

init

(double)doubleValue

Returns the DBValue's value converted to a double-precision floating-point number. If the conversion is not performed, a DB\_COERCION\_ERROR exception is raised.

(float)floatValue

Returns the DBValue's value converted to a single-precision floating-point number. If the conversion is not performed, a DB\_COERCION\_ERROR exception is raised.

free

Frees the DBValue.

init

The designated initializer for the DBValue class, init initializes a newly allocated DBValue object.

(int)intValue

Returns the DBValue's value converted to an integer. If the conversion can't be performed, a DB\_COERCION\_ERROR exception is raised.

(BOOL)isEqual:(DBValue \*)anotherValue

Compares the DBValue with anotherValue and returns YES or NO as their values are or aren't equal. If the objects' types needn't be the same the method will convert the argument's value to that of the receiver, if necessary, and then perform the comparison. A DB\_COERCION\_ERROR exception is raised if the conversion is not supported.

object value

Returns the DBValue's value. The value must be an object, otherwise a DB\_COERCION\_ERROR

read:(NXTypedStream \*)stream

Reads the DBValue from the typed stream stream. Returns self.

setDoubleValue:(double)aDouble

Sets the DBValue's value to aDouble and declares its type to be a double. Returns self.

setFloatValue:(float)aFloat

Sets the DBValue's value to aFloat and declares its type to be a float. Returns self.

setIntValue:(int)anInt

Sets the DBValue's value to anInt and declares its type to be an integer. Returns self.

setNull

Sets the DBValue's value and type to NULL. Returns self.

setObjectValue:(id)anObject

Sets the DBValue's value to a copy of anObject and declares its type to be an object. Returns self.

setObjectValueNoCopy:(id)anObject

Sets the DBValue's value to anObject and declares its type to be an object. Returns self.

setStringValue:(const char \*)aString

Sets the DBValue's value to a copy of aString and declares its type to be a string. Returns self.

setStringValueNoCopy:(const char \*)aString

`(const char *)stringValue`

Returns the DBValue's value converted to a string. If the conversion can't be performed, a DB\_C exception is raised.

`(id <DBTypes>)valueType`

Returns a private, DBTypes-conforming object that stores the DBValue's type. To get a string that Objective C data type from this object, you would send it an objcType message. The following table shows string representations of the DBValue types:

`write:(NXTypedStream *)stream`

Writes the DBValue object to the typed stream stream. Returns self.