Поддержка Linq в 1С

# Операции сравнения

Поддерживаются следующие операции сравнения: равно, неравно, больше, больше или равно, меньше, меньше или равно

# Логические операции

Поддерживаются следующие логические операции: NOT, AND, OR.

!с => NOT c

c1 && c2 => c1 AND c2

c1 & c2 => c1 AND c2

c1 || c2 => c1 OR c2

c1 | c2 => c1 OR c2

# Проверка на NULL

value == null => value IS NULL

value != null => value IS NOT NULL

# Арифметические операции

Бинарные операции: +, -, \*, /

Унарная операция отрицания: -

# Операция проверки на тип

value is Type => value REFS DataSource(Type)

# Условие IN со списком значений

values.Contains(value) => value IN (values[0],…,values[N])

!values.Contains(value) => value NOT IN (values[0],…,values[N])

OneSSqlFunctions.In(value, values) => value IN (values[0],…,values[N])

! OneSSqlFunctions.In(value, values) => value NOT IN (values[0],…,values[N])

OneSSqlFunctions.InHierarchy(value, values) => value IN HIERARCHY (values[0],…,values[N])

! OneSSqlFunctions.InHierarchy(value, values) => value NOT IN HIERARCHY (values[0],…,values[N])

# Приведение типа

(bool)value => CAST (value AS BOOLEAN)

SqlFunctions.ToBoolean(value) => CAST (value AS BOOLEAN)

(numberType)value => CAST (value AS NUMBER) , где numberType : short, int, long, float, double

SqlFunctions.ToInt16 (value, length = null) => CAST (value AS NUMBER[(length)])

SqlFunctions.ToInt32 (value, length = null) => CAST (value AS NUMBER[(length)])

SqlFunctions.ToInt64 (value, length = null) => CAST (value AS NUMBER[(length)])

SqlFunctions.ToSingle (value, length = null, precision = null) => CAST (value AS NUMBER[(length, [precision])])

SqlFunctions.ToDouble (value, length = null, precision = null) => CAST (value AS NUMBER[(length, [precision])])

SqlFunctions.ToDecimal(value, length = null, precision = null) => CAST (value AS NUMBER[(length, [precision])])

(string)value => CAST (value AS STRING)

SqlFunctions.ToString(value, length = null) => CAST (value AS STRING[(length)])

(DateTime)value => CAST(value AS DATE)

SqlFunctions.ToDateTime(value) => CAST (value AS DATE)

(RecordType)value => CAST(value AS tableName), где tableName := DataSourceName(RecordType)

SqlFunctions.ToDataRecord(value, tableName) : OneSDataRecord => CAST(value AS tableName)