

Пример 2. Заполнение реквизита ФИО по данным реквизитов : Ф. | И. | О.

✓ используем метод обращения – «ЭтотОбъект», т.к. он обеспечивает получение данных из реквизитов в *открытом экземпляре* объекта конфигурации

✓ через точку указываем конкретный реквизит – для данной задачи

✓ Используем методы: СтрРазделить(), Лев()

✓ Реализуем процедуру ...

```
Процедура ПередЗаписью(Отказ)
МассивСтрок = СтрРазделить(ЭтотОбъект.Наименование, " ");

ЭтотОбъект.ФИО_Сокращенно= МассивСтрок[0] + " "
                        + Лев(МассивСтрок[1],1) + "."
                        + Лев(МассивСтрок[2], 1) + ".";

КонецПроцедуры
```

Пример 3 **Расчет суммы по строкам каждой табличной части и итогового значения по двум табличным частям**

Варианты решения:

- 3.1 В модуле объекта** – процедура «ПередЗаписью()»
- 3.2 В модуле формы** – обработчик события «ПриИзменении(Элемент)»

Вариант решения 3.1

- 1) организуем построчный подсчет суммы:
 - цикл Для Каждого
 - обращение к элементам таблицы через метод **ЭтотОбъект** (ЭтотОбъект.СписокТоваров)
 - **ЭтотОбъект** - служебное слово (метод), актуальное для данного модуля, означает обращение к конкретной ячейке БД
 - реализуем расчет : $\text{Сумм} = \text{Цена} * \text{Количество}$

```
Процедура ПередЗаписью(Отказ, РежимЗаписи, РежимПроведения)
  Для Каждого СтрокаТаблицы из ЭтотОбъект.СписокТоваров Цикл
    СтрокаТаблицы.Сумма = СтрокаТаблицы.Цена * СтрокаТаблицы.Количество;
  КонечЦикла;
КонечПроцедуры
```

- 2) Для расчета итогового значения табличной части:
 - а) вводим реквизит объекта «ИтоговаяСумма»
 - б) включаем этот реквизит в ранее написанную процедуру:

```
Процедура ПередЗаписью(Отказ, РежимЗаписи, РежимПроведения)
  ЭтотОбъект.ИтоговаяСумма=0;

  Для Каждого СтрокаТаблицы из ЭтотОбъект.СписокТоваров Цикл
    СтрокаТаблицы.Сумма = СтрокаТаблицы.Цена * СтрокаТаблицы.Количество;

    ЭтотОбъект.ИтоговаяСумма=ЭтотОбъект.ИтоговаяСумма + СтрокаТаблицы.Сумма;
  КонечЦикла;
КонечПроцедуры
```

- 3) В случае двух и более табличных делаем дополнительно циклы аналогично п.2

Пример 3 Расчет суммы по строкам каждой табличной части и итогового значения по двум табличным частям

Вариант решения 3.2 : в модуле формы – обработчик события «ПриИзменении(Элемент)»

событие – изменение цены или количества в форме документа ЗаказНаряд

вид модуля – модуль ФормаДокумента в Документе ЗаказНаряд

процедура – расчет суммы по строкам формы (каждой позиции в документе ЗаказКлиента) при изменении одного или двух параметров:
 $\text{ТоварыСумма} = \text{ТоварыКоличество} * \text{ТоварыЦена}$

используем созданную ранее Форму Документа

переходим в свойства элемента ТовариЦена (таблицы Товары)

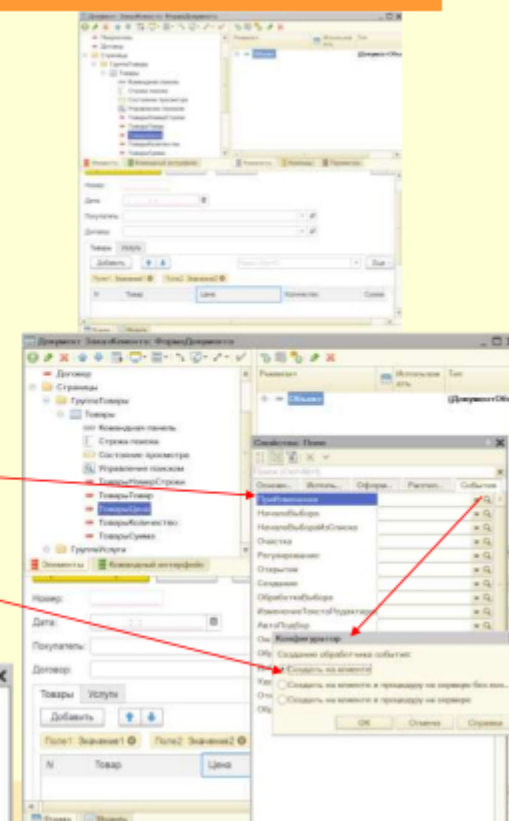
на вкладке «События» в Палитре свойств этого элемента выбираем поле ПриИзменении

в открывшемся окне конструктора «Создание обработчика события» оставляем стандартный выбор «Создать на клиенте»

конструктор «Создание обработчика события» сформирует шаблон процедуры и откроет Модуль формы:

```

Документ ЗаказКлиента: Форма/Документа
{
  &НаКлиенте
  Процедура ТовариЦенаПриИзменении(Элемент)
  // Вставить содержимое обработчика.
  КонецПроцедуры
}
    
```



В этом модуле необходимо описать процедуру **ТоварыЦенаПриИзменении(Элемент)**

&НаКлиенте

Процедура ТоварыЦенаПриИзменении(Элемент)

Элементы.Товары.ТекущиеДанные.Сумма = Элементы.Товары.ТекущиеДанные.Количество * Элементы.Товары.ТекущиеДанные.Цена;
КонецПроцедуры

Эта процедура позволит автоматически пересчитывать Сумму при изменении Цены товара в форме документа ЗаказКлиента (в пользовательском режиме)

Аналогично создаем процедуру **ТоварыКоличествоПриИзменении(Элемент)**, которая позволит пересчитывать Сумму при изменении Количества товара

&НаКлиенте

Процедура ТоварыЦенаПриИзменении(Элемент)

Элементы.Товары.ТекущиеДанные.Сумма = Элементы.Товары.ТекущиеДанные.Количество * Элементы.Товары.ТекущиеДанные.Цена;
КонецПроцедуры

«Элементы» - т.к. нужно обращаться ко всем элементам формы (строкам)

«Товары» – это название табличной части формы

«ТекущиеДанные» - свойство, определяющее возможность обращения ко всем реквизитам этой табличной части формы – в т.ч. Сумма, Цена, Количество

«Сумма» / «Количество» / «Цена» - <имя> реквизита табличной части документа ЗаказКлиента

(!): **обработчик события** рекомендуется создавать для каждого элемента формы (методическое обоснование ...)

можно принять из контекстной подсказки

можно «перетащить» из Древа конфигурации

Оптимизируем составленный код:

- 1 - Оптимизация (экономия) объема программного кода;
- 2 - Оптимизация структуры программного кода
- 3 – Структурирование программного кода

1. Оптимизация (экономия) объема программного кода:

- а) в исходном варианте используется (дублируется) в нескольких местах одна и та же конструкция: «Элементы.Товары.ТекущиеДанные»
- б) можно заменить эту конструкцию специально введенной переменной; например, «СтрокаРасчета=Элементы.Товары.ТекущиеДанные» (произвольно принятое наименование)
- в) заменим в исх.коде эту конструкцию на переменную, что приведет к сокращению объема кода (происходит т.н. связь по ссылке)

```
&НаКлиенте
[ Процедура ТоварыКоличествоПриИзменении (Элемент)
  СтрокаРасчета=Элементы.Товары.ТекущиеДанные;
  СтрокаРасчета.Сумма = СтрокаРасчета.Количество * СтрокаРасчета.Цена;
  КонечПроцедуры

&НаКлиенте
[ Процедура ТоварыЦенаПриИзменении (Элемент)
  СтрокаРасчета=Элементы.Товары.ТекущиеДанные;
  СтрокаРасчета.Сумма = СтрокаРасчета.Количество * СтрокаРасчета.Цена;
  КонечПроцедуры
```

2. Оптимизация структуры программного кода:

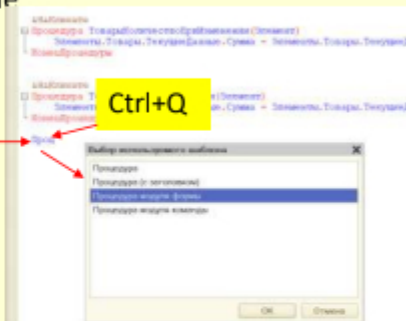
а) в исходном варианте – две процедуры, которые содержат одинаковые наборы команд

б) для исключения такого дублирования рекомендуется создавать дополнительную процедуру (для этого модуля формы) – **«РасчетСуммы»**

в эту процедуру помещаем строки кода:

СтрокаРасчета=Элементы.Товары.ТекущиеДанные;
СтрокаРасчета.Сумма = СтрокаРасчета.Количество * СтрокаРасчета.Цена;

в) в процедурах-обработчиках событий «ТоварыЦенаПриИзменении..» и «ТоварыКоличествоПриИзменении..» вызываем процедуру «РасчетСуммы»



```

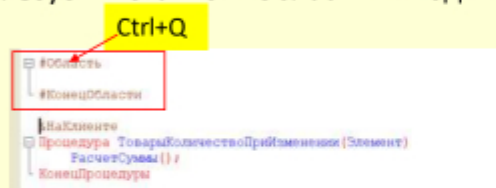
«НаКлиенте
  Процедура ТоварыКоличествоПриИзменении (Элемент)
    РасчетСуммы ();
  КонечПроцедуры

«НаКлиенте
  Процедура ТоварыЦенаПриИзменении (Элемент)
    РасчетСуммы ();
  КонечПроцедуры

«НаКлиенте
  Процедура РасчетСуммы ()
    СтрокаРасчета=Элементы.Товары.ТекущиеДанные;
    СтрокаРасчета.Сумма = СтрокаРасчета.Количество * СтрокаРасчета.Цена;
  КонечПроцедуры
  
```

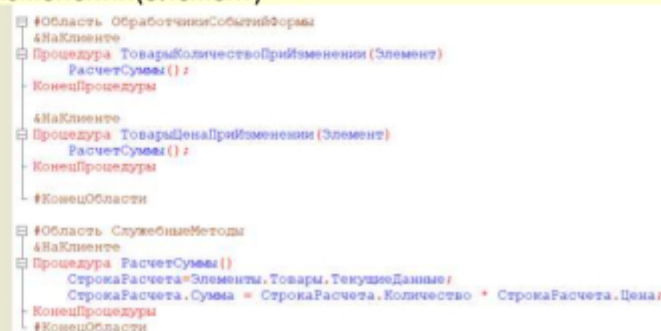
3. Структурирование программного кода:

а) используем механизм Области – вводим две Области: «ОбработчикиСобытий» (для «основных» процедур) «СлужебныеМетоды» (для доп. проц. «РасчетСуммы»)



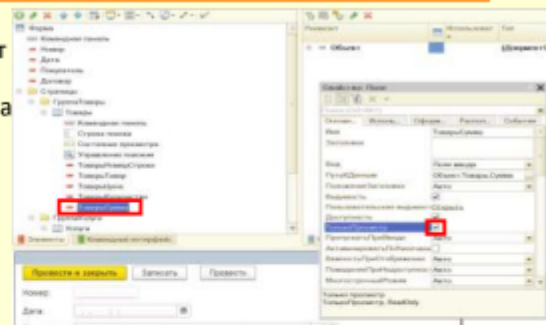
б) в Область «ОбработчикиСобытий» переносим процедуры ТоварыЦенаПриИзменении(Элемент) и ТоварыКоличествоПриИзменении(Элемент)

в) Область «СлужебныеМетоды» - процедуру РасчетСуммы()

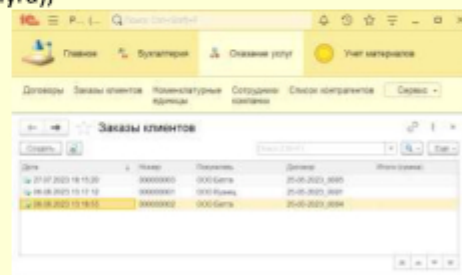


После введения Обработчика события «Расчет суммы заказа...», который позволяет автоматизировать вывод реквизита Сумма в форме документа «Заказ клиента», желательно **исключить возможность редактирования** значений этого реквизита (в **пользовательском режиме**):

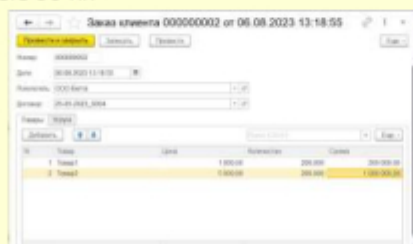
на вкладке Формы документа ЗаказКлиента
редактируем настройку элемента «ТоварыСумма» –
в Палитре свойств ставим флаг в поле «ТолькоПросмотр»



Сформированный документ ЗаказКлиента позволяет выполнять функции (в пользовательском режиме):
ввод данных о товаре, количестве и цене – в разрезе номенклатурных групп (Товар, Услуга);
автоматический расчет суммы по элементам заказа;
формировать сводный список заказов со следующими реквизитами:
дата заказа / номер заказа / Покупатель / договор

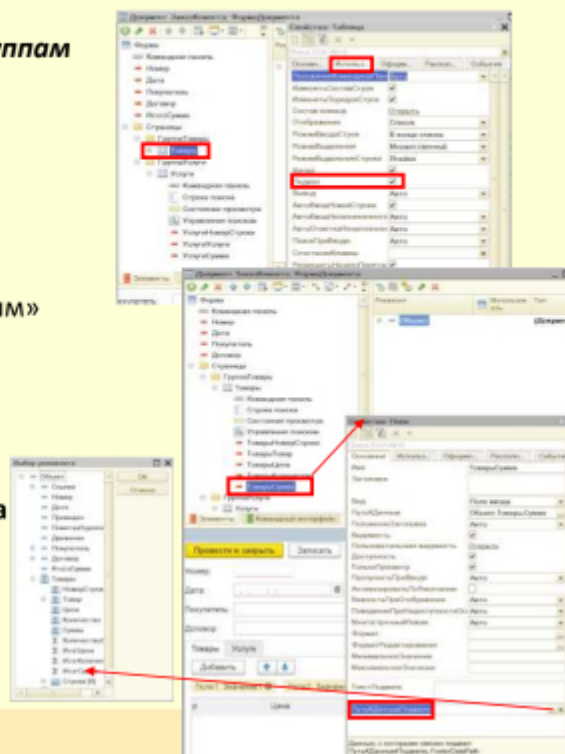


(-) : в списке заказов отсутствуют данные об их
стоимости в карточке заказа
=> нужен доп.параметр –
итоговая сумма



Включим функцию расчета итоговой суммы по номенклатурным группам (на вкладке «Товары» и вкладке «Услуги» в карточке заказа):

- обращаемся к объекту конфигурации «Документы / ЗаказКлиента / Формы / ФормаДокумента»
 - в Форме документа, в табличной части для группы «Товары» **активируем итоговую строку**: в Палитре свойств на вкладке Использование ставим флажок в поле «Подвал» (после этого у элементов табличной части появляется свойство «ПутьКДанным»)
 - в окне редактора элементов формы выбираем реквизит «ТоварыСумма» (в табличной части, в группе Товары)
 - открываем палитру свойств, вкладку Основные
 - в поле **ПутьКДаннымПодвала** выбираем нужную функцию – **ИтогСумма**
 - в поле ТекстПодвала можно сделать комментирующую запись, например «Итоговая сумма:» (эта запись м.б. сделана и в «подвале» по др. реквизиту)
- (аналогично настраиваем расчет в табличной части «Услуги»)



Сформируем процедуру расчета суммы в документе ЗаказКлиента:

- создаем дополнительный реквизит в Документе ЗаказКлиента (в дополнение к двум созданным ранее) - «ИтоговаяСумма»

(этот реквизит будет хранить итоговую сумму для каждого документа)

- установим Тип данных - *ОпределяемыйТип.ДенежныеСредства* (ранее созданный нами)

- создаем процедуру, призванную выполнять функцию расчета суммы в каждом документе ЗаказКлиента:

используемый вид модуля – Модуль объекта (находим на вкладке «Прочие» в окне редактирования объекта ЗаказКлиента);

в модуле создаем область: #Область ОбработчикиСобытий

вызываем список доступных для этого модуля процедур (Ctrl+Alt+P)

используем процедуру «ПередЗаписью» (процедура расчета суммы включится после ввода данных и нажатия кнопки «Записать» (в пользовательском режиме)

пишем код в «Модуле объекта» документа «ЗаказКлиента»:

ЭтотОбъект.ИтоговаяСумма=ЭтотОбъект.Товары.Итог("Сумма")+ ЭтотОбъект.Услуги.Итог("Сумма");

ЭтотОбъект служебное слово (метод), актуальное для данного модуля, означает обращение к конкретной ячейке БД

ИтоговаяСумма имя реквизита соответствующего объекта конфигурации (Документ «ЗаказКлиента»), для которого описываем алгоритм расчета искомой величины

Товары имя табличной части объекта конфигурации, в которой находятся данные для расчета

Итог функция (выбор из открывающегося списка)

"Сумма" наименование столбца

Механизм Ввод на основании

механизм «**Ввод на основании**»

позволяет переносить данные из



Механизм Ввод на основании

Пример:



Документ.ЗаказКлиента

1. Реквизиты документа –
Номер / Дата / Покупатель / Договор / Итого
2. Реквизит «Покупатель» -
должен быть связан со справочником Контр
3. Реквизит «Договор» -
должен быть связан с существующим справ
возможность выбора и отбор договоров для
4. Табличные части –
«СписокТоваров», «Услуги»
5. Реквизиты табличной части «СписокТоваров» -
Номер / НаименованиеТовара / Цена / Коли
6. Реквизит «НаименованиеТовара » в табличной ч
должен быть связан с существующим справ
7. Реквизит «Сумма» в табличной части «СписокТо
должен обеспечивать автоматический расче
8. Реквизиты табличной части «Услуги»:
аналогично пп. 5-7

