```
DBMS_XMLDOM.GETLENGTH (
    nl      IN      DOMNODELIST)
RETURN NUMBER;
```

Parameters

Table 233-58 GETLENGTH Function Parameters

| Parameter | Description |
|-----------|------------------|
| cd | DOMCHARACTERDATA |
| nnm | DOMNAMEDNODEMAP |
| nl | DOMNODELIST |

GETLOCALNAME Procedure and Functions

This function is overloaded as a procedure and two functions. The specific forms of functionality are described alongside the syntax declarations.

Syntax

Retrieves the local part of the node's qualified name (See Also: [DOMNode Subprograms](#)):


```
DBMS_XMLDOM.GETLOCALNAME(  
    n          IN      DOMNODE,  
    data       OUT     VARCHAR2);
```

Returns the local name of the DOMAttr (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.GETLOCALNAME(  
    a          IN      DOMATTR)  
RETURN VARCHAR2;
```

Returns the local name of the DOMELEMENT (See Also: [DOMELEMENT Subprograms](#))

```
DBMS_XMLDOM.GETLOCALNAME(  
    elem       IN      DOMELEMENT)  
RETURN VARCHAR2;
```

Parameters

Table 233-59 GETLOCALNAME Procedure and Function Parameters

| Parameter | Description |
|-----------|----------------------|
| n | DOMNode |
| data | Returned local name. |
| a | DOMAttr. |
| elem | DOMELEMENT. |

GETNAME Functions

This function is overloaded. The specific forms of functionality are described with the syntax declarations.

Syntax

Returns the name of this attribute (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.GETNAME(  
    a          IN      DOMATTR)  
RETURN VARCHAR2;
```

Retrieves the name of DTD, or the name immediately following the DOCTYPE keyword (See Also: [DOMDocumentType Subprograms](#)):

```
DBMS_XMLDOM.GETNAME(  
    dt         IN      DOMDOCUMENTTYPE)  
RETURN VARCHAR2;
```

Parameters

Table 233-60 GETNAME Function Parameters

| Parameter | Description |
|-----------|-----------------|
| a | DOMATTR |
| dt | DOMDOCUMENTTYPE |

GETNAMEDITEM Function

GETNAMEDITEM retrieves a node specified by name.



See Also:

[DOMNamedNodeMap Subprograms](#)

Syntax

Retrieves a node specified by name:

```
DBMS_XMLDOM.GETNAMEDITEM(  
    nnm      IN  DOMNAMEDNODEMAP,  
    name     IN  VARCHAR2)  
RETURN DOMNODE;
```

Retrieves a node specified by name and namespace URI:

```
DBMS_XMLDOM.GETNAMEDITEM(  
    nnm      IN  DOMNAMEDNODEMAP,  
    name     IN  VARCHAR2,  
    ns       IN  VARCHAR2)  
RETURN DOMNODE;
```

Parameters

Table 233-61 GETNAMEDITEM Function Parameters

| Parameter | Description |
|-----------|----------------------------------|
| nnm | DOMNAMEDNODEMAP |
| name | Name of the item to be retrieved |
| ns | Namespace |

GETNAMESPACE Procedure and Functions

This subprogram is overloaded as a procedure and two functions. The specific forms of functionality are described alongside the syntax declarations.

Syntax

Retrieves the namespace URI associated with the node (See Also: [DOMNode Subprograms](#)):

```
DBMS_XMLDOM.GETNAMESPACE(  
    n      IN  DOMNODE,  
    data   OUT VARCHAR2);
```

Retrieves the namespace of the DOMATTR (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.GETNAMESPACE(  
    a      IN  DOMATTR)  
RETURN VARCHAR2;
```

Retrieves the namespace of the `DOMELEMENT` (See Also: [DOMElement Subprograms](#)):

```
DBMS_XMLDOM.GETNAMESPACE(  
    elem      IN      DOMELEMENT)  
RETURN VARCHAR2;
```


Parameters

Table 233-62 GETNAMESPACE Procedure and Function Parameters

| Parameter | Description |
|-----------|------------------------|
| n | DOMNODE |
| data | Returned namespace URI |
| a | DOMATTR |
| elem | DOMELEMENT |

GETNEXTSIBLING Function

This function retrieves the node immediately following this node. If there is no such node, this returns `NULL`.

 See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETNEXTSIBLING(  
    n      IN      DOMNODE)  
RETURN DOMNode;
```

Parameters

Table 233-63 GETNEXTSIBLING Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETNODETYPE Function

This function retrieves a code representing the type of the underlying object.

 See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETNODETYPE (  
    n          IN      DOMNODE)  
RETURN NUMBER;
```

Parameters

Table 233-64 GETNODETYPE Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETNODENAME Function

This function gets the name of the node depending on its type.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETNODENAME (  
    n          IN      DOMNODE)  
RETURN VARCHAR2;
```

Parameters

Table 233-65 GETNODENAME Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETNODEVALUE Function

This function gets the value of this node, depending on its type.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETNODEVALUE (  
    n          IN      DOMNODE)  
RETURN VARCHAR2;*
```

Parameters

Table 233-66 GETNODEVALUE Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETNODEVALUEASBINARYSTREAM Function & Procedure

The operation of these subprograms is described with each syntax implementation.



See Also:

[DOMNode Subprograms](#)

Syntax

This function returns an instance of the PL/SQL `XMLBinaryInputStream`. The node datatype must be `RAW` or `BLOB` – if not an exception is raised.

```
DBMS_XMLDOM.GETNODEVALUEASBINARYSTREAM (  
    n          IN      DOMNODE)  
RETURN SYS.UTL_BINARYINPUTSTREAM;
```

Using this procedure, the application passes an implementation of `SYS.UTL_BINARYOUTPUTSTREAM` into which XDB writes the contents of the node. The datatype of the node must be `RAW` or `CLOB` – if not an exception is raised.

```
DBMS_XMLDOM.GETNODEVALUEASBINARYSTREAM (  
    n          in      DOMNODE,  
    value      in      SYS.UTL_BINARYOUTPUTSTREAM);
```

Parameters

Table 233-67 GETNODEVALUEASBINARYSTREAM Function & Procedure Parameters

| Parameter | Description |
|-----------|--------------------|
| n | DOMNODE |
| value | BINARYOUTPUTSTREAM |

GETNODEVALUEASCHARACTERSTREAM Function & Procedure

The operation of these subprograms is described with each syntax implementation.



See Also:

[DOMNode Subprograms](#)

Syntax

This function returns an instance of the PL/SQL `XMLCharacterInputStream`. If the node data is character it is converted to the current session character set. If the node data is not character data, it is first converted to character data.

```
DBMS_XMLDOM.GETNODEVALUEASCHARACTERSTREAM (
    n          IN      DOMNODE)
RETURN SYS.UTL_CHARACTERINPUTSTREAM;
```

Using this procedure, the node data is converted, as necessary, to the session character set and then "pushed" into the `SYS.UTL_CHARACTEROUTPUTSTREAM`.

```
DBMS_XMLDOM.GETNODEVALUEASCHARACTERSTREAM (
    n          IN      DOMNODE,
    value      IN      SYS.UTL_CHARACTEROUTPUTSTREAM);
```

Parameters

Table 233-68 GETNODEVALUEASCHARACTERSTREAM Function & Procedure Parameters

| Parameter | Description |
|-----------|-----------------------|
| n | DOMNODE |
| value | CHARACTEROUTPUTSTREAM |

GETNOTATIONNAME Function

This function returns the notation name of the `DOMENTITY`.



See Also:

[DOMEntity Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETNOTATIONNAME (
    ent      IN      DOMENTITY)
RETURN VARCHAR2;
```


Parameters

Table 233-69 GETNOTATIONNAME Function Parameters

| Parameter | Description |
|-----------|-------------|
| ent | DOMENTITY |

GETNOTATIONS Function

This function retrieves a `DOMNAMEDNODEMAP` containing the notations declared in the DTD.

 **See Also:**

[DOMDocumentType Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETNOTATIONS (  
    dt          IN      DOMDOCUMENTTYPE)  
RETURN DOMNAMEDNODEMAP;
```


Parameters

Table 233-70 GETNOTATIONS Function Parameters

| Parameter | Description |
|-----------|-----------------|
| dt | DOMDOCUMENTTYPE |

GETTARGET Function

This function returns the target of the `DOMPROCESSINGINSTRUCTION`.

 **See Also:**

[DOMProcessingInstruction Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETTARGET (  
    pi          IN      DOMPROCESSINGINSTRUCTION)  
RETURN VARCHAR2;
```


Parameters

Table 233-71 GETTARGET Function Parameters

| Parameter | Description |
|-----------|--------------------------|
| pi | DOMPROCESSINGINSTRUCTION |

GETOWNERDOCUMENT Function

This function retrieves the Document object associated with this node. This is also the Document object used to create new nodes. When this node is a Document or a Document Type that is not used with any Document yet, this is `NULL`.

 **See Also:**

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETOWNERDOCUMENT (
    n          IN      DOMNODE)
RETURN DOMDOCUMENT;
```


Parameters

Table 233-72 GETOWNERDOCUMENT Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETOWNERELEMENT Function

This function retrieves the Element node to which the specified Attribute is attached.

 **See Also:**

[DOMAttr Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETOWNERELEMENT (
    a          IN      DOMATTR)
RETURN DOMELEMENT;
```


Parameters

Table 233-73 GETOWNERELEMENT Function Parameters

| Parameter | Description |
|-----------|-------------|
| a | Attribute |

GETPARENTNODE Function

This function retrieves the parent of this node. All nodes, except `Attr`, `Document`, `DocumentFragment`, `Entity`, and `Notation` may have a parent. However, if a node has just been created and not yet added to the tree, or if it has been removed from the tree, this is `NULL`.

 **See Also:**

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETPARENTNODE (  
    n            IN      DOMNODE)  
RETURN DOMNODE;
```


Parameters

Table 233-74 GETPARENTNODE Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETPREFIX Function

This function retrieves the namespace prefix of the node.

 **See Also:**

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETPREFIX (  
    n            IN      DOMNODE)  
RETURN VARCHAR2;
```


Parameters

Table 233-75 GETPREFIX Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETPREVIOUSIBLING Function

This function retrieves the node immediately preceding this node. If there is no such node, this returns NULL.

 **See Also:**
[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETPREVIOUSIBLING (
    n          IN      DOMNODE)
RETURN DOMNODE;
```

Parameters

Table 233-76 GETPREVIOUSIBLING Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETPUBLICID Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Returns the public identifier of the specified DTD (See Also: [DOMDocumentType Subprograms](#)):

```
DBMS_XMLDOM.GETPUBLICID (
    dt          IN      DOMDOCUMENTTYPE)
RETURN VARCHAR2;
```

Returns the public identifier of the DOMENTITY (See Also: [DOMEntity Subprograms](#)):

```
DBMS_XMLDOM.GETPUBLICID (
    ent         IN      DOMENTITY)
RETURN VARCHAR2;
```

Returns the public identifier of the DOMNOTATION (See Also: [DOMNotation Subprograms](#)):

```
DBMS_XMLDOM.GETPUBLICID (
    n           IN      DOMNOTATION)
RETURN VARCHAR2;
```

Parameters

Table 233-77 GETPUBLICID Function Parameters

| Parameter | Description |
|-----------|-------------|
| dt | The DTD |
| ent | DOMENTITY |
| n | DOMNOTATION |

GETQUALIFIEDNAME Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Returns the qualified name of the `DOMATTR` (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.GETQUALIFIEDNAME (  
    a          IN      DOMATTR)  
RETURN VARCHAR2;
```

Returns the qualified name of the `DOMElement` (See Also: [DOMElement Subprograms](#)):

```
DBMS_XMLDOM.GETQUALIFIEDNAME (  
    elem      IN      DOMELEMENT)  
RETURN VARCHAR2;
```

Parameters

Table 233-78 GETQUALIFIEDNAME Functions Parameters

| Parameter | Description |
|-----------|-------------|
| a | DOMATTR |
| elem | DOMELEMENT |

GETSCHEMANODE Function

This function retrieves the schema URI associated with the node.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETSCHEMANODE (  
    n          IN      DOMNODE)  
RETURN DOMNODE;
```

Parameters

Table 233-79 GETSCHEMANODE Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

GETSPECIFIED Function

If this attribute was explicitly specified, a value in the original document, this is true; otherwise, it is false.



See Also:

[DOMAttr Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETSPECIFIED (
    a          IN      DOMATTR)
RETURN BOOLEAN;
```

Parameters

Table 233-80 GETSPECIFIED Function Parameters

| Parameter | Description |
|-----------|-------------|
| a | DOMATTR |

GETSTANDALONE Function

This function returns the standalone property associated with the DOMDOCUMENT.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETSTANDALONE (
    doc        IN      DOMDOCUMENT)
RETURN VARCHAR2;
```

Parameters

Table 233-81 GETSTANDALONE Function Parameters

| Parameter | Description |
|-----------|--------------|
| doc | DOMDOCUMENT. |

GETSYSTEMID Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Returns the system id of the specified DTD (See Also: [DOMDocumentType Subprograms](#)):

```
DBMS_XMLDOM.GETSYSTEMID(  
    dt          IN      DOMDOCUMENTTYPE)  
RETURN VARCHAR2;
```

Returns the system identifier of the DOMENTITY (See Also: [DOMEntity Subprograms](#)):

```
DBMS_XMLDOM.GETSYSTEMID(  
    ent        IN      DOMENTITY)  
RETURN VARCHAR2;
```

Returns the system identifier of the DOMNOTATION (See Also: [DOMNotation Subprograms](#)):

```
DBMS_XMLDOM.GETSYSTEMID(  
    n          IN      DOMNOTATION)  
RETURN VARCHAR2;
```

Parameters

Table 233-82 GETSYSTEMID Function Parameters

| Parameter | Description |
|-----------|--------------|
| dt | The DTD. |
| ent | DOMEntity. |
| n | DOMNotation. |

GETTAGNAME Function

This function returns the name of the DOMELEMENT.



See Also:

[DOMELEMENT Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETTAGNAME (  
    elem      IN      DOMELEMENT)  
RETURN VARCHAR2;
```

Parameters

Table 233-83 GETTAGNAME Function Parameters

| Parameter | Description |
|-----------|----------------|
| elem | The DOMELEMENT |

GETVALUE Function

This function retrieves the value of the attribute.



See Also:

[DOMAttr Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETVALUE (  
    a      IN      DOMATTR)  
RETURN VARCHAR2;
```

Parameters

Table 233-84 GETVALUE Function Parameters

| Parameter | Description |
|-----------|-------------|
| a | DOMATTR |

GETVERSION Function

This function returns the version of the DOMDOCUMENT.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETVERSION (  
    doc      IN      DOMDOCUMENT)  
RETURN VARCHAR2;
```

Parameters

Table 233-85 GETVERSION Function Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDOCUMENT |

GETXMLTYPE Function

This function returns the `XMLType` associated with the `DOMDOCUMENT`.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.GETXMLTYPE (  
    doc      IN      DOMDOCUMENT)  
RETURN SYS.XMLTYPE;
```

Parameters

Table 233-86 GETXMLTYPE Function Parameters

| Parameter | Description |
|-----------|-------------|
| doc | DOMDOCUMENT |

HASATTRIBUTE Functions

Verifies whether an attribute has been defined for `DOMELEMENT`, or has a default value.



See Also:

[DOMElement Subprograms](#)

Syntax

Verifies whether an attribute with the specified name has been defined for `DOMElement`:

```
DBMS_XMLDOM.HASATTRIBUTE (  
    elem      IN  DOMELEMENT,  
    name      IN  VARCHAR2)  
RETURN VARCHAR2;
```

Verifies whether an attribute with specified name and namespace URI has been defined for `DOMELEMENT`; namespace enabled:

```
DBMS_XMLDOM.HASATTRIBUTE (
    elem      IN  DOMELEMENT,
    name      IN  VARCHAR2,
    ns        IN  VARCHAR2)
RETURN VARCHAR2;
```

Parameters

Table 233-87 HASATTRIBUTE Function Parameters

| Parameter | Description |
|-----------|---|
| elem | The DOMELEMENT |
| name | Attribute name; * matches any attribute |
| ns | Namespace |

HASATTRIBUTES Function

This function returns whether this node has any attributes.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.HASATTRIBUTES (
    n          IN      DOMNODE)
RETURN BOOLEAN;
```

Parameters

Table 233-88 HASATTRIBUTES Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

HASCHILDNODES Function

This function determines whether this node has any children.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.HASCHILDNODES (  
    n          IN      DOMNODE)  
RETURN BOOLEAN;
```

Parameters

Table 233-89 HASCHILDNODES Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

HASFEATURE Function

This function tests if the `DOMIMPLEMENTATION` implements a specific feature.



See Also:

[DOMImplementation Subprograms](#)

Syntax

```
DBMS_XMLDOM.HASFEATURE (  
    di          IN      DOMIMPLEMENTATION,  
    feature     IN      VARCHAR2,  
    version     IN      VARCHAR2)  
RETURN BOOLEAN;
```

Parameters

Table 233-90 HASFEATURE Function Parameters

| Parameter | Description |
|-----------|------------------------------------|
| di | DOMIMPLEMENTATION |
| feature | The feature to check for |
| version | The version of the DOM to check in |

IMPORTNODE Function

This function imports a node from an external document and returns this new node.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.IMPORTNODE (  
    doc           IN  DOMDOCUMENT,  
    importedNode  IN  DOMNODE,  
    deep          IN  BOOLEAN)  
RETURN DOMNODE;
```

Parameters

Table 233-91 IMPORTNODE Function Parameters

| Parameter | Description |
|--------------|--|
| doc | Document from which the node is imported |
| importedNode | Node to import |
| deep | Setting for recursive import. <ul style="list-style-type: none">If this value is TRUE, the entire subtree of the node will be imported with the node.If this value is FALSE, only the node itself will be imported. |

Usage Notes

Note that the [ADOPTNODE Function](#) removes the node from the source document while the [IMPORTNODE Function](#) clones the node in the source document.

INSERTBEFORE Function

This function inserts the node `newchild` before the existing child node `refchild`. If `refchild` is **NULL**, insert `newchild` at the end of the list of children.

If `newchild` is a **DOCUMENTFRAGMENT** object, all of its children are inserted, in the same order, before `refchild`. If the `newchild` is already in the tree, it is first removed.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.INSERTBEFORE (  
    n           IN  DOMNODE,  
    newchild    IN  DOMNODE,  
    refchild    IN  DOMNODE)  
RETURN DOMNode;
```

Parameters

Table 233-92 INSERTBEFORE Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNode |

Table 233-92 (Cont.) INSERTBEFORE Function Parameters

| Parameter | Description |
|-----------|--|
| newChild | The child to be inserted in the DOMNODE |
| refChild | The reference node before which the newchild is to be inserted |

INSERTDATA Procedure

This procedure inserts a string at the specified character offset.



See Also:

[DOMCharacterData Subprograms](#)

Syntax

```
DBMS_XMLDOM.INSERTDATA (  
    cd          IN      DOMCHARACTERDATA,  
    offset      IN      NUMBER,  
    arg         IN      VARCHAR2);
```

Parameters

Table 233-93 INSERTDATA Procedure Parameters

| Parameter | Description |
|-----------|--|
| cd | DOMCHARACTERDATA |
| offset | The offset at which to insert the data |
| arg | The value to be inserted |

ISNULL Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Checks if the specified DOMNODE is NULL. Returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMNode Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (  
    n          IN      DOMNODE)  
RETURN BOOLEAN;
```

Checks that the specified DOMATTR is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (  
    a          IN      DOMATTR)  
RETURN BOOLEAN;
```

Checks that the specified DOMCDATASECTION is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMCDATASection Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    cds      IN      DOMCDATASECTION)
RETURN BOOLEAN;
```

Checks that the specified DOMCHARACTERDATA is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMCharacterData Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    cd       IN      DOMCHARACTERDATA)
RETURN BOOLEAN;
```

Checks that the specified DOMCOMMENT is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMComment Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    com      IN      DOMCOMMENT)
RETURN BOOLEAN;
```

Checks that the specified DOMDOCUMENT is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    doc      IN      DOMDOCUMENT)
RETURN BOOLEAN;
```

Checks that the specified DOMDOCUMENTFRAGMENT is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMDocumentFragment Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    df       IN      DOMDOCUMENTFRAGMENT)
RETURN BOOLEAN;
```

Checks that the specified DOMDOCUMENTTYPE is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMDocumentType Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    dt       IN      DOMDOCUMENTTYPE)
RETURN BOOLEAN;
```

Checks that the specified DOMELEMENT is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMELEMENT Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    elem     IN      DOMELEMENT)
RETURN BOOLEAN;
```

Checks that the specified DOMENTITY is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMEntity Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    ent      IN      DOMENTITY)
RETURN BOOLEAN;
```

Checks that the specified DOMENTITYREFERENCE is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMEntityReference Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    eref     IN      DOMENTITYREFERENCE)
RETURN BOOLEAN;
```

Checks that the specified DOMIMPLEMENTATION is NULL; returns TRUE if it is NULL (See Also: [DOMImplementation Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    di          IN      DOMIMPLEMENTATION)
RETURN BOOLEAN;
```

Checks that the specified DOMNAMEDNODEMAP is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMNamedNodeMap Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    nnm         IN      DOMNAMEDNODEMAP)
RETURN BOOLEAN;
```

Checks that the specified DOMNODELIST is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMNodeList Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    nl          IN      DOMNODELIST)
RETURN BOOLEAN;
```

Checks that the specified DOMNOTATION is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMNotation Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    n           IN      DOMNOTATION)
RETURN BOOLEAN;
```

Checks that the specified DOMPROCESSINGINSTRUCTION is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMProcessingInstruction Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    pi          IN      DOMPROCESSINGINSTRUCTION)
RETURN BOOLEAN;
```

Checks that the specified DOMTEXT is NULL; returns TRUE if it is NULL, FALSE otherwise (See Also: [DOMText Subprograms](#)):

```
DBMS_XMLDOM.ISNULL (
    t           IN      DOMTEXT)
RETURN BOOLEAN;
```

Parameters

Table 233-94 ISNULL Function Parameters

| Parameter | Description |
|-----------|------------------------------|
| n | DOMNODE to check |
| a | DOMATTR to check |
| cds | DOMCDATASECTION to check |
| cd | DOMCHARACTERDATA to check |
| com | DOMCOMMENT to check |
| doc | DOMDOCUMENT to check |
| dF | DOMDOCUMENTFRAGMENT to check |
| dt | DOMDOCUMENTTYPE to check |
| elem | DOMELEMENT to check |

Table 233-94 (Cont.) ISNULL Function Parameters

| Parameter | Description |
|-----------|-----------------------------------|
| ent | DOMENTITY to check |
| eref | DOMENTITYREFERENCE to check |
| di | DOMIMPLEMENTATION to check |
| nnm | DOMNAMENODEMAP to check |
| nl | DOMNODELIST to check |
| n | DOMNOTATION to check |
| pi | DOMPROCESSINGINSTRUCTION to check |
| t | DOMTEXT to check |

ITEM Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Returns the item in the map which corresponds to the `INDEX` parameter. If `INDEX` is greater than or equal to the number of nodes in this map, this returns `NULL` (See Also: [DOMNamedNodeMap Subprograms](#)):

```
DBMS_XMLDOM.ITEM (
    nnm      IN      DOMNAMEDNODEMAP,
    index    IN      NUMBER)
RETURN DOMNODE;
```

Returns the item in the collection which corresponds to the `INDEX` parameter. If `index` is greater than or equal to the number of nodes in the list, this returns `NULL` (See Also: [DOMNodeList Subprograms](#)):

```
DBMS_XMLDOM.ITEM (
    nl      IN      DOMNODELIST,
    index   IN      NUMBER)
RETURN DOMNODE;
```


Parameters

Table 233-95 ITEM Function Parameters

| Parameter | Description |
|-----------|--|
| nnm | DOMNAMEDNODEMAP |
| index | The index in the node map at which the item is to be retrieved |
| nl | DOMNODELIST |
| index | The index in the <code>NodeList</code> used to retrieve the item |

MAKEATTR Function

This function casts a specified DOMNODE to a DOMATTR, and returns the DOMATTR.

 See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEATTR (  
    n          IN      DOMNODE)  
RETURN DOMATTR;
```


Parameters

Table 233-96 MAKEATTR Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKECDATASECTION Function

This function casts a specified DOMNODE to a DOMCDATASECTION.

 See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKECDATASECTION (  
    n          IN      DOMNODE)  
RETURN DOMCDATASECTION;
```

Parameters

Table 233-97 MAKECDATASECTION Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKECHARACTERDATA Function

This function casts a specified `DOMNODE` to a `DOMCHARACTERDATA`, and returns the `DOMCHARACTERDATA`.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKECHARACTERDATA (  
    n          IN      DOMNode)  
RETURN DOMCharacterData;
```

Parameters

Table 233-98 MAKECHARACTERDATA Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKECOMMENT Function

This function casts a specified `DOMNODE` to a `DOMCOMMENT`, and returns the `DOMCOMMENT`.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKECOMMENT (  
    n          IN      DOMNODE)  
RETURN DOMCOMMENT;
```

Parameters

Table 233-99 MAKECOMMENT Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKEDOCUMENT Function

This function casts a specified DOMNODE to a DOMDOCUMENT, and returns the DOMDOCUMENT.

 See Also:
[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEDOCUMENT (  
    n          IN      DOMNODE)  
RETURN DOMDocument;
```


Parameters

Table 233-100 MAKEDOCUMENT Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKEDOCUMENTFRAGMENT Function

This function casts a specified DOMNODE to a DOMDOCUMENTFRAGMENT, and returns the DOMDOCUMENTFRAGMENT.

 See Also:
[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEDOCUMENTFRAGMENT (  
    n          IN      DOMNODE)  
RETURN DOMDOCUMENTFRAGMENT;
```

Parameters

Table 233-101 MAKEDOCUMENTFRAGMENT Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKEDOCUMENTTYPE Function

This function casts a specified `DOMNODE` to a `DOMDOCUMENTTYPE` and returns the `DOMDOCUMENTTYPE`.

 **See Also:**

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEDOCUMENTTYPE (  
    n          IN      DOMNODE)  
RETURN DOMDOCUMENTTYPE;
```


Parameters

Table 233-102 MAKEDOCUMENTTYPE Function Parameters

| Parameter | Description |
|-----------|------------------|
| n | DOMNODE to cast. |

MAKEELEMENT Function

This function casts a specified `DOMNODE` to a `DOMELEMENT`, and returns the `DOMELEMENT`.

 **See Also:**

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEELEMENT (  
    n          IN      DOMNODE)  
RETURN DOMELEMENT;
```


Parameters

Table 233-103 MAKEELEMENT Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKEENTITY Function

This function casts a specified DOMNODE to a DOMENTITY, and returns the DOMENTITY.

 See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEENTITY (  
    n          IN      DOMNODE)  
RETURN DOMENTITY;
```


Parameters

Table 233-104 MAKEENTITY Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKEENTITYREFERENCE Function

This function casts a specified DOMNODE to a DOMENTITYREFERENCE, and returns the DOMENTITYREFERENCE.

 See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEENTITYREFERENCE (  
    n          IN      DOMNODE)  
RETURN DOME
```

Parameters

Table 233-105 MAKEENTITYREFERENCE Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKENODE Functions

This function is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Casts specified DOMATTR to a DOMNODE, and returns the DOMNODE (See Also: [DOMAttr Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    a          IN      DOMATTR)  
RETURN DOMNODE;
```

Casts the DOMCDATASECTION to a DOMNODE, and returns that DOMNODE (See Also: [DOMCDATASection Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    cds        IN      DOMCDATASECTION)  
RETURN DOMNODE;
```

Casts the specified DOMCHARACTERDATA as a DOMNODE, and returns that DOMNODE (See Also: [DOMCharacterData Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    cd         IN      DOMCHARACTERDATA)  
RETURN DOMNODE;
```

Casts the specified DOMCOMMENT to a DOMNODE, and returns that DOMNODE (See Also: [DOMComment Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    com        IN      DOMCOMMENT)  
RETURN DOMNODE;
```

Casts the DOMDOCUMENT to a DOMNODE, and returns that DOMNODE (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    doc        IN      DOMDOCUMENT)  
RETURN DOMNODE;
```

Casts the specified DOMDOCUMENTFRAGMENT to a DOMNODE, and returns that DOMNODE (See Also: [DOMDocumentFragment Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    df         IN      DOMDOCUMENTFRAGMENT)  
RETURN DOMNode;
```

Casts the specified DOMDOCUMENTTYPE to a DOMNODE, and returns that DOMNODE (See Also: [DOMDocumentType Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (  
    dt         IN      DOMDOCUMENTTYPE)  
RETURN DOMNODE;
```

Casts the specified DOMELEMENT to a DOMNODE, and returns that DOMNODE (See Also: [DOMELEMENT Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (
    elem      IN      DOMELEMENT)
RETURN DOMNODE;
```

Casts specified **DOMENTITY** to a **DOMNODE**, and returns that **DOMNODE** (See Also: [DOMEntity Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (
    ent      IN      DOMENTITY)
RETURN DOMNODE;
```

Casts the **DOMENTITYREFERENCE** to a **DOMNODE**, and returns that **DOMNODE** (See Also: [DOMEntityReference Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (
    eref     IN      DOMENTITYREFERENCE)
RETURN DOMNODE;
```

Casts the **DOMNOTATION** to a **DOMNODE**, and returns that **DOMNODE** (See Also: [DOMNotation Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (
    n        IN      DOMNOTATION)
RETURN DOMNODE;
```

Casts the **DOMPROCESSINGINSTRUCTION** to a **DOMNODE**, and returns the **DOMNODE** (See Also: [DOMProcessingInstruction Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (
    pi       IN      DOMPROCESSINGINSTRUCTION)
RETURN DOMNODE;
```

Casts the **DOMTEXT** to a **DOMNODE**, and returns that **DOMNODE** (See Also: [DOMText Subprograms](#)):

```
DBMS_XMLDOM.MAKENODE (
    t        IN      DOMTEXT)
RETURN DOMNODE;
```

Parameters

Table 233-106 MAKENODE Function Parameters

| Parameter | Description |
|-----------|-----------------------------|
| a | DOMATTR to cast |
| cds | DOMCDATASECTION to cast |
| cd | DOMCHARACTERDATA to cast |
| com | DOMCOMMENT to cast |
| doc | DOMDOCUMENT to cast |
| df | DOMDOCUMENTFRAGMENT to cast |
| dt | DOMDOCUMENTTYPE to cast |
| elem | DOMELEMENT to cast |
| ent | DOMENTITY to cast |
| eref | DOMENTITYREFERENCE to cast |
| n | DOMNOTATION to cast |

Table 233-106 (Cont.) MAKENODE Function Parameters

| Parameter | Description |
|-----------|----------------------------------|
| pi | DOMPROCESSINGINSTRUCTION to cast |
| t | DOMTEXT to cast |

MAKENOTATION Function

This function casts a specified DOMNODE to a DOMNOTATION, and returns the DOMNOTATION.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKENOTATION (  
    n          IN      DOMNODE)  
RETURN DOMNOTATION;
```

Parameters

Table 233-107 MAKENOTATION Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKEPROCESSINGINSTRUCTION Function

This function casts a specified DOMNODE to a DOMPROCESSINGINSTRUCTION, and returns the Domprocessinginstruction.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKEPROCESSINGINSTRUCTION (  
    n          IN      DOMNODE)  
RETURN DOMPROCESSINGINSTRUCTION;
```

Parameters

Table 233-108 MAKEPROCESSINGINSTRUCTION Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

MAKETEXT Function

This function casts a specified DOMNODE to a DOMTEXT, and returns the DOMTEXT.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.MAKETEXT (  
    n          IN      DOMNODE)  
RETURN DOMTEXT;
```

Parameters

Table 233-109 MAKETEXT Function Parameters

| Parameter | Description |
|-----------|-----------------|
| n | DOMNODE to cast |

NEWDOMDOCUMENT Functions

This function returns a new DOMDOCUMENT instance.



See Also:

[DOMDocument Subprograms](#)

Syntax

Returns a new DOMDOCUMENT instance:

```
DBMS_XMLDOM.NEWDOMDOCUMENT  
RETURN DOMDOCUMENT;
```

Returns a new DOMDOCUMENT instance created from the specified XMLType object:

```
DBMS_XMLDOM.NEWDOMDOCUMENT (  
    xmldoc      IN SYS.XMLTYPE)  
RETURN DOMDOCUMENT;
```

Returns a new `DOMDOCUMENT` instance created from the specified `CLOB`:

```
DBMS_XMLDOM.NEWDOMDOCUMENT (  
    cl          IN    CLOB)  
RETURN DOMDOCUMENT;
```

Parameters

Table 233-110 NEWDOMDOCUMENT Function Parameters

| Parameter | Description |
|-----------|---|
| xmlDoc | XMLType source for the <code>DOMDOCUMENT</code> |
| cl | CLOB source for the <code>DOMDOCUMENT</code> |

NORMALIZE Procedure

This procedure normalizes the text children of the `DOMELEMENT`.



See Also:

[DOMElement Subprograms](#)

Syntax

```
DBMS_XMLDOM.NORMALIZE (  
    elem          IN    DOMELEMENT);
```

Parameters

Table 233-111 NORMALIZE Procedure Parameters

| Parameter | Description |
|-----------|-----------------------------|
| elem | The <code>DOMELEMENT</code> |

REMOVEATTRIBUTE Procedures

This procedure removes an attribute from the `DOMELEMENT` by name.



See Also:

[DOMElement Subprograms](#)

Syntax

Removes the value of a `DOMELEMENT`'s attribute by name:

```
DBMS_XMLDOM.REMOVEATTRIBUTE (  
    elem          IN    DOMELEMENT,  
    name          IN    VARCHAR2);
```


Removes the value of a `DOMELEMENT`'s attribute by name and namespace URI.

```
DBMS_XMLDOM.REMOVEATTRIBUTE (  
    elem      IN      DOMELEMENT,  
    name      IN      VARCHAR2,  
    ns        IN      VARCHAR2) ;
```

Parameters

Table 233-112 REMOVEATTRIBUTE Procedure Parameters

| Parameter | Description |
|-----------|-----------------------------|
| elem | The <code>DOMELEMENT</code> |
| name | Attribute name |
| ns | Namespace |

REMOVEATTRIBUTENODE Function

This function removes the specified attribute node from the `DOMELEMENT`. The method returns the removed node.



See Also:

[DOMElement Subprograms](#)

Syntax

```
DBMS_XMLDOM.REMOVEATTRIBUTENODE (  
    elem      IN      DOMELEMENT,  
    oldAttr   IN      DOMATTR)  
RETURN DOMAttr;
```

Parameters

Table 233-113 REMOVEATTRIBUTENODE Function Parameters

| Parameter | Description |
|-----------|--------------------------------|
| elem | The <code>DOMELEMENT</code> . |
| oldAttr | The old <code>DOMATTR</code> . |

REMOVECHILD Function

This function removes the child node indicated by `oldchild` from the list of children, and returns it.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.REMOVECHILD(  
    n            IN      DOMNode,  
    oldchild    IN      DOMNode)  
RETURN DOMNODE;
```

Parameters

Table 233-114 REMOVECHILD Function Parameters

| Parameter | Description |
|-----------------------|--|
| <code>n</code> | DOMNode |
| <code>oldChild</code> | The child of the node <code>n</code> to be removed |

REMOVENAMEDITEM Function

This function removes a node, specified by name, from the map and returns this node.

When this map contains the attributes attached to an element, if the removed attribute is known to have a default value, an attribute immediately appears containing the default value as well as the corresponding namespace URI, local name, and prefix when applicable.



See Also:

[DOMNamedNodeMap Subprograms](#)

Syntax

Removes a node specified by name:

```
DBMS_XMLDOM.REMOVENAMEDITEM(  
    nnm        IN      DOMNamedNodeMap,  
    name       IN      VARCHAR2)  
RETURN DOMNode;
```

Removes a node specified by name and namespace URI:

```
DBMS_XMLDOM.REMOVENAMEDITEM(  
    nnm        IN      DOMNamedNodeMap,  
    name       IN      VARCHAR2,
```

```
    ns          IN      VARCHAR2)  
RETURN DOMNode;
```

Parameters

Table 233-115 REMOVENAMEDITEM Function Parameters

| Parameter | Description |
|-----------|---|
| nnm | DOMNamedNodeMap |
| name | The name of the item to be removed from the map |
| ns | Namespace |

REPLACECHILD Function

This function replaces the child node `oldchild` with `newchild` in the list of children, and returns the `oldchild` node.

If `newchild` is a `DocumentFragment` object, `oldchild` is replaced by all of the `DocumentFragment` children, which are inserted in the same order. If the `newchild` is already in the tree, it is first removed.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.REPLACECHILD(  
    n          IN      DOMNode,  
    newchild   IN      DOMNode,  
    oldchild   IN      DOMNode)  
RETURN DOMNode;
```

Parameters

Table 233-116 REPLACECHILD Function Parameters

| Parameter | Description |
|-----------|---|
| n | DOMNode |
| newchild | The new child which is to replace the old child |
| oldchild | The child of the node n which is to be replaced |

REPLACEDATA Procedure

This procedure changes a range of characters in the node. Upon success, data and length reflect the change.



See Also:

[DOMCharacterData Subprograms](#)

Syntax

```
DBMS_XMLDOM.REPLACEDATA (  
    cd          IN      DOMCHARACTERDATA,  
    offset      IN      NUMBER,  
    cnt         IN      NUMBER,  
    arg         IN      VARCHAR2);
```

Parameters

Table 233-117 REPLACEDATA Procedure Parameters

| Parameter | Description |
|-----------|-------------------------------------|
| cd | DOMCHARACTERDATA |
| offset | The offset at which to replace |
| cnt | The number of characters to replace |
| arg | The value to replace with |

RESOLVENAMESPACEPREFIX Function

This function resolves the specified namespace prefix, and returns the resolved namespace.



See Also:

[DOMElement Subprograms](#)

Syntax

```
DBMS_XMLDOM.RESOLVENAMESPACEPREFIX (  
    elem      IN      DOMELEMENT,  
    prefix    IN      VARCHAR2)  
RETURN VARCHAR2;
```

Parameters

Table 233-118 RESOLVENAMESPACEPREFIX Function Parameters

| Parameter | Description |
|-----------|------------------|
| elem | The DOMELEMENT |
| prefix | Namespace prefix |

SETATTRIBUTE Procedures

This procedure sets the value of a DOMELEMENT's attribute by name.



See Also:

[DOMELEMENT Subprograms](#)

Syntax

Sets the value of a DOMELEMENT's attribute by name:

```
DBMS_XMLDOM.SETATTRIBUTE(  
    elem      IN  DOMELEMENT,  
    name      IN  VARCHAR2,  
    newvalue  IN  VARCHAR2);
```

Sets the value of a DOMELEMENT's attribute by name and namespace URI:

```
DBMS_XMLDOM.SETATTRIBUTE(  
    elem      IN  DOMELEMENT,  
    name      IN  VARCHAR2,  
    newvalue  IN  VARCHAR2,  
    ns        IN  VARCHAR2);
```

Parameters

Table 233-119 SETATTRIBUTE Procedure Parameters

| Parameter | Description |
|-----------|-----------------|
| elem | The DOMELEMENT |
| name | Attribute name |
| newvalue | Attribute value |
| ns | Namespace |

SETATTRIBUTENODE Functions

This function adds a new attribute node to the `DOMELEMENT`.



See Also:

[DOMElement Subprograms](#)

Syntax

Adds a new attribute node to the `DOMELEMENT`:

```
DBMS_XMLDOM.SETATTRIBUTENODE (
    elem      IN  DOMELEMENT,
    newAttr   IN  DOMATTR)
RETURN DOMATTR;
```

Adds a new attribute node to the `DOMElement`; namespace enabled:

```
DBMS_XMLDOM.SETATTRIBUTENODE (
    elem      IN  DOMELEMENT,
    newAttr   IN  DOMATTR,
    ns        IN  VARCHAR2)
RETURN DOMATTR;
```

Parameters

Table 233-120 SETATTRIBUTENODE Function Parameters

| Parameter | Description |
|----------------------|------------------------------|
| <code>elem</code> | The <code>DOMELEMENT</code> |
| <code>newAttr</code> | The new <code>DOMATTR</code> |
| <code>ns</code> | The namespace |

SETCHARSET Procedure

This function sets the character set of the DOM document.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.SETCHARSET (
    doc      IN  DOMDocument,
    charset  IN  VARCHAR2);
```

Parameters

Table 233-121 SETCHARSET Procedure Parameters

| Parameter | Description |
|-----------|---------------|
| doc | DOM document |
| charset | Character set |

Usage Notes

This is used for [WRITETOFILE Procedures](#) if not explicitly specified at that time.

SETDATA Procedures

This overloaded procedure sets character data or `DOMPROCESSINGINSTRUCTION` content data. The specific functionality is described in the syntax declarations.

Syntax

Sets the character data of the node that implements this interface (See Also: [DOMCharacterData Subprograms](#)):

```
DBMS_XMLDOM.SETDATA (  
    cd          IN      DOMCHARACTERDATA,  
    data        IN      VARCHAR2);
```

Sets the content data of the `DOMPROCESSINGINSTRUCTION` (See Also: [DOMProcessingInstruction Subprograms](#)):

```
DBMS_XMLDOM.SETDATA (  
    pi          IN      DOMPROCESSINGINSTRUCTION,  
    data        IN      VARCHAR2);
```

Parameters

Table 233-122 SETDATA Procedure Parameters

| Parameter | Description |
|-----------|---|
| cd | DOMCHARACTERDATA |
| data | The data to which the node is set |
| pi | DOMPROCESSINGINSTRUCTION |
| data | New processing instruction content data |

SETDOCTYPE Procedure

Given a DOM document, this procedure creates a new DTD with the specified name, system id and public id and sets it in the document.

This DTD can later be retrieved using the [GETDOCTYPE Function](#).

Syntax

```
DBMS_XMLDOM.SETDOCTYPE (  
    doc      IN    DOMDocument,  
    name     IN    VARCHAR2,  
    sysid    IN    VARCHAR2,  
    pubid    IN    VARCHAR2);
```

Parameters

Table 233-123 SETDOCTYPE Procedure Parameters

| Parameter | Description |
|-----------|---|
| doc | The document whose DTD has to be set |
| name | The name that the doctype needs to be initialized with |
| sysid | The system ID that the doctype needs to be initialized with |
| pubid | The public ID that the doctype needs to be initialized with |

SETNAMEDITEM Function

This function adds a node using its `nodeName` attribute.

If a node with that name is already present in this map, it is replaced by the new one. The old node is returned on replacement; if no replacement is made, `NULL` is returned.

As the `nodeName` attribute is used to derive the name under which the node must be stored, multiple nodes of certain types, those that have a "special" string value, cannot be stored because the names would clash. This is seen as preferable to allowing nodes to be aliased.



See Also:

[DOMNamedNodeMap Subprograms](#)

Syntax

Adds a node using its `nodeName` attribute:

```
DBMS_XMLDOM.SETNAMEDITEM(  
    nnm      IN    DOMNAMEDNODEMAP,  
    arg      IN    DOMNODE)  
RETURN DOMNode;
```

Adds a node using its `nodeName` attribute and namespace URI:

```
DBMS_XMLDOM.SETNAMEDITEM(  
    nnm      IN    DOMNAMEDNODEMAP,  
    arg      IN    DOMNODE,  
    ns       IN    VARCHAR2)  
RETURN DOMNode;
```


Parameters

Table 233-124 SETNAMEDITEM Function Parameters

| Parameter | Description |
|-----------|--|
| nnm | DOMNAMEDNODEMAP |
| arg | The Node to be added using its <code>nodeName</code> attribute |
| ns | Namespace |

SETNODEVALUE Procedure

This procedure sets the value of this node, depending on its type. When it is defined to be `NULL`, setting it has no effect.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.SETNODEVALUE (  
    n          IN      DOMNODE,  
    nodeValue IN      VARCHAR2);
```

Parameters

Table 233-125 SETNODEVALUE Procedure Parameters

| Parameter | Description |
|-----------|--------------------------------|
| n | DOMNode |
| nodeValue | The value to which node is set |

SETNODEVALUEASBINARYSTREAM Function & Procedure

The operation of these subprograms is described in the syntax section.



See Also:

[DOMNode Subprograms](#)

Syntax

This function returns an instance of the PL/SQL `XMLBINARYOUTPUTSTREAM` into which the caller can write the node value. The datatype of the node must be `RAW` or `BLOB` – if not, an exception is raised.

```
DBMS_XMLDOM.SETNODEVALUEASBINARYSTREAM (
    n          IN      DOMNODE)
RETURN SYS.UTL_BINARYOUTPUTSTREAM;
```

Using this procedure, the application passes in an implementation of `sys.utl_BinaryInputStream` from which XDB reads data to populate the node. The datatype of the node must be RAW or BLOB – if not an exception is raised.

```
DBMS_XMLDOM.SETNODEVALUEASBINARYSTREAM (
    n          in      DOMNODE,
    value      in      SYS.UTL_BINARYINPUTSTREAM);
```

Parameters

Table 233-126 SETNODEVALUEASBINARYSTREAM Function & Procedure Parameters

| Parameter | Description |
|-----------|-------------------|
| n | DOMNODE |
| value | BINARYINPUTSTREAM |

SETNODEVALUEASCHARACTERSTREAM Function & Procedure

The operation of these subprograms is described in the syntax section.



See Also:

[DOMNode Subprograms](#)

Syntax

This function returns an instance of the PL/SQL `XMLCHARACTEROUTPUTSTREAM` type into which the caller can write the node value. The datatype of the node can be any valid XDB datatype. If the type is not character or CLOB, the character data written to the stream is converted to the node datatype. If the datatype of the node is character or CLOB, then the character data written to the stream is converted from PL/SQL session character set to the character set of the node.

```
DBMS_XMLDOM.SETNODEVALUEASCHARACTERSTREAM (
    n          IN      DOMNODE)
RETURN SYS.UTL_CHARACTEROUTPUTSTREAM;
```

Using this procedure, the application passes in an implementation of `SYS.UTL_CHARACTERINPUTSTREAM` from which XDB reads to populate the node. The datatype of the node may be any valid type supported by XDB. If a non-character datatype, the character data read from the stream is converted to the datatype of the node. If the datatype of the node is either character or CLOB, then no conversion occurs and the character set of the node becomes the character set of the PL/SQL session.

```
DBMS_XMLDOM.SETNODEVALUEASCHARACTERSTREAM (
    n          IN      DOMNODE,
    value      IN      SYS.UTL_CHARACTERINPUTSTREAM);
```

Parameters

Table 233-127 SETNODEVALUEASCHARACTERSTREAM Function & Procedure Parameters

| Parameter | Description |
|-----------|----------------------|
| n | DOMNODE |
| value | CHARACTERINPUTSTREAM |

SETPREFIX Procedure

This procedure sets the namespace prefix for this node to the specified value.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.SETPREFIX(  
    n          IN      DOMNODE,  
    prefix     IN      VARCHAR2);
```

Parameters

Table 233-128 SETPREFIX Procedure Parameters

| Parameter | Description |
|-----------|--|
| n | DOMNODE |
| prefix | The value for the namespace prefix of the node |

SETSTANDALONE Procedure

This procedure sets the standalone property of the DOMDOCUMENT.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.SETSTANDALONE(  
    doc          IN      DOMDOCUMENT,  
    newvalue     IN      VARCHAR2);
```

Parameters

Table 233-129 SETSTANDALONE Procedure Parameters

| Parameter | Description |
|-----------|--|
| doc | DOMDOCUMENT |
| newvalue | Value of the standalone property of the document |

SETVALUE Procedure

This procedure sets the value of the attribute.



See Also:

[DOMAttr Subprograms](#)

Syntax

```
DBMS_XMLDOM.SETVALUE (
    a          IN      DOMATTR,
    value      IN      VARCHAR2);
```

Parameters

Table 233-130 SETVALUE Procedure Parameters

| Parameter | Description |
|-----------|---|
| a | DOMATTR |
| value | The value to which to set the attribute |

SETVERSION Procedure

This procedure sets the version of the DOMDOCUMENT.



See Also:

[DOMDocument Subprograms](#)

Syntax

```
DBMS_XMLDOM.SETVERSION (
    doc        IN      DOMDOCUMENT,
    version    IN      VARCHAR2);
```

Parameters

Table 233-131 SETVERSION Procedure Parameters

| Parameter | Description |
|-----------|-----------------------------|
| doc | DOMDOCUMENT |
| version | The version of the document |

SPLITTEXT Function

This function breaks this DOMTEXT node into two DOMTEXT nodes at the specified offset.



See Also:

[DBMS_XMLDOM DOMText Subprograms](#)

Syntax

```
DBMS_XMLDOM.SPLITTEXT (  
    t          IN      DOMTEXT,  
    offset     IN      NUMBER)  
RETURN DOMText;
```

Parameters

Table 233-132 SPLITTEXT Function Parameters

| Parameter | Description |
|-----------|--------------------------|
| t | DOMTEXT |
| offset | Offset at which to split |

SUBSTRINGDATA Function

This function extracts a range of data from the node.



See Also:

[DOMCharacterData Subprograms](#)

Syntax

```
DBMS_XMLDOM.SUBSTRINGDATA (  
    cd          IN      DOMCHARACTERDATA,  
    offset      IN      NUMBER,
```

```
    cnt      IN      NUMBER)  
RETURN VARCHAR2;
```

Parameters

Table 233-133 SUBSTRINGDATA Function Parameters

| Parameter | Description |
|-----------|---|
| cd | DOMCHARACTERDATA |
| offset | The starting offset of the data from which to get the data |
| cnt | The number of characters (from the offset) of the data to get |

USEBINARYSTREAM Function

This function returns `TRUE` if the datatype of the node is `RAW` or `BLOB`, so that the node value may be read or written using an `UTL_BINARYINPUTSTREAM` or `UTL_BINARYOUTPUTSTREAM`.

If a value of `FALSE` is returned, the node value may only be accessed through an `UTL_CHARACTERINPUTSTREAM` or `UTL_CHARACTEROUTPUTSTREAM`.



See Also:

[DOMNode Subprograms](#)

Syntax

```
DBMS_XMLDOM.USEBINARYSTREAM  (  
    n      IN      DOMNODE)  
RETURN BOOLEAN;
```

Parameters

Table 233-134 USEBINARYSTREAM Function Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

WRITETOBUFFER Procedures

`WRITETOBUFFER` is an overloaded procedure that writes an XML node, XML document, or a document fragment to a specified buffer.

This procedure is overloaded. The specific forms of functionality are described along with the syntax declarations.

Syntax

Writes XML node to specified buffer using the database character set (See Also: [DOMNode Subprograms](#)):

```
DBMS_XMLDOM.WRITETOBUFFER(  
    n          IN          DOMNODE,  
    buffer     IN OUT     VARCHAR2);
```

Writes XML document to a specified buffer using database character set (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.WRITETOBUFFER(  
    doc        IN          DOMDOCUMENT,  
    buffer     IN OUT     VARCHAR2);
```

Writes the contents of the specified document fragment into a buffer using the database character set (See Also: [DOMDocumentFragment Subprograms](#)):

```
DBMS_XMLDOM.WRITETOBUFFER(  
    df         IN          DOMDOCUMENTFRAGMENT,  
    buffer     IN OUT     VARCHAR2);
```

Parameters

Table 233-135 WRITETOBUFFER Procedure Parameters

| Parameter | Description |
|-----------|--------------------------|
| n | DOMNODE |
| buffer | Buffer to which to write |
| doc | DOMDOCUMENT |
| df | DOM document fragment |

WRITETOCLOB Procedures

WRITETOCLOB is an overloaded procedure that writes an XML node or document to a specified CLOB.

The specific forms of functionality are described along with the syntax declarations.

Syntax

Writes XML node to specified CLOB using the database character set (See Also: [DOMNode Subprograms](#)):

```
DBMS_XMLDOM.WRITETOCLOB(  
    n          IN          DOMNODE,  
    cl         IN OUT     CLOB);
```

Writes XML document to a specified CLOB using database character set (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.WRITETOCLOB(  
    doc        IN          DOMDOCUMENT,  
    cl         IN OUT     CLOB);
```

Parameters

Table 233-136 WRITETOCLOB Procedure Parameters

| Parameter | Description |
|-----------|------------------------|
| n | DOMNODE |
| cl | CLOB to which to write |
| doc | DOMDOCUMENT |

WRITETOFILE Procedures

This overloaded procedure writes an XML node or XML document to a specified node.

The specific forms of functionality are described along with the syntax declarations.

Syntax

Writes XML node to specified file using the database character set (See Also: [DOMNode Subprograms](#)):

```
DBMS_XMLDOM.WRITETOFILE (  
    n            IN      DOMNODE,  
    fileName     IN      VARCHAR2);
```

Writes XML node to specified file using the specified character set, which is passed in as a separate parameter (See Also: [DOMNode Subprograms](#)):

```
DBMS_XMLDOM.WRITETOFILE (  
    n            IN      DOMNODE,  
    fileName     IN      VARCHAR2,  
    charset      IN      VARCHAR2);
```

Writes an XML document to a specified file using database character set (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.WRITETOFILE (  
    doc          IN      DOMDOCUMENT,  
    filename     IN      VARCHAR2);
```

Writes an XML document to a specified file using specified character set (See Also: [DOMDocument Subprograms](#)):

```
DBMS_XMLDOM.WRITETOFILE (  
    doc          IN      DOMDOCUMENT,  
    fileName     IN      VARCHAR2,  
    charset      IN      VARCHAR2);
```

Parameters

Table 233-137 WRITETOFILE Procedure Parameters

| Parameter | Description |
|-----------|-------------|
| n | DOMNODE |

Table 233-137 (Cont.) WRITETOFILE Procedure Parameters

| Parameter | Description |
|-----------|---|
| fileName | File to which to write. The filename should be in the format of database_directory_object_name/filename, for example mydir/filename (on windows, use \ instead of /). |
| charset | specified character set |
| doc | DOMDOCUMENT |
| charset | Character set |