



□ Как получить доступ к реквизитам объекта?

Статья носит справочный характер. Предназначена для начинающих программистов.

Получить массив реквизитов объекта можно при помощи самого объекта:

```
{codecitation style="brush: xml;"}
    ДокСсылка = СсылкаНаДокумент(ТипДокумента, НомерДок);
    ДокОбъект = ДокСсылка.ПолучитьОбъект();

    МассивРеквизитовОбъекта = ДокОбъект.Метаданные().Реквизиты;

{/codecitation}
```

Получив массив реквизитов, можно организовать Цикл для просмотра имен реквизитов:

```
{codecitation style="brush: xml;"}  
  
Для каждого Реквизит Из ДокОбъект.Метаданные().Реквизиты Цикл  
Сообщить(Реквизит.Имя + " :" + Реквизит.Представление());  
КонецЦикла;  
  
{/codecitation}
```

Для того, чтобы получить значения реквизитов объекта вообще говоря надо обратиться с запросом к БД, типа:

```
{codecitation style="brush: xml;"}  
  
Запрос = "ВЫБРАТЬ  
|      ИмяРеквизита,  
|      ИЗ  
|      Объект.ТипОбъекта КАК ОбъектТипОбъекта  
|      ГДЕ  
|      ОбъектТипОбъекта.Ссылка = &СсылкаНаОбъект  
  
Запрос.УстановитьПараметр("СсылкаНаОбъект", СсылкаНаОбъект);
```

```
// Возвращает структуру, содержащую значения реквизитов прочитанные из
информационной базы
// по ссылке на объект.
//
// Если доступа к одному из реквизитов нет, возникнет исключение прав доступа.
// Если необходимо зачитать реквизит независимо от прав текущего пользователя,
// то следует использовать предварительный переход в привилегированный режим.
//
// Параметры:
// Ссылка      - ссылка на объект, - элемент справочника, документ, ...
// ИменаРеквизитов - Строка или Структура - Если Строка, то имена реквизитов,
//              перечисленные через запятую, в формате требований к свойствам структуры.
```

```
//      Например, "Код, Наименование, Родитель".
//      Если Структура, то в качестве ключа передается псевдоним поля для
//      возвращаемой структуры с результатом, а в качестве значения (опционально)
//      - фактическое имя поля в таблице.
//      Если значение не определено, то имя поля берется из ключа.
//
// Возвращаемое значение:
// Структура  - содержит список свойств, как список имен в строке
//      ИменаРеквизитов, со значениям реквизитов, прочитанных
//      из информационной базы.
//
Функция ПолучитьЗначенияРеквизитов(Ссылка, ИменаРеквизитов) Экспорт
```

```
    Если ТипЗнч(ИменаРеквизитов) = Тип("Структура") Тогда
        СтруктураРеквизитов = ИменаРеквизитов;
    ИначеЕсли ТипЗнч(ИменаРеквизитов) = Тип("Строка") Тогда
        СтруктураРеквизитов = Новый Структура(ИменаРеквизитов);;
    Иначе
        ВызватьИсключение
СтроковыеФункцииКлиентСервер.ПодставитьПараметрыВСтроку(
        НСтр("ru = 'Неверный тип второго параметра
ИменаИлиСтруктураРеквизитов: %1'",
        Строка(ТипЗнч(ИменаРеквизитов)));
    КонецЕсли;
```

```
    ТекстПолей = "";
    Для каждого КлючИЗначение Из СтруктураРеквизитов Цикл
        ИмяПоля  = ?(ЗначениеЗаполнено(КлючИЗначение.Значение),
СокрЛП(КлючИЗначение.Значение), СокрЛП(КлючИЗначение.Ключ));
        Псевдоним = СокрЛП(КлючИЗначение.Ключ);
        ТекстПолей = ТекстПолей + ?(ПустаяСтрока(ТекстПолей), "", ",") + "
        |      " + ИмяПоля + " КАК " + Псевдоним;
    КонецЦикла;
```

```
    Запрос = Новый Запрос(
        "ВЫБРАТЬ
        |" + ТекстПолей + "
        |ИЗ
        |      " + Ссылка.Метаданные().ПолноеИмя() + " КАК
ПсевдонимЗаданнойТаблицы
```

|ГДЕ

| ПсевдонимЗаданнойТаблицы.Ссылка = &Ссылка

|");

Запрос.УстановитьПараметр("Ссылка", Ссылка);

Выборка = Запрос.Выполнить().Выбрать();

Выборка.Следующий();

Результат = Новый Структура;

Для каждого КлючИЗначение Из СтруктураРеквизитов Цикл

Результат.Вставить(КлючИЗначение.Ключ);

КонецЦикла;

ЗаполнитьЗначенияСвойств(Результат, Выборка);

Возврат Результат;

КонецФункции

{/codecitation}

Значение реквизитов в этой функции получено через запрос к БД. Возвращаемое значение - структура (ключ-значение). Пример использования:

{codecitation style="brush: xml;"}

```
ДокСсылка = СсылкаНаДокумент(ТипДокумента, НомерДок);
```

```
СтруктураНастроек = Новый Структура;
```

```
//инициируем структуру настроек именами реквизитов  
Для каждого Реквизит Из ДокОбъект.Метаданные().Реквизиты Цикл  
    СтруктураНастроек.Вставить(Реквизит.Имя, Реквизит.Имя);  
КонецЦикла;
```

```
ЗначенияРеквизитов = ПолучитьЗначенияРеквизитов(ДокСсылка ,  
СтруктураНастроек);  
СканПростойСтруктуры(ЗначенияРеквизитов); //вывести структуру для просмотра
```

```
{/codecitation}
```

Поскольку в функцию **ПолучитьЗначенияРеквизитов()** во второй параметр надо передать структуру, то мы создали структуру и организовали Цикл ее инициализации данными из массива имен реквизитов объекта. Возвращает функция так же структуру (ключ-значение), с которой удобно работать. Для просмотра этой структуры можно воспользоваться циклом, либо вызвать функцию

СканПростойСтруктуры()

, которой передать структуру для просмотра.

```
{codecitation style="brush: xml;"}  

```

```
Функция СканПростойСтруктуры(Структура)
    Для каждого Элемент из Структура Цикл
        Сообщить(Элемент.Ключ + ": " + Элемент.Значение);
    КонецЦикла;
КонецФункции
```

{/codecitation}

Если исследуемый объект использует так же еще и общие реквизиты, и их необходимо найти, то можно воспользоваться функцией **МассивИменРеквизитовОбъекта()**, которая находит не только реквизиты объекта но и общие и даже стандартные реквизиты, которые использует объект:

```
{codecitation style="brush: xml;"}

```

```
// Возвращает массив имен всех реквизитов переданного объекта
//
Функция МассивИменРеквизитовОбъекта(Объект) Экспорт
```

```
    МассивИменРеквизитов = Новый Массив;
```

```
Если ТипЗнч(Объект) = Тип("ОбъектМетаданных") Тогда
    МетаданныеОбъекта = Объект;
Иначе
    МетаданныеОбъекта = Метаданные.НайтиПоТипу(ТипЗнч(Объект));
    Если МетаданныеОбъекта = Неопределено Тогда
        Возврат МассивИменРеквизитов;
    КонецЕсли;
КонецЕсли;

Для индекс = 0 по 1 Цикл
//    КоллекцияРеквизитов = ?(индекс = 0,
МетаданныеОбъекта.СтандартныеРеквизиты, МетаданныеОбъекта.Реквизиты);
    КоллекцияРеквизитов = МетаданныеОбъекта.Реквизиты;

    Для каждого Реквизит Из КоллекцияРеквизитов Цикл
        МассивИменРеквизитов.Добавить(Реквизит.Имя);
    КонецЦикла;

КонецЦикла;

Для каждого ОбщийРеквизит Из Метаданные.ОбщиеРеквизиты Цикл
    Если ИспользуетсяОбщийРеквизит(ОбщийРеквизит, МетаданныеОбъекта) Тогда
        МассивИменРеквизитов.Добавить(ОбщийРеквизит.Имя);
    КонецЕсли;
КонецЦикла;

Возврат МассивИменРеквизитов;

КонецФункции

{/codecitation}
```


Тогда программа будет выглядеть так:

```
{codecitation style="brush: xml;"}  
  

```

```
ДокСсылка = СсылкаНаДокумент(ТипДокумента, НомерДок);  
ДокОбъект = ДокСсылка.ПолучитьОбъект();
```

```
мМассивИмен = МассивИменРеквизитовОбъекта(ДокОбъект);
```

```
СтруктураНастроек = Новый Структура;  
Для каждого эл Из мМассивИмен Цикл  
    СтруктураНастроек.Вставить(эл, эл);  
КонецЦикла;
```

```
ЗначенияРеквизитов =  
ОбщегоНазначения.ПолучитьЗначенияРеквизитов(ДокСсылка , СтруктураНастроек);  
СканПростойСтруктуры(ЗначенияРеквизитов);
```

```
{/codecitation}
```

```
{codecitation style="brush: xml;"}
```

```

Если ТипЗнч(ОбщийРеквизит) = Тип("ОбъектМетаданных") Тогда
    МетаданныеОбщегоРеквизита = ОбщийРеквизит;
Иначе
    МетаданныеОбщегоРеквизита =
Метаданные.ОбщиеРеквизиты.Найти(ОбщийРеквизит);
    Если МетаданныеОбщегоРеквизита = Неопределено Тогда
        Возврат Ложь;
    КонецЕсли;
КонецЕсли;

```

```
ЭлементСостава =  
МетаданныеОбщегоРеквизита.Состав.Найти(МетаданныеОбъекта);
```

```
Если ЭлементСостава = Неопределено Тогда  
    Возврат Ложь;  
КонецЕсли;
```

```
пИспользованиеОбщегоРеквизита =  
Метаданные.СвойстваОбъектов.ИспользованиеОбщегоРеквизита;
```

```
Если ЭлементСостава.Использование =  
пИспользованиеОбщегоРеквизита.Использовать Тогда  
    Возврат Истина;  
ИначеЕсли ЭлементСостава.Использование =  
пИспользованиеОбщегоРеквизита.НеИспользовать Тогда  
    Возврат Ложь;  
Иначе  
    пАвтоИспользованиеОбщегоРеквизита =  
Метаданные.СвойстваОбъектов.АвтоИспользованиеОбщегоРеквизита;
```

```
Если МетаданныеОбщегоРеквизита.АвтоИспользование =  
пАвтоИспользованиеОбщегоРеквизита.Использовать Тогда  
    Возврат Истина;  
Иначе  
    Возврат Ложь;  
КонецЕсли;  
КонецЕсли;
```

```
КонецФункции
```

```
{/codecitation}
```

Иногда необязательно получать значения всего массива реквизитов. Программист как правило знает имя реквизита, значение которого необходимо получить. К отдельному реквизиту объекта можно обратиться просто по его имени:

Поставщик = ДокОбъект.Метаданные().Реквизиты.Поставщик;

```
{/codecitation}
```

Получить значение этого реквизита можно так же через запрос, написанный специально для этого реквизита. Но можно воспользоваться все той же функцией **ПолучитьЗначенияРеквизитов()**

. Цикл для настройки структуры в этом случае не нужен, так как нам надо найти значение всего одного элемента:

```
{codecitation style="brush: xml;"}
```

```
ДокСсылка = СсылкаНаДокумент(ТипДокумента, НомерДок);  
ДокОбъект = ДокСсылка.ПолучитьОбъект();
```

```
Организация = Строка(ДокОбъект.Метаданные().Реквизиты.Организация);  
СтруктураНастроек = Новый Структура;  
СтруктураНастроек.Вставить(Организация, Организация);
```

```
ЗначенияРеквизитов =  
ОбщегоНазначения.ПолучитьЗначенияРеквизитов(ДокСсылка , СтруктураНастроек);  
СканПростойСтруктуры(ЗначенияРеквизитов);
```

{/codecitation}

Обратите внимание что здесь реквизит который необходимо найти преобразовывается к типу Строка, потому, что в структуру настроек надо передать именно строку.

Как найти имена реквизитов в таб части объекта?

Функция определяет есть ли реквизит в таб части документа:

```
{codecitation style="brush: xml;"}  
  

```

```
// Позволяет определить есть ли среди реквизитов табличной части документа  
// реквизит с переданным именем.
```

```
//  
// Параметры:  
// ИмяРеквизита - строковое имя искомого реквизита,  
// МетаданныеДокумента - объект описания метаданных документа, среди реквизитов  
// которого производится поиск.  
// ИмяТабЧасти - строковое имя табличной части документа, среди реквизитов которого  
// производится поиск  
//  
// Возвращаемое значение:  
// Истина - нашли реквизит с таким именем, Ложь - не нашли.  
//
```

Функция ЕстьРеквизитТабЧастиДокумента(ИмяРеквизита, МетаданныеДокумента,
ИмяТабЧасти) Экспорт

```
ТабЧасть = МетаданныеДокумента.ТабличныеЧасти.Найти(ИмяТабЧасти);
```

```
Если ТабЧасть = Неопределено Тогда // Нет такой таб. части в документе  
    Возврат Ложь;  
Иначе  
    Возврат НЕ (ТабЧасть.Реквизиты.Найти(ИмяРеквизита) = Неопределено);
```

```
КонецЕсли;
```

```
КонецФункции
```

```
{/codecitation}
```

Пример использования:

{/codecitation}

```
{codecitation style="brush: xml;"}
```


Функция ПолучитьЗначенияРеквизитовТабЧасти(Ссылка, ИменаРеквизитов, ИмяТабЧасти) Экспорт

```
ИмяТЧ = ИмяТабЧасти;
ПолноеИмя = Ссылка.Метаданные().ПолноеИмя();

Если ТипЗнч(ИменаРеквизитов) = Тип("Структура") Тогда
    СтруктураРеквизитов = ИменаРеквизитов;
ИначеЕсли ТипЗнч(ИменаРеквизитов) = Тип("Строка") Тогда
    СтруктураРеквизитов = Новый Структура(ИменаРеквизитов);;
Иначе
    ВызватьИсключение
СтроковыеФункцииКлиентСервер.ПодставитьПараметрыВСтроку(
    НСтр("ru = 'Неверный тип второго параметра
ИменаИлиСтруктураРеквизитов: %1'",
        Строка(ТипЗнч(ИменаРеквизитов))));
КонецЕсли;

ТекстПолей = "";
Для Каждого КлючИЗначение Из СтруктураРеквизитов Цикл
    ИмяПоля = ?(ЗначениеЗаполнено(КлючИЗначение.Значение),
СокрЛП(КлючИЗначение.Значение), СокрЛП(КлючИЗначение.Ключ));
    Псевдоним = СокрЛП(КлючИЗначение.Ключ);
    ТекстПолей = ТекстПолей + ?(ПустаяСтрока(ТекстПолей), "", ",") + "
        | " + ИмяПоля + " КАК " + Псевдоним;
КонецЦикла;

Запрос = Новый Запрос(
    "ВЫБРАТЬ
    | " + ТекстПолей + "
    | ИЗ
    | " + ПолноеИмя + "." + ИмяТЧ + " КАК ПсевдонимЗаданнойТаблицы
    | ГДЕ
    | ПсевдонимЗаданнойТаблицы.Ссылка = &Ссылка
    |");

Запрос.УстановитьПараметр("Ссылка", Ссылка);
Результат = Запрос.Выполнить().Выгрузить();

Возврат Результат;
```

{/codecitation}

```
{codecitation style="brush: xml;"}

```

Сообщить("Имя таб части: " + ТабЧасть);

{/codecitation}

Вспомогательная функция:

```
{codecitation style="brush: xml;"}
```

функция ПоказатьТаблицу(Табл)

Сообщить("///Функция ПоказатьТаблицу///");

```
Количество = Табл.Колонки.Количество();
```

Для каждого Строка из Табл Цикл

Для Индекс = 0 По Количество-1 Цикл

Сообщить(СокрЛП(Индекс)+""+СокрЛП(Табл.Колонки[Индекс].Имя)+": "+

```
Строка[СокрЛП(Табл.Колонки[Индекс].Имя)];  
    КонецЦикла;  
    КонецЦикла;  
    Сообщить("///Конец Функция ПоказатьТаблицу///");  
КонецФункции
```

{/codecitation}

Результат для документа "Реализация товаров услуг":

Имя таб части: Товары
///Функция ПоказатьТаблицу///
0 ЕдиницаИзмерения: шт
1 ЕдиницаИзмеренияМест:
2 Качество: Новый
3 Количество: 1
4 КоличествоМест: 0
5 Коэффициент: 1
6 Номенклатура: Вентилятор настольный
7 ПроцентСкидкиНаценки: 0
8 СерияНоменклатуры: Н-908, С-890 от 01.01.2003
9 СпособСписанияОстаткаТоваров: Со склада
10 СтавкаНДС: Без НДС
11 Сумма: 445,5
12 СуммаНДС: 0
13 ХарактеристикаНоменклатуры:
14 Цена: 450
15 Склад: Магазин "Бытовая техника"
16 ПроцентАвтоматическихСкидок: 1
17 УсловиеАвтоматическойСкидки: По виду дисконтных карт
18 ЗначениеУсловияАвтоматическойСкидки: Серебряная карта
19 КлючСтроки: 3
20 КлючСвязи: 0
21 ЗаказПокупателя: Заказ покупателя ТД0000000001 от 08.06.2008 12:00:00
0 ЕдиницаИзмерения: шт
1 ЕдиницаИзмеренияМест:
2 Качество: Новый
3 Количество: 1

4 КоличествоМест: 0
5 Коэффициент: 1
6 Номенклатура: Набор вентиляторов
7 ПроцентСкидкиНаценки: 0
8 СерияНоменклатуры:
9 СпособСписанияОстаткаТоваров: Со склада
10 СтавкаНДС: Без НДС
11 Сумма: 1948,32
12 СуммаНДС: 0
13 ХарактеристикаНоменклатуры:
14 Цена: 1968
15 Склад: Магазин "Бытовая техника"
16 ПроцентАвтоматическихСкидок: 1
17 УсловиеАвтоматическойСкидки: По виду дисконтных карт
18 ЗначениеУсловияАвтоматическойСкидки: Серебряная карта
19 КлючСтроки: 4
20 КлючСвязи: 0
21 ЗаказПокупателя: Заказ покупателя ТД000000001 от 08.06.2008 12:00:00
///Конец Функция ПоказатьТаблицу///

Имя таб части: ВозвратнаяТара
///Функция ПоказатьТаблицу///
///Конец Функция ПоказатьТаблицу///

Имя таб части: Услуги
///Функция ПоказатьТаблицу///
///Конец Функция ПоказатьТаблицу///

Имя таб части: СоставНабора
///Функция ПоказатьТаблицу///
0 Номенклатура: Вентилятор BINATONE ALPINE 160вт, напольный ,
1 ХарактеристикаНоменклатуры:
2 СерияНоменклатуры: 01234/11020/7654321, БОЛГАРИЯ
3 Количество: 1
4 ЕдиницаИзмерения: шт
5 Склад: Магазин "Бытовая техника"
6 Качество: Новый
7 СпособСписанияОстаткаТоваров: Со склада
8 КлючСтроки: 4
9 Цена: 1068

10 КлючСвязи: 0
0 Номенклатура: Вентилятор JIPONIC (Тайв.),
1 ХарактеристикаНоменклатуры:
2 СерияНоменклатуры: 11234/11020/7654321, ТАЙВАНЬ (КИТАЙ)
3 Количество: 1
4 ЕдиницаИзмерения: шт
5 Склад: Магазин "Бытовая техника"
6 Качество: Новый
7 СпособСписанияОстаткаТоваров: Со склада
8 КлючСтроки: 4
9 Цена: 900
10 КлючСвязи: 0
///Конец Функция ПоказатьТаблицу///

Имя таб части: СерийныеНомера

///Функция ПоказатьТаблицу///
///Конец Функция ПоказатьТаблицу///

Имя таб части: СерийныеНомераСоставНабора

///Функция ПоказатьТаблицу///
///Конец Функция ПоказатьТаблицу///

Имя таб части: ДокументыРасчетовСКонтрагентом

///Функция ПоказатьТаблицу///
///Конец Функция ПоказатьТаблицу///