

Автоматизация бизнес-процессов

Второй отборочный этап

Задача IV.1. Создание бизнес-приложения для туристической компании (внесение справочной информации) (10 баллов)

Условие

Помогите Грише!

Каждый из нас хотел бы, чтобы трудовая деятельность, которая занимает большую часть жизни человека, приносила радость, была любимым делом. Алла Волкова всегда мечтала, чтобы ее жизнь была связана с путешествиями. Ей удалось осуществить свою мечту и создать свою небольшую туристическую компанию. Но оказалось, что ведение бизнеса — это очень непростой процесс и большой труд.

Ей понадобилось хранить большое количество информации об отелях, с которыми она начала сотрудничать, о наличии в них свободных номеров. Нужно было где-то фиксировать информацию о клиентах, их потребностях, заказах, оплатах.

У Аллы голова шла кругом — она путалась в бесконечных документах, файлах, папках. И вдруг поняла, что нужно с этим что-то делать. В сети Интернет Алла прочла, что есть компьютерные программы по автоматизации бизнес-процессов, которые позволяют сохранить всю информацию в одном месте, и часть ручного труда будет выполнять компьютер. При помощи таких программ можно анализировать информацию и грамотно принимать решения по развитию своей компании.

Алла решила, что ей нужна такая программа. Но сама она в компьютерных программах ничего не понимает. Тогда она вспомнила о своем однокласснике Грише, который, по слухам, стал крутым программистом.

Григорий согласился помочь. Но слухи оказались слишком преувеличены — программист из Гриши вышел никудышный. Ему очень стыдно перед Аллой, и он просит вас ему помочь — решить несколько задач.

Кстати сказать, Алла, в отличие от Гриши, оказалась очень аккуратной и умной. Она подробно описала, что ей нужно — получилось практически техническое задание, разбитое на 5 задач.

Задача для Гриши №1

Необходимо хранить информацию об отелях, с которыми заключен договор о сотрудничестве.

Техническое задание

Структура места хранения (справочника) следующая:

- «Название отеля» — строка до 150 символов, хранит наименование отеля.
- «Местонахождение» — ссылка на элемент другой таблицы (справочника или классификатора), где хранится список всех регионов. Это необходимо для того, чтобы при заведении нового отеля в систему вручную избежать ошибок в указании регионов — «Москва», «г. Сочи» и т. д.

- «Контактный телефон» — строка.
- «Ф. И. О. управляющего» — ссылка на элемент другой таблицы (справочника). Управляющий может уйти из отеля, но связи с ним полезно сохранить. Поэтому удобно в системе иметь справочник, в котором будет храниться информация о контактных лицах, их ролях и их личных контактов — телефон и e-mail.
- «Описание отеля» — строка до 500 символов, многострочный режим. Хранит краткое описание особенностей отеля.

Критерии оценивания

Критерий	Базовый балл	Прибавка	Макс. балл
Созданная информационная база	1		1
Созданный справочник «Отели»	1		1
Реквизит справочника «Название отеля» — строка до 200 символов, хранит наименование отеля.	1		1
Реквизит справочника «Местонахождение»	1	Если реализована ссылка на справочник «Регионы» +1	2
Реквизит справочника «Контактный телефон»	1		1
Реквизит справочника «Ф. И. О. управляющего»	1	Если реализована ссылка на справочник «Контактные лица» +1	2
Реквизит «Описание отеля» — строка до 500 символов, многострочный режим.	1		1
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		+1	1

Решение

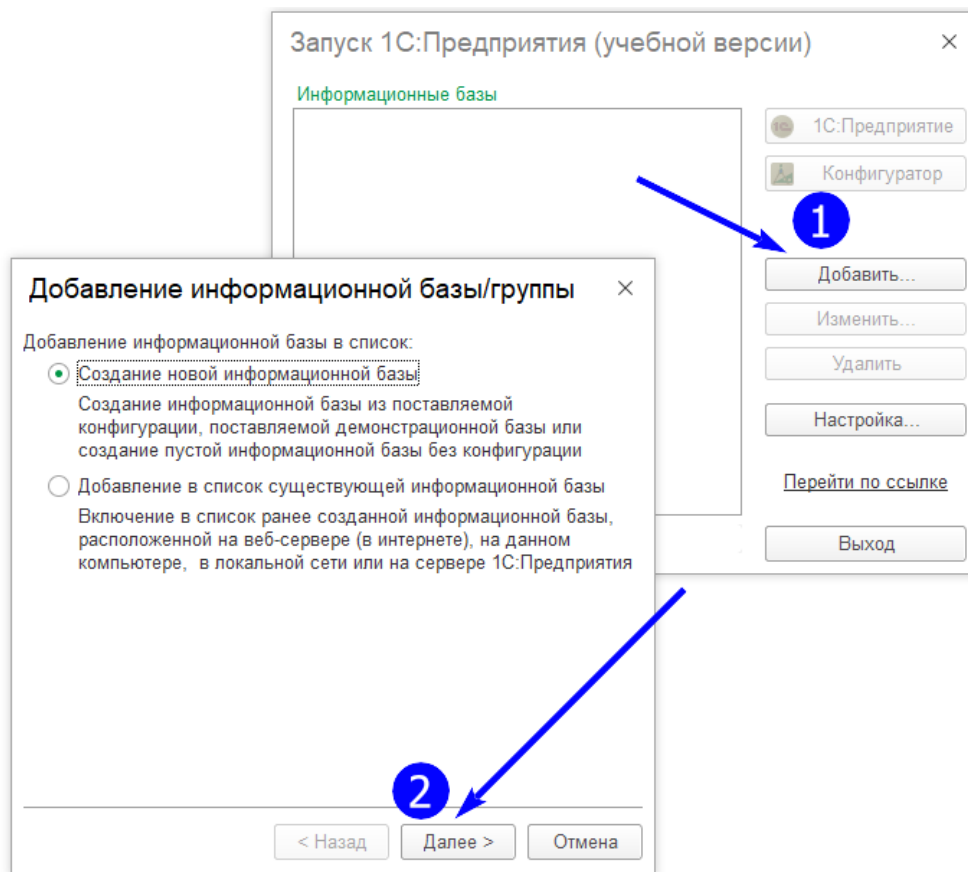
Рекомендуем для решения задач использовать технологическую платформу «1С:Предприятие 8» — это современное средство для создания приложений для автоматизации ведения бизнеса.

Существует учебная версия платформы, которой достаточно для решения наших задач. Скачать ее можно совершенно бесплатно здесь: <https://uc1.1c.ru/uchebnaya-versiya-1s/>.

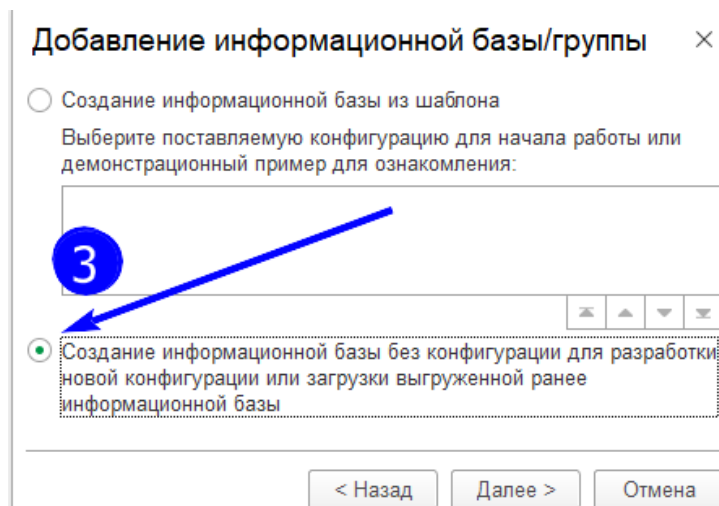
Устанавливается она совершенно стандартно с помощью мастера установки (как и все остальные программы).

Обучающие материалы по работе с платформой у нас выложены на странице Олимпиады:

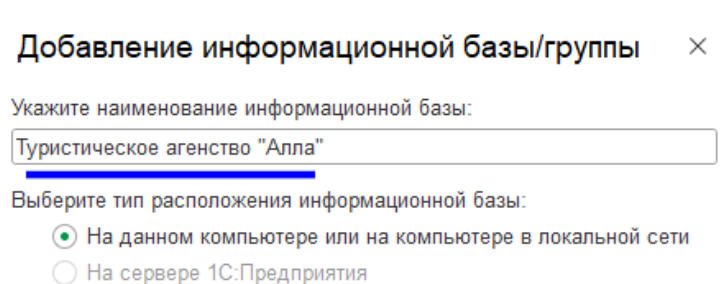
1. После запуска платформы (после установки создается соответствующий ярлычок) необходимо создать информационную базу туристической компании.



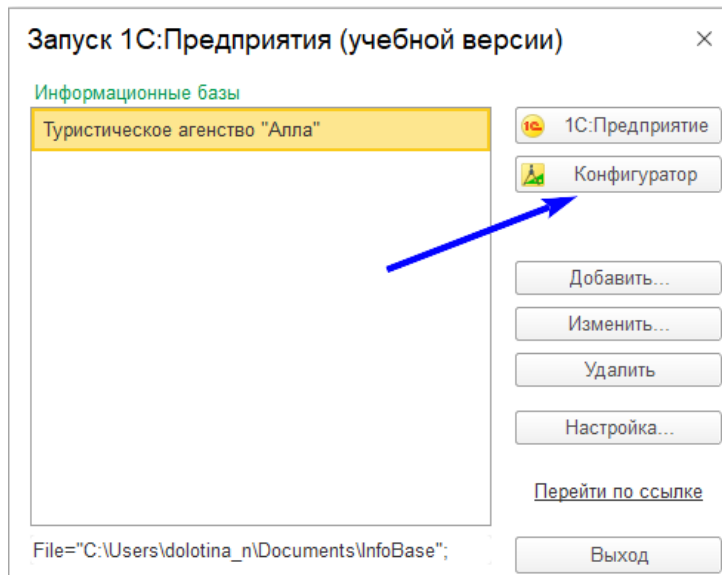
2. Выбираем переключатель «Создание информационной базы из шаблона».



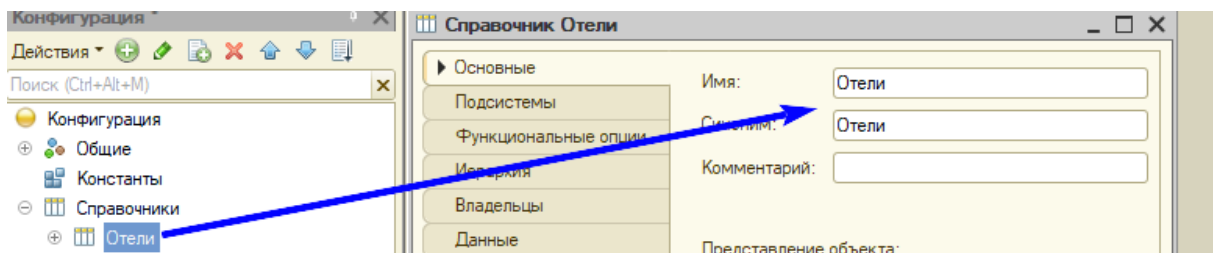
3. Дать название, например.



4. Запустить информационную базу в режиме «Конфигуратор» и начать разработку.

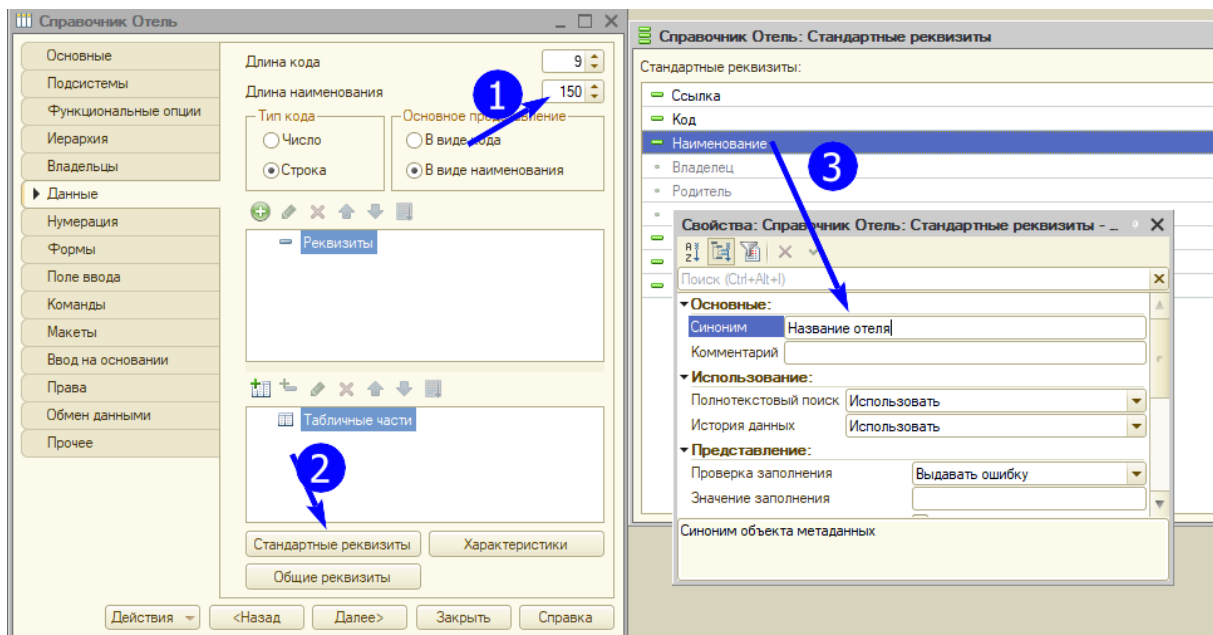


5. Создать справочник «Отели».

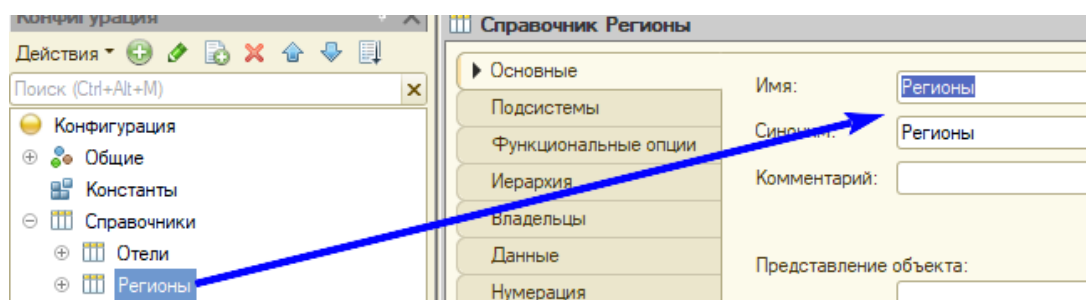


6. Перейти на закладку «Данные» и создать реквизиты справочника согласно описанной выше структуре.

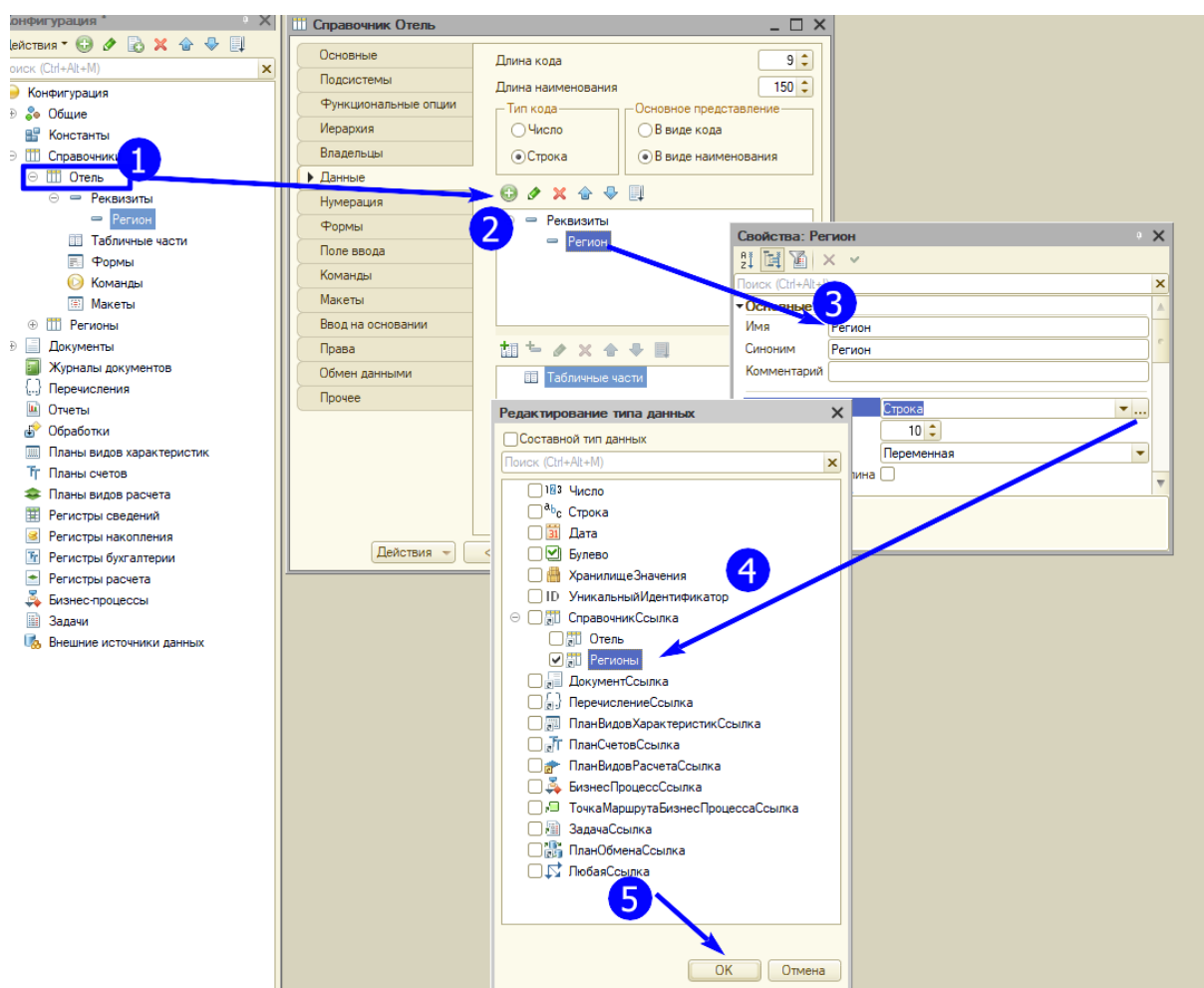
- 6.1. Реквизит «Название отеля» — стандартный реквизит «Наименование». Указываем в поле «Длина наименования» — 150 символов. Можно задать синоним (отображение для пользователя) — «Название отеля».



6.2. При создании реквизита «Местонахождение» сталкиваемся с тем, что нам потребуется еще один справочник — «Регионы», создаем его. Для этого справочника достаточно стандартных реквизитов.

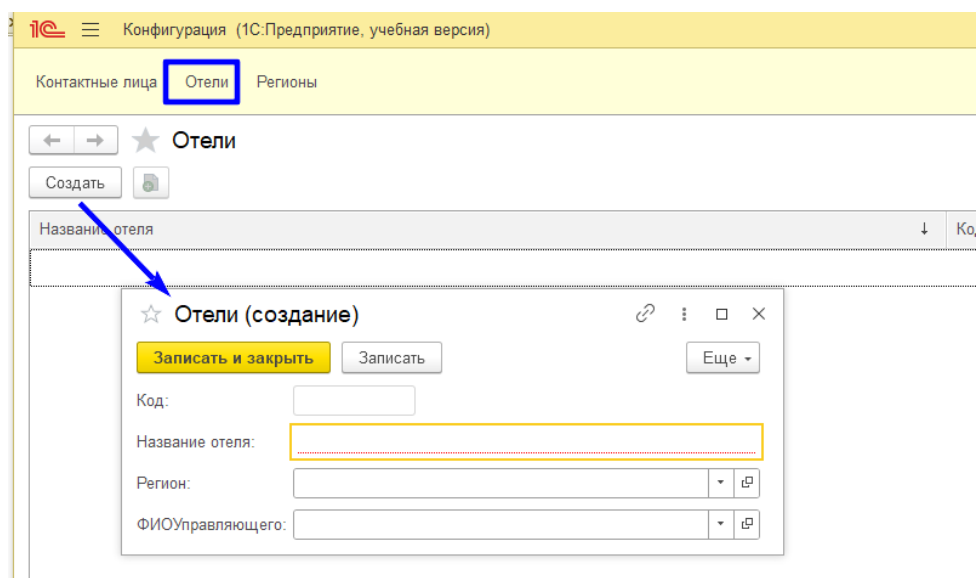


Выбираем новый справочник в качестве ссылки для реквизита «Местонахождение».



6.3. реквизит «Ф. И. О. управляющего» также будет иметь тип данных «ссылка». В качестве ссылки используется справочник «Контактные лица». Его нужно также создать. В стандартный реквизит «Наименование» будет вводиться Ф. И. О. контактного лица, а для хранения телефона и e-mail нужно создать отдельные реквизиты. Чтобы не смущать пользователя, следует для стандартного реквизита «Наименование» задать синоним — Ф. И. О.

По умолчанию платформа «1С:Предприятие 8» сама создает интерфейс приложения.



Можно попробовать уже на начальном этапе поработать над ним — определить цветовую гамму приложения, добавить логотип, название приложения и т. д.

Задача IV.2. Формирование туров (продолжение работы со справочной информацией) (20 баллов)

Условие

Задача для Гриши №2

Туры в компании формируются самостоятельно, вручную, информацию о них необходимо где-то хранить. Кроме этого, в приложении необходимо сохранять информацию о клиентах. Туры продаются как простым людям, так и компаниям — турагентствам. Это все должно быть отражено в программе.

Техническое задание

Создать некий справочник под названием «Туры», который будет хранить следующую информацию:

- «Отель» — отель, с которым заказчик сотрудничает по данной программе. Целесообразно отель выбирать из уже имеющегося списка отелей.
- Даты заезда и выезда — тип «Дата».
- Количество дней/ночей, так удобно ориентироваться заказчикам туров, автоматический расчет на основании дат заезда и выезда.
- Вид питания — выбирается из вариантов: без питания, с завтраком, 3-разовое.
- Стоимость тура — тип «Число», 2 знака после запятой. Удобно рядом указать денежную единицу (руб.).
- Описание тура — строка до 500 символов, многострочный режим. Хранит краткое описание особенностей отеля.

Структура справочника «Клиенты» должна выглядеть следующим образом:

- Наименование — строка до 200 символов, хранит имя клиента.
- Контактное лицо — ссылка на справочник «Контактные лица». При выборе контактного лица должен здесь же автоматически отображаться его телефон,

чтобы не искать его по справочникам.

- Вид клиента — один из двух вариантов: физическое лицо или юридическое лицо.

Критерии оценивания

Критерий	Базовый балл	Прибавка
Созданный справочник «Продажа туров» со стандартными реквизитами	1	
Созданный реквизит «Отель»	1	+1 если реализована ссылка на справочник «Отели»
Реквизиты «Дата заезда» и «Дата выезда» — формат «Дата» (можно «Дата и время» — не снимаем за это баллы)	1	
Реквизит «Вид питания»	1	+1 Если реализован в виде перечисления или в виде справочника
Реквизит «Стоимость тура» с типом «Число»	1	
Реквизит «Описание тура» — строка, многострочный режим	1	
Реализован механизм расчета количества ночей (дней) — достаточно любого параметра. Если есть код, но ошибка в формуле, то выставляется половина базового балла. Если код есть, но в нем ошибки, ставится 1 балл за попытку реализации	1 (если есть попытка создания кода)	+1 если механизм работает, но ошибка в формуле +2 если механизм работает верно
Созданный справочник «Клиенты» со стандартными реквизитами	1	
Созданный реквизит «Контактное лицо»	1	+1 Если реализован в виде перечисления или в виде справочника
Реквизит «Телефон»	1	+2 если реализован механизм подстановки телефона из справочника контактного лица +1 если реализован механизм подстановки телефона с ошибками
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		до +3, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего

Если в сопроводительной записке у участника прописаны дополнительные улучшения, например:

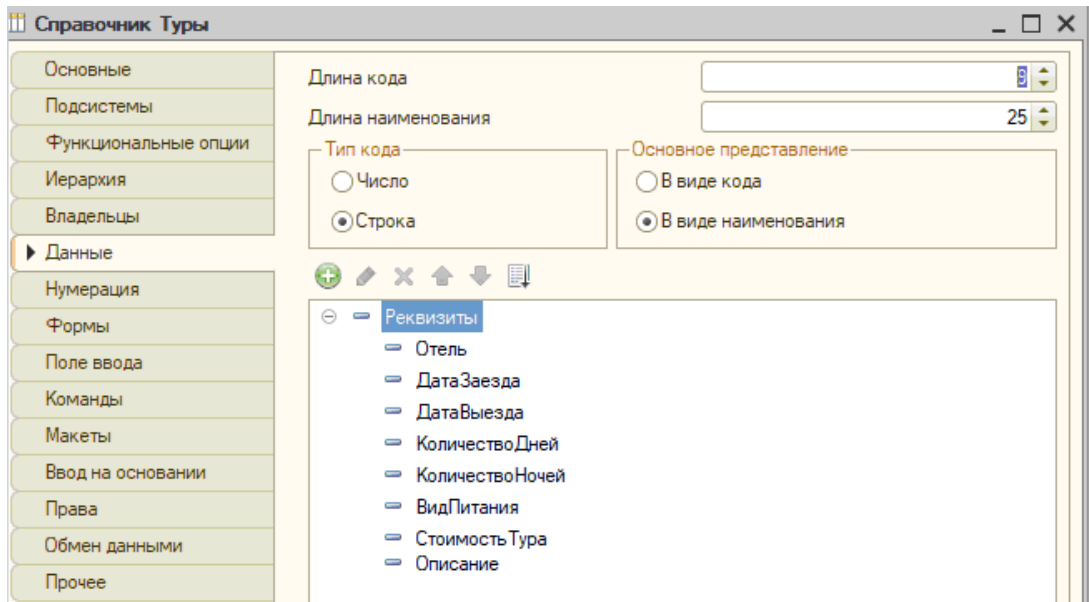
- Запрет ввода прошедшей даты.
- Маска для ввода телефона в определенном формате и т. д.

В этом случае слушателю можно добавить до 3-х бонусных баллов.

Решение

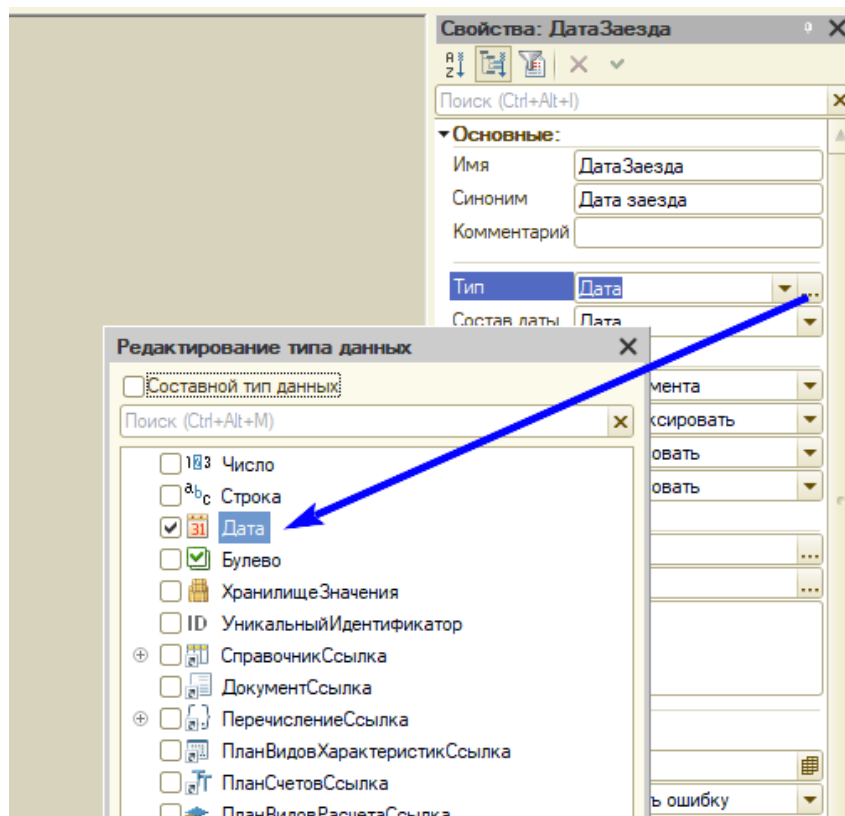
Основные этапы решения задачи представлены на базе платформы «1С:Предприятие 8».

1. Создать справочник «Туры» с перечисленной в задаче структурой.



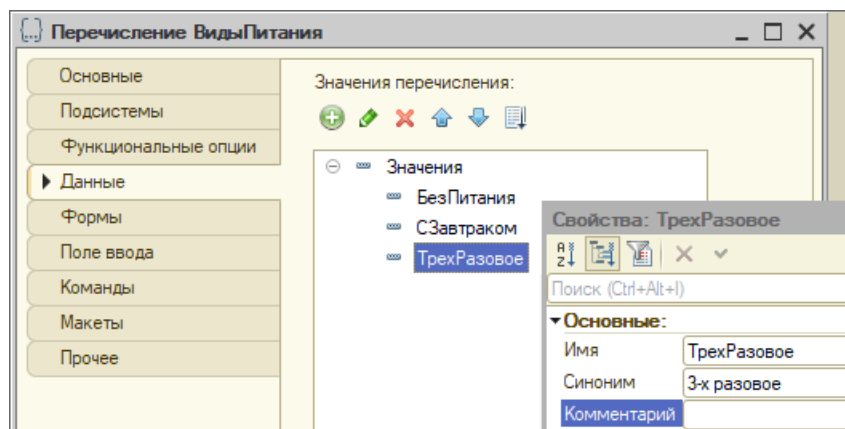
2. Не забываем указывать типы данных:

- 2.1. Тип реквизита «Отель» — ссылка на уже созданный справочник «Отели».
- 2.2. Для реквизитов «ДатаЗаезда» и «ДатаВыезда» указываем тип «Дата».

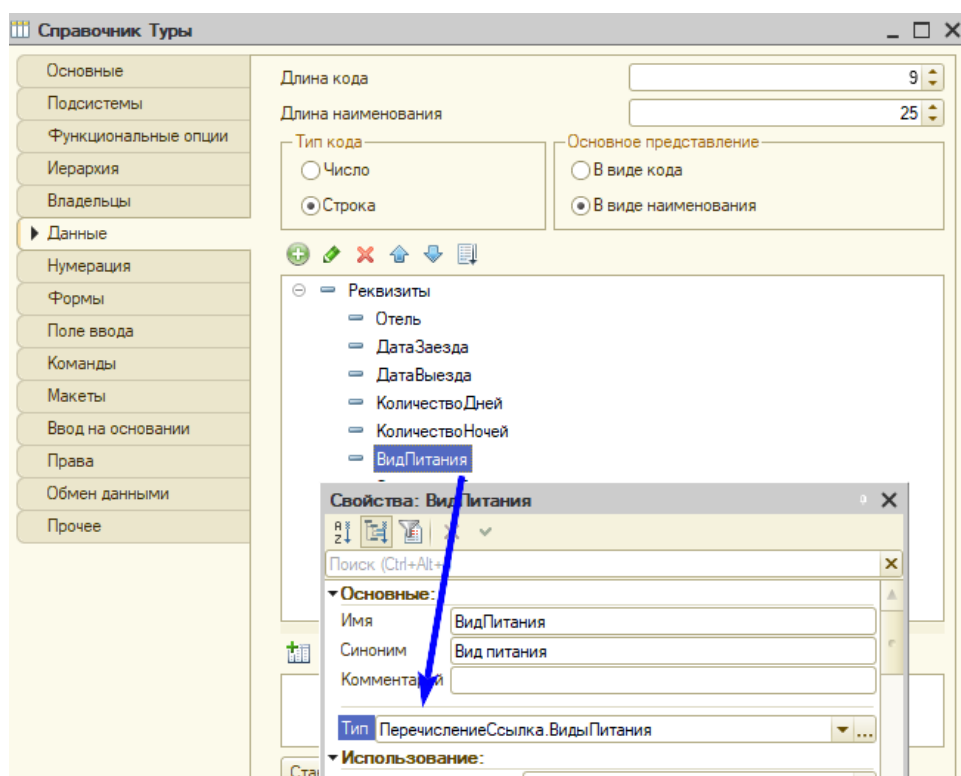


- 2.3. Для реквизитов «Количество дней» и «Количество ночей» указываем тип «Число».

- 2.4. Для реквизита «Описание» указываем тип «Строка», многострочный режим.
- 2.5. Тип реквизита «Вид питания» — т. к. это практически неизменяемая информация, то логично создать новый объект «Перечисления» под названием «Виды питания» и внести туда сразу варианты питания.

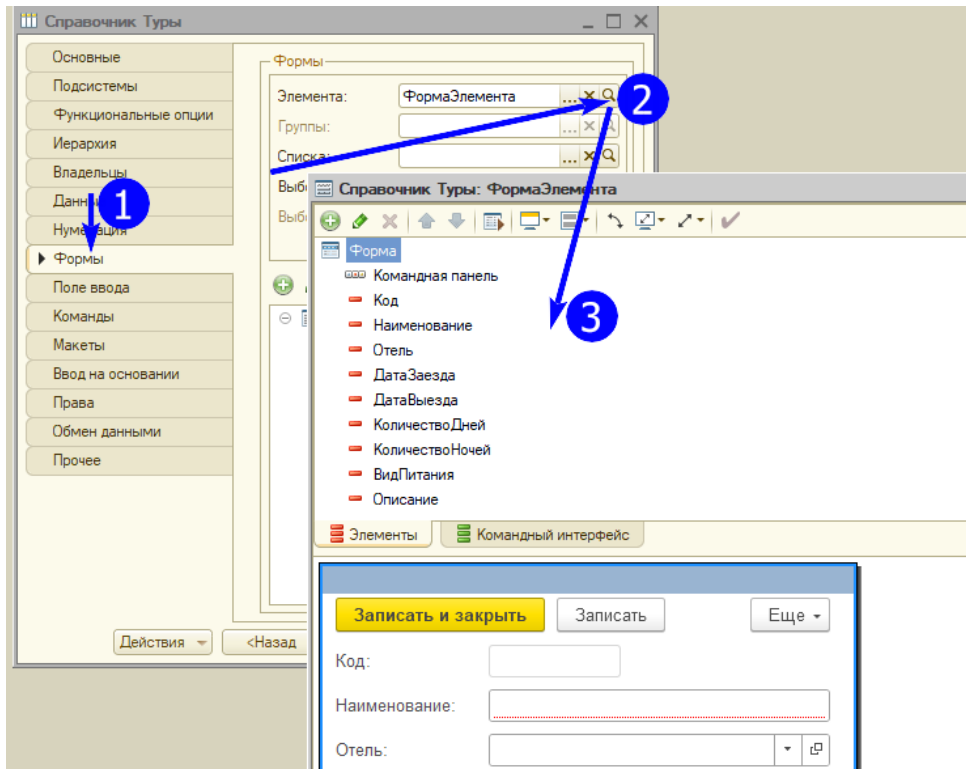


Данное перечисление указываем в качестве типа данных для реквизита «ВидПитания» справочника «Туры».

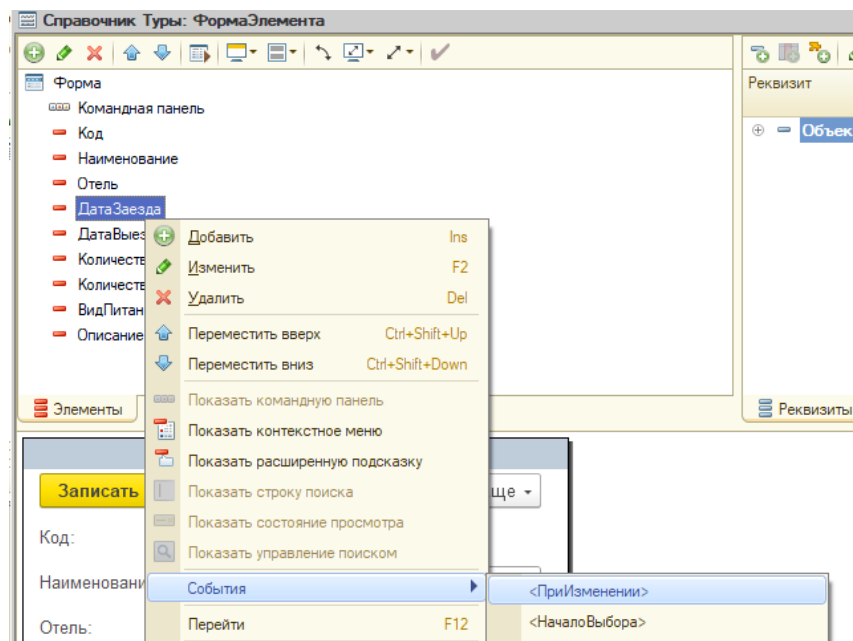


- 2.6. Необходимо реализовать автоматический расчет количества дней и ночей при вводе дат заезда и выезда. Встроенными алгоритмами это уже не реализовать, необходимо написать программный код на встроенном языке «1С:Предприятие 8». Программный код пишется в специальных текстовых редакторах — модулях. Модуль играет роль листочка с инструкциями для системы. Модули бывают различными по назначению. Поскольку дату заезда и выезда мы вносим в форму элемента, то и расчет количества дней и ночей мы также будем осуществлять в форме элемента, и в качестве «листка с инструкциями» нам понадобится модуль формы.

По умолчанию система сама создает интерфейс и формы объектов. Если нам нужно внести какие-то изменения, мы создаем свою форму. В нашем случае форму элемента (это форма карточки самого тура, где в пользовательском режиме вводятся данные).



2.7. Теперь нужно подумать, как писать программный код? Очевидно, что расчет количества дней и ночей происходит при изменении значения реквизитов «ДатаЗаезда» и «ДатаВыезда». Это события. Вообще, программирование в 1С носит, как правило, событийный характер. Определить событие для реквизита «ДатаЗаезда» необходимо следующим образом.



2.8. Система сама перебросит вас в модуль формы, где необходимо прописать алгоритм для данной процедуры. Ниже приведен вариант реализации с комментариями (обращаем внимание, что это не единственный вариант, у вас может быть свой).

```
1 &НаКлиенте
2 Процедура ДатаЗаездаПриИзменении(Элемент)
3 //Проверяем заполнение значений
4 Если ЗначениеЗаполнено(Этаформа.Объект.ДатаЗаезда) и
   ↳ ЗначениеЗаполнено(Этаформа.Объект.ДатаВыезда) тогда
5 //Если значения заполнены, то проверяем нет ли такой ситуации, когда
   ↳ дата заезда больше даты выезда:
6 Если Этаформа.Объект.ДатаЗаезда > Этаформа.Объект.ДатаВыезда тогда
7 //если да - то сообщаем об этом пользователю и сбрасываем даты
8 Сообщить("Дата заезда не может быть больше даты выезда!");
9 Этаформа.Объект.КоличествоДней = 0;
10 Этаформа.Объект.КоличествоНочей = 0;
11 Иначе
12 //если все в порядке (дата заезда меньше даты выезда), то рассчитываем
   ↳ количество дней и ночей
13 Этаформа.Объект.КоличествоДней = (Этаформа.Объект.ДатаВыезда -
   ↳ Этаформа.Объект.ДатаЗаезда) / (24 * 60 * 60) + 1;
14 Этаформа.Объект.КоличествоНочей = (Этаформа.Объект.ДатаВыезда -
   ↳ Этаформа.Объект.ДатаЗаезда) / (24 * 60 * 60);
15 КонецЕсли;
16 Иначе
17 Этаформа.Объект.КоличествоДней = 0;
18 Этаформа.Объект.КоличествоНочей = 0;
19 КонецЕсли;
20 КонецПроцедуры
```

2.9. Аналогично создаем событие для даты выезда.

```
1 &НаКлиенте
2 Процедура ДатаВыездаПриИзменении(Элемент)
3 Если ЗначениеЗаполнено(Этаформа.Объект.ДатаЗаезда) и
   ↳ ЗначениеЗаполнено(Этаформа.Объект.ДатаВыезда) тогда
4 ситуации, когда дата заезда больше даты выезда:
5 Если Этаформа.Объект.ДатаЗаезда > Этаформа.Объект.ДатаВыезда тогда
6 Сообщить("Дата заезда не может быть больше даты выезда!");
7 Этаформа.Объект.КоличествоДней = 0;
8 Этаформа.Объект.КоличествоНочей = 0;
9 Иначе
10 Этаформа.Объект.КоличествоДней = (Этаформа.Объект.ДатаВыезда -
   ↳ Этаформа.Объект.ДатаЗаезда) / (24 * 60 * 60) + 1;
11 Этаформа.Объект.КоличествоНочей = (Этаформа.Объект.ДатаВыезда -
   ↳ Этаформа.Объект.ДатаЗаезда) / (24 * 60 * 60);
12 КонецЕсли;
13 Иначе
14 Этаформа.Объект.КоличествоДней = 0;
15 Этаформа.Объект.КоличествоНочей = 0;
16 КонецЕсли;
17 КонецПроцедуры
```

2.10. Раз уж мы взялись изменять интерфейс данного объекта, то можно «поиграться» с формой. Например, сделать так, чтобы поля «Количество дней» и «Количество ночей» нельзя было изменить в пользовательском режиме (это делается в свойствах элемента), расставить их нужным образом и т. д. Ниже приводим пример интерфейса одного из участников Олимпиады.

☆ Назад в будущее (Туры) 🔗 ⋮ □ ×

Записать и закрыть

Наименование: Код:

Отель:

Дата заезда: Дата выезда:

Количество дней: 2 Количество ночей: 1

Вид питания:

Стоимость тура: Валюта:

Описание:

Тур по нефтяной столице: экскурсия по научно-технологическому центру в сочетании с повествованием об истории города.

Видим, что участник добавил еще один реквизит — валюта. Почему нет? Любые улучшения всегда приветствуются и награждаются бонусами.

3. Создать справочник «Клиенты» с перечисленной в задаче структурой.

Справочник Клиенты

Основные
Подсистемы
Функциональные опции
Иерархия
Владельцы

▶ Данные
Нумерация
Формы
Поле ввода
Команды
Макеты
Ввод на основании

Длина кода
Длина наименования

Тип кода: Число Строка

Основной вид: В ви. В ви.

Реквизиты

- КонтактноеЛицо
- ВидКлиента
- Телефон

3.1. Для реквизита «КонтактноеЛицо» тип данных — ссылка на справочник «КонтактныеЛица».

3.2. Тип данных для реквизита «ВидКлиента» лучше реализовать через перечисления.

Перечисление ВидКлиента

Основные
Подсистемы
Функциональные опции

▶ Данные
Формы
Поле ввода
Команды
Макеты
Прочее

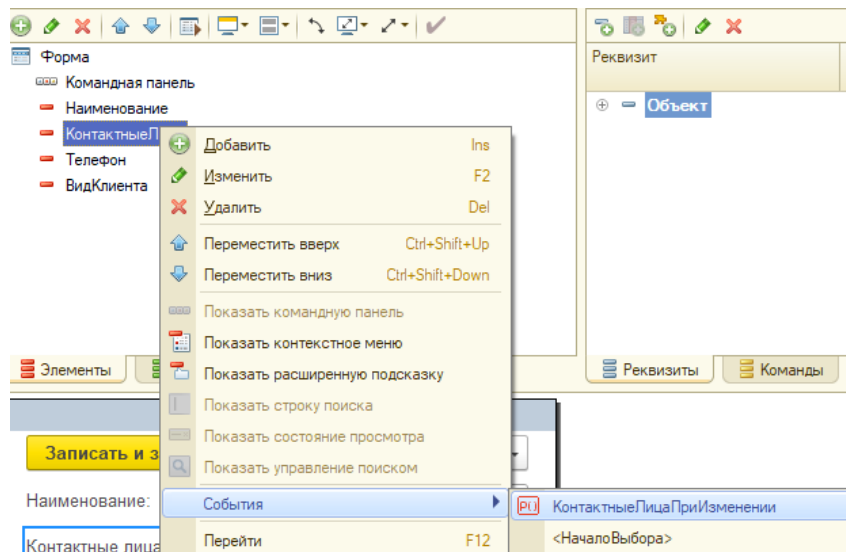
Значения перечисления:

Значения

- ФизическоеЛицо
- ЮридическоеЛицо

3.3. При выборе контактного лица необходимо реализовать алгоритм автоматического заполнения реквизита «Телефон». Для этого нам также надо

создать форму элемента данного справочника и определить событие — это событие при изменении значения поля «КонтактноеЛицо».



3.4. Ниже приводим один из вариантов реализации алгоритма.

```

&НаСервере
[
  Процедура КонтактныеЛицаПриИзмененииНаСервере ()
  // Вставить содержимое обработчика.
  Объект.Телефон = Объект.КонтактныеЛица.Телефон;
  Объект.Наименование = Объект.КонтактныеЛица.Наименование;
  КонецПроцедуры
]

&НаКлиенте
[
  Процедура КонтактныеЛицаПриИзменении (Элемент)
  КонтактныеЛицаПриИзмененииНаСервере ();
  КонецПроцедуры
]
  
```

3.5. Далее также можно поработать над внешним видом формы — например, реализовать поле «Вид клиента» в виде переключателя. Разделить на группы — например, отдельно выделить «Наименование» и «Вид клиента», и отдельно «Контактная информация». Ниже приведен вариант формы от одного из участников Олимпиады.

Задача IV.3. Формирование туров. Регистрация заказов (35 баллов)

Условие

Задача для Гриши №3

Клиент заинтересовался конкретным туром, выразил намерение в будущем его приобрести. Необходимо зафиксировать этот интерес в приложении, чтобы потом работать с клиентом и продать ему тур.

Техническое задание

Интерес клиента к конкретному туру — это событие в жизни компании. Его можно назвать «Заказ тура». Заказ должен содержать следующую информацию:

- «Клиент» — ссылка на справочник «Клиенты».
- «Вид оплаты» — одно из двух вариантов: предоплата или кредит.
- Информация о продаваемых турах в виде таблицы, состоящей из следующих колонок:
 - Тур — ссылка на справочник с турами.
 - Цена — стоимость тура, тип «Число», 2 знака после запятой. Она подставляется при выборе конкретного тура. Но может корректироваться и вручную.
 - Количество человек — тип «Число», количество знаков после запятой — 0.
 - Стоимость — тип «Число», 2 знака после запятой. Рассчитывается автоматически. Пересчитывается при изменении цены или количества человек.
 - «Общая стоимость заказа» — тип «Число», 2 знака после запятой. Рассчитывается как сумма стоимости всех выбранных заказчиком туров.

Критерии оценивания

Критерий	Базовый балл	Прибавка	Макс. балл
Созданный документ «Заказ туров»	1		1
Созданный реквизит «Клиент» со ссылкой на справочник «Клиенты»	1	+1 со ссылкой на справочник «Клиенты»	2
Реквизит «Вид оплаты»	1	+1 Если реализован в виде перечисления или в виде справочника	2
Табличная часть «Туры» с колонками «Название тура», «Цена», «Количество», «Стоимость»	5	+1 Если «Название тура» ссылка на справочник с турами +1 если есть попытка подстановки цены из справочника +4 если подстановка работает корректно	10

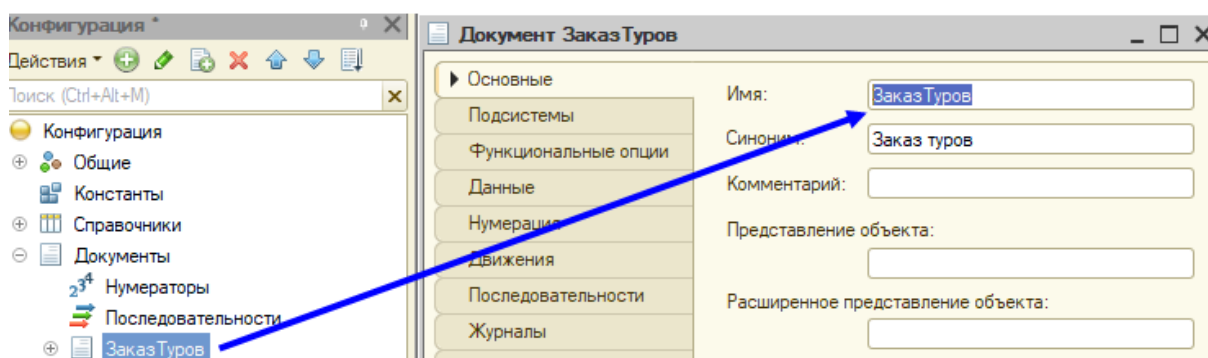
Критерий	Базовый балл	Прибавка	Макс. балл
Расчет стоимости тура (цена · количество)	3 (если есть попытка создания кода)	+1 если механизм работает, но ошибка в формуле +4 если механизм работает корректно	7
Расчет общей стоимости заказа	1 (если создан реквизит или итоговое поле в таблице)	+1 если есть попытка создания кода +2 если механизм работает, но ошибка в формуле +6 если механизм работает верно	7
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		до +6, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	6

Если в сопроводительной записке у участника прописаны дополнительные улучшения, и они работают в программе, можно добавить до 3-х бонусных баллов.

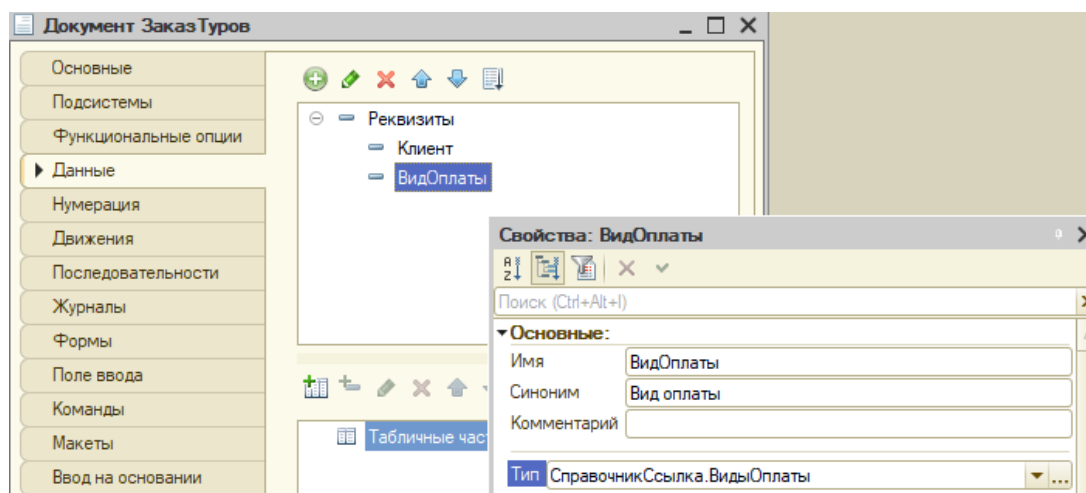
Решение

Основные этапы решения задачи представлены на базе платформы «1С:Предприятие 8».

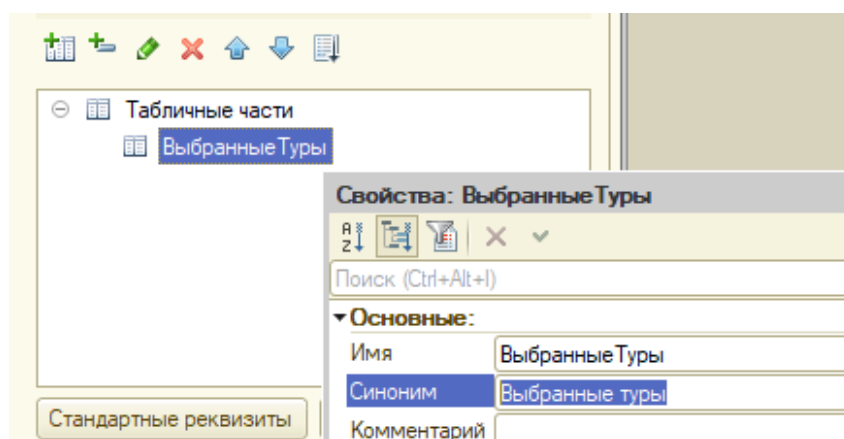
- Здесь следует обратить внимание на то, что заказ клиента, выражающий заинтересованность в приобретении какого-либо тура — это событие в жизни компании. Появился клиент, который может купить тур! Событие происходит в определенную дату. Вот он звонит сегодня и интересуется туром. Для регистрации событий в «1С:Предприятие 8» существует специальный объект под названием «Документ». Именно им мы и воспользуемся. Создать новый документ «Заказ туров» можно в дереве конфигурации.



- Дальше надо заполнить вкладку «Данные» по описанной в условии задачи структуре.
 - Создать два реквизита документа:
 - «Клиент» — ссылка на справочник «Клиенты».
 - Тип реквизита «Вид оплаты» — ссылка на справочник «Виды оплаты», который надо создать дополнительно.

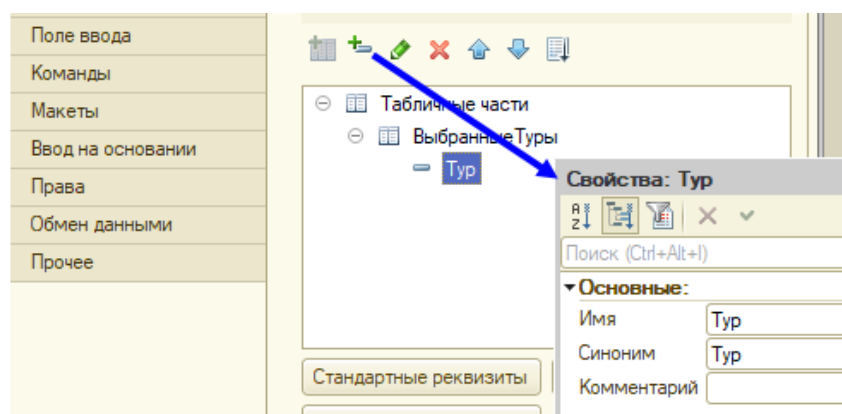


2.2. Информация о турах, интересующая клиента, будет храниться в табличной части документа. Это удобно тем, что клиента может заинтересовать сразу несколько туров. Поэтому здесь же, на закладке «Данные» нам необходимо создать табличную часть «Выбранные туры».



2.3. Она состоит из колонок (реквизитов табличной части):

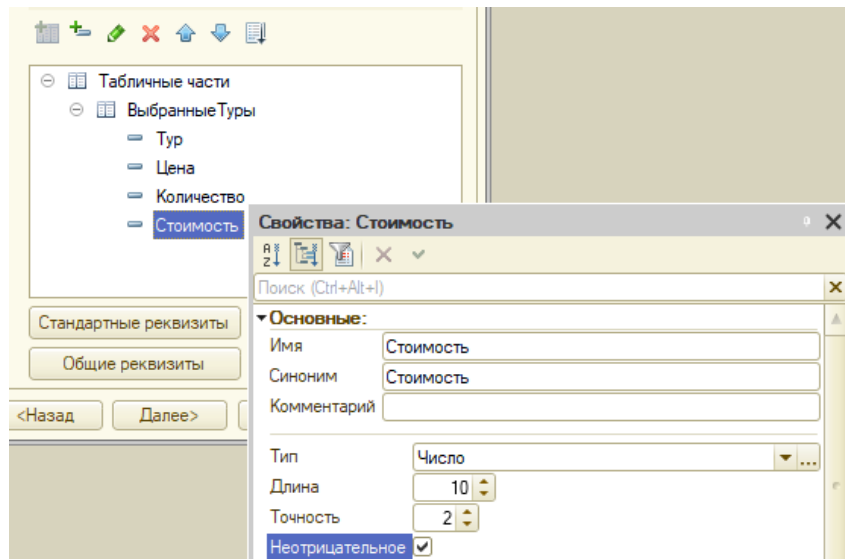
2.3.1. «Тур» — ссылка на справочник «Туры».



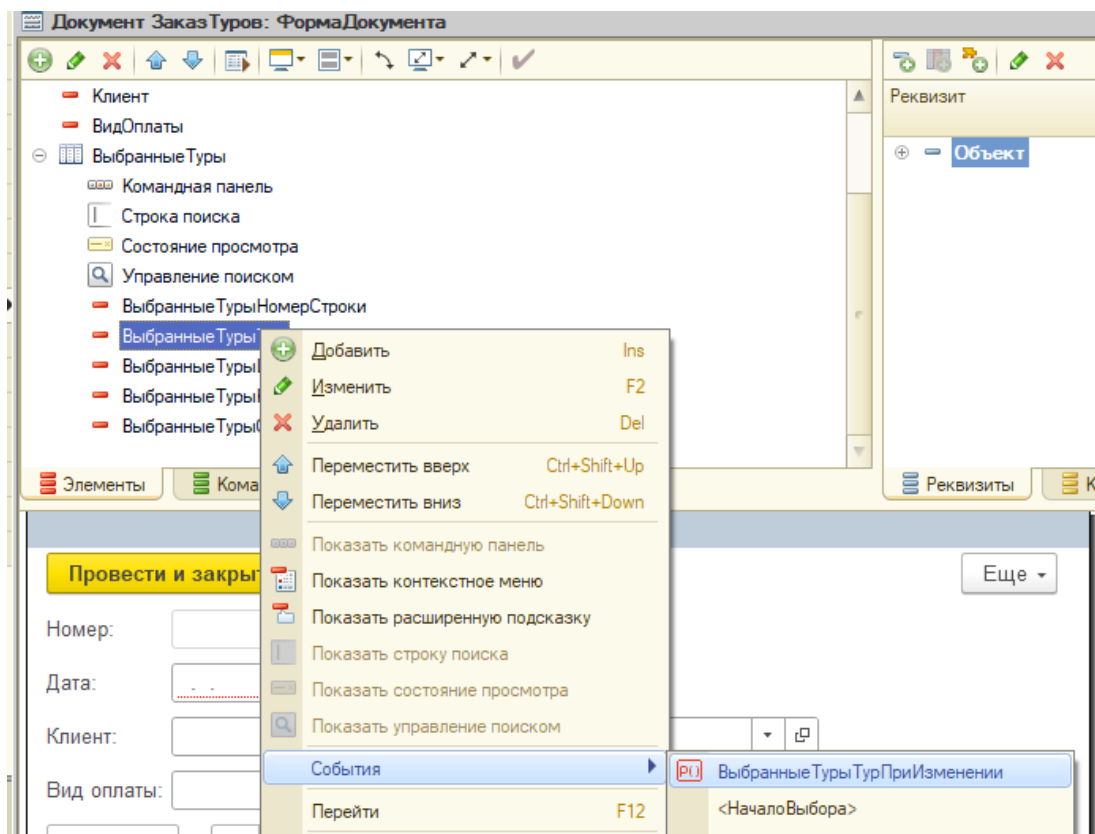
2.3.2. «Цена» — цены можно указывать вручную, можно подставлять из справочника «Туры», тип «Число».

2.3.3. Количество — тип «Число».

2.3.4. Стоимость — тип «Число».



2.4. На следующем этапе необходимо разработать алгоритм подстановки цены из справочника «Туры». Для этого нужно создать форму документа (на закладке «Формы», аналогично форме справочника, см. решение предыдущей задачи) и здесь, в модуле формы, прописать алгоритмы для соответствующего события. Цена подставляется при выборе определенного тура. Значит, нужное нам событие — при изменении данных в колонке «Туры».



2.5. Система «перебросит» вас в «Модуль формы».

```
Документ ЗаказТуров: ФормаДокумента
&НаКлиенте
  Процедура ВыбранныеТурыТурПриИзменении(Элемент)
    // Вставить содержимое обработчика.
  КонецПроцедуры
```

2.6. Снова представим, что происходит в пользовательском режиме — менеджер, добавил новую строку таблицы — это строка активна, так называемая «Текущая строка». В этой текущей строке в колонке «Тур» он выбрал конкретный тур — элемент справочника «Туры», в котором указана его стоимость. Вот к этому элементу справочника мы должны обратиться, чтобы «достать» цену и поместить ее в нужную колонку. Таким образом, программный код будет выглядеть так.

```
1 &НаКлиенте
2 Процедура ВыбранныеТурыТурПриИзменении(Элемент)
3   //Обращаемся к текущей строке
4   ТекСтр = Элементы.ВыбранныеТуры.ТекущиеДанные;
5   //Получаем цену при помощи специальной функции и записываем в колонку
   ↳ "Цена" текущей строки
6   Элементы.ВыбранныеТуры.ТекущиеДанные.Цена = ПолучитьСтоимость(ТекСтр.Тур);
7   КонецПроцедуры
```

2.7. «ПолучитьСтоимость(<Параметр>)» — это специальная функция, которая получает стоимость выбранного в текущей строке тура. Т. к. функция предназначена для получения данных, то для ее описания необходимо указать директиву — «НаСервере». Код функции будет выглядеть так.

```
1 //Тур - параметр функции (туда будет передаваться значение выбранного тура в
   ↳ текущей строке)
2 &НаСервере
3 Функция ПолучитьСтоимость(Тур)
4   //Обращаемся к значению реквизита "СтоимостьТура" выбранного элемента
   ↳ справочника "Туры"
5   Возврат Тур.СтоимостьТура;
6   Конецфункции
```

2.8. Необходимо разработать алгоритм изменения стоимости тура при изменении количества или цены. Здесь тоже будем писать алгоритм для событий элементов формы. Когда будет вестись пересчет стоимости? Очевидно, когда меняется цена или количество. Поэтому нужно создать два события — «При изменении цены» и «При изменении количества».

```
&НаКлиенте
  Процедура ВыбранныеТурыЦенаПриИзменении(Элемент)
    // Вставить содержимое обработчика.
  КонецПроцедуры

&НаКлиенте
  Процедура ВыбранныеТурыКоличествоПриИзменении(Элемент)
    // Вставить содержимое обработчика.
  КонецПроцедуры
```

2.9. Стоимость тура, очевидно, равна цене, умноженной на количество. Поскольку считаться стоимость будет при наступлении более, чем одного события, можно ее расчет вынести в отдельную процедуру. И тогда код будет выглядеть следующим образом.

```
1 &НаКлиенте
2 Процедура ВыбранныеТурыЦенаПриИзменении(Элемент)
3   //Обращение к текущей строке
```

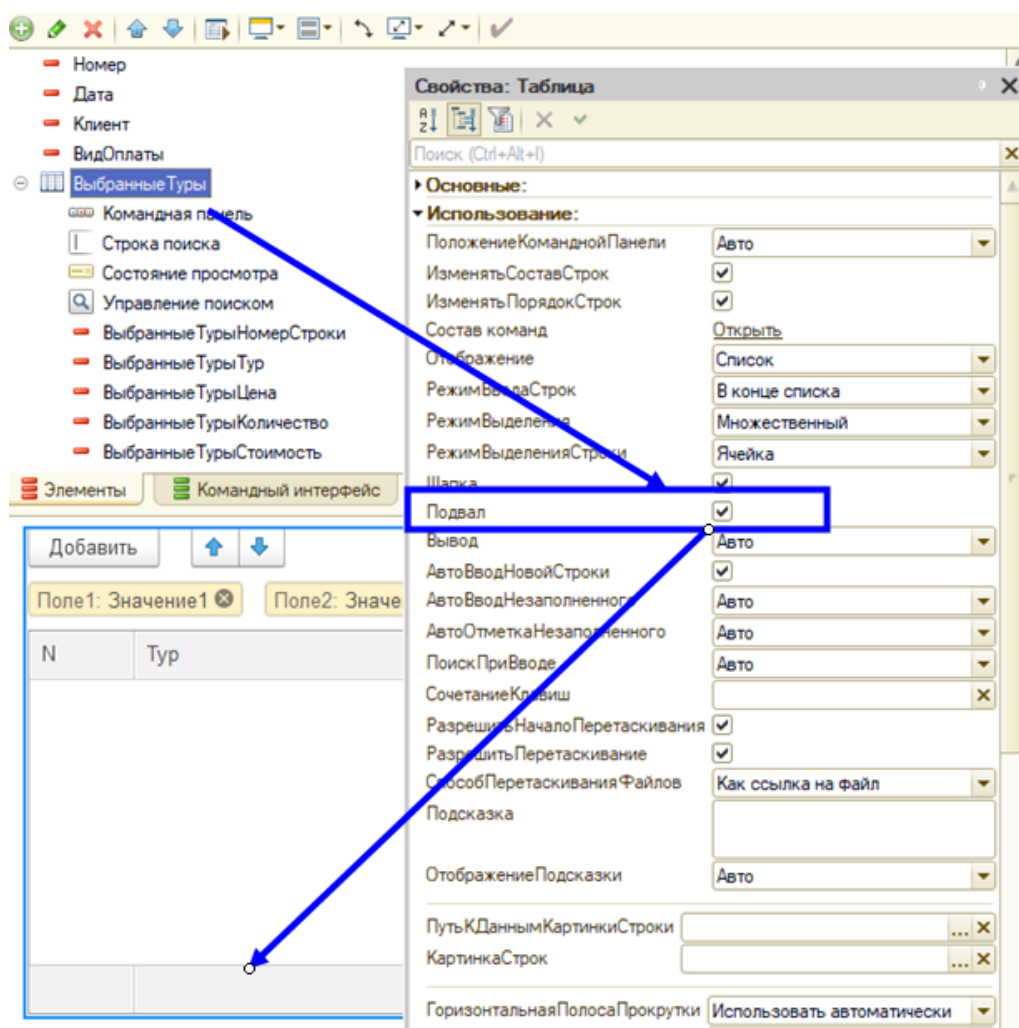
```

4   РасчетСтоимости (Элементы.ВыбранныеТуры.ТекущиеДанные)
5   КонецПроцедуры
6
7   &НаКлиенте
8   Процедура ВыбранныеТурыКоличествоПриИзменении(Элемент)
9       //Обращение к текущей строке
10      РасчетСтоимости (Элементы.ВыбранныеТуры.ТекущиеДанные)
11   КонецПроцедуры
12
13   &НаКлиенте
14   Процедура РасчетСтоимости(ТекущаяСтрока)
15       ТекущаяСтрока.Стоимость = ТекущаяСтрока.Количество*ТекущаяСтрока.Цена;
16   КонецПроцедуры // РасчетСтоимости()

```

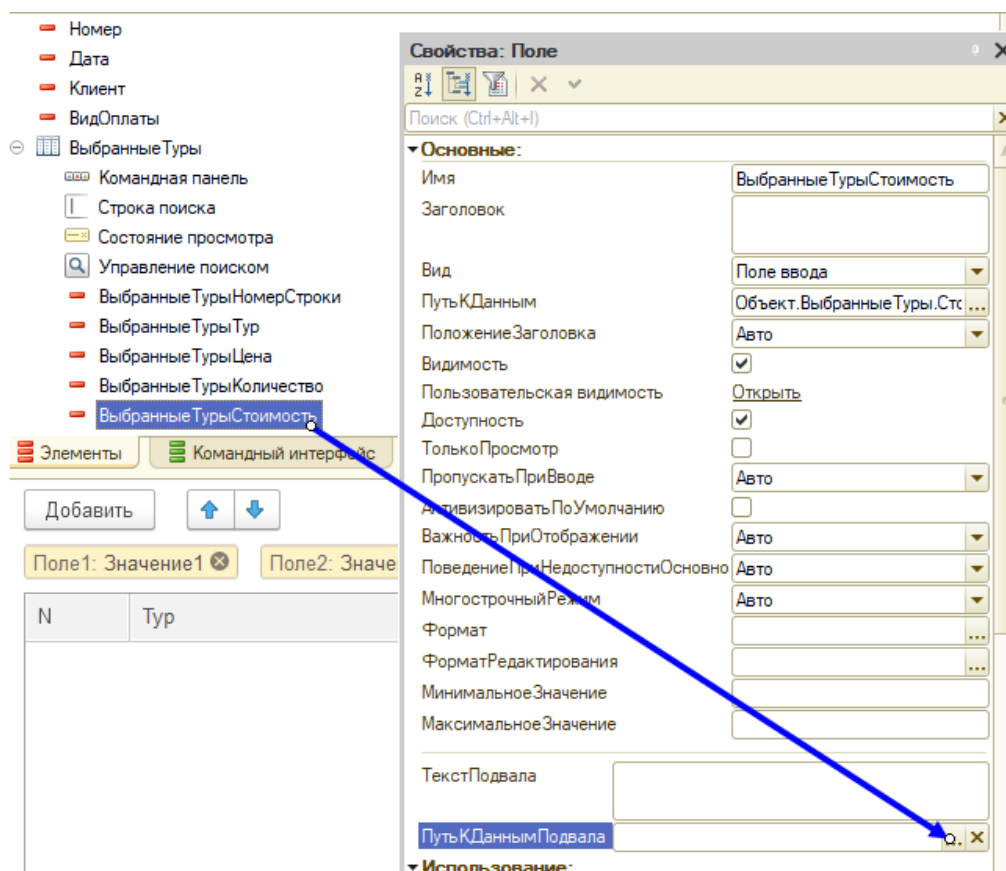
Еще раз обращаем внимание, что приводим всего лишь один из вариантов реализаций.

- 2.10. Далее необходимо внизу таблицы рассчитать общую стоимость заказа. Можно это сделать различными способами. Например, добавить реквизит «ОбщаяСуммаЗаказа» и туда записать общую стоимость. А можно обойтись стандартными средствами, которые уже есть в «1С:Предприятие 8». У табличной части есть такая часть, которая называется «Подвал» — она предназначена для вывода итогов по колонкам таблицы. «Подвал» можно добавить на форму, указав его необходимость в свойствах табличной части.

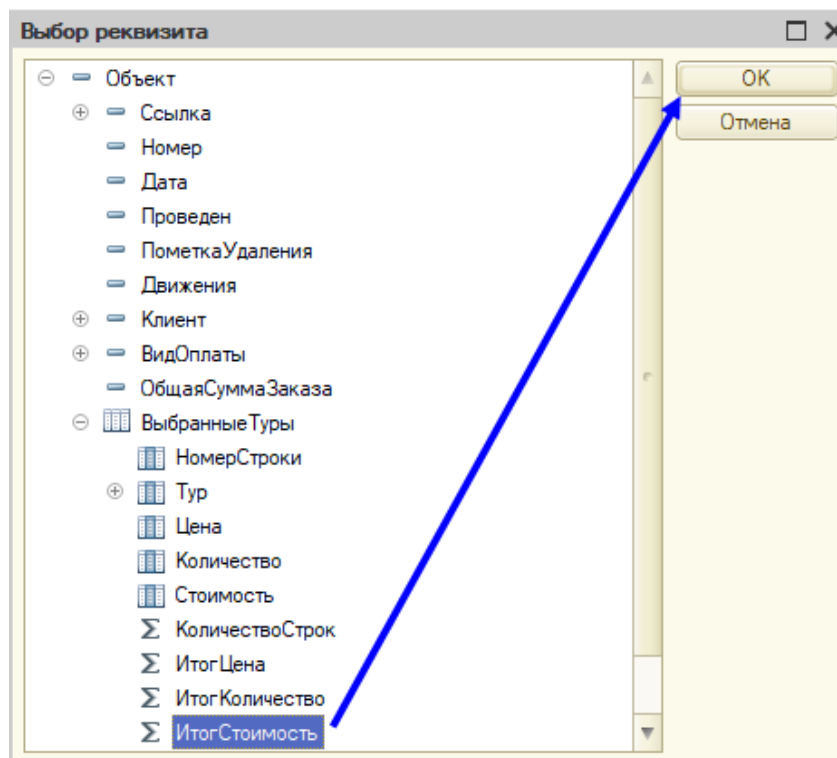


- 2.11. Нас интересует подвал колонки «Стоимость». Открываем свойства этого

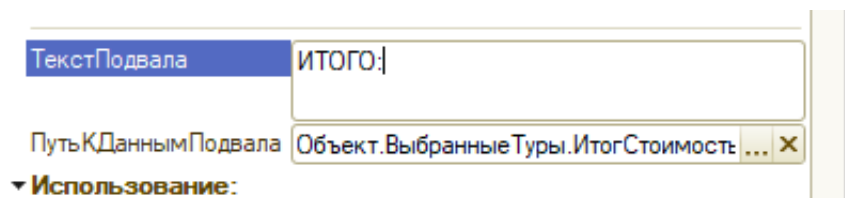
элемента. Нас интересуют свойства, связанные с подвалом, а точнее — это свойство «ПутьКДаннымПодвала».



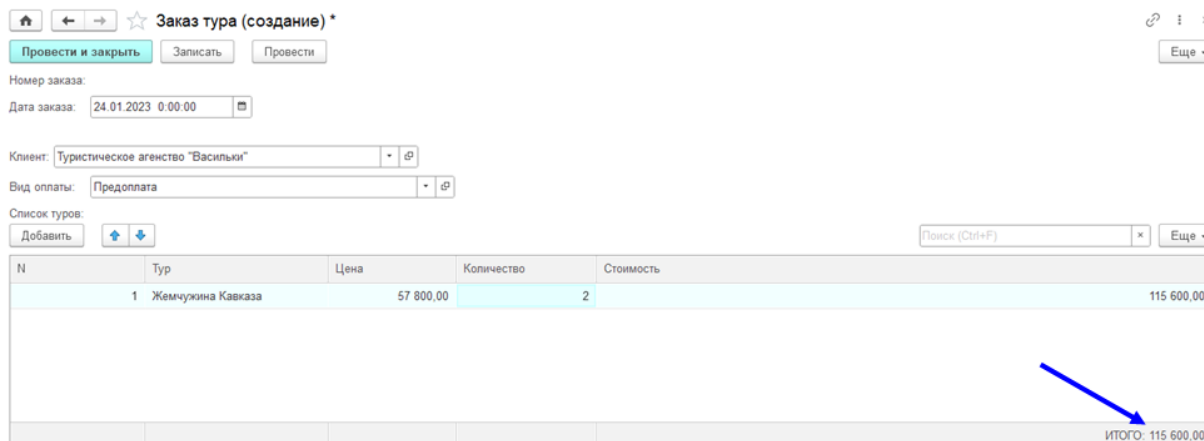
2.12. А здесь есть возможность выбрать «Итог» по колонке «Стоимость».



2.13. Чтобы было понятно, что это общая сумма заказа, можно в свойстве «ТекстПодвала» добавить: «ИТОГО:».



2.14. Ниже приводим вариант реализации одного из участников Олимпиады.



2.15. Также самое время заняться интерфейсом приложения: цветовой гаммой, логотипом, разделением приложения на подсистемы, интерфейсом форм объектов, в т. ч. документа «Заказ клиента», справочника «Туры» и т. д.

2.16. Кроме этого, предусмотрены бонусные баллы за реализацию дополнительных механизмов, например, системы скидок.

Задача IV.4. Регистрация продажи туров (40 баллов)

Условие

Задача для Гриши №4

Интерес клиента может завершиться продажей тура и получением оплаты. Эти два события: продажу и оплату — необходимо фиксировать в программе.

Техническое задание

Зафиксировать событие «Оплата тура» со следующими данными:

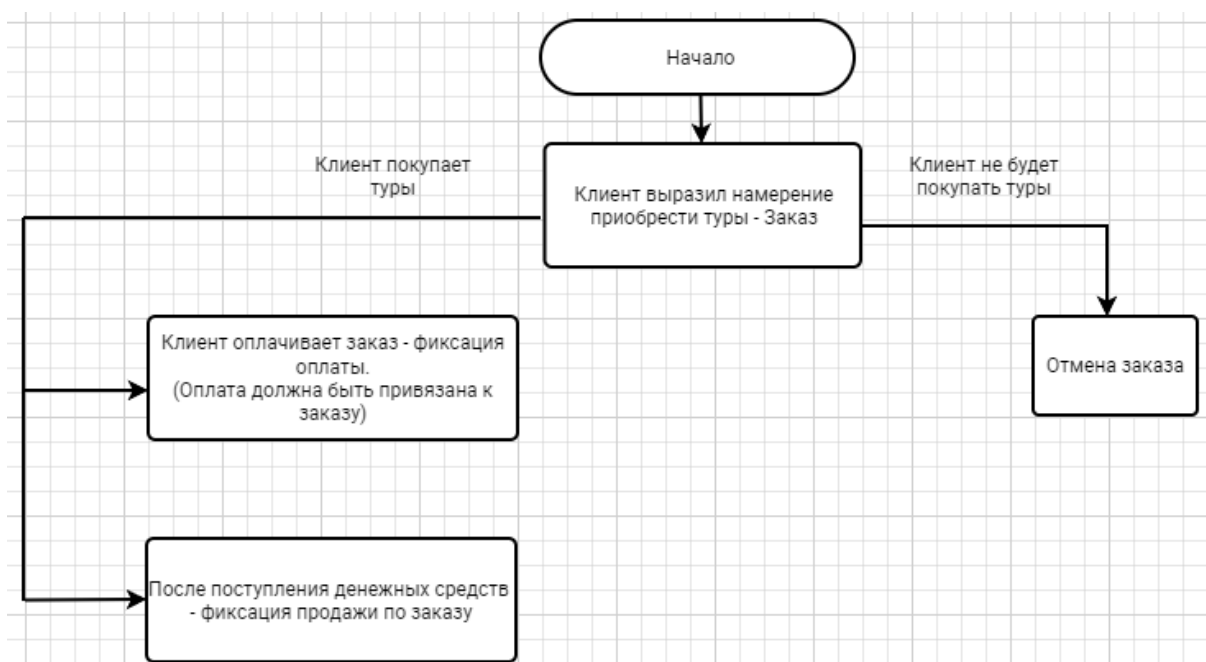
- «Заказ» — ссылка на конкретный заказ клиента.
- Сумма оплаты — копируется из «Заказа».

Зафиксировать событие «Продажа тура»:

- Структура документа «Продажа тура» совпадает со структурой документа «Заказ тура», но добавляется поле «Бронь номеров подтверждена отелем» с вариантами значений «да/нет». Всю совпадающую информацию можно перенести из «Заказа тура» при создании «Продажи тура».

Все три события в системе должны быть связаны между собой.

Т. е. бизнес-процесс продажи туров должен выглядеть следующим образом.



Отмену заказов можно сделать любым способом — либо отдельным документом, либо в виде статусов заказа — «Действует/Завершен(положительно)/Отменен».

Но в любом случае хотелось бы видеть, какие заказы отменены, а какие действуют.

Пожелание: в системе стало достаточно много различных объектов. Для того чтобы было удобно в них ориентироваться, необходимо их сгруппировать, разбить на подсистемы и так далее. Также хочется, чтобы решение было красивым — продумать фон, пиктограммы и т. д. Если ты, Гриша, еще этим не занимался, конечно. Если уже начал — продолжай, без вознаграждения не останешься точно!

Критерии оценивания

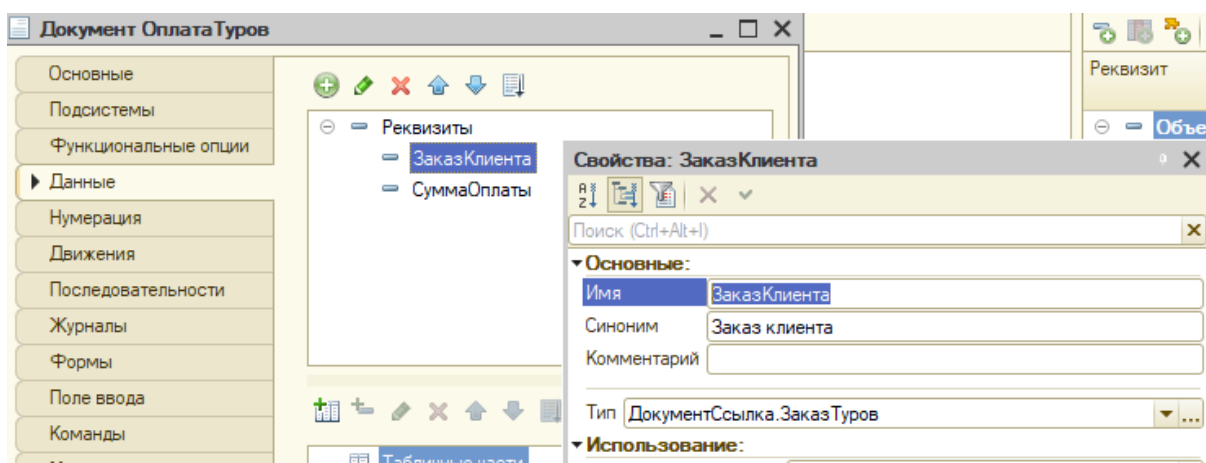
Критерий	Базовый балл	Прибавка	Макс. балл
Созданный документ «Продажа тура»	1		1
Документ создается на основании заказа или заполняется данными из заказа	1 если есть попытка реализации механизма, пусть с ошибками	+4 на основании заполняются не все данные +12 если все работает корректно	13
Созданный документ «Оплата»	1		1
Документ «Оплата» можно ввести на основании заказа	1 если есть попытка реализации механизма, пусть с ошибками	+5 если в документ оплата копируется сумма из заказа, но нет поля с данными о заказе (реквизит, ссылка) или сведения о заказе есть, но не подставляется оплата +12 если поля «Заказ» и «Оплата» заполняются автоматически.	13

Критерий	Базовый балл	Прибавка	Макс. балл
Реализация статусов заказов и фиксация отмены заказов			4
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовков приложения, какие-либо вставки картинок, форма и т. д.		До +6, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	8

Если в сопроводительной записке у участника прописаны дополнительные улучшения, и они работают — в этом случае слушателю можно добавить до 5-ти бонусных баллов.

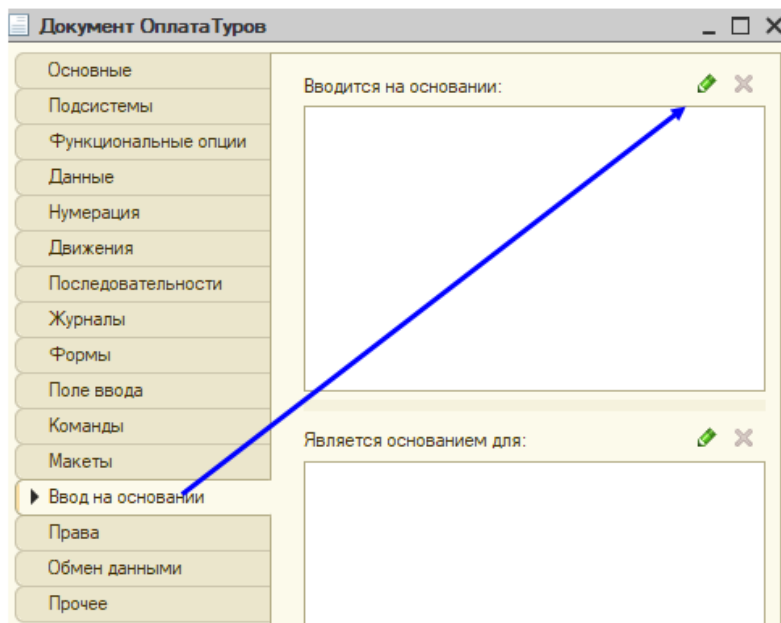
Решение

1. Создать документ «Оплата тура» по описанной в условии задачи структуре.

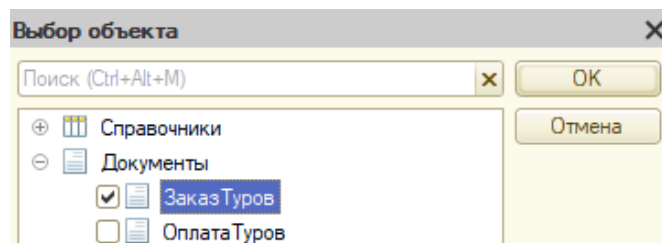


- 1.1. Далее необходимо реализовать взаимосвязь документов «Заказ клиента» и документа оплаты. Т. е. клиент выразил намерение приобрести тур и оплачивает его. В жизни компании происходит новое событие, которое мы отражаем в программе в виде документа «Оплата туров». Документ можно «привязать к заказу» — сделать так, чтобы он вводился на основании заказа. Для этого можно воспользоваться специальным механизмом — «Ввод на основании».

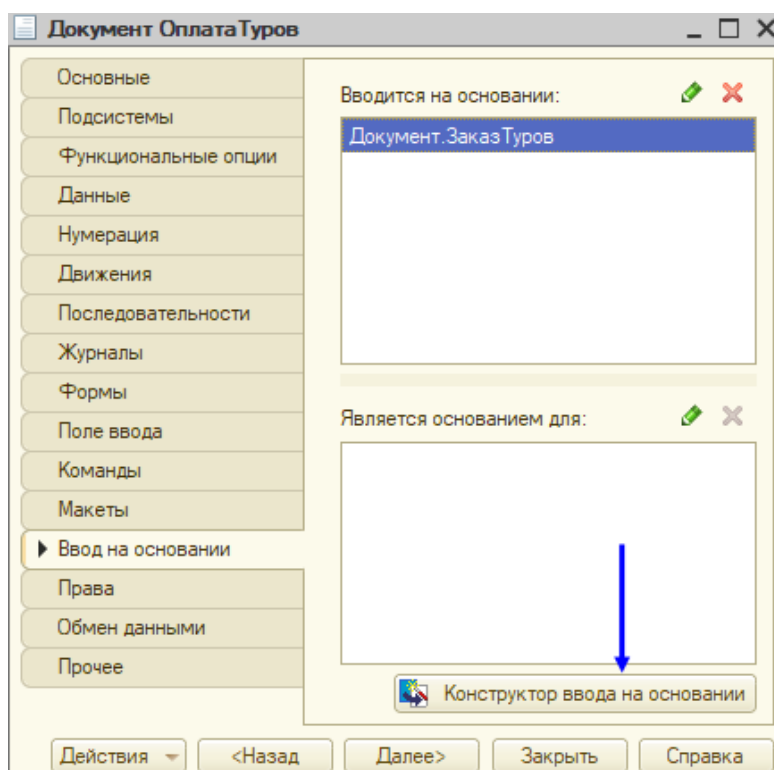
При реализации такого механизма можно сделать так, чтобы данные из объекта одного вида копировались в объект другого вида. Мы сделаем так, чтобы из документа «Заказ клиента» копировался сам заказ в виде ссылки на него и общая сумма заказа. В окне редактирования объекта «Оплата туров» есть специальная вкладка «Ввод на основании». Нужно на нее перейти. Поскольку именно «Оплата туров» вводится на основании заказа, то на этой вкладке нам понадобится соответствующий раздел.



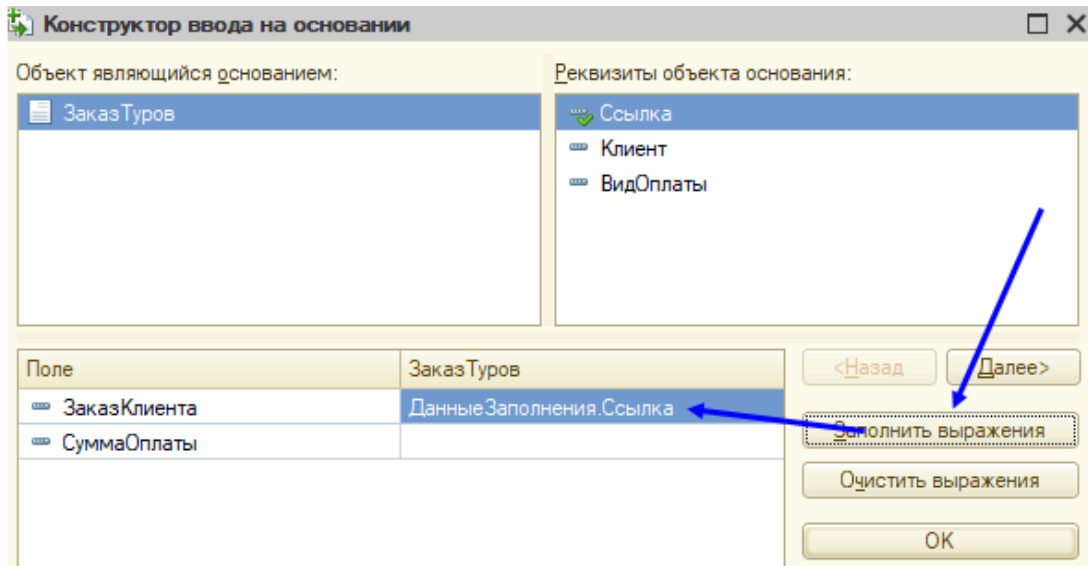
1.2. Здесь выбираем нужный документ.



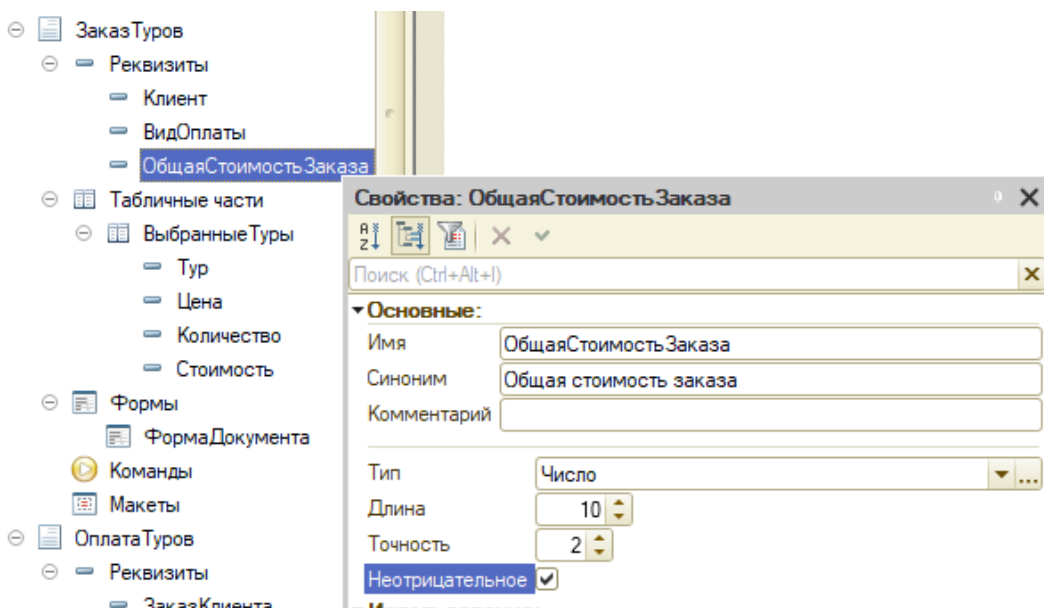
1.3. Далее нам необходимо описать, какие действия будут осуществляться при вводе на основании — т. е. задать алгоритм. Это можно сделать без задания кода при помощи специального механизма «Конструктор движения». Открывается он на этой же вкладке по соответствующей кнопке.



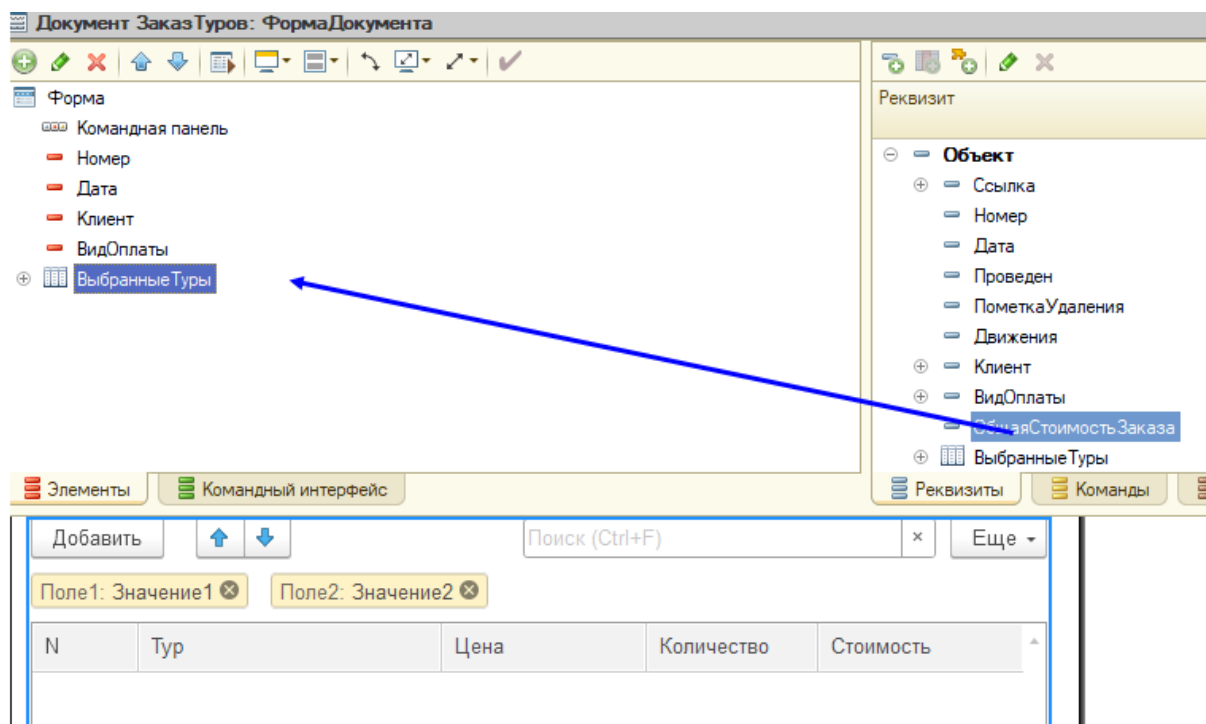
- 1.4. В открывшемся окне необходимо заполнить данные для заполнения. Можно нажать на кнопку «Заполнить выражения» и система сама сопоставит реквизиты двух видов документа.



- 1.5. Сумму оплаты в прошлой задаче мы отображали в «Подвале» таблицы. Оттуда данные взять не получится, т. к. это данные формы. Необходимо сумму записывать в дополнительный реквизит документа «Заказ клиента» — «ОбщаяСтоимостьЗаказа». Его можно не отображать на форме документа, но он должен быть. Нажмем на кнопку «ОК» в «Конструкторе ввода» на основании, система нас перебросит в «Модуль объекта», и мы увидим созданный алгоритм, который мы не набирали сами. Его создала система. Пока отвлечемся от него, перейдем к «Заказу клиента» и создадим там новый реквизит «ОбщаяСуммаЗаказа», тип «Число», точность — 2 знака после запятой.



- 1.6. Откроем форму «Заказа клиента». Т. к. реквизит был создан позднее, чем форма, по умолчанию он не отображается на самой форме. По желанию его можно перенести.



1.7. Перейдем на закладку «Модуль» данной формы. И опишем процедуру получения итоговой суммы заказа с директивой «&НаСервере».

```

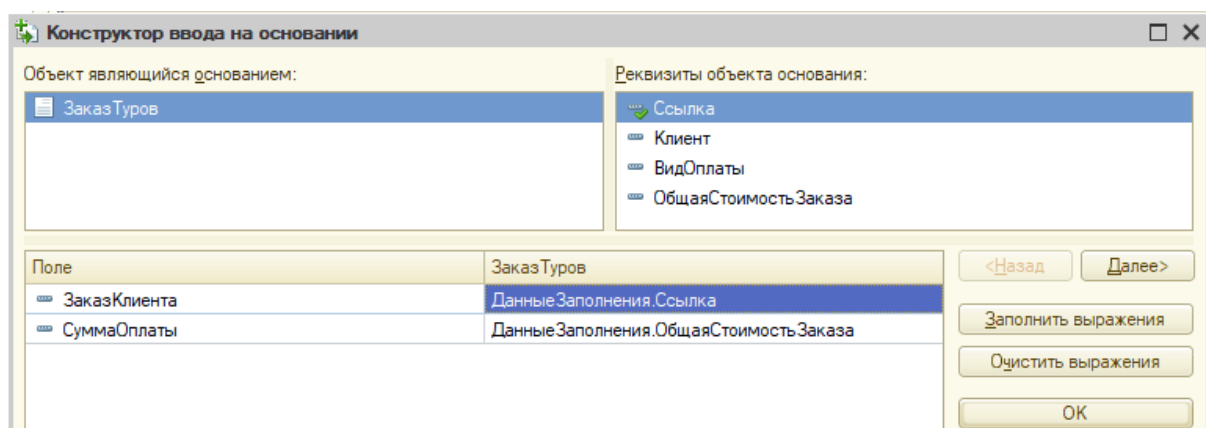
1 &НаСервере
2 Процедура РассчитатьОбщуюСуммуЗаказа()
3 //Обращаемся к реквизиту "ОбщаяСтоимостьЗаказа"
4 //К табличной части "ВыбранныеТуры" применяем имеющийся в платформе метод
  ↪ "Итог(<Колонка>)"
5 //Описание метода может найти в Синтакс-помощнике
6 Объект.ОбщаяСтоимостьЗаказа=Объект.ВыбранныеТуры.Итог ("Стоимость")
7 КонецПроцедуры
  
```

1.8. Эту процедуру нужно вызвать при расчете стоимости выбранного тура.

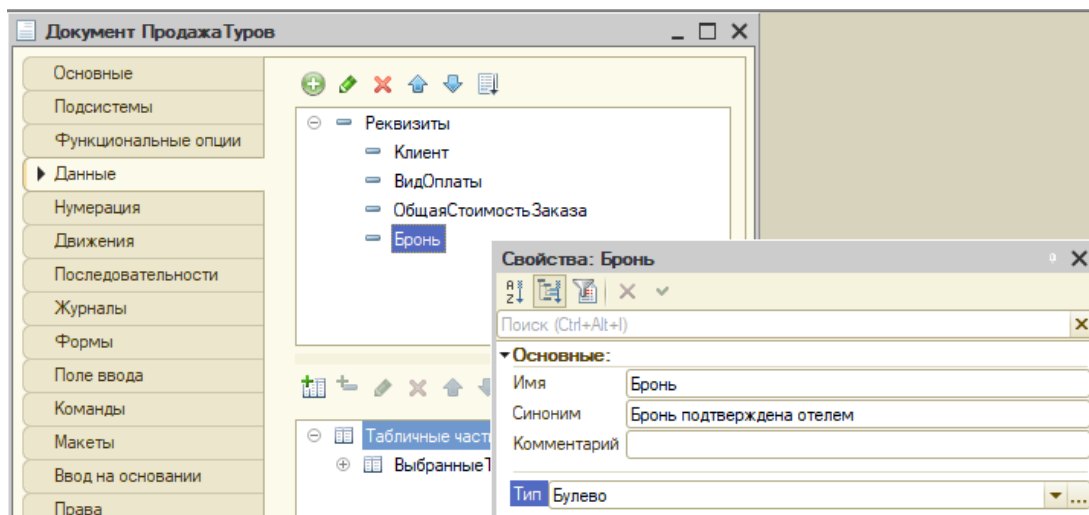
```

1 &НаКлиенте
2 Процедура РасчетСтоимости(ТекущаяСтрока)
3 ТекущаяСтрока.Стоимость = ТекущаяСтрока.Количество*ТекущаяСтрока.Цена;
4 РассчитатьОбщуюСуммуЗаказа()
5 КонецПроцедуры
  
```

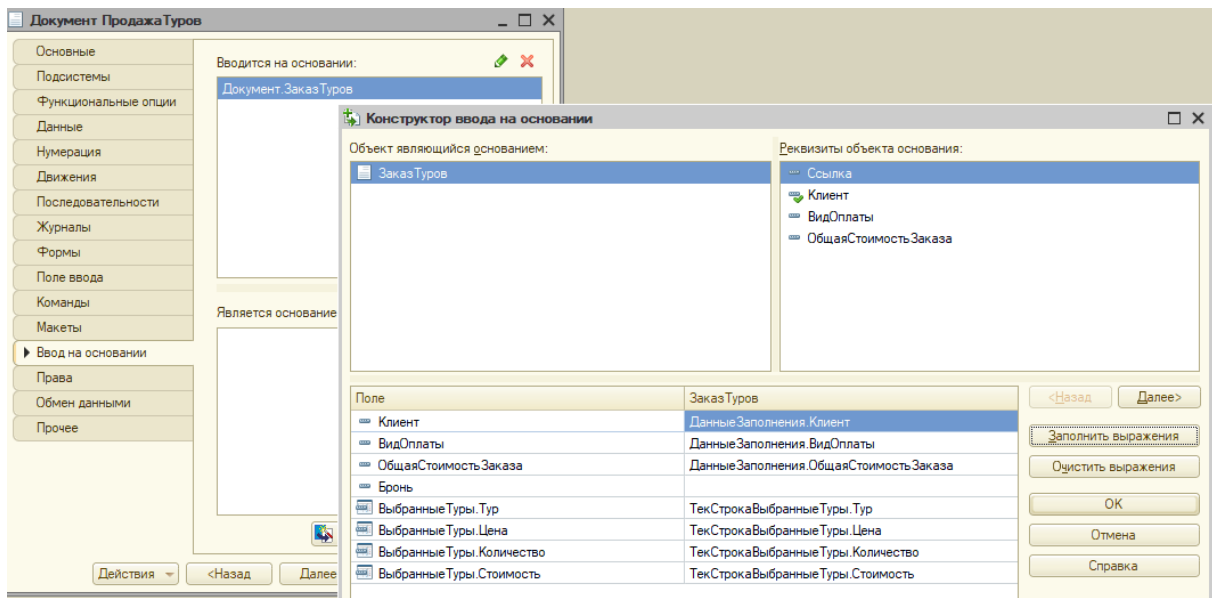
1.9. Теперь возвращаемся к документу оплаты — а точнее к «Конструктору ввода» на основании. В открывшемся окне для поля «СуммаОплаты» указываем, что надо взять значение из реквизита «ОбщаяСуммаЗаказа», нажимаем «ОК».



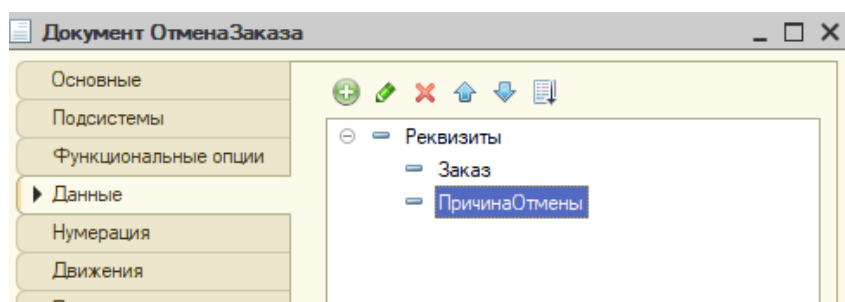
- 1.10. Проверяем, работает ли механизм в режиме «1С:Предприятие 8».
2. Переходим к созданию документа «Продажа тура». Здесь следует учесть, что он вводится на основании документа «Заказ тура» и необходимо также реализовать механизм ввода на основании, чтобы вся общая информация была скопирована из «Заказа» в «Документ продажи». Поскольку структура документа продажи будет совпадать с документом заказа, то можно создать документ копированием заказа клиента, просто переименовав его.
- 2.1. Добавляем в документ продажи тура дополнительный реквизит «Бронь» типа «Булево».



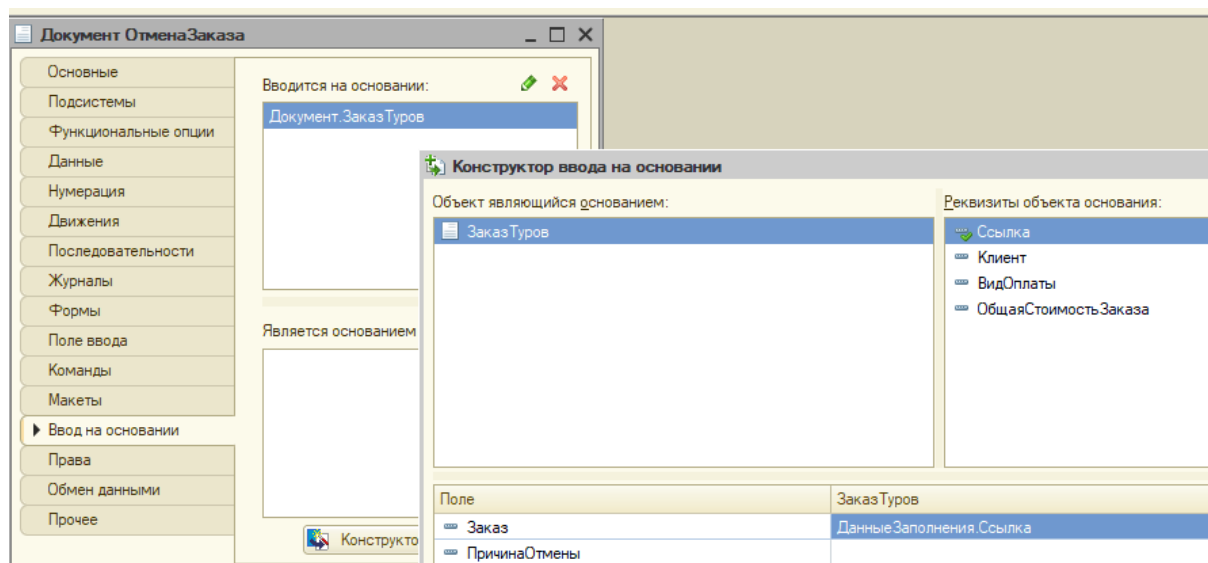
- 2.2. Переходим к описанию механизма ввода на основании. Должно получиться вот так (после нажатия на кнопку «Заполнить выражения» в «Конструкторе ввода на основании» на основании).



3. Осталось реализовать «Отмену заказов». Можно ее реализовать в виде статусов, можно отдельным документом, связанным с заказом. Мы выберем второй вариант.
- 3.1. Создадим документ «ОтменаЗаказа» с двумя реквизитами: «Заказ» (ссылка на Заказ клиента) и «ПричинаОтмены» (строка, многострочный режим).



3.2. Причина отмены, очевидно, будет заполняться менеджером вручную, а заказ — при создании документа отмены заказа. Поэтому необходимо описать механизм ввода на основании (соответствующая закладка окна редактирования объекта).



4. Также необходимо поработать над интерфейсом как самого приложения, так и отдельных объектов, а также получить бонусы за дополнительные улучшения и механизмы.

Задача IV.5. Создание отчетов *Создание отчетов (20 баллов)*

Условие

Задача для Гриши №5

Очень хочется получить сводную информацию о том, сколько было заказано туров, а сколько продано. Также хотелось бы знать, сколько денег есть на счету компании.

Техническое задание

Для анализа заказов и продаж туров необходимо отображать данные в виде таблицы приблизительно такого вида.

Период

Тур	Заказано	Продано

Также хочется видеть, сколько денег сейчас на счету.

Количество денежных средств

Остаток на счете	<input type="text"/>
------------------	----------------------

Критерии оценивания

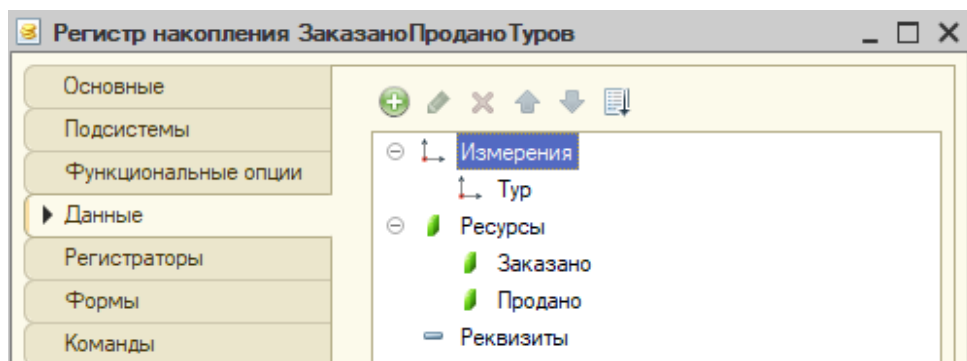
Критерий	Базовый балл	Прибавка	Макс. балл
Созданный отчет о заказах и продажах	3	+5 данные корректны +7 данные корректны и можно посмотреть данные о продажах за период	10
Созданный отчет об остатках ДС	3	+5 данные в отчете корректны	8
Интерфейс		+7 если он реализован	7

+1 б. за каждое улучшение в отчетах: фильтры, диаграмма как вариант отчета и т. д.

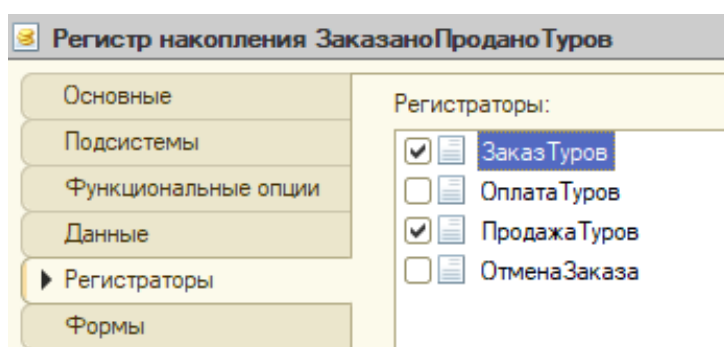
Решение

Для того чтобы данные из разных источников удобно было выводить в отчет, следует поместить их в специальные таблицы — регистры. В данном случае потребуются регистры накопления.

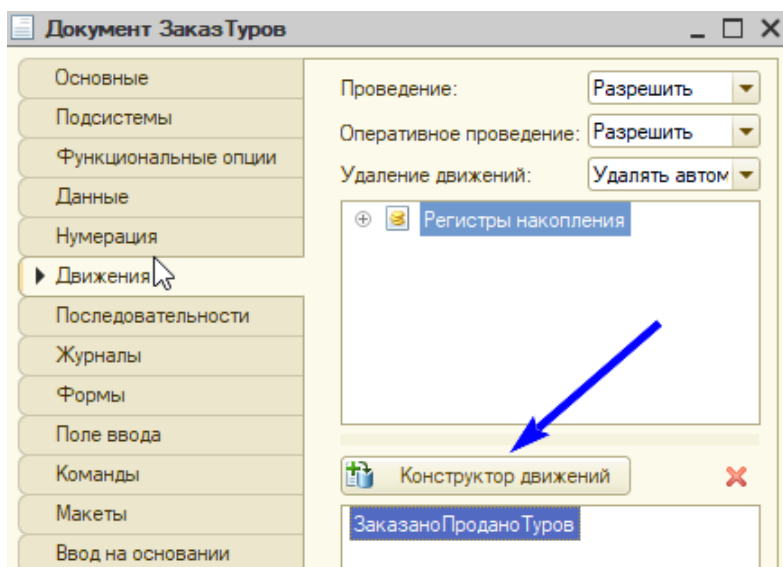
1. Создаем регистр накопления под названием «ЗаказаноПроданоТуров». У регистра есть измерения (то, в разрезе чего мы собираем данные) и ресурсы — непосредственно сами данные. В нашем случае измерением будет «Тур» (ссылка на справочник «Туры»), а ресурсами — количество заказанных и проданных туров (тип — число).



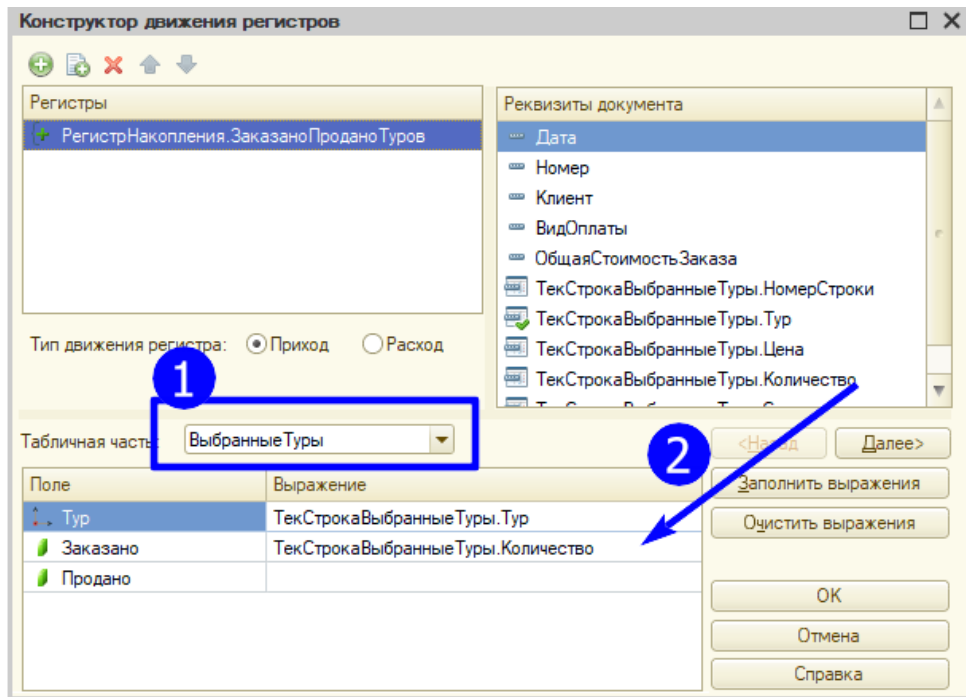
1.1. Регистры накопления не заполняются вручную. Они заполняются данными из документов. Очевидно, что в нашем случае регистраторами будут являться документы заказов и продаж. Перейдем на закладку «Регистраторы» и укажем их.



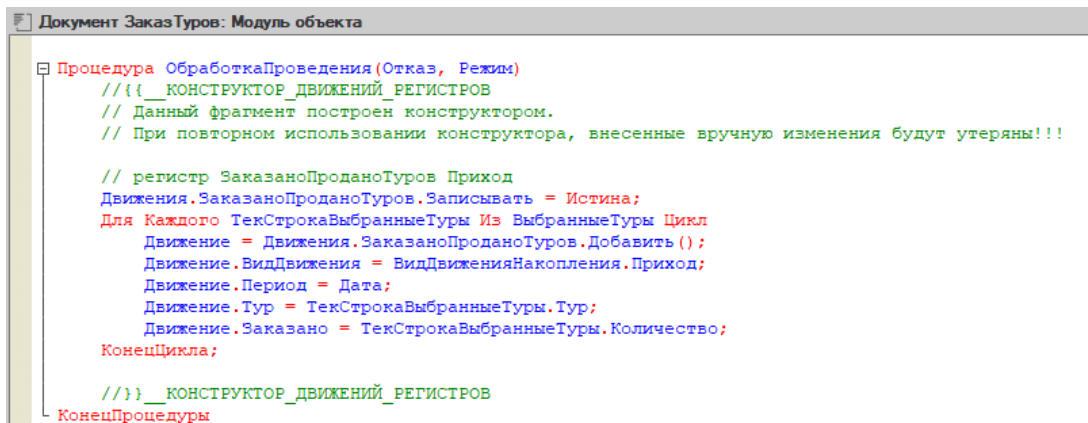
1.2. Теперь надо описать правила записи данных в регистр накопления. Поскольку у нас два документа являются регистраторами, надо описать правило записи для каждого документа. Откроем окно редактирования документа «Заказ клиента» и перейдем на закладку «Движения». Здесь необходимо перейти к «Конструктору движений».



1.3. В открывшемся окне нужно описать правило записи данных. Это будет выглядеть так.

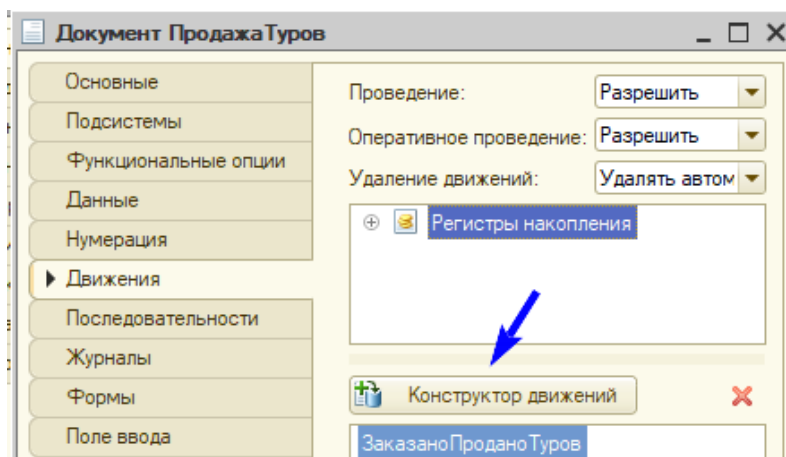


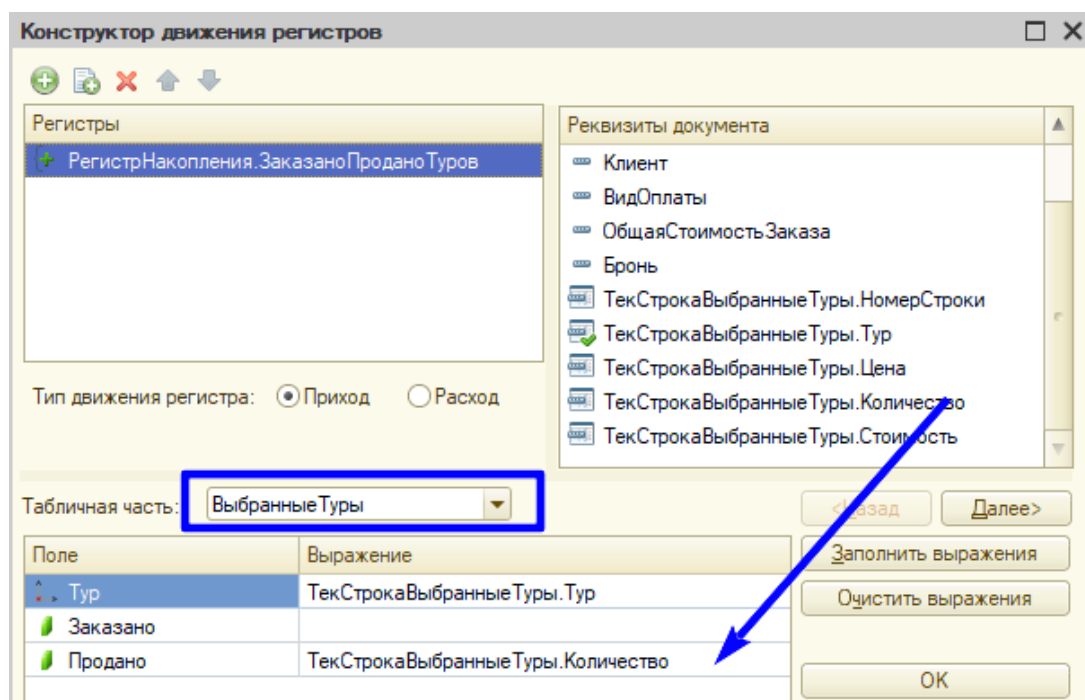
1.4. После нажатия на кнопку «ОК» система сама создаст алгоритм обработки проведения документа.



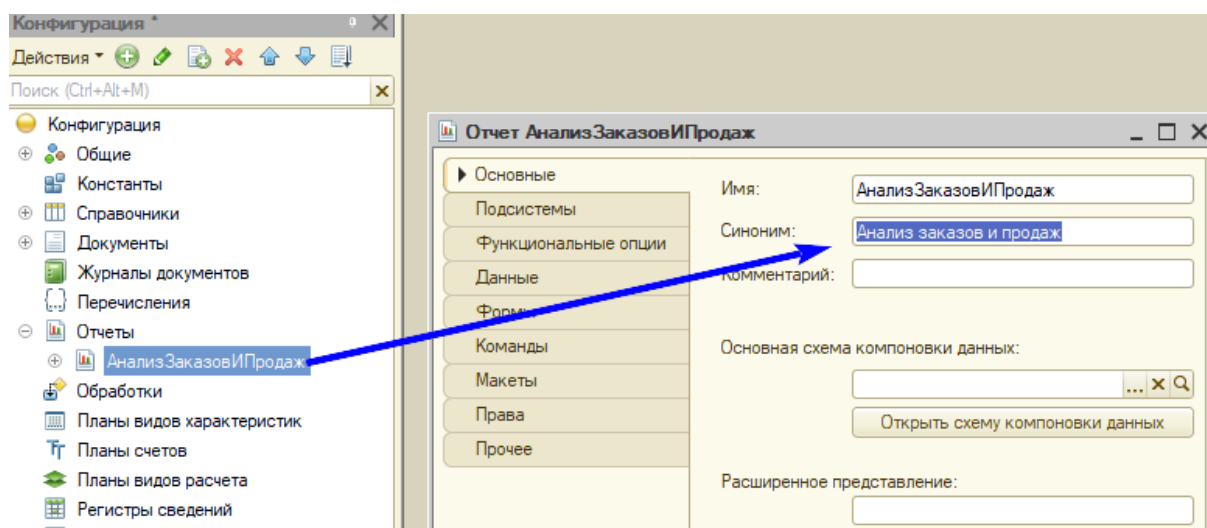
1.5. Т. е. запись данных в регистр накопления происходит при проведении документа «Заказ клиента».

1.6. Аналогично переходим к документу продажи и описываем алгоритм записи данных в регистр накопления.

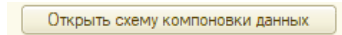




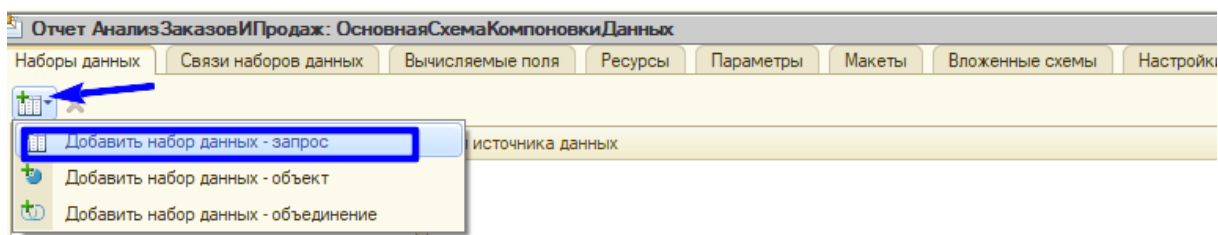
2. Далее переходим непосредственно к созданию отчета. Находим объект «Отчет» в дереве конфигурации и создаем новый отчет под названием «АнализЗаказовИПродаж».



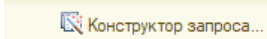
- 2.1. Теперь при помощи специального механизма под названием «Схема компоновки данных» определим, какие данные и в каком виде будут отображаться в отчете.

Для того чтобы открыть «Конструктор схемы компоновки данных» необходимо нажать на соответствующую кнопку — 

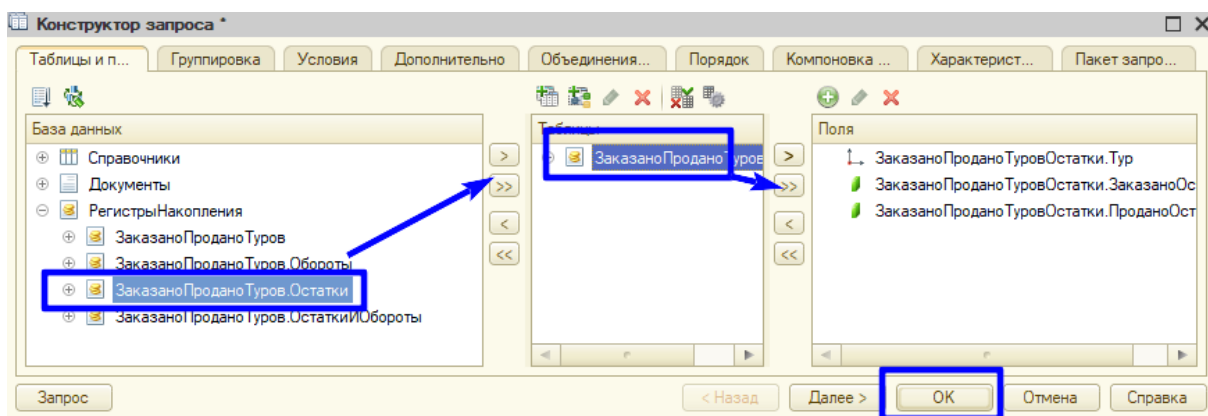
- 2.2. В открывшемся окне «Конструктора» определим способ получения данных для отчета. Мы воспользуемся «Запросом» (обращение к БД за выборкой данных).



2.3. Запрос также можно создать при помощи «Конструктора» — «Конструктора запросов». Открыть этот «Конструктор» можно при помощи кнопки

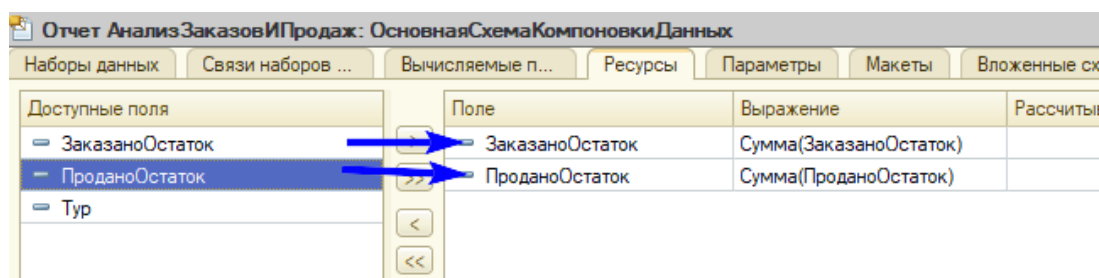



2.4. В открывшемся окне определим таблицу, из которой будем брать данные.

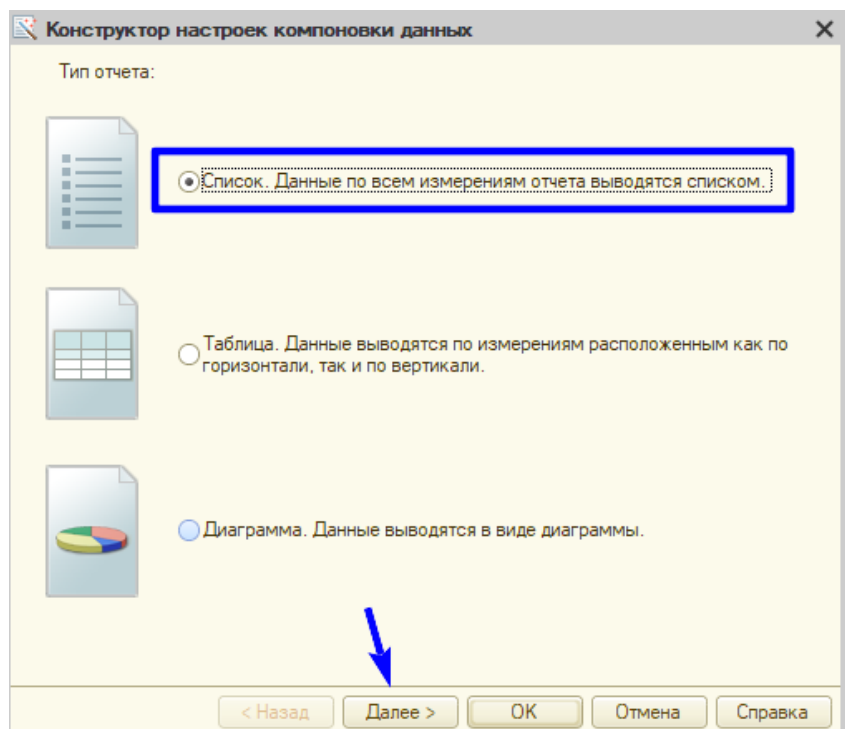


2.5. Был создан запрос на специальном языке — языке запросов для выбора данных из таблицы регистра накопления. Мы его создали при помощи конструктора без знания самого языка запросов.

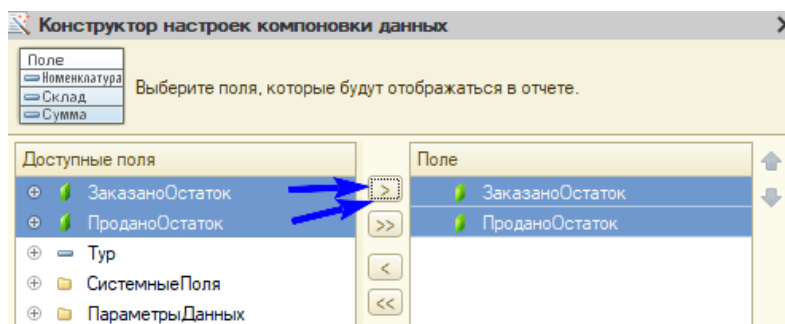
2.6. Далее переходим на закладку «Ресурсы» «Конструктора компоновки данных» и указываем, что нам необходимо суммарное количество заказов и продаж по турам.



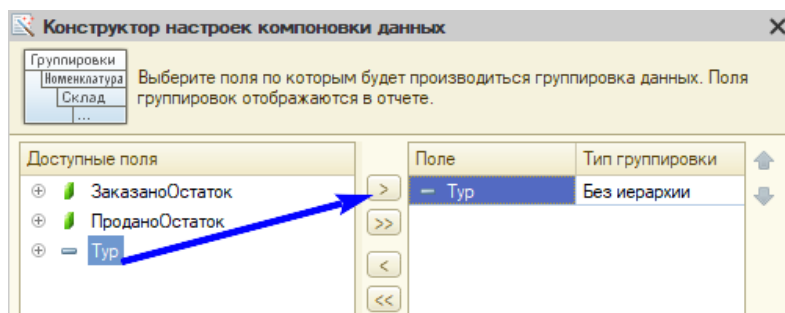
2.7. Теперь переходим на закладку «Настройки» «Конструктора схемы компоновки данных» и настраиваем отчет также при помощи специального «Конструктора» (кнопка ). В открывшемся окне выбираем переключатель и движемся по кнопке «Далее».



2.8. На следующем шаге выбираем поля, которые будут отображаться в отчете.

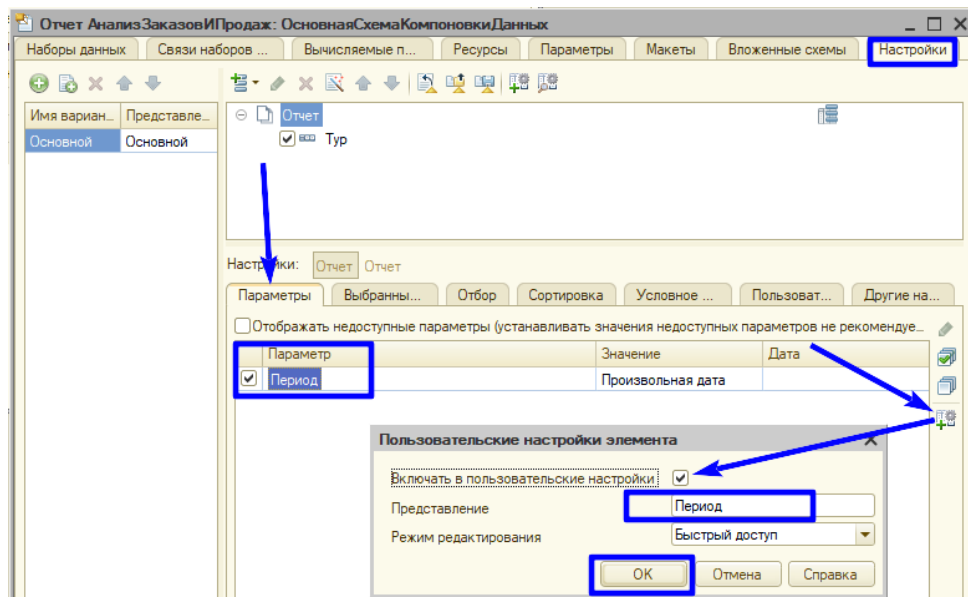


2.9. Далее выбираем группировочное поле.

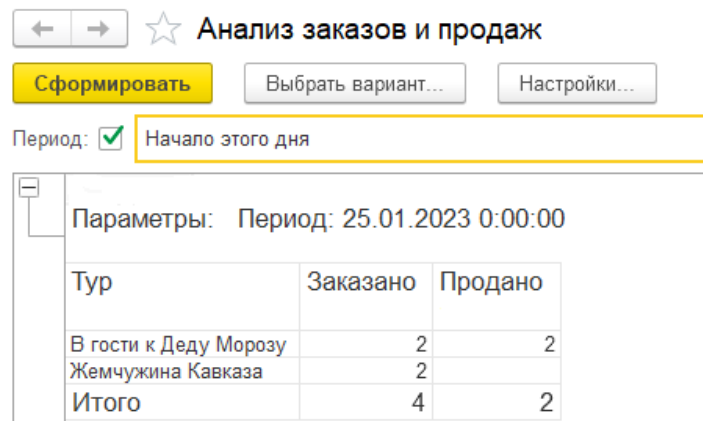


2.10. На последнем шаге нажимаем кнопку 

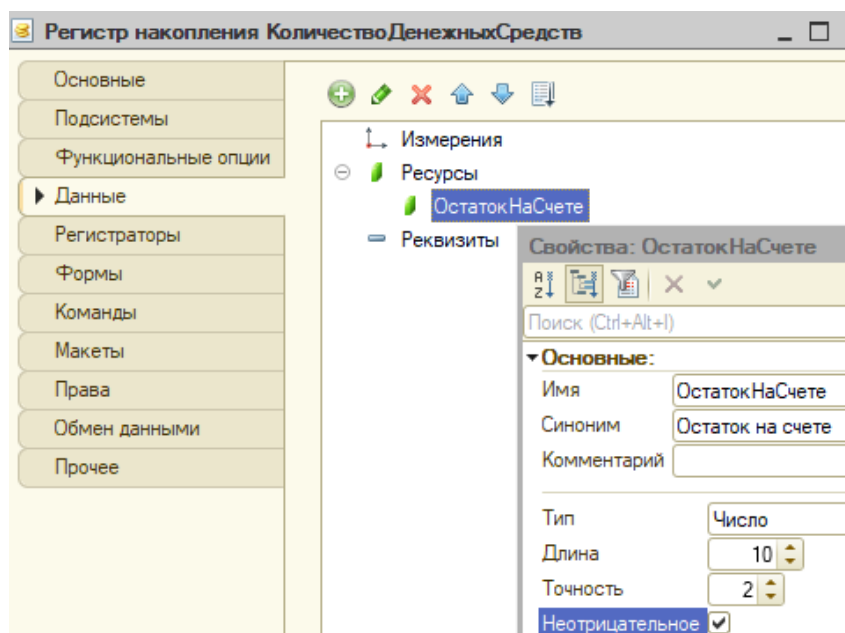
2.11. Отчет готов. Можно сделать так, чтобы данные в нем можно было смотреть на определенную дату. Для этого на закладке «Параметры» нужно сделать следующие настройки.



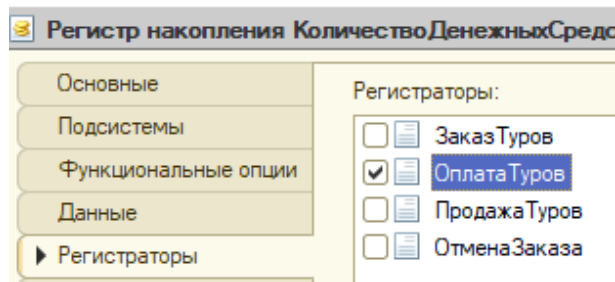
3. Отчет готов. Можно перейти в режим «1С:Предприятие 8» и посмотреть его.



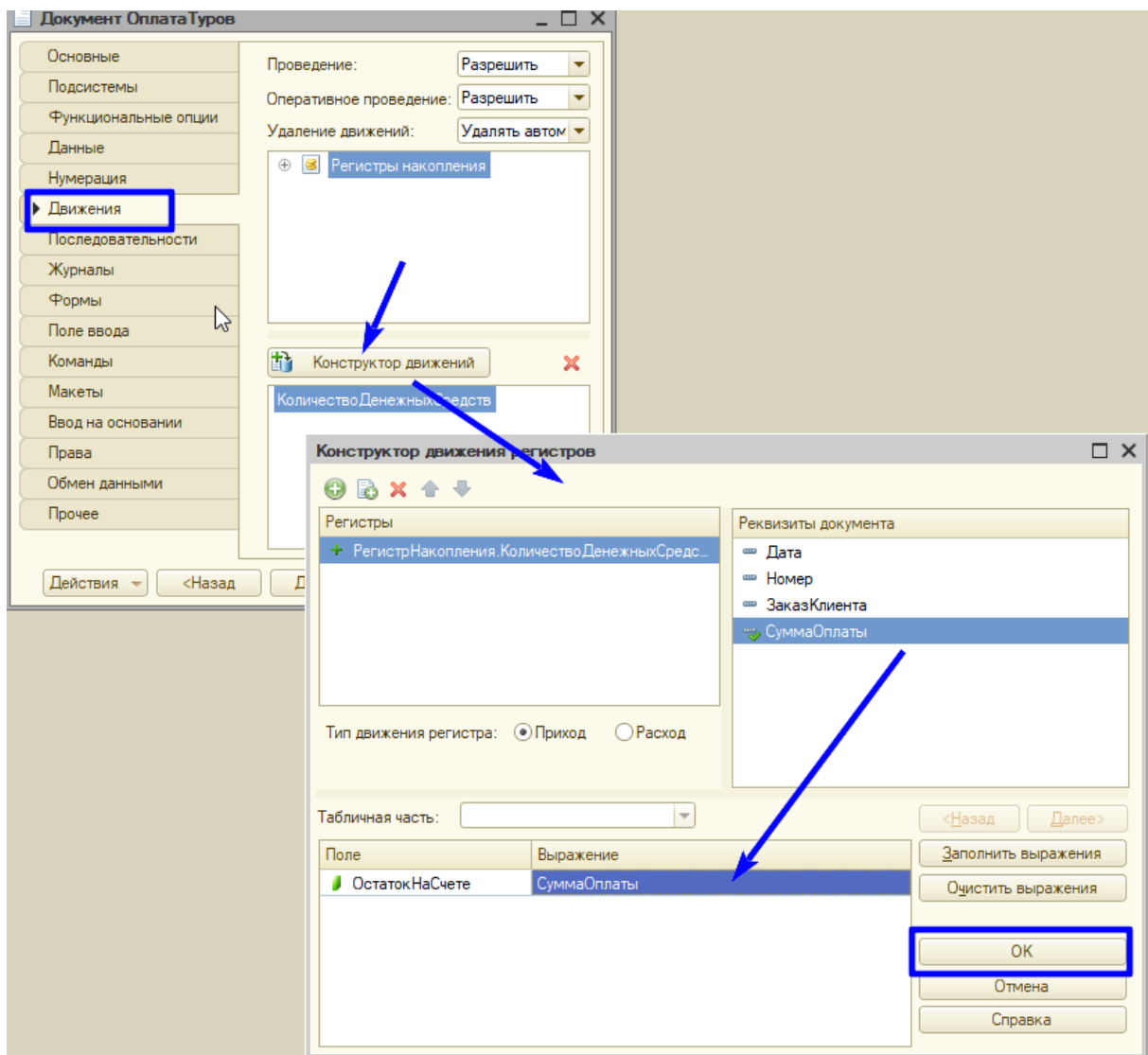
4. Переходим к отчету о количестве денежных средств. Также необходимо создать регистр накопления под названием «КоличествоДенежныхСредств».



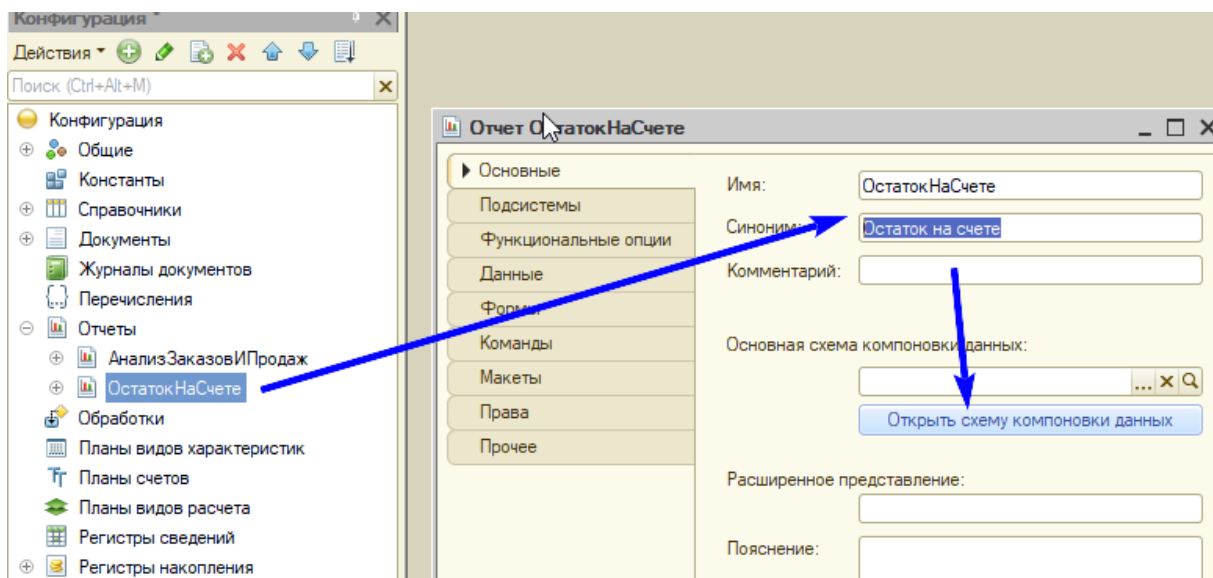
4.1. Регистратором для него будет являться документ «ОплатаТуров».



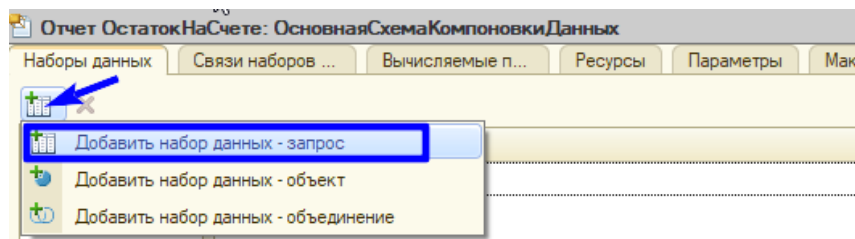
4.2. Переходим к документу «ОплатаТуров» и описываем движения.



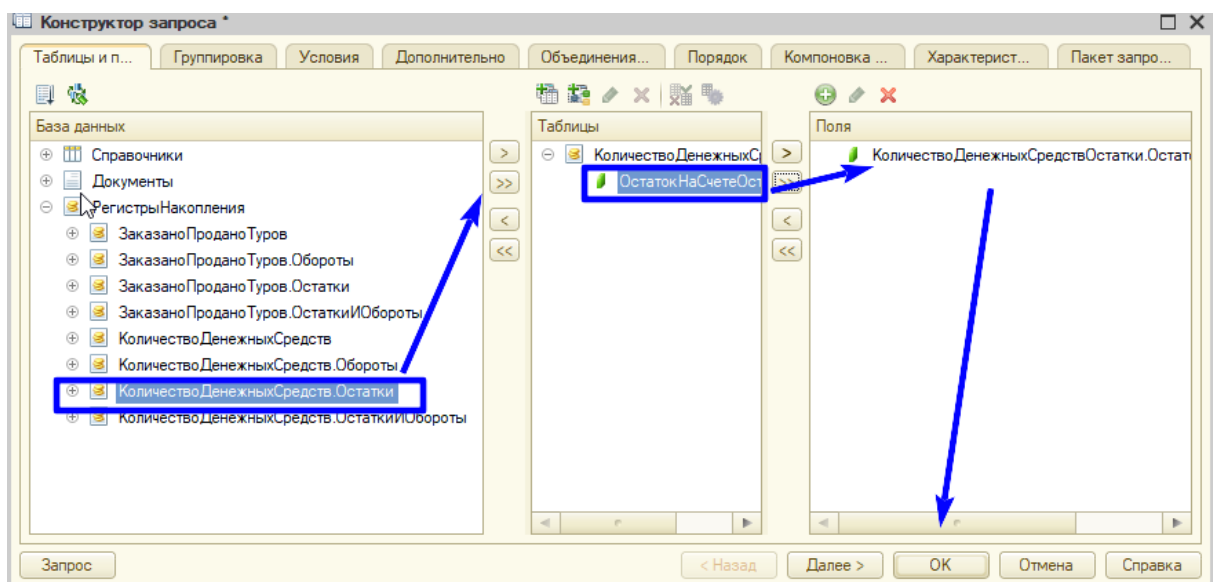
5. Осталось создать в дереве конфигурации отчет под названием «ОстатокНаСчете».




5.1. В открывшемся окне «Конструктора» определяем, что данные будут выбираться при помощи запроса.



5.2. Далее при помощи «Конструктора запроса» (Конструктор запроса...) определяем нужную таблицу и поля.



5.3. Теперь перейдем на закладку «Настройки» «Конструктора схемы компоновки данных» и определим внешний вид вашего отчета. Сделать это можно по кнопке . Внешний вид настраивается также, как и для предыдущего отчета.

-
6. За различные доработки отчетов, например, вариант отчета о заказах и продажах в виде диаграммы, добавление в отчет данных о сумме заказанных и проданных туров, новые дополнительные отчеты и т. д., предполагаются бонусы +1 балл за каждое улучшение. Здесь можно пофантазировать.
 7. Также нужно закончить работу над интерфейсом, чтобы приложение имело законченный вид.