

1. Основы языка запросов 1С

- назначение
- операторы
- консоль запросов
- конструктор ЗАПРОСОВ
- запросы из нескольких таблиц: Соединение, Объединение; Срез последних

2. Использование запроса в программном коде

- Запрос – как объект программного кода
- порядок «встраивания» текста и результатов запроса в программный код

1. Основы языка запросов 1С



Для извлечения и обработки данных из БД в 1С используют запросы, которые пишутся на собственном (SQL-подобном) языке запросов.

Механизм запросов в 1С отвечает только за *получение* данных и не вносит изменения в БД: посредством запросов реализуется обращение к системе для *выбора* определенной информации из базы данных и, возможно, некоторой ее обработки: группировка, сортировка, некоторых видов вычислений

ВЫБРАТЬ	означает начало запроса: возможности : выбор <i>всех полей</i> таблицы-источника (определяется через *); применимо для реальных таблиц перечень полей – через запятую выбор <i>определенных полей</i> таблицы-источника; применимо как для реальных , так и виртуальных таблиц
ИЗ	указание объекта запроса
КАК	присваивание псевдонима полям и таблицам-источникам данных
ГДЕ	ключевое слово блока Условие (в языке запросов)
и др.	

<https://its.1c.ru/db/metod8dev/content/2590/hdoc>

Для конструирования, отладки и просмотра результатов выполнения запросов в режиме 1С:Предприятие (пользовательском клиентском приложении) м.б. использоваться Инструмент «Консоль запросов»

«Консоль запросов» -- это специальная *внешняя обработка* (файл с расширением .erf)

Существуют готовые решения по Консоли запросов (м. получить на сайте ИТС)

- 1) Скачиваем с сайта ИТС (файл .erf)
- 2) «встраиваем» в информационную базу:
тип объекта «Обработки» (в составе дерева конфигурации);
контекстное меню – «Вставить внешнюю ...»
- 3) Запускаем из пользовательского приложения (кн. Сервис)

ПРИМЕР1 -- запуск запроса и создание поля со значением «5»

ВЫБРАТЬ

5

Поле1
5

ПРИМЕР2 -- запуск запроса и создание полей со значениями разл. типов

ВЫБРАТЬ

5,

ложь,

"текст.стр."

Запрос: (Записей в результате: 1)

Поле1	Поле2	Поле3
5	Нет	текст.стр.

Пример3 -- запуск запроса, создание полей со значениями разл. Типов, присваивание псевдонимов полям

ВЫБРАТЬ

5 КАК ПолеСЧислом,

ложь КАК ПолеСБулево,

"текст.стр." КАК ПолеСТекстом

Запрос: (Записей в результате: 1)

ПолеСЧислом	ПолеСБулево	ПолеСТекстом
5	Нет	текст.стр.

ПРИМЕР4 -- использование данных инф.базы :
запрос всех данных из объекта Справочник.НоменклатураТоваров

исп.:

- знак *
- слово ИЗ

Запрос: СПРАВОЧНИК.НоменклатураТоваров (Записей в результате: 4)

Ссылка	Версия	Данные	Томета	Удаление	Код	Наименование	Предопределенный	Бит	Гр
Чайник	AAAAAQAAAA	Нет			1	Чайник	Нет		
Утюг	AAAAAQAAAA	Нет			2	Утюг	Нет		
Кофейник	AAAAAQAAAA	Нет			3	Кофейник	Нет		
Телевизор	AAAAAQAAAA	Нет			4	Телевизор	Нет		

ПРИМЕР5 -- запрос данных из объекта Справочник.НоменклатураТоваров - по полям Код, Наименование

Запрос: Справочник.НоменклатураТоваров (Зап

Код	Наименование
1	Чайник
2	Утюг
3	Кофейник
4	Телевизор

ПРИМЕР6-- присваивание псевдонимов наименованиям полей

Запрос: Справочник.НоменклатураТ

Код	Наименование
1	Чайник
2	Утюг
3	Кофейник
4	Телевизор

ПРИМЕР7 -- присваивание псевдонима источнику данных запроса, т.е. таблице —объекту конфигурации (это м.б. востребовано, например, в случае, когда запрос содержит обращение к неск. таблицам)

ВЫБРАТЬ

НашаНоменклатура.Код как НашКод,

НашаНоменклатура.Наименование как НашеНаименование

из

Справочник.НоменклатураТоваров КАК НашаНоменклатура

ПРИМЕР8 -- использование блока условия (ГДЕ) – выбор из Справочник.НоменклатураТоваров элементов с кодом >2

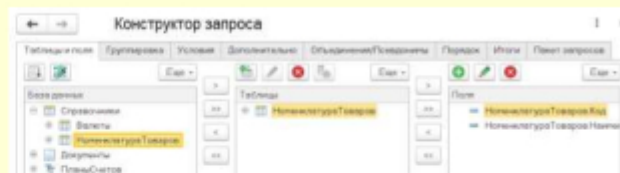
Запрос: Справочник.НоменклатураТ

Код	Наименование
2	Утюг
3	Кофейник
4	Телевизор

!!!
ГДЕ –
после
из

!!!
КОД –
числ.
тип

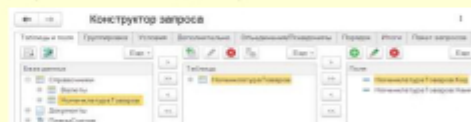
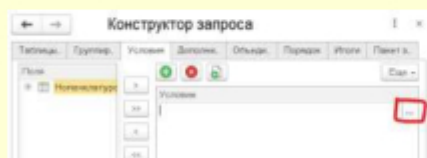
- запуск конструктора – в пустом поле из контекстного меню
- выбор н. табл.и полей
- запуск констр-ра - Ok



ПРИМЕР9 –

- 1) Выполнить задание по прим. 5, 8 с использованием конструктора запросов (запрос по полям КОД, НАИМЕНОВАНИЕ; использование блока условие : код >2) :
 - вкл. Таблицы и поля – выбор таблицы-источника и полей

- вкл. Условие –



!!!
Ранее
написанный
тест
запроса –
лучше
убрать

- 2) Провести анализ кода, сгенерированного конструктором

Использование ПАРАМЕТРОВ

ПАРАМЕТРЫ – это некоторая находящаяся внутри запроса переменная, которая задается извне и в зависимости от ее значения б. выводиться результат запроса – каждый раз новый при одном и том же тексте запроса

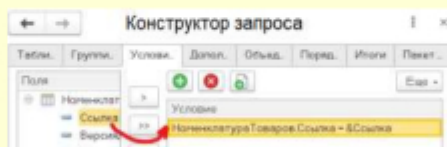
(ПАРАМЕТР обозначается в тексте запроса символом &)

Пример10

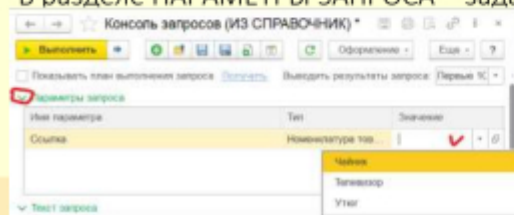
-Продолжение прим.9 :

Включение в запрос параметра ССЫЛКА

- 1) На вкл. Условие – задание параметра генерация текста запроса



- 2) В разделе ПАРАМЕТРЫ ЗАПРОСА – задание значения параметра:



- значение параметра в консоли (раздел "Параметры запроса") задаем в два шага:

- 1) нажать "Выполнить" – подгружаются параметры,
- 2) задаем значения параметров – поле Значение

Шаталова О.М. Учетные системы

Запросы из нескольких таблиц

часто н. получать данные из нескольких таблиц,

чтобы совместить эти данные используются конструкции *Соединение, Объединение*.

Соединения - выполняется в одном запросе: сопоставляются (соединяются) строки таблиц по определенному признаку.

Объединение - выполняется двух разных запросов : результаты объединяются в один общий (Объединение «сливает» два результата и группирует повторяющие строки)

Соединение

- Запрос собирает данные из нескольких таблиц, соединяя данные из них в результирующей таблице
- Таблицы-источники данных указываются в запросе после ключевого слова **ИЗ**:
 - а) сначала указывается «первичная» таблица
 - б) затем, после задания **вида соединения**, указывается «вторичная» (присоединяемая) таблица
(виды соединения: Внутреннее, Полное, Левое / Правое, Полное, Перекрестное (декартово произведение) – задаются с использованием соответствующих ключевых слов)
- При этом обычно требуется связать данные таблиц по некоторому общему полю
- Соединение таблиц в запросах обычно производится по полю, имеющему ссылочный тип
- Для каждой записи в исходных таблицах производится проверка условия (на равенство, неравенство) значений полей, выступающих условием связи;
 - это условие задаётся в запросе после ключевого слова **ПО** в предложении **ИЗ**

ВЫБРАТЬ

(имя полей)

ИЗ

(1) – имя таблицы

ВНУТРЕННЕЕ СОЕДИНЕНИЕ / ЛЕВОЕ СОЕДИНЕНИЕ / ... (указание вида соединения)

(2) – имя таблицы

ПО

(условие соединения (= - во); исп. поля, имеющее ссылочный тип)

УПОРЯДОЧИТЬ ПО

(используется, если алгоритм обработки результатов запроса зависит от порядка записей в запросе или если результат обработки запроса в той или иной форме представляется пользователю)

Соединение : Виды

1) ВНУТРЕННЕЕ СОЕДИНЕНИЕ

только совпадающие строки (по заданному нами условию)

2) ЛЕВОЕ

вся табл.1 + совпадающие знач. из табл.2 (левое – табл.1, правое – табл.2)

3) ПРАВОЕ

вся табл.2 + совпадающие из табл.1

4) ПОЛНОЕ СОЕДИНЕНИЕ

вся табл.1 + вся табл.2

ПОЛНОЕ СОЕДИНЕНИЕ = ВНУТРЕННЕЕ СОЕДИНЕНИЕ + НЕДОСТАЮЩИЕ ЗАПИСИ ИЗ ПЕРВОЙ ТАБЛИЦЫ + НЕДОСТАЮЩИЕ ЗАПИСИ ИЗ ВТОРОЙ ТАБЛИЦЫ

5) ПЕРЕКРЕСТНОЕ

это декартово произведение; возникает, если не задано условие соединения либо в условии соединения указано: СОЕДИНЕНИЕ по ИСТИНА

(чаще так возникает по ошибке; т.к. практических применений не возникает, кроме нпр турнирная таблица)

Пример11:

Запрос данных из таблиц:

1) Справочник.НоменклатураТоваров

2) Регистр сведений «ЦеныФактические» (РЕГИСТР СВЕДЕНИЙ д.б. периодический)

Назначение запроса – соединение данных справочника и с данными о *последних* (актуальных) ценах

Запрос: Справочник.Номенклатура

Наименование	Товары	Цена
Кофейник		
Телевизор		
Утюг	Утюг	1 000
Чайник	Чайник	6 000

ПРИМЕР11(1) (ПОЛНОЕ СОЕДИНЕНИЕ)

ВЫБРАТЬ

СПРАВОЧНИК.НОМЕНКЛАТУРАТОВАРОВ.КОД,
 СПРАВОЧНИК.НОМЕНКЛАТУРАТОВАРОВ.НАИМЕНОВАНИЕ,
 РегистрСведений.ЦеныФактические.Товары,
 РегистрСведений.ЦеныФактические.Цена

ИЗ

СПРАВОЧНИК.НОМЕНКЛАТУРАТОВАРОВ
 ПОЛНОЕ СОЕДИНЕНИЕ
 РегистрСведений.ЦеныФактические

ПО

СПРАВОЧНИК.НОМЕНКЛАТУРАТОВАРОВ.ссылка= РегистрСведений.ЦеныФактические.Товары

Запрос: СПРАВОЧНИК.НОМЕНКЛА

Код	Наименование	Товары	Цена
1	Чайник	Чайник	10 000
1	Чайник	Чайник	10 000
1	Чайник	Чайник	6 000
2	Утюг	Утюг	1 000
3	Кофейник		
4	Телевизор		

Результат запроса – вывод ВСЕХ цен
 вывод пустых строк
 неупорядоченные строки

ИСПОЛЬЗУЕМ ВИРТУАЛЬНУЮ ТАБЛИЦУ **СрезПоследних** (для получения данных о последних ценах)

ИСПОЛЬЗУЕМ «УПОРЯДОЧИТЬ ПО»

ИСПОЛЬЗУЕМ Вид соединения – «Внутреннее СОЕДИНЕНИЕ»

Запрос: Справочник.Номенкла

Наименование	Товары	Цена
Утюг	Утюг	1 000
Чайник	Чайник	6 000

ПРИМЕР11(2) (СРЕЗ ПОСЛЕДНИХ, ВНУТРЕННЕЕ СОЕДИНЕНИЕ)

ВЫБРАТЬ

Справочник.НоменклатураТоваров.Наименование,
РегистрСведений.ЦеныФактические.СрезПоследних.Товары,
РегистрСведений.ЦеныФактические.СрезПоследних.Цена

ИЗ

Справочник.НоменклатураТоваров

ВНУТРЕННЕЕ СОЕДИНЕНИЕ

РегистрСведений.ЦеныФактические.СрезПоследних

ПО

Справочник.НоменклатураТоваров.Ссылка =
РегистрСведений.ЦеныФактические.СрезПоследних.Товары

УПОРЯДОЧИТЬ ПО

Справочник.НоменклатураТоваров.Наименование

ПРИМЕР11

ВЫБРАТЬ

Справочник.НоменклатураТоваров.Наименование,
РегистрСведений.ЦеныФактические.СрезПоследних.Товары,
РегистрСведений.ЦеныФактические.СрезПоследних.Цена

ИЗ

Справочник.НоменклатураТоваров

ПОЛНОЕ СОЕДИНЕНИЕ

РегистрСведений.ЦеныФактические.СрезПоследних

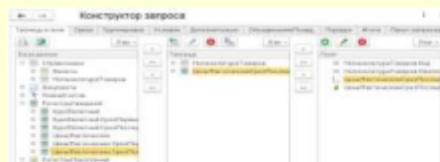
ПО

Справочник.НоменклатураТоваров.Ссылка =
РегистрСведений.ЦеныФактические.СрезПоследних.Товары

УПОРЯДОЧИТЬ ПО

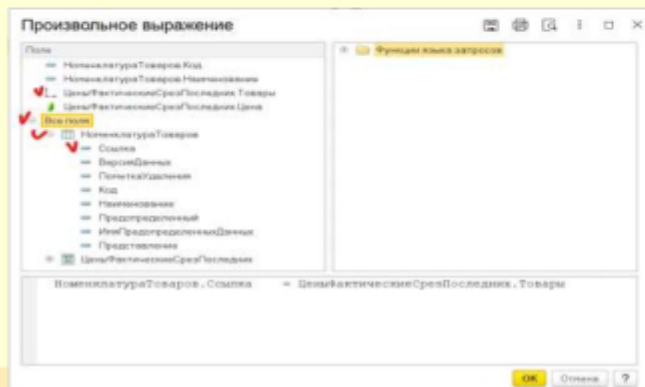
Справочник.НоменклатураТоваров.Наименование

- 1) вызов конструктора
- 2) Выбор таблиц и полей :



- 3) задаем Связи:

- а) устанавливаем последовательность таблиц: 1 – Спр.Номенклатура; 2 – Рег.ЦеныФакт. (перетаскиваем) ;
- б) делаем "подчиненность"
- в) устанавливаем вид соединения ЛЕВОЕ (прим.1)
- г) вв.условие соединения (вместо ИСТИНА) :



РЕЗУЛЬТАТ генерации кода запроса Конструктором ЗАПРОСОВ

-Присвоены псевдонимы (полям и таблицам)

-Установлен вид соединения

-Установлено условие соединения

Запрос: Справочник.Номенклат

Код	Наименование	Товары	Цена
1	Чайник	Чайник	3 000
2	Утюг	Утюг	1 000
3	Кофейник		
4	Телевизор		

ВЫБРАТЬ

НоменклатураТоваров.Код КАК Код,

НоменклатураТоваров.Наименование КАК Наименование,

ЦеныФактическиеСрезПоследних.Товары КАК Товары,

ЦеныФактическиеСрезПоследних.Цена КАК Цена

ИЗ

Справочник.НоменклатураТоваров КАК НоменклатураТоваров

ЛЕВОЕ СОЕДИНЕНИЕ РегистрСведений.ЦеныФактические.СрезПоследних КАК ЦеныФактическиеСрезПоследних

ПО (НоменклатураТоваров.Ссылка = ЦеныФактическиеСрезПоследних.Товары)

Исключим поле (дублирующее):

удаление строки кода : " ЦеныФактическиеСрезПоследних.Товары КАК Товары,"

2. Использование запроса в программном коде



ЗАПРОС – как объект программного кода:

- запрос позволяет получить определенную совокупность данных
- для использования этой совокупности данных в целях разработки конфигурации (реализации необходимой программной логики) требуется встроить текст запроса в программный код

Объект "Запрос" имеет свойство "Текст",
содержит строку с текстом запроса на языке запросов.

Этому свойству, во-первых, можно явно присвоить значение,
во-вторых, воспользоваться конструктором объекта "Запрос".

Результатом ЗАПРОСА становится виртуальный объект, который является «черным ящиком»,
Результат м.б. выдан в оперативную память а) построчно, б) в форме таблицы значений
т.е. в работе с ЗАПРОСАМИ используется Коллекция значений -- тип Структура (значения ключа –
строка.типа)

Обращение к элементам Структуры – через цикл **Для Каждого , Пока**

что такое запрос как объект в программном коде?

-это коллекция значений, к. состоит из текста запроса, параметров запроса, встроенных методов и др.:

текст запроса (Текст)	параметры	...	...
" ... (текст запроса) ..."	Структура со значениями параметров	...	...

текст запроса - тип Строковый

параметр_i - тип Структура, каждый элемент – тип строка

Процедура ПриЗаписи(Отказ)

```

ЭтотОбъект.Сумма=ЭтотОбъект.Цена*ЭтотОбъект.Количество;

Запрос = Новый Запрос;
Запрос.Текст =
    "ВЫБРАТЬ
    | КурсВалютныйСрезПоследних.Курс КАК Курс
    | ИЗ
    | РегистрСведений.КурсВалютный.СрезПоследних (&Период, ВидВалюты = &ВидВалюты) КАК КурсВалютныйСрезПоследних";

Запрос.УстановитьПараметр ("ВидВалюты", ЭтотОбъект.ВидВалюты);
Запрос.УстановитьПараметр ("Период", ЭтотОбъект.Дата);

РезультатЗапроса = Запрос.Выполнить();

// ===== случай построчного метода выдачи результата запроса

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
    ЭтотОбъект.СуммаВВалюте=ЭтотОбъект.Сумма / ВыборкаДетальныеЗаписи.Курс;

КонецЦикла;

КонецПроцедуры

```

1. инициализация объекта Запрос :	<code>Запрос = Новый Запрос;</code>
2. присваивание объекту Запрос значения по ключу Текст :	<code>Запрос.Текст = "ВЫВРАТЬ КурсВалютныйСрезПоследних.Курс КАК Курс ИЗ РегистрСведений.КурсВалютный.СрезПоследних (&Период, ВидВалюты = &ВидВалюты) КАК КурсВалютныйСрезПоследних"</code> (знак - означает перенос строки)
3. установка параметров запроса – исп. метод <code>УстановитьПараметр()</code>	<code>Запрос.УстановитьПараметр("ВидВалюты", ЭтотОбъект.ВидВалюты); Запрос.УстановитьПараметр("Период", ЭтотОбъект.Дата);</code> (в данном примере использовано два параметра; м.б. несколько параметров; м. не быть параметров) первый аргумент – наименование параметра ("ВидВалюты"), оно д.совпадать с наим.после & в тексте запроса; второй аргумент – присваивание значения этого параметра (в данном случае - по имени реквизита объекта);

4.

запрос запускается на выполнение с помощью метода "**Выполнить()**" объекта "Запрос". Этот метод возвращает другой объект "РезультатЗапроса", содержащий выбранные данные из базы данных (м.б. инициализирована с др. именем, которой присваивается результат запроса)

```
РезультатЗапроса = Запрос.Выполнить ();
```

Результатом ЗАПРОСА становится виртуальный объект, который является «черным ящиком», Результат м.б. выдан в оперативную память а) построчно, б) в форме таблицы значений, в) дерево, и др.

5.

выдача результата запроса

построчно:

-за выборку построчно отвечает метод **Выбрать()**;

-Цикл ПОКА:

обход результата – т.е. строк в составе

ВыборкаДетальныеЗаписи

-Использование результата запроса в алгоритме обработки

```
ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать ();
Пока ВыборкаДетальныеЗаписи.Следующий ()
    Цикл
        ЭтотОбъект.СуммаВВалюте=ЭтотОбъект.Сумма/ВыборкаДетальныеЗаписи.Курс;
    КонечЦикла;
```

(пример выдачи рез-та запроса в форме **таблицы значений**:

```
TS=Новый ТаблицаЗначений;
TS=РезультатЗапроса.Выгрузить ();
Для Каждого Строка из TS
    Цикл ЭтотОбъект.СуммаВВалюте=ЭтотОбъект.Сумма / Строка.Курс;
    КонечЦикла;
```

- инициац.нов.Табл.Знач.;
- м-д Выгрузить();
- обход – в цикле Для го

Задание:

1) используем объекты:

Документ.Закупки,

РегистрСведений.КурсВалютный

2) в Документ.Закупки организовать расчет суммы в валюте (с использованием Конструктора запросов – из модуля объекта)

Общая схема выполнения запроса :

1. Создание объекта "Запрос" с нужным текстом запроса на специальном языке запросов.
2. Установка параметров запроса с помощью метода УстановитьПараметр.
3. Выполнение запроса, получение результата.
4. Получение выборки из результата запроса или выгрузка результата в таблицу значений / дерево значений. Также есть возможность использовать результат запроса для формирования сводной таблицы.
5. Обработка выборки или таблицы значений (например, перебор строк) и выполнение действий, для которых был нужен запрос, например, вывод области при формировании отчета



<https://its.1c.ru/db/metod8dev/content/2582/hdoc>

Организовать:

1) ввод данных о цене реализации товара –
независимый)

2) заполнение табличной части расходной н

Наименование

Количество

Цена реализации (заполнение – посредством

Сумма

