

#T5

Универсальные коллекции: ТаблицаЗначений и ДеревоЗначений

Программист 1С: с нуля до профи



Коллекции

примитивный тип

данных



`A = 5;`



одно значение



коллекция



`A = Новый <ТипКоллекции>;`



много значений

<КлючЭлемента1>	5
<КлючЭлемента2>	«красный»
<КлючЭлемента3>	01.02.1991
...	...
<КлючЭлементаN>	12

Коллекции

ключ типа
«Число»



0	5
1	6
2	7
...	...
N-1	114

ключ типа
«Строка»



«ПервоеЧисло»	5
«ВтороеЧисло»	6
«ТретьеЧисло»	7
...	...
«ПоследнееЧисло»	114

ключ типа
«Дата»



01.05.2020	5
01.06.2020	6
01.07.2020	7
...	...
31.12.2020	114

Как получить первое значение из коллекции?

A[0]

A[«ПервоеЧисло»]

A['20200501']

Универсальные коллекции

- 1 Массив
- 2 Структура
- 3 Соответствие
- 4 Список значений
- 5 Таблица значений
- 6 Дерево значений

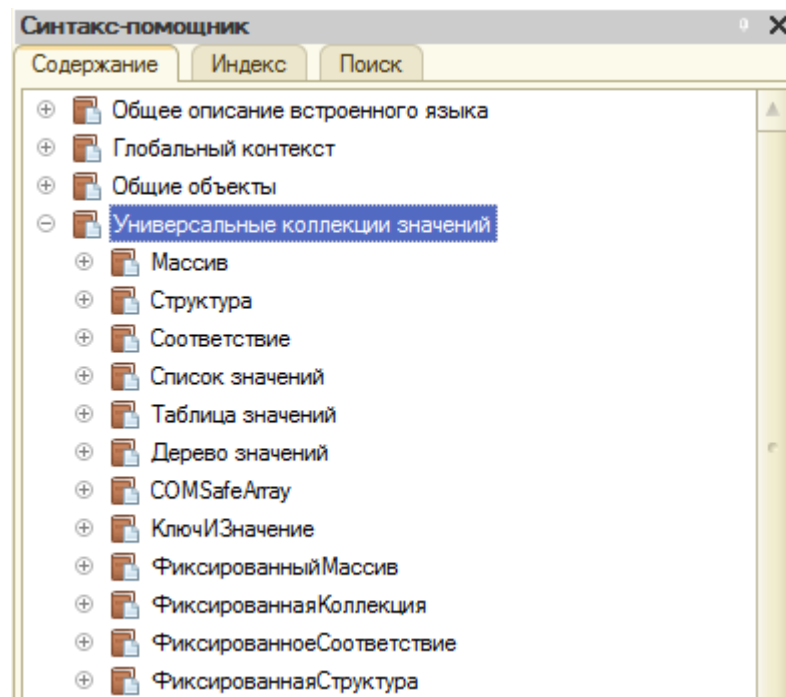


Таблица значений

Номенклатура	Склад	В наличии	В резерве	Остаток
Слива "Овация", Не годен	Торговый зал	2,000		2,000
Вентилятор JIPONIC, напольный	Магазин "Бытовая техника"	3,000		3,000
Капуста белокачаная	Магазин "Продукты"	2 000,000		2 030,000
Капуста белокачаная	Торговый зал	949,140		998,920
Капуста белокачаная	Оптовый склад	980,100		980,100
Хлеб "Водар"	Магазин "Продукты"	20,000		20,000
Хлеб "Балтийский" нарезной	Магазин "Продукты"	20,000		20,000
Кондиционер для белья	Оптовый склад	4,000		4,000
Порошок стиральный	Оптовый склад	16,000		16,000
Стол кухонный "Стандарт Плюс"	Мебельный склад	10,000		10,000
Стул "Премьер"	Мебельный склад	7,000		7,000
A-21212 ARISTON однокамерный, авторазморозка Италия	Оптовый склад	2,000	2,000	
Мойка металлическая	Мебельный склад	12,000		12,000
Мишка (конфеты)	Мой магазин			60,000
Мойка мраморная	Мебельный склад	11,000		11,000
Мебельный гарнитур "Романтика"	Оптовый склад	1,000	1,000	
Кабель NYM (Витебскабель) 1x4	Оптовый склад	19 820,000	5 110,000	14 710,000

Таблица значений

ЧИСЛОВОЙ



Индекс	ИмяКолонки1	ИмяКолонки2	...	ИмяКолонкиM
0	«116216»	«Погремушка»
1	«Вт»	«Вторник»
2	«Ср»	«Среда»
...
N - 1	«Значение[N-1, 1]»	«Значение[N-1, 2]»	...	«Значение[N-1, M]»

N – количество строк в таблице

M – количество колонок в таблице

Доступность типов

Обработка УниверсальныеКоллекции: Форма

```
&НаКлиенте
☒ Процедура РаботаСМассивом(Команда) ...

&НаКлиенте
☒ Процедура РаботаСоСтруктурой(Команда) ...

&НаКлиенте
☒ Процедура РаботаССоответствием(Команда) ...

&НаКлиенте
☒ Процедура РаботаСоСпискомЗначений(Команда) ...

&НаКлиенте ←
☐ Процедура РаботаСТаблицейЗначений(Команда)
    ТаблицаЗначений = Новый ТаблицаЗначений;
КонецПроцедуры
```

Служебные сообщения

```
*** {Обработка.УниверсальныеКоллекции.Форма.Форма.Форма(326,26)}: Тип не определен (ТаблицаЗначений)
    ТаблицаЗначений = Новый <<?>>ТаблицаЗначений; (Проверка: Тонкий клиент)
```

Доступность типов

Массив (Array)

Элементы коллекции:

Произвольный

Для объекта доступен обход коллекции посредством операции `foreach`. При обходе выбираются значения элементов. Возможно обращение к значению элемента посредством индекса. В качестве аргумента передается индекс значения (номер элемента).

Методы:

[ВГраница \(UBound\)](#)

[Вставить \(Insert\)](#)

[Добавить \(Add\)](#)

[Количество \(Count\)](#)

[Найти \(Find\)](#)

[Очистить \(Clear\)](#)

[Получить \(Get\)](#)

[Удалить \(Delete\)](#)

[Установить \(Set\)](#)

Конструкторы:

[На основании фиксированного массива](#)

[По количеству элементов](#)

Описание:

Предназначен для доступа к элементам массива, его методы...

Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, внешнее соединение, мобильное приложение (клиент), мобильное приложение (сервер), мобильный автономный сервер.

Возможен обмен с сервером. Сериализуется. Данный объект может быть сериализован в/из XDTO. Тип XDTO, соответствующий данному объекту, определяется в пространстве имен `{http://schemas.datacontract.org/2004/07/}`. Имя типа

Структура (Structure)

Элементы коллекции:

[КлючИЗначение](#)

Для объекта доступен обход коллекции посредством операции `foreach`. При обходе выбираются элементы. ... Из ... Цикл. При обходе выбираются элементы.

Свойства:

[Имя ключа \(<Key name>\)](#)

Методы:

[Вставить \(Insert\)](#)

[Количество \(Count\)](#)

[Очистить \(Clear\)](#)

[Свойство \(Property\)](#)

[Удалить \(Delete\)](#)

Конструкторы:

[На основании фиксированной структуры](#)

[По ключам и значениям](#)

Описание:

Представляет собой коллекцию пар [КлючИЗначение](#). Ключ должен быть только строковым и должен удовлетворять требованиям к именованию переменных. К значениям структуры можно обращаться как к свойствам. Ключ используется как имя свойства.

Структура используется обычно для хранения значений, каждое из которых имеет некоторую структуру.

Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, внешнее соединение, мобильное приложение (клиент), мобильное приложение (сервер), мобильный автономный сервер.

ДиалогВыбораФайла (FileDialog)

Свойства:

[ВыбранныеФайлы \(SelectedFiles\)](#)

[Заголовок \(Title\)](#)

[ИндексФильтра \(FilterIndex\)](#)

[Каталог \(Directory\)](#)

[МножественныйВыбор \(Multiselect\)](#)

[ПолноеИмяФайла \(FullFileName\)](#)

[ПредварительныйПросмотр \(Preview\)](#)

[ПроверитьСуществованиеФайла \(CheckFileExistence\)](#)

[Расширение \(DefaultExt\)](#)

[Режим \(Mode\)](#)

[Фильтр \(Filter\)](#)

Методы:

[Выбрать \(Choose\)](#)

[ВыбратьАсинх \(ChooseAsync\)](#)

[Показать \(Show\)](#)

Конструкторы:

[С заданным режимом работы](#)

Описание:

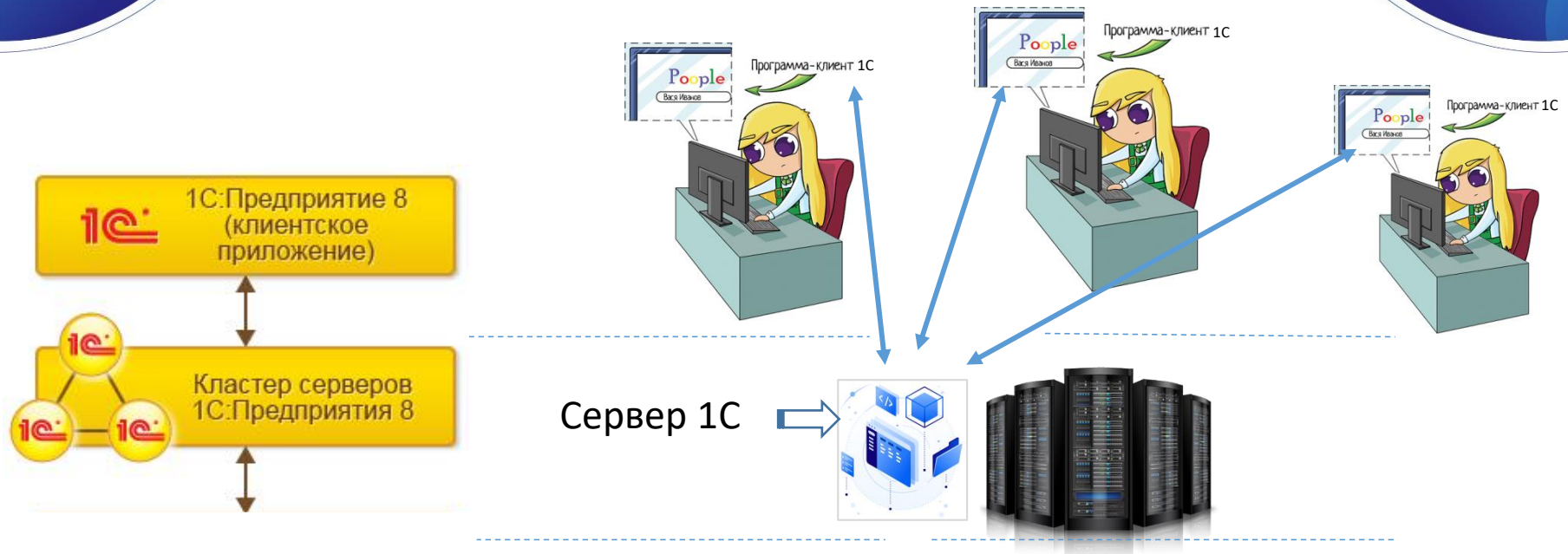
Предназначен для работы с диалогом специального вида для открытия файла или нескольких файлов, сохранения файла или выбора каталога.

В веб-клиенте, если расширение работы с файлами не установлено, может быть использован только для установки заголовка диалога или множественного выбора в методах [НачатьПомещениеФайла](#) и [НачатьПомещениеФайлов](#).

Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, толстый клиент, мобильное приложение (клиент).

«Клиент» и «Сервер»



• • •

• • •

Директивы компиляции

Директива компиляции – специальная конструкция, которая указывается перед описанием процедуры (функции) и определяет среду исполнения данной процедуры (функции)

- В частности – в какой части приложения (клиентской или серверной) будет выполняться код, описанный внутри процедуры (функции)

```
&НаКлиенте ←
□ Процедура РаботаСТаблицейЗначений (Команда)
    ПроцедураНаСервере ();
КонецПроцедуры

&НаСервере ←
□ Процедура ПроцедураНаСервере ()
    //...
КонецПроцедуры
```

Как понимать «Доступность»

Описание:
Таблица значений предназначена для хранения значений в табличном виде. Все основные операции с таблицей производятся именно через этот объект. Он позволяет манипулировать строками таблицы значений и предоставляет доступ к коллекции колонок. Колонки могут быть разных типов (в том числе множественных).

Доступность:
Сервер, толстый клиент, внешнее соединение, мобильное приложение (сервер), мобильный автономный сервер.
Возможен обмен с сервером. Сериализуется. Данный объект может быть сериализован в/из XDTO. Тип XDTO, соответствующий данному объекту, определяется в пространстве имен {http://v8.1c.ru/8.1/data/core}. Имя типа XDTO: ValueTable.

нельзя

можно

```
&НаКлиенте
- Процедура ПроцедураНаКлиенте (Команда)
  //...
- КонечПроцедуры

&НаСервере
- Процедура ПроцедураНаСервере ()
  //...
- КонечПроцедуры
```

Описание:
Предназначен для работы с диалогом специального вида для открытия файла или нескольких файлов, сохранения файла или выбора каталога. В веб-клиенте, если расширение работы с файлами не установлено, может быть использован только для установки заголовка диалога или множественного выбора в методах [НачатьПомещениеФайла](#) и [НачатьПомещениеФайлов](#).

Доступность:
Тонкий клиент, веб-клиент, мобильный клиент, толстый клиент, мобильное приложение (клиент).

можно

нельзя

```
&НаКлиенте
- Процедура ПроцедураНаКлиенте (Команда)
  //...
- КонечПроцедуры

&НаСервере
- Процедура ПроцедураНаСервере ()
  //...
- КонечПроцедуры
```

Как понимать «Доступность»

Массив (Array)

Элементы коллекции:

Произвольный

Для объекта доступен обход коллекции посредством оператора Для каждого ... Из ... Цикл. При обходе выбираются значения элементов массива.

Возможно обращение к значению элемента посредством оператора [...]. В качестве аргумента передается индекс значения (нумерация с 0).

Методы:

[ВГраница \(UBound\)](#)

[Вставить \(Insert\)](#)

[Добавить \(Add\)](#)

[Количество \(Count\)](#)

[Найти \(Find\)](#)

[Очистить \(Clear\)](#)

[Получить \(Get\)](#)

[Удалить \(Delete\)](#)

[Установить \(Set\)](#)

Конструкторы:

[На основании фиксированного массива](#)

[По количеству элементов](#)

Описание:

Предназначен для доступа к элементам массива, его методам и конструктору.

Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение (клиент), мобильное приложение (сервер), мобильный автономный сервер.

Возможен обмен с сервером. Сериализуется. Данный объект может быть сериализован в/из XDTO. Тип XDTO, соответствующий данному объекту, определяется в

&НаКлиенте

Процедура ПроцедураНаКлиенте (Команда)

//...

КонецПроцедуры

&НаСервере

Процедура ПроцедураНаСервере ()

//...

КонецПроцедуры

МОЖНО

МОЖНО

Таблица значений

ЧИСЛОВОЙ



Индекс	ИмяКолонки1	ИмяКолонки2	...	ИмяКолонкиМ
0	«116216»	«Погремушка»
1	«Вт»	«Вторник»
2	«Ср»	«Среда»
...
N - 1	«Значение[N-1, 1]»	«Значение[N-1, 2]»	...	«Значение[N-1, M]»

N – количество строк в таблице

M – количество колонок в таблице

Таблица значений

Номенклатура	Склад	В наличии	В резерве	Остаток
Слива "Овация", Не годен	Торговый зал	2,000		2,000
Вентилятор JIPONIC, напольный	Магазин "Бытовая техника"	3,000		3,000
Капуста белокачаная	Магазин "Продукты"	2 000,000		2 030,000
Капуста белокачаная	Торговый зал	949,140		998,920
Капуста белокачаная	Оптовый склад	980,100		980,100
Хлеб "Водар"	Магазин "Продукты"	20,000		20,000
Хлеб "Балтийский" нарезной	Магазин "Продукты"	20,000		20,000
Кондиционер для белья	Оптовый склад	4,000		4,000
Порошок стиральный	Оптовый склад	16,000		16,000
Стол кухонный "Стандарт Плюс"	Мебельный склад	10,000		10,000
Стул "Премьер"	Мебельный склад	7,000		7,000

Таблица значений. Основные операции

- Как **создать** таблицу значений
- Как **добавить колонки** в таблицу значений
- Как **добавить** элементы (строки) в таблицу значений
- Как получить **количество элементов** в таблице значений
- Как **найти** один элемент в таблице значений
- Как **найти все элементы** таблицы значений, которые удовлетворяют условию поиска
- Как **создать копию** таблицы значений (**с данными**)
- Как **создать копию** таблицы значений (**без данных**)
- Как **удалить** элемент таблицы значений
- Как **очистить** таблицу значений
- Как выполнить **сортировку** в таблице значений
- Как **получить итог** по колонке числового типа
- **Обход** всех элементов (строк) таблицы значений

Таблица значений. Создание

Этапы создания:

1. Создание коллекции ТаблицаЗначений

```
ТаблицаТоваров = Новый ТаблицаЗначений;
```

2. Добавление колонок

```
ТаблицаТоваров.Колонки.Добавить ("Номенклатура");  
ТаблицаТоваров.Колонки.Добавить ("Склад");  
ТаблицаТоваров.Колонки.Добавить ("ВНаличии");  
ТаблицаТоваров.Колонки.Добавить ("ВРезерве");  
ТаблицаТоваров.Колонки.Добавить ("Остаток");
```

3. Добавление элементов (строк)

```
СтрокаТаблицы = ТаблицаТоваров.Добавить ();  
СтрокаТаблицы.Номенклатура = "Капуста белокачаная";  
СтрокаТаблицы.Склад = "Магазин ""Продукты""";  
СтрокаТаблицы.ВНаличии = 2000;  
СтрокаТаблицы.ВРезерве = 0;  
СтрокаТаблицы.Остаток = 2000;
```

Таблица значений. Поиск

Метод Найти():

- Подходит, если нужно найти одну строку в таблице по значению в одной из колонок

Пример:

Найти строку с товаром «*Капуста белокачанная*»

`НайденнаяСтрока = ТаблицаТоваров.Найти("Капуста белокачанная", "Номенклатура");`

Номенклатура	Склад	В наличии	В резерве	Остаток
Слива "Овация", Не годен	Торговый зал	2,000		2,000
Вентилятор JIPONIC, напольный	Магазин "Бытовая техника"	3,000		3,000
Капуста белокачанная	Магазин "Продукты"	2 000,000		2 030,000
Хлеб "Водар"	Магазин "Продукты"	20,000		20,000

Таблица значений. Поиск

Метод НайтиСтроки():

- Подходит, если нужно найти несколько строк в таблице по комбинации значений в нескольких колонках

Пример:

Найти все строки с товаром «Капуста белокачанная»

СтруктураПоиска = Новый Структура ("Номенклатура", "Капуста белокачанная");
МассивНайденныхСтрок = ТаблицаТоваров.НайтиСтроки(СтруктураПоиска);

Номенклатура	Склад	В наличии	В резерве	Остаток
Слива "Овация", Не годен	Торговый зал	2,000		2,000
Вентилятор JIPONIC, напольный	Магазин "Бытовая техника"	3,000		3,000
Капуста белокачаная	Магазин "Продукты"	2 000,000		2 030,000
Капуста белокачаная	Торговый зал	949,140		998,920
Капуста белокачаная	Оптовый склад	980,100		980,100



Таблица значений. Метод «Свернуть()»

Колонки группировки

Колонка
суммирования

Контрагент	Договор	Сумма
Рога и копыта	Д11	100
Рога и копыта	Д12	150
Федоров и КО	Д21	99
Федоров и КО	Д22	100



Контрагент	Сумма
Рога и копыта	250
Федоров и КО	199

Таблица значений
ДО свертки

Таблица значений
ПОСЛЕ свертки

Дерево значений

Склад	В наличии	В резерве	Остаток
Номенклатура			
[-] Арендванный склад Вега-Транс			
Ботинки женские демисезонные	6,000		6,000
Вентилятор BINATONE ALPINE 160вт, напольный, оконный	3,000		3,000
Вентилятор оконный, модель 902	1,000		1,000
Вентилятор JIPONIC, напольный	1,000		1,000
S-545454 Samsung двухкамерный, авторазморозка РОССИЯ	18,000		18,000
S-3243234 Samsung однокамерный, авторазморозка РОССИЯ	3,000		3,000
Босоножки женские	2,000		2,000
[-] Ларек "Розница"			
Мука гречневая, 900 г.	70,000		170,000
Кофеварка BRAUN KF22R	1,000		1,000
[-] Магазин "Бытовая техника"	159,000	2,000	177,000
Комбайн кухонный BINATONE FP 67	9,000		9,000
Мясорубка MOULINEX A 15	20,000	1,000	19,000
Пылесос "SAMSUNG"	1,000		2,000
Пылесос "Энергия-SANYO"	1,000		1,000
Чайник MOULINEX L 1,3	7,000		7,000
Вентилятор ОРБИТА STERLING	4,000		4,000
Вентилятор BINATONE ALPINE 160вт, напольный, оконный	4,000		4,000
Вентилятор настольный, Модель 901	2,000		2,000
Вентилятор оконный, модель 902	2,000		2,000
Вентилятор JIPONIC, напольный	3,000		3,000
Чайник BINATONE EWK-3000, 2л	17,000		17,000

Дерево значений

Номенклатура

- Мониторы
- Системные блоки и комплектующие
- Принтеры
 - Лазерный принтер Canon LBP-810
 - Лазерный принтер 5250197-203 Minolta-QMS
 - Лазерный принтер HP LaserJet 2200
- Мыши
- Клавиатуры
- Программное обеспечение
 - Windows
 - 1С:Бухгалтерия 7.7 Базовая версия
 - 1С:Бухгалтерия 7.7 Стандартная версия
 - 1С:Бухгалтерия ПРОФ версия 7.7
 - 1С:Аспект 7.7
 - 1С:Торговля и Склад 7.7 Проф
- Услуги
- Ноутбуки
- Телефоны
- Копировальные аппараты

Склад	Номенклатура	КоличествоОст...	Факт. остаток
Основной склад		530	34,00
Основной склад	Мониторы	41	16,00
Основной склад	Монитор 15" LG Studioworks 575N	12	
Основной склад	Монитор 17" Philips 107S20	6	6,00
Основной склад	Монитор 19" Hitachi CM715ET	17	10,00
Основной склад	Монитор LCD 22" M8537ZM/A	6	
Основной склад	Системные блоки и комплектующие	48	18,00
Основной склад	Сист. блок Hewlett-Packard Brio BA410	16	
Основной склад	Сист. блок Hewlett-Packard Vectra VL420	8	8,00
Основной склад	Сист. блок IBM NetVista A22p	13	10,00
Основной склад	Сист. блок IBM NetVista M41	11	
Основной склад	Принтеры	26	
Основной склад	Лазерный принтер Canon LBP-810	4	
Основной склад	Лазерный принтер 5250197-203 Minolta-QMS	12	
Основной склад	Лазерный принтер HP LaserJet 2200	10	
Основной склад	Мыши	191	
Основной склад	Мышь 2-кноп. A4Tech PS/2	53	

Дерево значений

КАКОЙ СТРОКЕ
ПОДЧИНЕНА
ТЕКУЩАЯ
ЧИСЛОВОЙ



Индекс	Родитель	ИмяКолонки1	ИмяКолонки2	...	ИмяКолонкиM
0		«Квартал1»	
1	0	«Январь»	31
2	0	«Февраль»	28
...	
N - 1	18	«Значение[N-1, 1]»	«Значение[N-1, 2]»	...	«Значение[N-1, M]»

N – количество строк в таблице

M – количество колонок в таблице

Дерево значений

Склад	В наличии	В резерве	Остаток
Номенклатура			
Арендованный склад Вега-Транс	34,000		34,000
Ботинки женские демисезонные	6,000		6,000
Вентилятор BINATONE ALPINE 160вт, напольный, оконный	3,000		3,000
Вентилятор оконный, модель 902	1,000		1,000
Вентилятор JIPONIC, напольный	1,000		1,000
S-545454 Samsung двухкамерный, авторазморозка РОССИЯ	18,000		18,000
S-3243234 Samsung однокамерный, авторазморозка РОССИЯ	3,000		3,000
Босоножки женские	2,000		2,000
Ларек "Розница"	171,000		171,000
Мука гречневая, 900 г.	70,000		170,000
Кофеварка BRAUN KF22R	1,000		1,000
Магазин "Бытовая техника"	159,000	2,000	177,000
Комбайн кухонный BINATONE FP 67	9,000		9,000
Мясорубка MOULINEX A 15	20,000	1,000	19,000

Дерево значений.

Основные операции

- Как **создать** дерево значений
- Как **добавить колонки** в дерево значений
- Как **добавить** элементы (строки) в дерево значений
- Как получить **количество элементов** в дереве значений
- Как **найти** один элемент в дереве значений
- Как **найти все элементы** дерева значений, которые удовлетворяют условию поиска
- Как **создать копию** дерева значений (**с данными**)
- Как **удалить** элемент дерева значений
- Как **очистить** дерево значений
- Как выполнить **сортировку** в дереве значений
- **Обход** всех элементов дерева значений

Дерево значений. Создание

Этапы создания:

1. Создание коллекции Деревозначений

```
ДеревоТоваров = Новый Деревозначений;
```

2. Добавление колонок

```
ДеревоТоваров.Колонки.Добавить ("Склад");  
ДеревоТоваров.Колонки.Добавить ("Номенклатура");  
ДеревоТоваров.Колонки.Добавить ("ВНаличии");  
ДеревоТоваров.Колонки.Добавить ("ВРезерве");  
ДеревоТоваров.Колонки.Добавить ("Остаток");
```

3. Добавление элементов (строк)

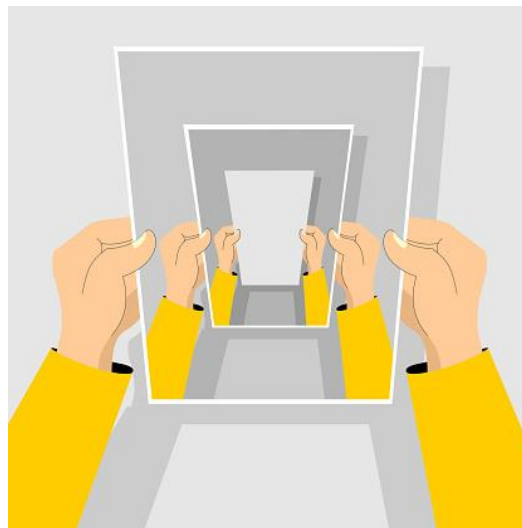
```
//добавление строки верхнего уровня  
СтрокаДереваСклад = ДеревоТоваров.Строки.Добавить();  
СтрокаДереваСклад.Склад = "Магазин ""Продукты""";  
СтрокаДереваСклад.ВНаличии = 34;  
СтрокаДереваСклад.ВРезерве = 0;  
СтрокаДереваСклад.Остаток = 34;  
  
//добавление строки 1-го уровня (подчиненной строки)  
СтрокаДереваТовар = СтрокаДереваСклад.Строки.Добавить();  
СтрокаДереваТовар.Склад = "Магазин ""Продукты""";  
СтрокаДереваТовар.Номенклатура = "Капуста белокачанная";  
СтрокаДереваТовар.ВНаличии = 34;  
СтрокаДереваТовар.ВРезерве = 0;  
СтрокаДереваТовар.Остаток = 34;
```

Дерево значений. Рекурсия

Рекурсия - вычислительный процесс, направленный на решение определенной задачи таким образом, что само решение использует этот же процесс, решающий аналогичную подзадачу.

Простыми словами:

Рекурсия – это функция, которая сама вызывает себя



Дерево значений. Рекурсия

□ Процедура ОсновнаяПроцедура ()

```
Сумма = 0;
```

```
Число = 1;
```

```
УвеличитьСуммуНаЧисло (Сумма, Число);
```

```
Сообщить (Сумма);
```

└─ КонечПроцедуры

□ Процедура УвеличитьСуммуНаЧисло (Сумма, Число)

```
Если Сумма >= 1000 Тогда
```

```
    Возврат;
```

```
КонецЕсли;
```

```
Сумма = Сумма + Число;
```

```
УвеличитьСуммуНаЧисло (Сумма, Число);
```

└─ КонечПроцедуры

← Рекурсивная процедура

Конец 5-го занятия

Подписывайтесь на нас в социальных сетях 😊



<https://www.youtube.com/channel/UC124tAuP11GlaGpK3YGHzw>



<https://www.facebook.com/ironskillsby/>



https://www.instagram.com/ironskills_/