

Справочники и документы как основа учетных задач

ЦЕАБ
МНОГОПРОФИЛЬНЫЙ
ЦЕНТР КВАЛИФИКАЦИИ
В СОСТАВЕ **SDG**

Справочники – что это и с чем едят

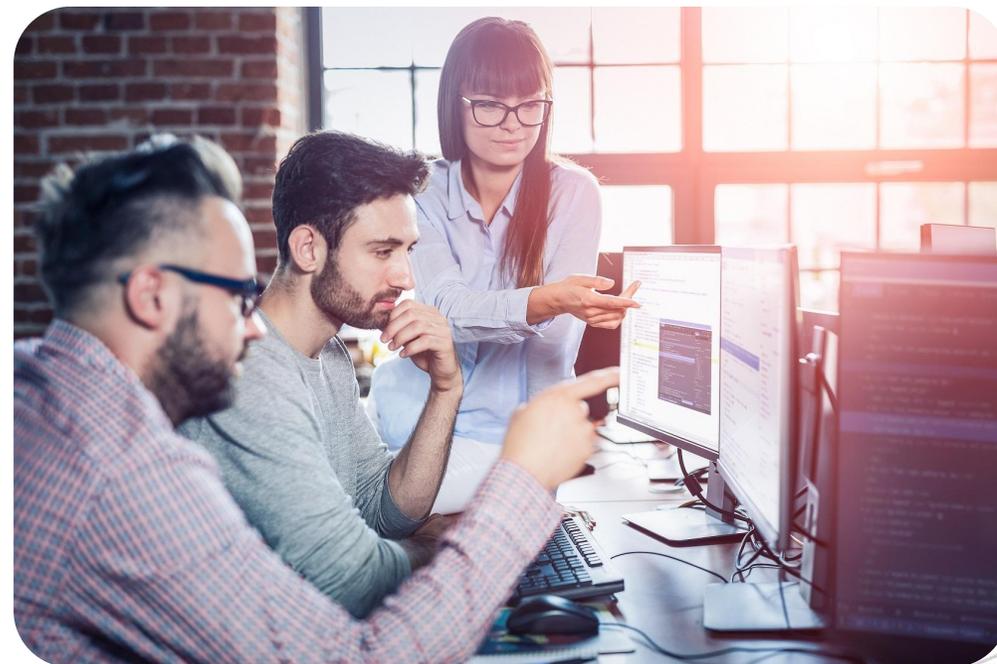


С теоретической точки зрения, **Справочники** – это прикладные объекты конфигурации



Они позволяют хранить в информационной базе данные, имеющие одинаковую структуру и списочный характер

Это может быть, например, список сотрудников, перечень товаров, список поставщиков или покупателей



Структура справочника

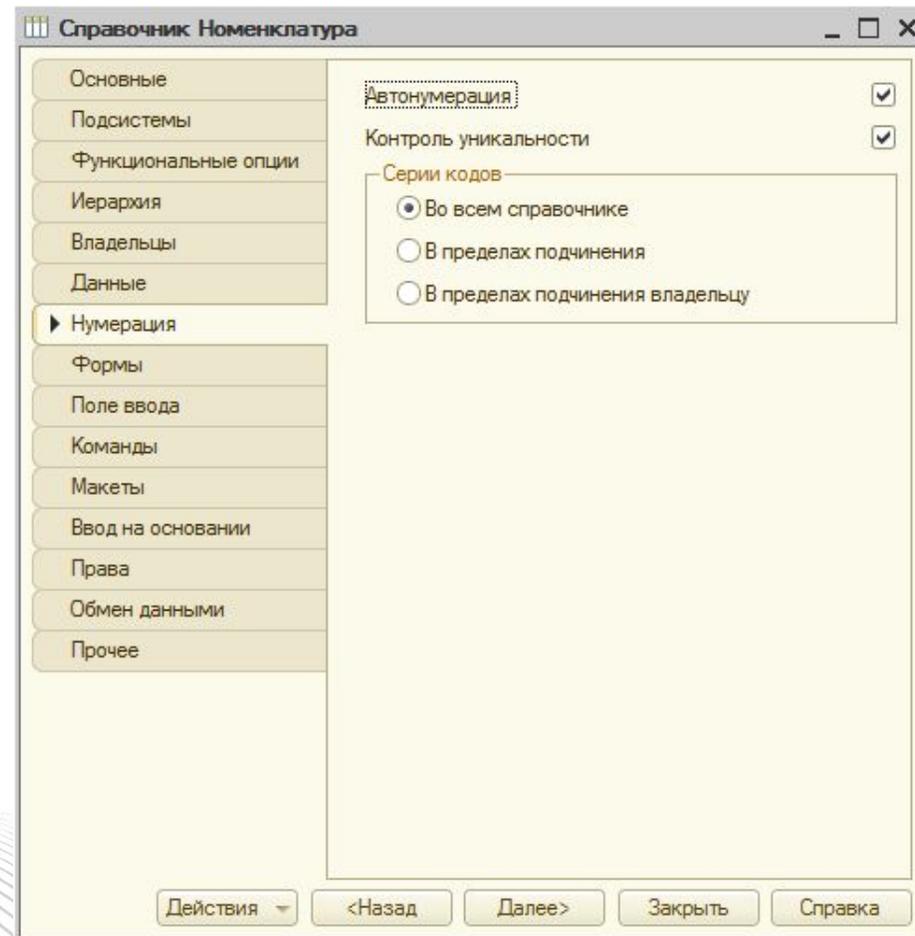


Каждый элемент справочника характеризуется **кодом** и **наименованием**



Система поддерживает режим автоматической нумерации элементов, при котором она самостоятельно может генерировать код для нового элемента справочника

Например, для товара это может быть информация об артикуле, упаковке и т. п.



Реквизиты справочника

Поле базы данных, характеризующее все элементы данных некоторого **объекта конфигурации**.

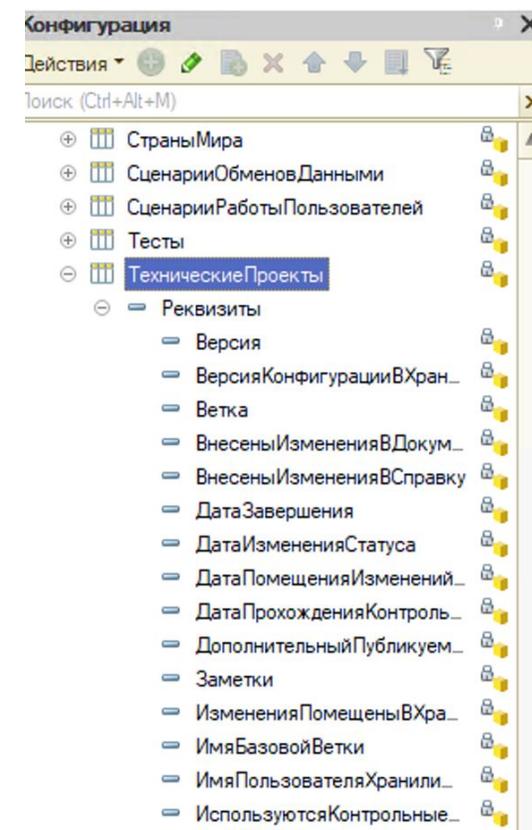
Например, **объект конфигурации Справочник**.

Товары может иметь реквизит **Производитель**. Это означает, что для каждого товара из этого **справочника** можно указать производителя.

Реквизиты могут создаваться разработчиком. Могут

существовать **стандартные реквизиты**, созданные **платформой**, например, **Наименование**, **Код**, **ПометкаУдаления** и другие.

При работе с данными из **встроенного языка** реквизиты **объектов конфигурации** доступны как свойства **объектов встроенного языка**.



Табличные части справочника

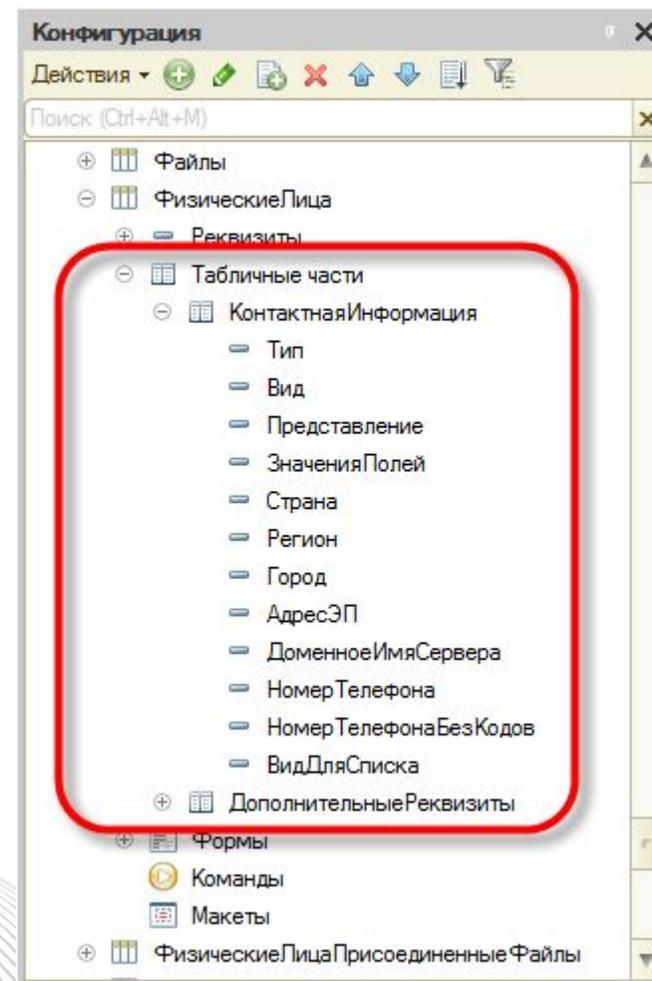


Кроме этого, каждый элемент справочника может содержать некоторый набор информации, которая одинакова по своей структуре, но различна по количеству, для разных элементов справочника

Например, для каждого сотрудника в справочнике **Физические лица** это может быть контактная информация или информация о составе семьи, образовании



Для хранения подобных данных служат табличные части справочника



Иерархия справочников



Иерархия – это возможность связать между собой записи, находящиеся в рамках одной таблицы.



Справочники могут поддерживать иерархическое расположение элементов

Например, в справочнике Номенклатура могут быть созданы группы: Бытовая техника, Обувь, Продукты и т. д., в которых располагаются элементы, относящиеся к этим группам.



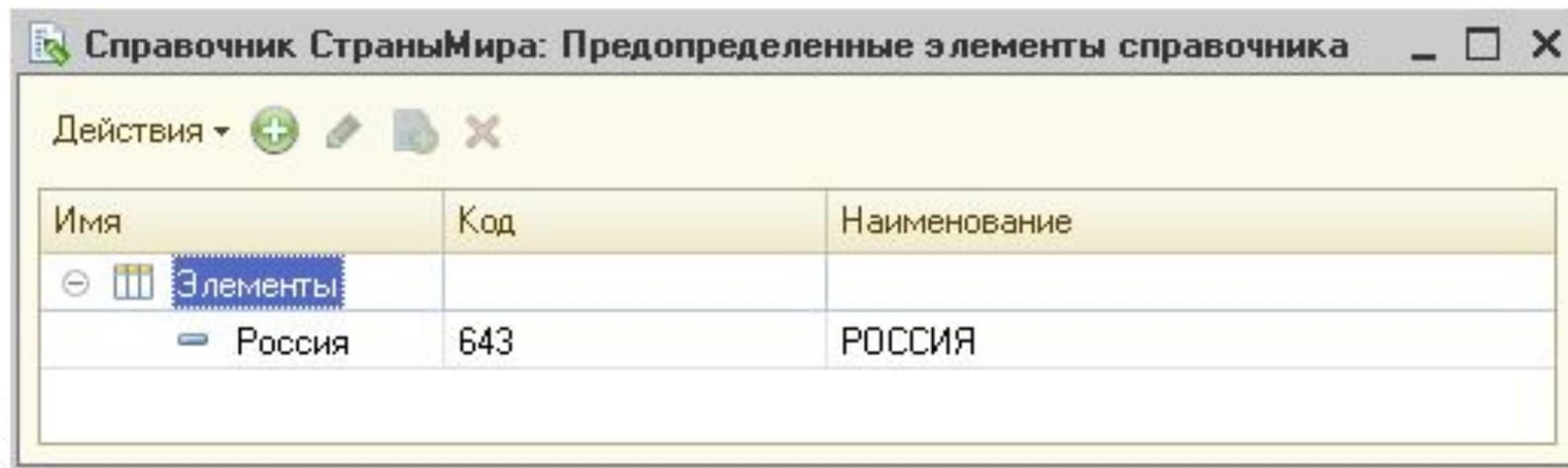
Кроме того группы справочника могут включать в себя и другие группы, создавая тем самым многоуровневую иерархическую структуру.



Предопределенные элементы

Такие элементы справочника создаются разработчиком при разработке прикладного решения и не могут быть удалены или перемещены пользователем

Например, если в основном прикладное решение будет внедряться на предприятиях одного государства, России, в справочник Страны мира сразу же можно добавить предопределенный элемент Россия, поскольку он понадобится подавляющему большинству пользователей



Имя	Код	Наименование
Элементы		
Россия	643	РОССИЯ

Объектная модель

Работаем как с объектом, у которого есть:
данные элемента, стандартные методы,
не стандартные методы, ссылка

```
НовыйЭлемент = Справочники.Контрагенты.СоздатьЭлемент(); // Созда  
ние элемента  
НовыйЭлемент.Родитель = Справочники.Контрагенты.НайтиПоНаименован  
ию("Покупатели"); // Заполнение родителя  
НовыйЭлемент.Наименование = "Ромашка 000"; // Заполнение наименов  
ания  
НовыйЭлемент.ИНН = "123456789"; // Заполнение реквизита
```

Табличная модель

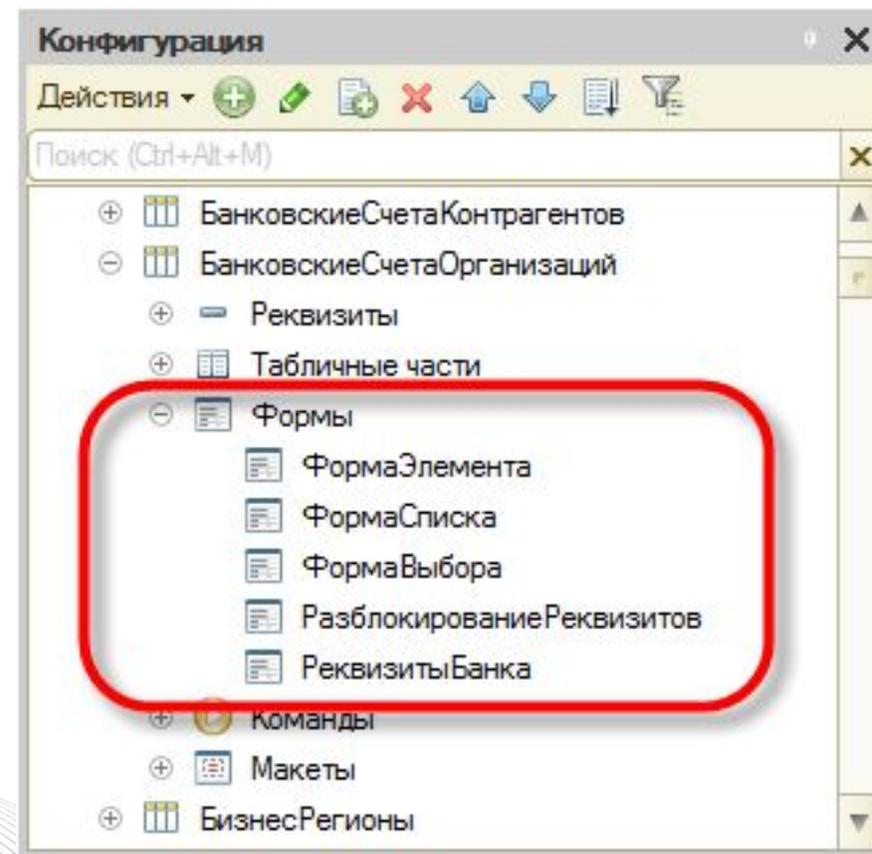
Все данные в 1С хранятся в базе данных.
База данных – набор таблиц.
Соответственно «физически» объект 1С
раскладывается на набор строк в этих
таблицах

```
ТекущийЭлементНоменклатуры = ЭлементНоменклатуры;  
Запрос = Новый Запрос("ВЫБРАТЬ  
| Номенклатура.Родитель,  
| Номенклатура.Родитель.Родитель,  
| Номенклатура.Родитель.Родитель.Родитель,  
| Номенклатура.Родитель.Родитель.Родитель.Родитель,  
| Номенклатура.Родитель.Родитель.Родитель.Родитель.Родитель  
| ИЗ  
| Справочник.Номенклатура КАК Номенклатура  
| ГДЕ  
| Номенклатура.Ссылка = «ТекущийЭлементНоменклатуры»;  
Пока Истина Цикл  
Запрос.УстановитьПараметр("ТекущийЭлементНоменклатуры", ТекущийЭлементНоменклатуры);  
Результат = Запрос.Выполнить();  
Если Результат.Пустой() Тогда  
Прервать;  
КонецЕсли;  
Выборка = Результат.Выбрать();  
Выборка.Следующий();  
Для НомерКолонки = 0 По Результат.Колонки.Количество() - 1 Цикл  
ТекущийЭлементНоменклатуры = Выборка[НомерКолонки];  
Если ТекущийЭлементНоменклатуры = Справочники.Номенклатура.ПустаяСсылка() Тогда  
Прервать;  
Иначе  
Сообщить(ТекущийЭлементНоменклатуры);  
КонецЕсли;  
КонецЦикла;
```

Работа с формами справочника

✓ Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в справочнике, система поддерживает несколько форм представления справочника. Система может автоматически генерировать все нужные формы справочника

✓ Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию



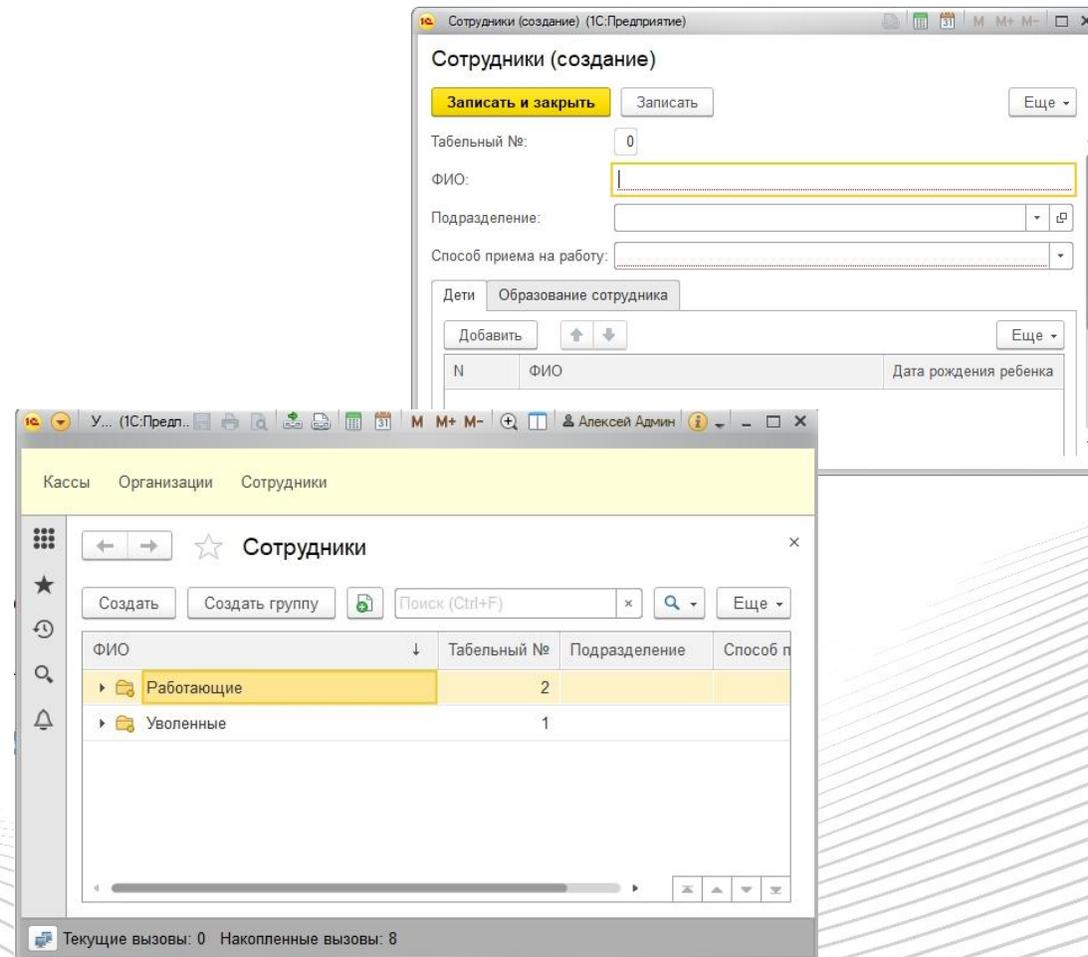
Работа с формами справочника. Форма элемента и форма списка

Форма элемента –

та форма, которая откроется пользователю, когда он начнет создавать новый элемент справочника, или редактировать текущий. Как правило, на ней должны быть поля, связанные с реквизитами справочника, которые может редактировать оператор

Форма списка –

форма, в которой перечислены все элементы справочника



Работа с формами справочника. Форма элемента и форма списка

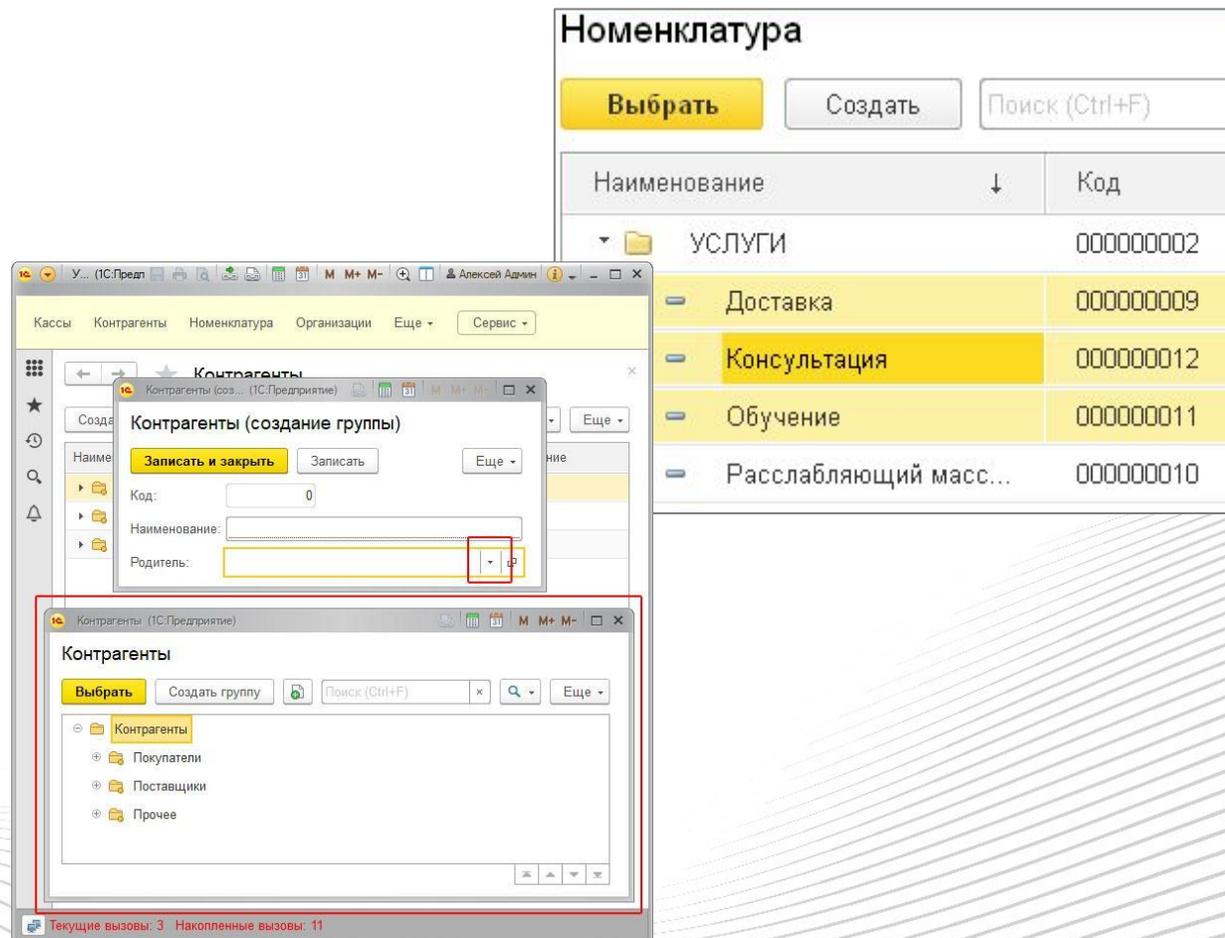
Форма выбора –

форма, которая откроется, когда пользователь будет выбирать нужный элемент справочника в каком-то поле

Форма группы (для иерархических справочников) – та форма, которая откроется пользователю, когда он начнет создавать новую группу справочника или редактировать текущую

Форма выбора группы

(для иерархических справочников) – форма, которая, откроется, когда пользователь будет выбирать нужную группу в каком-либо поле





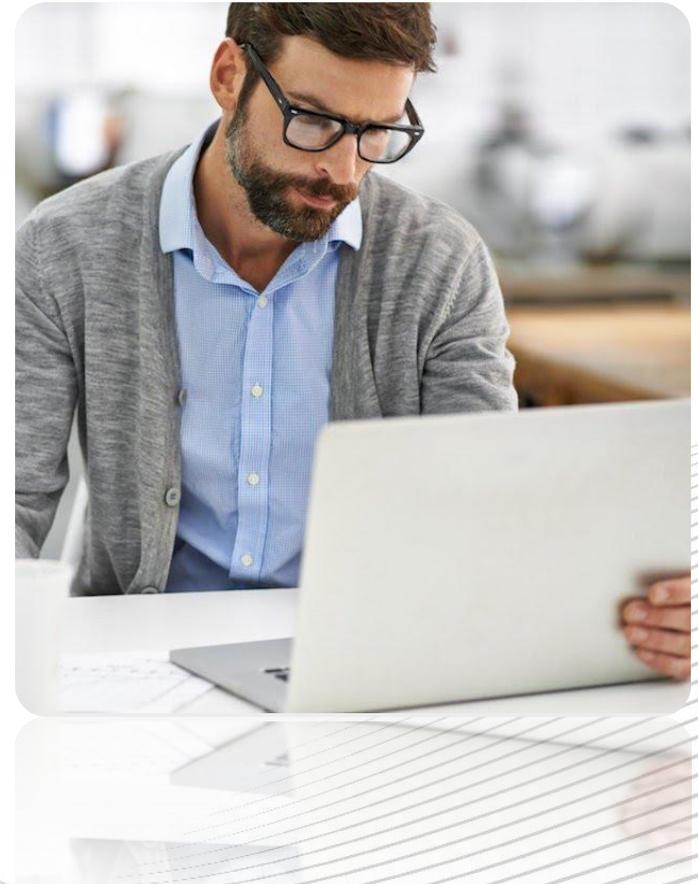
Используются при необходимости задать в системе некий линейный и не изменяемый список



В режиме исполнения список не подлежит ни исправлению, ни переопределению состава

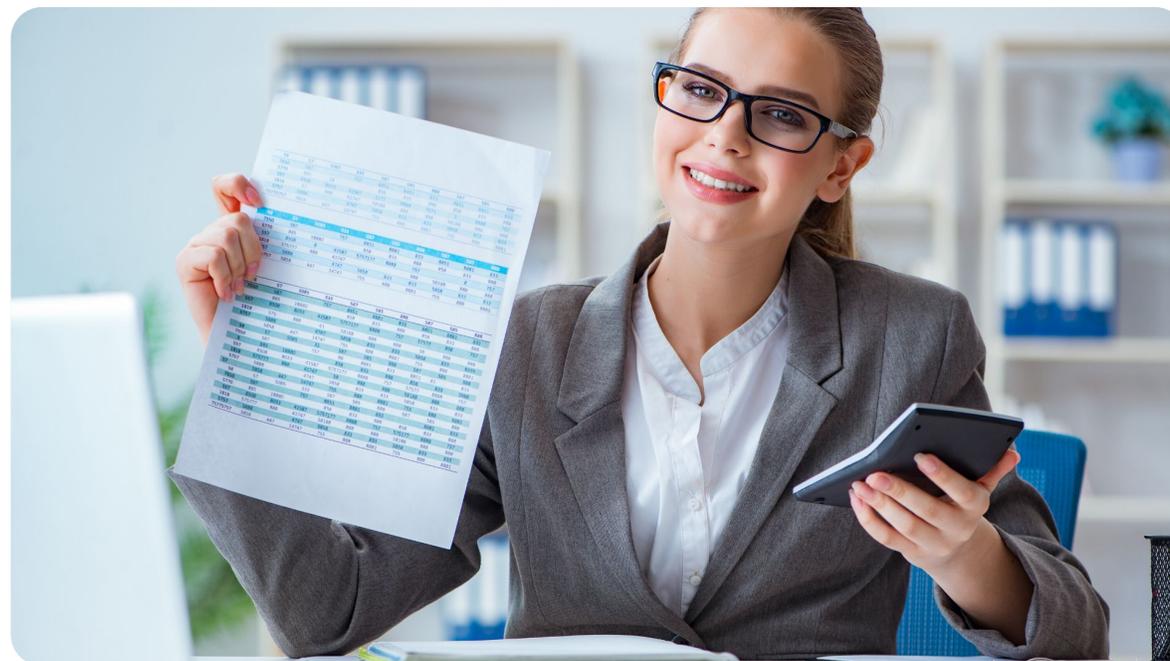


Пользователь может либо выбрать значение из этого списка, либо сбросить выбор



Для чего нужны печатные формы?

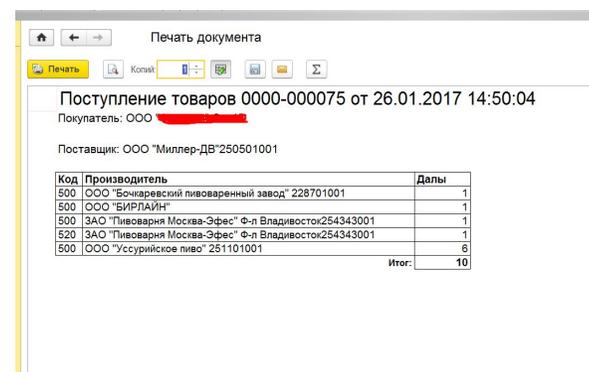
Без бумажки не может обойтись ни один серьезный бизнес. И когда мы говорим, что в 1С есть какие-то там электронные документы, то сразу же возникает вопрос как их распечатать в бумажном виде



Для чего нужны печатные формы?



Печатная форма – макет документа, который может автоматизировано заполнять параметры (поля) данными из информационной базы, для печати или сохранения в электронном варианте



Документы



Документ – это объект метаданных, с помощью которого в прикладном решении хранится информация о совершенных хозяйственных операциях или о событиях, произошедших в «жизни» предприятия вообще.

Это могут быть, например, приходные накладные, приказы о приеме на работу, счета, платежные поручения и т.д.



Каждый созданный документ в конфигурации создается для отражения своего типа операции (например, поступление товаров, списание товаров и т.д.).



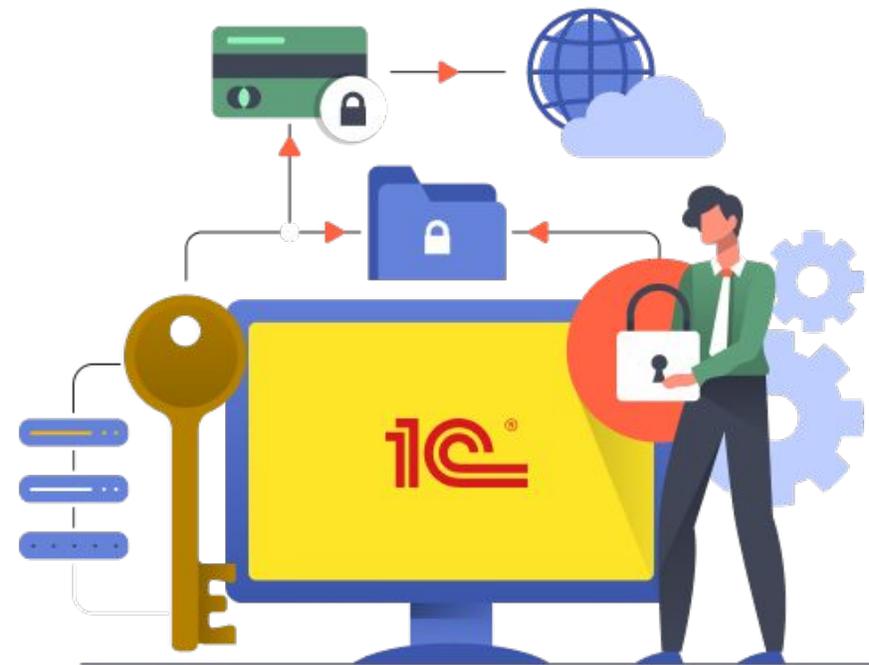
Состояния документа

Проведение документа – установка признака от том что документ принят к учету

- ✓ Не помечен на удаление и не проведен
- ✓ Не помечен на удаление и проведен
- ✓ Помечен на удаление и не проведен



Объект встроенного языка «Момент времени» содержит дату проведения документа и ссылку на него, что позволяет однозначно идентифицировать положение документа на временной оси



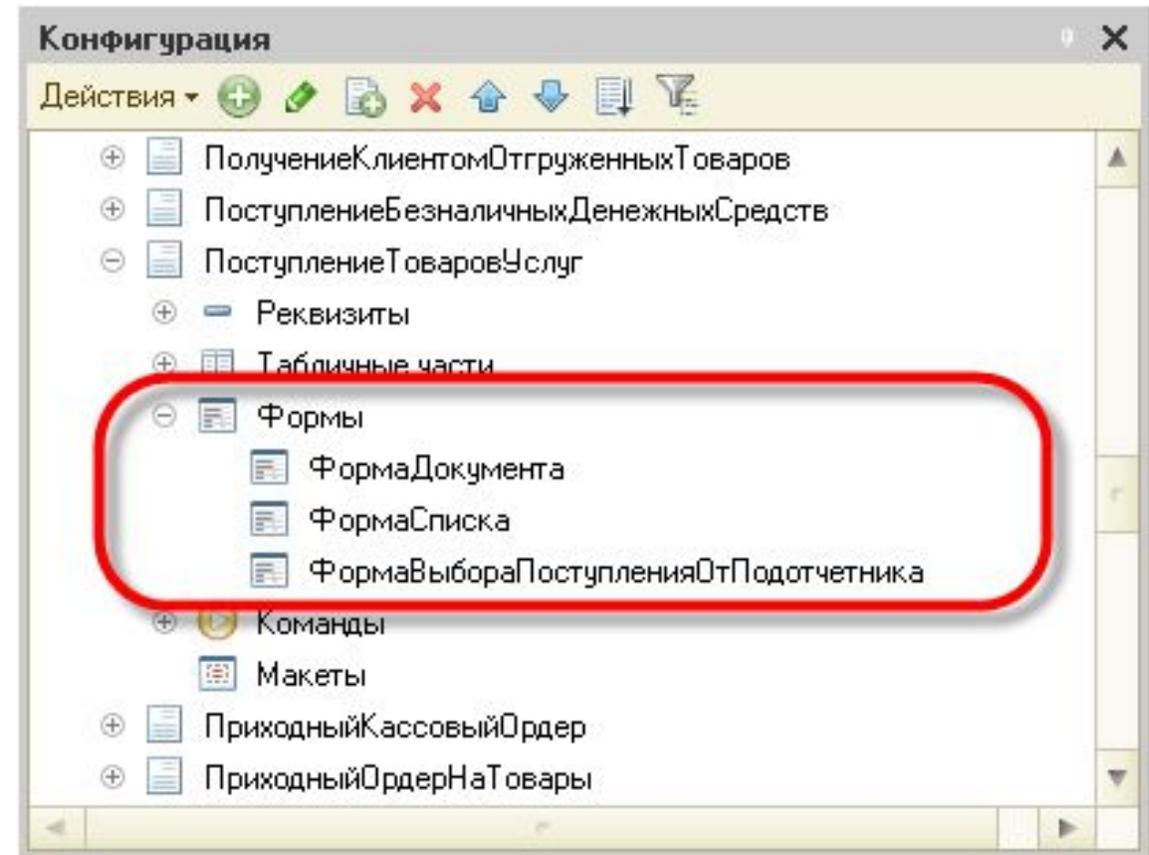
Формы документа



Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в документах, система поддерживает несколько форм представления документа. Система может автоматически генерировать все нужные формы документа



Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию

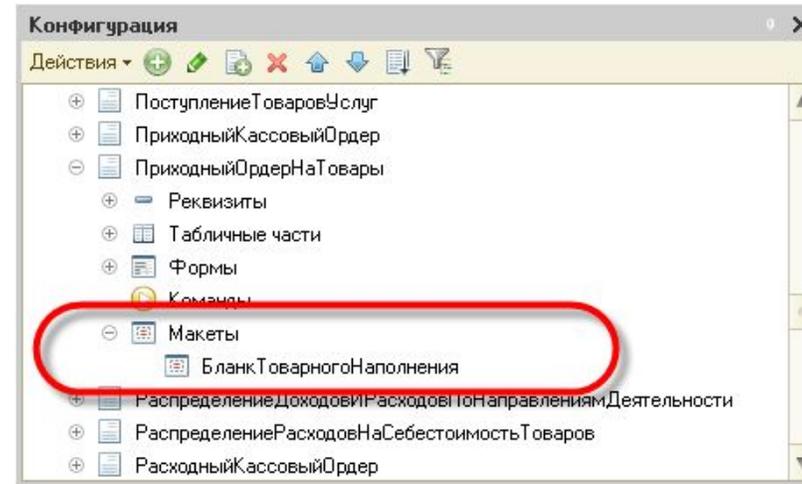


Макеты документа



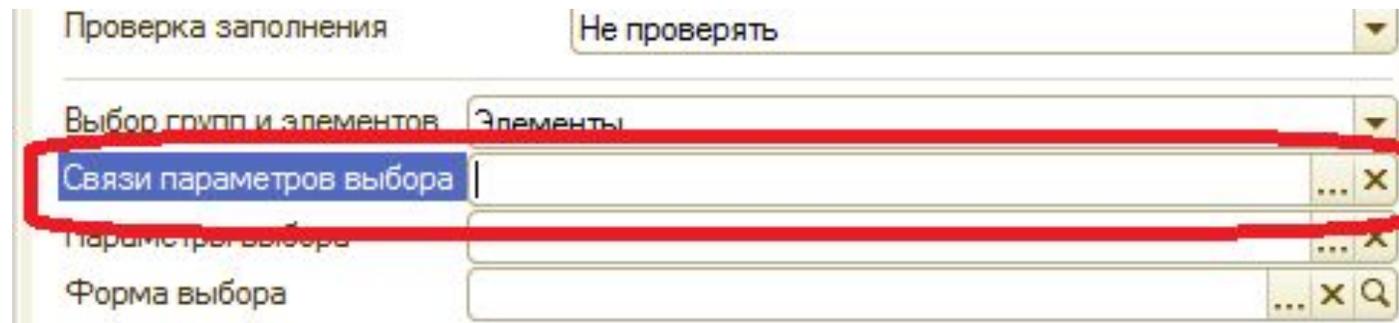
Документу могут быть сопоставлены несколько макетов, содержащих данные, необходимые для обеспечения работы документа

Макеты могут использоваться для формирования печатных форм документа или для отображения дополнительной информации, имеющей отношение к документу



	1	2	3	4
Заголовок	1			
	2			
	3			
	4			
	5			
	6			
СтрокаТовара	7	<Номенклатура>	<КоличествоУпаков>	<Упаковка>
	8			
ТочкаМаршрута	9	<ТочкаМаршрута>		
	10			
	11			

Связи параметров выбора



Зачастую в процессе работы пользователя с формами и элементами форм возникает необходимость ограничивать объем предлагаемых для выбора данных, другими словами, использовать фильтр

Вот именно одним из таких «фильтров» является свойство **«Связи параметров выбора»**

Это свойство предназначено для настройки фильтра формы выбора без написания кода программы, т.е. без программирования

Обсуждение

Вопросы, комментарии, возражения

