| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DynAnyOperations.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/omg/DynamicAny/DynAnyHelper.html)   [**NEXT CLASS**](http://docs.google.com/org/omg/DynamicAny/DynAnySeqHelper.html) | [**FRAMES**](http://docs.google.com/index.html?org/omg/DynamicAny/DynAnyOperations.html)    [**NO FRAMES**](http://docs.google.com/DynAnyOperations.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **org.omg.DynamicAny**

Interface DynAnyOperations

**All Known Subinterfaces:** [DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html), [DynArray](http://docs.google.com/org/omg/DynamicAny/DynArray.html), [DynArrayOperations](http://docs.google.com/org/omg/DynamicAny/DynArrayOperations.html), [DynEnum](http://docs.google.com/org/omg/DynamicAny/DynEnum.html), [DynEnumOperations](http://docs.google.com/org/omg/DynamicAny/DynEnumOperations.html), [DynFixed](http://docs.google.com/org/omg/DynamicAny/DynFixed.html), [DynFixedOperations](http://docs.google.com/org/omg/DynamicAny/DynFixedOperations.html), [DynSequence](http://docs.google.com/org/omg/DynamicAny/DynSequence.html), [DynSequenceOperations](http://docs.google.com/org/omg/DynamicAny/DynSequenceOperations.html), [DynStruct](http://docs.google.com/org/omg/DynamicAny/DynStruct.html), [DynStructOperations](http://docs.google.com/org/omg/DynamicAny/DynStructOperations.html), [DynUnion](http://docs.google.com/org/omg/DynamicAny/DynUnion.html), [DynUnionOperations](http://docs.google.com/org/omg/DynamicAny/DynUnionOperations.html), [DynValue](http://docs.google.com/org/omg/DynamicAny/DynValue.html), [DynValueBox](http://docs.google.com/org/omg/DynamicAny/DynValueBox.html), [DynValueBoxOperations](http://docs.google.com/org/omg/DynamicAny/DynValueBoxOperations.html), [DynValueCommon](http://docs.google.com/org/omg/DynamicAny/DynValueCommon.html), [DynValueCommonOperations](http://docs.google.com/org/omg/DynamicAny/DynValueCommonOperations.html), [DynValueOperations](http://docs.google.com/org/omg/DynamicAny/DynValueOperations.html) **All Known Implementing Classes:** [\_DynAnyStub](http://docs.google.com/org/omg/DynamicAny/_DynAnyStub.html), [\_DynArrayStub](http://docs.google.com/org/omg/DynamicAny/_DynArrayStub.html), [\_DynEnumStub](http://docs.google.com/org/omg/DynamicAny/_DynEnumStub.html), [\_DynFixedStub](http://docs.google.com/org/omg/DynamicAny/_DynFixedStub.html), [\_DynSequenceStub](http://docs.google.com/org/omg/DynamicAny/_DynSequenceStub.html), [\_DynStructStub](http://docs.google.com/org/omg/DynamicAny/_DynStructStub.html), [\_DynUnionStub](http://docs.google.com/org/omg/DynamicAny/_DynUnionStub.html), [\_DynValueStub](http://docs.google.com/org/omg/DynamicAny/_DynValueStub.html)

public interface **DynAnyOperations**

Any values can be dynamically interpreted (traversed) and constructed through DynAny objects. A DynAny object is associated with a data value which corresponds to a copy of the value inserted into an any.

A DynAny object may be viewed as an ordered collection of component DynAnys. For DynAnys representing a basic type, such as long, or a type without components, such as an empty exception, the ordered collection of components is empty. Each DynAny object maintains the notion of a current position into its collection of component DynAnys. The current position is identified by an index value that runs from 0 to n-1, where n is the number of components. The special index value -1 indicates a current position that points nowhere. For values that cannot have a current position (such as an empty exception), the index value is fixed at -1. If a DynAny is initialized with a value that has components, the index is initialized to 0. After creation of an uninitialized DynAny (that is, a DynAny that has no value but a TypeCode that permits components), the current position depends on the type of value represented by the DynAny. (The current position is set to 0 or -1, depending on whether the new DynAny gets default values for its components.)

The iteration operations rewind, seek, and next can be used to change the current position and the current\_component operation returns the component at the current position. The component\_count operation returns the number of components of a DynAny. Collectively, these operations enable iteration over the components of a DynAny, for example, to (recursively) examine its contents.

A constructed DynAny object is a DynAny object associated with a constructed type. There is a different interface, inheriting from the DynAny interface, associated with each kind of constructed type in IDL (fixed, enum, struct, sequence, union, array, exception, and value type).

A constructed DynAny object exports operations that enable the creation of new DynAny objects, each of them associated with a component of the constructed data value. As an example, a DynStruct is associated with a struct value. This means that the DynStruct may be seen as owning an ordered collection of components, one for each structure member. The DynStruct object exports operations that enable the creation of new DynAny objects, each of them associated with a member of the struct.

If a DynAny object has been obtained from another (constructed) DynAny object, such as a DynAny representing a structure member that was created from a DynStruct, the member DynAny is logically contained in the DynStruct. Calling an insert or get operation leaves the current position unchanged. Destroying a top-level DynAny object (one that was not obtained as a component of another DynAny) also destroys any component DynAny objects obtained from it. Destroying a non-top level DynAny object does nothing. Invoking operations on a destroyed top-level DynAny or any of its descendants raises OBJECT\_NOT\_EXIST. If the programmer wants to destroy a DynAny object but still wants to manipulate some component of the data value associated with it, then he or she should first create a DynAny for the component and, after that, make a copy of the created DynAny object.

The behavior of DynAny objects has been defined in order to enable efficient implementations in terms of allocated memory space and speed of access. DynAny objects are intended to be used for traversing values extracted from anys or constructing values of anys at runtime. Their use for other purposes is not recommended.

Insert and get operations are necessary to handle basic DynAny objects but are also helpful to handle constructed DynAny objects. Inserting a basic data type value into a constructed DynAny object implies initializing the current component of the constructed data value associated with the DynAny object. For example, invoking insert\_boolean on a DynStruct implies inserting a boolean data value at the current position of the associated struct data value. A type is consistent for inserting or extracting a value if its TypeCode is equivalent to the TypeCode contained in the DynAny or, if the DynAny has components, is equivalent to the TypeCode of the DynAny at the current position.

DynAny and DynAnyFactory objects are intended to be local to the process in which they are created and used. This means that references to DynAny and DynAnyFactory objects cannot be exported to other processes, or externalized with ORB.object\_to\_string(). If any attempt is made to do so, the offending operation will raise a MARSHAL system exception. Since their interfaces are specified in IDL, DynAny objects export operations defined in the standard org.omg.CORBA.Object interface. However, any attempt to invoke operations exported through the Object interface may raise the standard NO\_IMPLEMENT exception. An attempt to use a DynAny object with the DII may raise the NO\_IMPLEMENT exception.

| **Method Summary** | |
| --- | --- |
| void | [**assign**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#assign(org.omg.DynamicAny.DynAny))([DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) dyn\_any)            Initializes the value associated with a DynAny object with the value associated with another DynAny object. |
| int | [**component\_count**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#component_count())()            Returns the number of components of a DynAny. |
| [DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) | [**copy**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#copy())()            Creates a new DynAny object whose value is a deep copy of the DynAny on which it is invoked. |
| [DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) | [**current\_component**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#current_component())()            Returns the DynAny for the component at the current position. |
| void | [**destroy**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#destroy())()            Destroys a DynAny object. |
| boolean | [**equal**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#equal(org.omg.DynamicAny.DynAny))([DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) dyn\_any)            Compares two DynAny values for equality. |
| void | [**from\_any**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#from_any(org.omg.CORBA.Any))([Any](http://docs.google.com/org/omg/CORBA/Any.html) value)            Initializes the value associated with a DynAny object with the value contained in an any. |
| [Any](http://docs.google.com/org/omg/CORBA/Any.html) | [**get\_any**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_any())()            Extracts an Any value contained in the Any represented by this DynAny. |
| boolean | [**get\_boolean**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_boolean())()            Extracts the boolean value from this DynAny. |
| char | [**get\_char**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_char())()            Extracts the char value from this DynAny. |
| double | [**get\_double**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_double())()            Extracts the double value from this DynAny. |
| [DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) | [**get\_dyn\_any**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_dyn_any())()            Extracts the Any value contained in the Any represented by this DynAny and returns it wrapped into a new DynAny. |
| float | [**get\_float**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_float())()            Extracts the float value from this DynAny. |
| int | [**get\_long**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_long())()            Extracts the integer value from this DynAny. |
| long | [**get\_longlong**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_longlong())()            Extracts the long value from this DynAny. |
| byte | [**get\_octet**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_octet())()            Extracts the byte value from this DynAny. |
| [Object](http://docs.google.com/org/omg/CORBA/Object.html) | [**get\_reference**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_reference())()            Extracts the reference to a CORBA Object from this DynAny. |
| short | [**get\_short**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_short())()            Extracts the short value from this DynAny. |
| [String](http://docs.google.com/java/lang/String.html) | [**get\_string**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_string())()            Extracts the string value from this DynAny. |
| [TypeCode](http://docs.google.com/org/omg/CORBA/TypeCode.html) | [**get\_typecode**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_typecode())()            Extracts the TypeCode object from this DynAny. |
| int | [**get\_ulong**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_ulong())()            Extracts the integer value from this DynAny. |
| long | [**get\_ulonglong**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_ulonglong())()            Extracts the long value from this DynAny. |
| short | [**get\_ushort**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_ushort())()            Extracts the short value from this DynAny. |
| [Serializable](http://docs.google.com/java/io/Serializable.html) | [**get\_val**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_val())()            Extracts a Serializable object from this DynAny. |
| char | [**get\_wchar**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_wchar())()            Extracts the long value from this DynAny. |
| [String](http://docs.google.com/java/lang/String.html) | [**get\_wstring**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#get_wstring())()            Extracts the string value from this DynAny. |
| void | [**insert\_any**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_any(org.omg.CORBA.Any))([Any](http://docs.google.com/org/omg/CORBA/Any.html) value)            Inserts an Any value into the Any represented by this DynAny. |
| void | [**insert\_boolean**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_boolean(boolean))(boolean value)            Inserts a boolean value into the DynAny. |
| void | [**insert\_char**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_char(char))(char value)            Inserts a char value into the DynAny. |
| void | [**insert\_double**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_double(double))(double value)            Inserts a double value into the DynAny. |
| void | [**insert\_dyn\_any**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_dyn_any(org.omg.DynamicAny.DynAny))([DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) value)            Inserts the Any value contained in the parameter DynAny into the Any represented by this DynAny. |
| void | [**insert\_float**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_float(float))(float value)            Inserts a float value into the DynAny. |
| void | [**insert\_long**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_long(int))(int value)            Inserts an integer value into the DynAny. |
| void | [**insert\_longlong**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_longlong(long))(long value)            Inserts a long value into the DynAny. |
| void | [**insert\_octet**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_octet(byte))(byte value)            Inserts a byte value into the DynAny. |
| void | [**insert\_reference**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_reference(org.omg.CORBA.Object))([Object](http://docs.google.com/org/omg/CORBA/Object.html) value)            Inserts a reference to a CORBA object into the DynAny. |
| void | [**insert\_short**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_short(short))(short value)            Inserts a short value into the DynAny. |
| void | [**insert\_string**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_string(java.lang.String))([String](http://docs.google.com/java/lang/String.html) value)            Inserts a string value into the DynAny. |
| void | [**insert\_typecode**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_typecode(org.omg.CORBA.TypeCode))([TypeCode](http://docs.google.com/org/omg/CORBA/TypeCode.html) value)            Inserts a TypeCode object into the DynAny. |
| void | [**insert\_ulong**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_ulong(int))(int value)            Inserts an integer value into the DynAny. |
| void | [**insert\_ulonglong**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_ulonglong(long))(long value)            Inserts a long value into the DynAny. |
| void | [**insert\_ushort**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_ushort(short))(short value)            Inserts a short value into the DynAny. |
| void | [**insert\_val**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_val(java.io.Serializable))([Serializable](http://docs.google.com/java/io/Serializable.html) value)            Inserts a reference to a Serializable object into this DynAny. |
| void | [**insert\_wchar**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_wchar(char))(char value)            Inserts a char value into the DynAny. |
| void | [**insert\_wstring**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#insert_wstring(java.lang.String))([String](http://docs.google.com/java/lang/String.html) value)            Inserts a string value into the DynAny. |
| boolean | [**next**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#next())()            Advances the current position to the next component. |
| void | [**rewind**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#rewind())()            Is equivalent to seek(0). |
| boolean | [**seek**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#seek(int))(int index)            Sets the current position to index. |
| [Any](http://docs.google.com/org/omg/CORBA/Any.html) | [**to\_any**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#to_any())()            Creates an any value from a DynAny object. |
| [TypeCode](http://docs.google.com/org/omg/CORBA/TypeCode.html) | [**type**](http://docs.google.com/org/omg/DynamicAny/DynAnyOperations.html#type())()            Returns the TypeCode associated with this DynAny object. |

| **Method Detail** |
| --- |

### type

[TypeCode](http://docs.google.com/org/omg/CORBA/TypeCode.html) **type**()

Returns the TypeCode associated with this DynAny object. A DynAny object is created with a TypeCode value assigned to it. This TypeCode value determines the type of the value handled through the DynAny object. Note that the TypeCode associated with a DynAny object is initialized at the time the DynAny is created and cannot be changed during lifetime of the DynAny object.

**Returns:**The TypeCode associated with this DynAny object

### assign

void **assign**([DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) dyn\_any)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html)

Initializes the value associated with a DynAny object with the value associated with another DynAny object. The current position of the target DynAny is set to zero for values that have components and to -1 for values that do not have components.

**Parameters:**dyn\_any - **Throws:** TypeMismatch - if the type of the passed DynAny is not equivalent to the type of target DynAny

### from\_any

void **from\_any**([Any](http://docs.google.com/org/omg/CORBA/Any.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Initializes the value associated with a DynAny object with the value contained in an any. The current position of the target DynAny is set to zero for values that have components and to -1 for values that do not have components.

**Throws:** TypeMismatch - if the type of the passed Any is not equivalent to the type of target DynAny InvalidValue - if the passed Any does not contain a legal value (such as a null string)

### to\_any

[Any](http://docs.google.com/org/omg/CORBA/Any.html) **to\_any**()

Creates an any value from a DynAny object. A copy of the TypeCode associated with the DynAny object is assigned to the resulting any. The value associated with the DynAny object is copied into the any.

**Returns:**a new Any object with the same value and TypeCode

### equal

boolean **equal**([DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) dyn\_any)

Compares two DynAny values for equality. Two DynAny values are equal if their TypeCodes are equivalent and, recursively, all component DynAnys have equal values. The current position of the two DynAnys being compared has no effect on the result of equal.

**Returns:**true of the DynAnys are equal, false otherwise

### destroy

void **destroy**()

Destroys a DynAny object. This operation frees any resources used to represent the data value associated with a DynAny object. It must be invoked on references obtained from one of the creation operations on the ORB interface or on a reference returned by DynAny.copy() to avoid resource leaks. Invoking destroy on component DynAny objects (for example, on objects returned by the current\_component operation) does nothing. Destruction of a DynAny object implies destruction of all DynAny objects obtained from it. That is, references to components of a destroyed DynAny become invalid. Invocations on such references raise OBJECT\_NOT\_EXIST. It is possible to manipulate a component of a DynAny beyond the life time of the DynAny from which the component was obtained by making a copy of the component with the copy operation before destroying the DynAny from which the component was obtained.

### copy

[DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) **copy**()

Creates a new DynAny object whose value is a deep copy of the DynAny on which it is invoked. The operation is polymorphic, that is, invoking it on one of the types derived from DynAny, such as DynStruct, creates the derived type but returns its reference as the DynAny base type.

**Returns:**a deep copy of the DynAny object

### insert\_boolean

void **insert\_boolean**(boolean value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a boolean value into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_octet

void **insert\_octet**(byte value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a byte value into the DynAny. The IDL octet data type is mapped to the Java byte data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_char

void **insert\_char**(char value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a char value into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_short

void **insert\_short**(short value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a short value into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_ushort

void **insert\_ushort**(short value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a short value into the DynAny. The IDL ushort data type is mapped to the Java short data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_long

void **insert\_long**(int value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts an integer value into the DynAny. The IDL long data type is mapped to the Java int data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_ulong

void **insert\_ulong**(int value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts an integer value into the DynAny. The IDL ulong data type is mapped to the Java int data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_float

void **insert\_float**(float value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a float value into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_double

void **insert\_double**(double value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a double value into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_string

void **insert\_string**([String](http://docs.google.com/java/lang/String.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a string value into the DynAny. Both bounded and unbounded strings are inserted using this method.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 InvalidValue - if the string inserted is longer than the bound of a bounded string TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_reference

void **insert\_reference**([Object](http://docs.google.com/org/omg/CORBA/Object.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a reference to a CORBA object into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_typecode

void **insert\_typecode**([TypeCode](http://docs.google.com/org/omg/CORBA/TypeCode.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a TypeCode object into the DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_longlong

void **insert\_longlong**(long value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a long value into the DynAny. The IDL long long data type is mapped to the Java long data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_ulonglong

void **insert\_ulonglong**(long value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a long value into the DynAny. The IDL unsigned long long data type is mapped to the Java long data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_wchar

void **insert\_wchar**(char value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a char value into the DynAny. The IDL wchar data type is mapped to the Java char data type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_wstring

void **insert\_wstring**([String](http://docs.google.com/java/lang/String.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a string value into the DynAny. Both bounded and unbounded strings are inserted using this method.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 InvalidValue - if the string inserted is longer than the bound of a bounded string [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html)

### insert\_any

void **insert\_any**([Any](http://docs.google.com/org/omg/CORBA/Any.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts an Any value into the Any represented by this DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_dyn\_any

void **insert\_dyn\_any**([DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts the Any value contained in the parameter DynAny into the Any represented by this DynAny.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### insert\_val

void **insert\_val**([Serializable](http://docs.google.com/java/io/Serializable.html) value)  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Inserts a reference to a Serializable object into this DynAny. The IDL ValueBase type is mapped to the Java Serializable type.

**Throws:** InvalidValue - if this DynAny has components but has a current position of -1 TypeMismatch - if called on a DynAny whose current component itself has components

### get\_boolean

boolean **get\_boolean**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the boolean value from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_octet

byte **get\_octet**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the byte value from this DynAny. The IDL octet data type is mapped to the Java byte data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_char

char **get\_char**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the char value from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_short

short **get\_short**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the short value from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_ushort

short **get\_ushort**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the short value from this DynAny. The IDL ushort data type is mapped to the Java short data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_long

int **get\_long**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the integer value from this DynAny. The IDL long data type is mapped to the Java int data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_ulong

int **get\_ulong**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the integer value from this DynAny. The IDL ulong data type is mapped to the Java int data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_float

float **get\_float**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the float value from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_double

double **get\_double**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the double value from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_string

[String](http://docs.google.com/java/lang/String.html) **get\_string**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the string value from this DynAny. Both bounded and unbounded strings are extracted using this method.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_reference

[Object](http://docs.google.com/org/omg/CORBA/Object.html) **get\_reference**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the reference to a CORBA Object from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_typecode

[TypeCode](http://docs.google.com/org/omg/CORBA/TypeCode.html) **get\_typecode**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the TypeCode object from this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_longlong

long **get\_longlong**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the long value from this DynAny. The IDL long long data type is mapped to the Java long data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_ulonglong

long **get\_ulonglong**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the long value from this DynAny. The IDL unsigned long long data type is mapped to the Java long data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_wchar

char **get\_wchar**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the long value from this DynAny. The IDL wchar data type is mapped to the Java char data type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_wstring

[String](http://docs.google.com/java/lang/String.html) **get\_wstring**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the string value from this DynAny. Both bounded and unbounded strings are extracted using this method.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

### get\_any

[Any](http://docs.google.com/org/omg/CORBA/Any.html) **get\_any**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts an Any value contained in the Any represented by this DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_dyn\_any

[DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) **get\_dyn\_any**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts the Any value contained in the Any represented by this DynAny and returns it wrapped into a new DynAny.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### get\_val

[Serializable](http://docs.google.com/java/io/Serializable.html) **get\_val**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html),  
 [InvalidValue](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/InvalidValue.html)

Extracts a Serializable object from this DynAny. The IDL ValueBase type is mapped to the Java Serializable type.

**Throws:** TypeMismatch - if the accessed component in the DynAny is of a type that is not equivalent to the requested type. TypeMismatch - if called on a DynAny whose current component itself has components InvalidValue - if this DynAny has components but has a current position of -1

### seek

boolean **seek**(int index)

Sets the current position to index. The current position is indexed 0 to n-1, that is, index zero corresponds to the first component. The operation returns true if the resulting current position indicates a component of the DynAny and false if index indicates a position that does not correspond to a component. Calling seek with a negative index is legal. It sets the current position to -1 to indicate no component and returns false. Passing a non-negative index value for a DynAny that does not have a component at the corresponding position sets the current position to -1 and returns false.

### rewind

void **rewind**()

Is equivalent to seek(0).

### next

boolean **next**()

Advances the current position to the next component. The operation returns true while the resulting current position indicates a component, false otherwise. A false return value leaves the current position at -1. Invoking next on a DynAny without components leaves the current position at -1 and returns false.

### component\_count

int **component\_count**()

Returns the number of components of a DynAny. For a DynAny without components, it returns zero. The operation only counts the components at the top level. For example, if component\_count is invoked on a DynStruct with a single member, the return value is 1, irrespective of the type of the member.

* For sequences, the operation returns the current number of elements.
* For structures, exceptions, and value types, the operation returns the number of members.
* For arrays, the operation returns the number of elements.
* For unions, the operation returns 2 if the discriminator indicates that a named member is active, otherwise, it returns 1.
* For DynFixed and DynEnum, the operation returns zero.

### current\_component

[DynAny](http://docs.google.com/org/omg/DynamicAny/DynAny.html) **current\_component**()  
 throws [TypeMismatch](http://docs.google.com/org/omg/DynamicAny/DynAnyPackage/TypeMismatch.html)

Returns the DynAny for the component at the current position. It does not advance the current position, so repeated calls to current\_component without an intervening call to rewind, next, or seek return the same component. The returned DynAny object reference can be used to get/set the value of the current component. If the current component represents a complex type, the returned reference can be narrowed based on the TypeCode to get the interface corresponding to the to the complex type. Calling current\_component on a DynAny that cannot have components, such as a DynEnum or an empty exception, raises TypeMismatch. Calling current\_component on a DynAny whose current position is -1 returns a nil reference. The iteration operations, together with current\_component, can be used to dynamically compose an any value. After creating a dynamic any, such as a DynStruct, current\_component and next can be used to initialize all the components of the value. Once the dynamic value is completely initialized, to\_any creates the corresponding any value.

**Throws:** TypeMismatch - If called on a DynAny that cannot have components, such as a DynEnum or an empty exception

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DynAnyOperations.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/omg/DynamicAny/DynAnyHelper.html)   [**NEXT CLASS**](http://docs.google.com/org/omg/DynamicAny/DynAnySeqHelper.html) | [**FRAMES**](http://docs.google.com/index.html?org/omg/DynamicAny/DynAnyOperations.html)    [**NO FRAMES**](http://docs.google.com/DynAnyOperations.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).