

copy

NSMutableCopying mutableCopy

copyWithZone:

mutableCopyWithZone:

initWithBool:
initWithChar:
initWithDouble:
initWithFloat:
initWithInt:
initWithLong:
initWithLongLong:
initWithShort:
initWithUnsignedChar:
initWithUnsignedInt:
initWithUnsignedLong:
initWithUnsignedLongLong:
initWithUnsignedShort:

Accessing data boolValue

charValue
doubleValue
floatValue

Comparing data compare:

isEqual:

numberWithUnsignedChar:

numberWithUnsignedInt:

numberWithUnsignedLong:

numberWithUnsignedShort.

numberWithChar:

numberWithInt:

numberWithLong:

numberWithLongLong:

numberWithShort:

(BOOL)boolValue

Returns a BOOL value from a number object.

(char)charValue

Returns a char value from a number object.

unsignedCharValue

·NSOrderedDescending

NSOrderedDescending is also returned when other is not an NSNumber.

The compare: method conforms to the standard C rules for type conversion. For example, if you compare an integer value with a number object that has a floating point value, the integer value is converted to a floating point value.

Two number objects are equal if they have the same value and type.

For example, in this excerpt num1 and num2 evaluate as being equal.

(double)doubleValue

Returns a double value from a number object.

(float)floatValue

Returns a float value from a number object.

This excerpt creates two number objects: num1, which holds an integer value, and num2, which holds a floating point value. The excerpt then prints the floating point value of num1 and the integer value of num2.

initWithBool:(BOOL)value

Initializes the receiver, a newly allocated NSNumber, from value.

initWithChar:(char)value

`initWithFloat:(float)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithInt:(int)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithLong:(long)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithLongLong:(long long)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithShort:(short)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithUnsignedChar:(unsigned char)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithUnsignedInt:(unsigned int)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithUnsignedLong:(unsigned long)value`

Initializes the receiver, a newly allocated NSNumber, from value.

`initWithUnsignedLongLong:(unsigned long long)value`

Initializes the receiver, a newly allocated NSNumber, from value.

unsignedIntValue

(BOOL)isEqual:value

Returns YES if the receiver and value are equal otherwise returns NO. For NSNumber, isEqual: first compares the class of value and the receiver to verify that they are the same. If they are, isEqual: then invokes compare: on the receiver compare as NSOrderedSame, isEqual: returns YES.

(long long)longLongValue

Returns a long long value from a number object.

unsignedLongLongValue

(long)longValue

Returns a long value from a number object.

unsignedLongValue

(short)shortValue

Returns a short value from a number object.

unsignedShortValue

(NSString *)stringValue

Returns a pointer to an NSString object from a number object.

This excerpt creates a number object with an integer value, and then extracts its data as an NSString object.

(unsigned char)unsignedCharValue

Returns an unsigned char value from a number object.

charValue

Returns an unsigned long long value from a number object.

longLongValue

(unsigned long)unsignedLongValue

Returns an unsigned long value from a number object.

longValue

(unsigned short)unsignedShortValue

Returns an unsigned short value from a number object.

shortValue