

Datatype Functions

DoubleToBoolean(Double x)

Converts the given double value to a *boolean* value

DoubleToByte(Double x)

Converts the given double value to a *byte* value

DoubleToChar(Double x)

Converts the given double value to a *char* value

DoubleToShort(Double x)

Converts the given double value to a *short* value

DoubleToInteger(Double x)

Converts the given double value to a *integer* value

DoubleToLong(Double x)

Converts the given double value to a *long* value

DoubleToFloat(Double x)

Converts the given double value to a *float* value

isNull(Object o)

Checks whether the given object is NULL

ToBoolean(Object o)

Converts the given value to a *boolean* value.

ToByte(Object o)

Converts the given value to a *byte* value. This function converts any value to a *byte* value, thus it is more time consuming than the DoubleToByte function for double values.

ToChar(Object o)

Converts the given value to a *char* value. This function converts any value to a *char* value, thus it is more time consuming than the DoubleToChar function for double values.

ToShort(Object o)

Converts the given value to a *short* value. This function converts any value to a *short* value, thus it is more time consuming than the DoubleToShort function for double values.

ToInteger(Object o)

Converts the given value to a *integer* value. This function converts any value to a *integer* value, thus it is more time consuming than the DoubleToInteger function for double values.

ToLong(Object o)

Converts the given value to a *long* value. This function converts any value to a *long* value, thus it is more time consuming than the DoubleToLong function for double values.

ToFloat(Object o)

Converts the given value to a *float* value. This function converts any value to a *float* value, thus it is more time consuming than the DoubleToFloat function for double values.

- [DoubleToBoolean\(Double x\)](#)
- [DoubleToByte\(Double x\)](#)
- [DoubleToChar\(Double x\)](#)
- [DoubleToShort\(Double x\)](#)
- [DoubleToInteger\(Double x\)](#)
- [DoubleToLong\(Double x\)](#)
- [DoubleToFloat\(Double x\)](#)
- [isNull\(Object o\)](#)
- [ToBoolean\(Object o\)](#)
- [ToByte\(Object o\)](#)
- [ToChar\(Object o\)](#)
- [ToShort\(Object o\)](#)
- [ToInteger\(Object o\)](#)
- [ToLong\(Object o\)](#)
- [ToFloat\(Object o\)](#)
- [ToDouble\(Object o\)](#)
- [ToDate\(<Number x>|<String s, Pattern p>\)](#)
- [ToString\(Object o\)](#)
- [ToHex\(<Number x>|<String s>\)](#)
- [ToBinary\(<Number x>|<String s>\)](#)
- [ToTuple\(Object o1, .. , o10\)](#)
- [ToList\(Object o1, .. , o10\)](#)

ToDouble(Object o)

Converts the given value to a *double* value.

ToDate(<Number x>|<String s, Pattern p>)

Converts the given value to a *date* value. The format parameter is necessary if the first parameter is a string value.

ToString(Object o)

Converts the given value to a *string* value.

ToHex(<Number x>|<String s>)

Returns the hexadecimal string representation of the given value.

ToBinary(<Number x>|<String s>)

Returns the binary string representation of the given value.

ToTuple(Object o1, .. , o10)

Creates a tuple with the given values (max 10).

ToList(Object o1, .. , o10)

Creates a list with the given values (max 10).