

Нам в конфигурации понадобится справочник для хранения сведений об учениках. Каким он должен быть? Этот справочник должен содержать такую информацию, как фамилия, имя, отчество, пол, дата рождения, адрес и другая контактная информация. Но если нам придется создавать в нашей программе справочник для хранения сведений о родителях или учителях, то эти справочники будут содержать аналогичную информацию. Поэтому имеет смысл вынести такую информацию в отдельный справочник. Назовем его – *Физические лица*.

В отличие от других справочников, которые мы создавали, этот справочник будет иерархическим, т.е. будет иметь несколько уровней.

Существует два вида иерархии: **иерархия групп и элементов** и **иерархия элементов**. В первом случае все данные справочника будут разделяться на два вида: «просто» элементы справочника и группы справочника. **Группы** — это логическое объединение элементов справочника.

Использование иерархических справочников позволяет организовать ввод информации в справочник с нужной степенью детализации.

Элементы и группы элементов в иерархическом справочнике можно переносить из одной группы в другую.

Для справочников с видом иерархии *Иерархия элементов* группы как самостоятельный вид отсутствуют. Их роль выполняют сами элементы. Отличительной особенностью таких справочников является функциональность всех элементов.

Для иерархических справочников конфигуратор позволяет установить ограничение числа уровней справочников или сделать неограниченное число уровней вложенности.

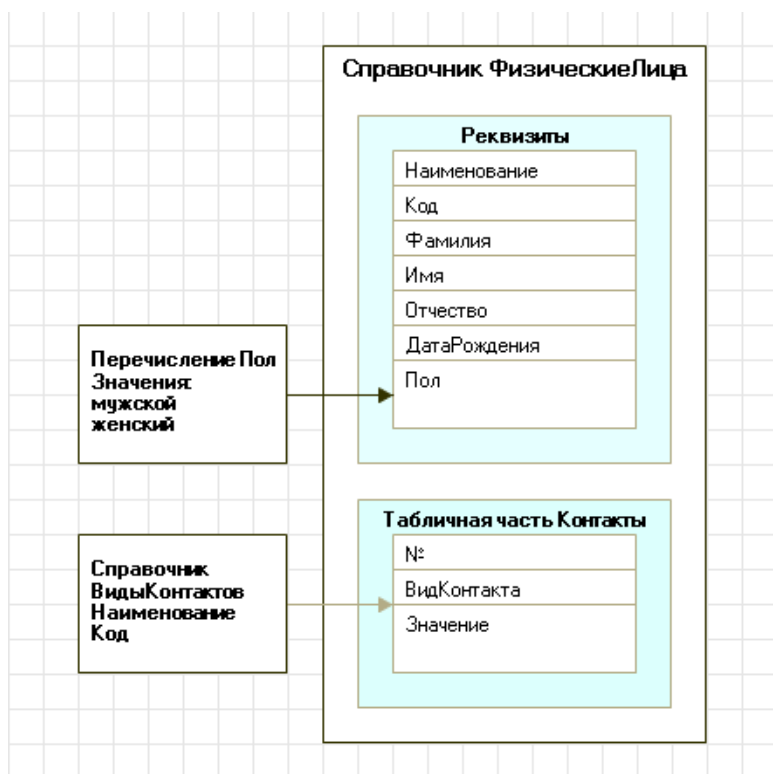
Помимо кода и наименования этот справочник будет иметь набор реквизитов: *Фамилия, Имя, Отчество, Пол, Дата* рождения. Для хранения такой информации, как адрес проживания, телефон, адрес электронной почты, создадим табличную часть *Контакты* с реквизитами *Вид контакта* и *Значение*. В табличных частях может храниться однотипная информация, число которой может быть переменным.

Поскольку количество видов контактов достаточно ограничено, то для реквизита Вид контакта лучше всего будет создать отдельный справочник *Виды Контакт*ов.

Реквизит справочника *Физические лица* *Пол* может принимать только одно из двух значений: *Мужской* и *Женский*. Для значений таких реквизитов лучше всего подходит тип объекта **Перечисление**.

Перечисления используются в системе «1С:Предприятие» для описания постоянных наборов значений, не изменяемых в процессе работы конфигурации. На этапе конфигурирования можно описать практически неограниченное количество видов перечислений. В отличие от справочника, значения перечислений задаются на этапе конфигурирования и не могут быть изменены на этапе исполнения.

Представим объекты, которые будем создавать на этом занятии схематически:



Задание 1.

Создайте справочник **Виды контактов**.

1. В дереве конфигурации добавим справочник, введем имя:

Справочник ВидыКонтактов

Основные

Подсистемы

Функциональные опции

Иерархия

Владельцы

Данные

Нумерация

Формы

Команды

Макеты

Ввод на основании

Права

Обмен данными

Прочее

Имя: ВидыКонтактов

Синоним: Виды контактов

Комментарий:

Представление объекта:

Расширенное представление объекта:

Представление списка:

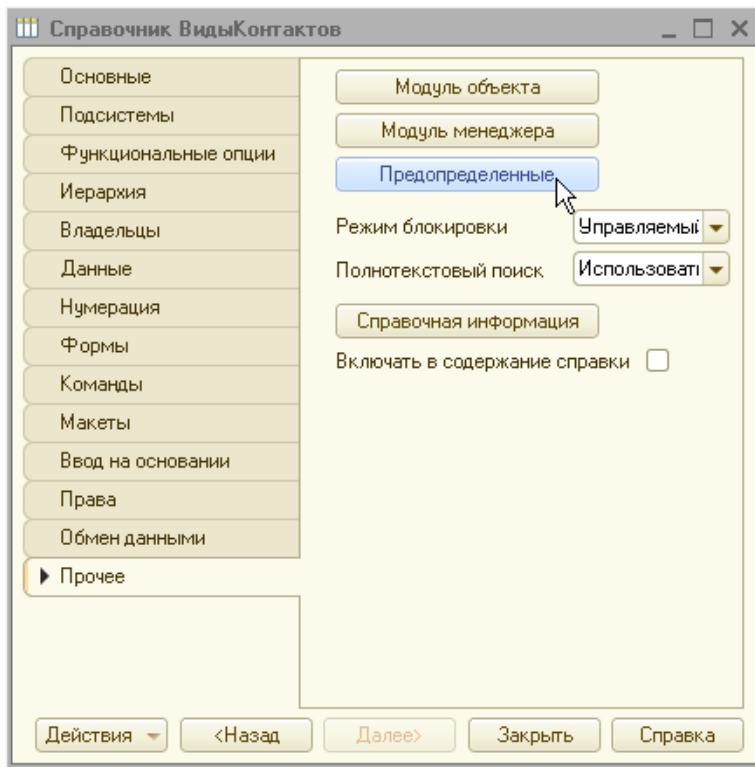
Расширенное представление списка:

Пояснение:

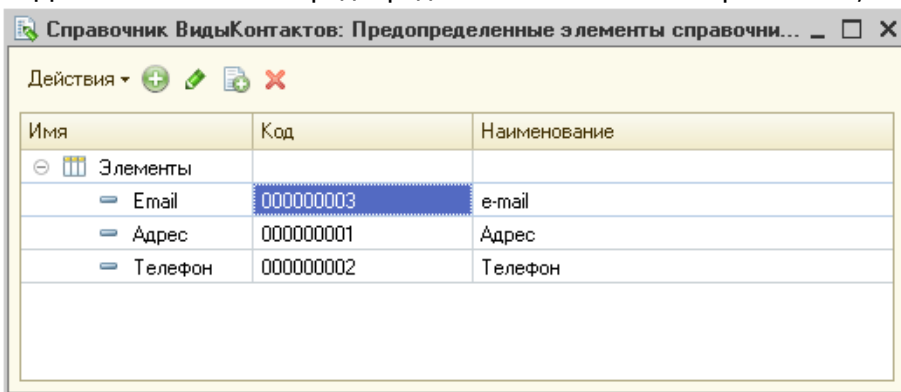
Действия <Назад Далее> Закреть Справка

2. На закладке *Подсистемы* укажем принадлежность этого справочника подсистеме *Состав школы*.

3. На закладке *Прочие* нажатием кнопки *Предопределенные* перейдем в список предопределенных элементов справочника:



4. Добавим несколько предопределенных элементов справочника, как показано на рисунке:



Теперь в справочнике в режиме 1С:Предприятие будут представлены эти виды контактов.

Задание 2

Создайте перечисление *Пол*.

1. Найдем в дереве конфигурации ветвь *Перечисления* и добавим новое перечисление.
2. В окне разработки перечисления на вкладке *Основные* введем имя - *Пол* :

Перечисление Пол

Основные

Подсистемы

Функциональные опции

Данные

Формы

Команды

Макеты

Прочее

Имя: Пол

Синоним: Пол

Комментарий:

Представление списка:

Расширенное представление списка:

Пояснение:

Действия <Назад Далее> Закреть Справка

3. На вкладке *Подсистемы* укажем подсистему *Состав школы* :

Перечисление Пол

Основные

Подсистемы

Функциональные опции

Данные

Формы

Команды

Макеты

Прочее

Подсистемы, в которых участвует перечисление:

☒ СоставШколы

☐ УчебныйПроцесс

☐ СтруктураШколы

Действия <Назад Далее> Закреть Справка

4. На вкладке *Данные* добавим значения *Мужской*, *Женский* :

Перечисление Пол

Основные

Подсистемы

Функциональные опции

Данные

Формы

Команды

Макеты

Прочее

Значения перечисления:

+ - X ↑ ↓

Значения

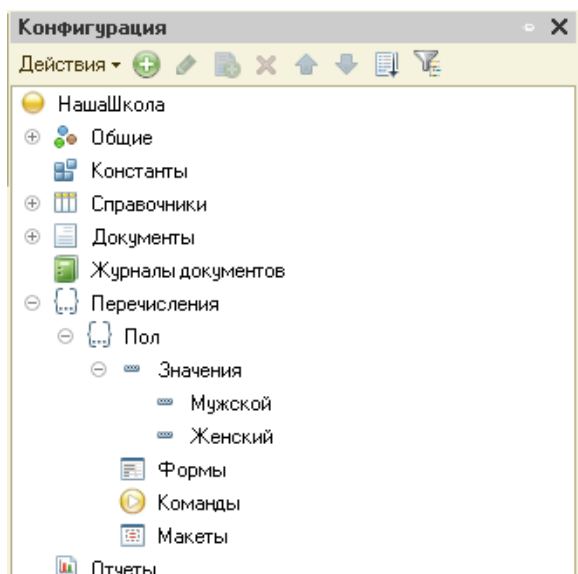
Мужской

Женский

Стандартные реквизиты Характеристики

Действия <Назад Далее> Закреть Справка

5. В дереве конфигурации перечисление Пол представлено следующим образом:

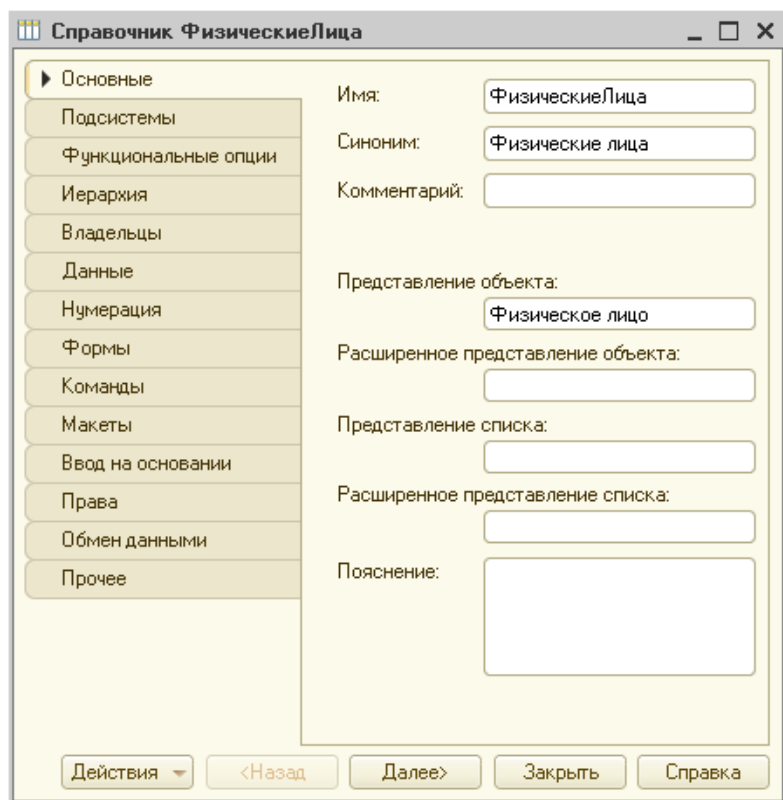


Задание 3

Разработайте справочник **Физические лица**.

1. Добавим новый справочник, зададим ему имя *ФизическиеЛица*, синоним – *Физические лица*.

Укажем представление объекта – *Физическое лицо*, так будет называться объект в единственном числе:



2. На закладке *Подсистемы* укажем принадлежность разделу *Состав школы*.

3. На закладке *Иерархия* установим флаг *Иерархический справочник* :

Справочник ФизическиеЛица

Основные
Подсистемы
Функциональные опции
Иерархия
Владельцы
Данные
Нумерация
Формы
Команды
Макеты
Ввод на основании
Права
Обмен данными
Прочее

Иерархический справочник ☒

Вид иерархии:
Иерархия групп и элементов

Размещать группы сверху ☒

Ограничение количества уровней иерархии ☐

Количество уровней иерархии 2

Действия <Назад Далее> Закрывать Справка

Это позволит группировать элементы справочника. Например, можно будет создать такие группы как *Ученики, Учителя, Родители*.

4. На закладке *Данные* увеличим длину наименования до 150 символов.

Добавим реквизиты:

- *Фамилия, Имя, Отчество*, тип – Строка, длина – 50 символов;
- *ДатаРождения*, тип – Дата;
- *Пол*, тип – ПеречислениеСсылка.Пол.

Справочник ФизическиеЛица

Основные
Подсистемы
Функциональные опции
Иерархия
Владельцы
Данные
Нумерация
Формы
Команды
Макеты
Ввод на основании
Права
Обмен данными
Прочее

Длина кода 9

Длина наименования 150

Тип кода
☐ Число
☒ Строка

Основное представление
☐ В виде кода
☒ В виде наименования

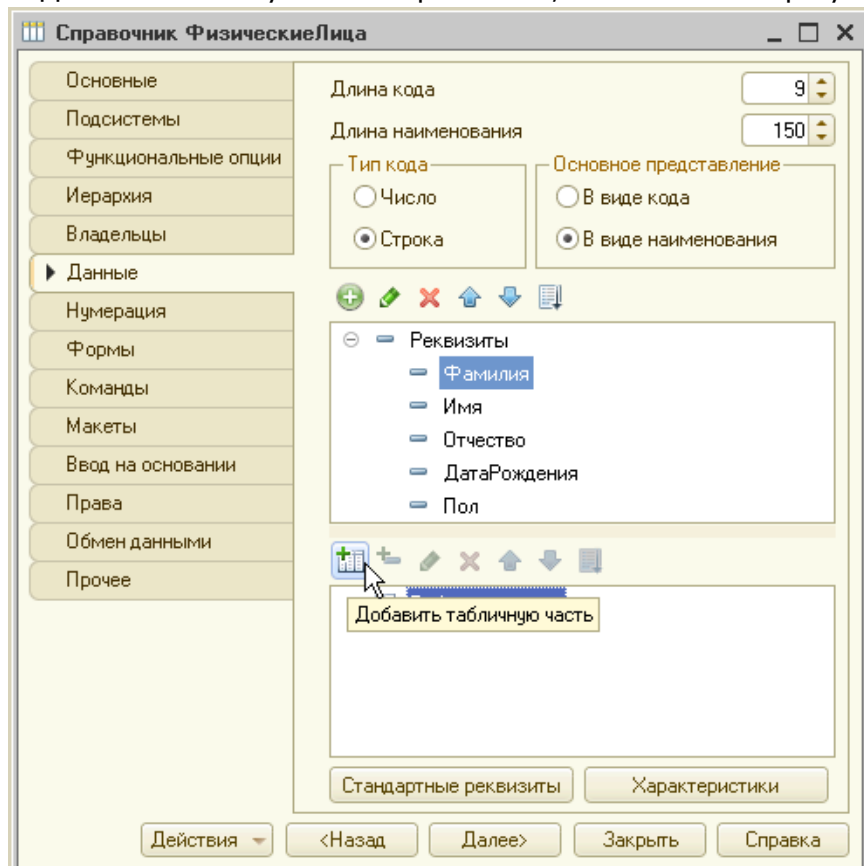
Реквизиты
Фамилия
Имя
Отчество
ДатаРождения
Пол

Табличные части

Стандартные реквизиты Характеристики

Действия <Назад Далее> Закрывать Справка

5. Добавим табличную часть справочника, как показано на рисунке:

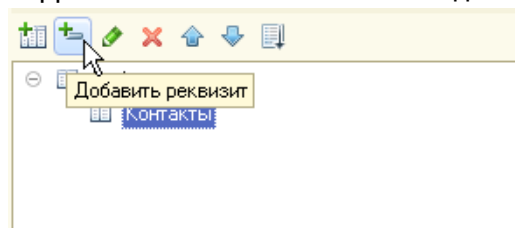


Укажем имя табличной части – *Контакты*. *Табличные части* используются для описания некоторых данных, относящихся к справочнику и не используемых самостоятельно. Например, как в нашем случае, состав контактной информации у каждого физического лица может быть различен, у кого-то может быть один телефон, у кого-то два. Контактная информация вряд ли будет использоваться отдельно от самого физического лица.

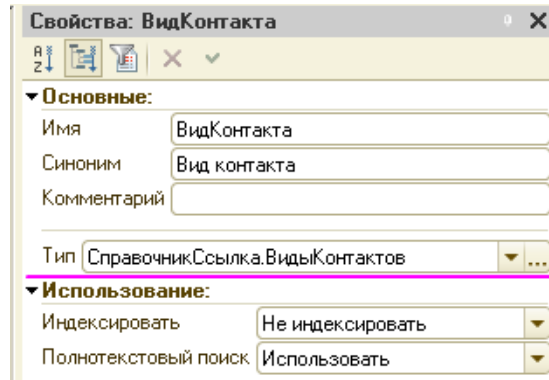
Если бы в приведенном примере сведения о контактной информации могли использоваться самостоятельно, то их можно было бы выделить в отдельный справочник, подчиненный справочнику *Физические лица*. В подчиненном справочнике каждый элемент «принадлежит», «прикреплен» к определенному элементу справочника-владельца.

Основное отличие табличной части от подчиненного справочника в том, что на элементы справочника можно ссылаться, а на строки табличной части нет. При обращении к элементу справочника он весь, вместе со всеми табличными частями, считывается из базы данных в память. Если табличная часть содержит достаточно большое количество строк, это может ухудшить производительность системы. Поэтому табличную часть стоит использовать, если не надо хранить ссылки на элементы, и количество элементов относительно невелико. Каждый справочник может иметь неограниченное число табличных частей.

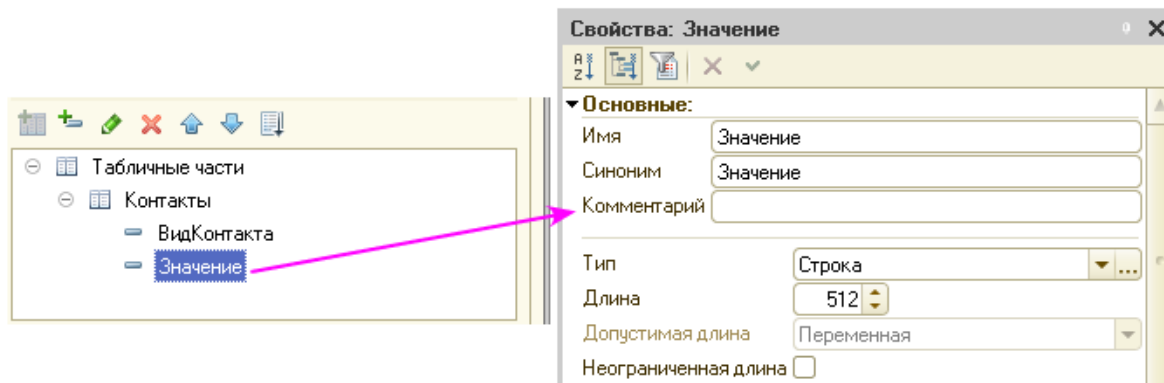
6. Для табличной части *Контакты* добавим реквизит:



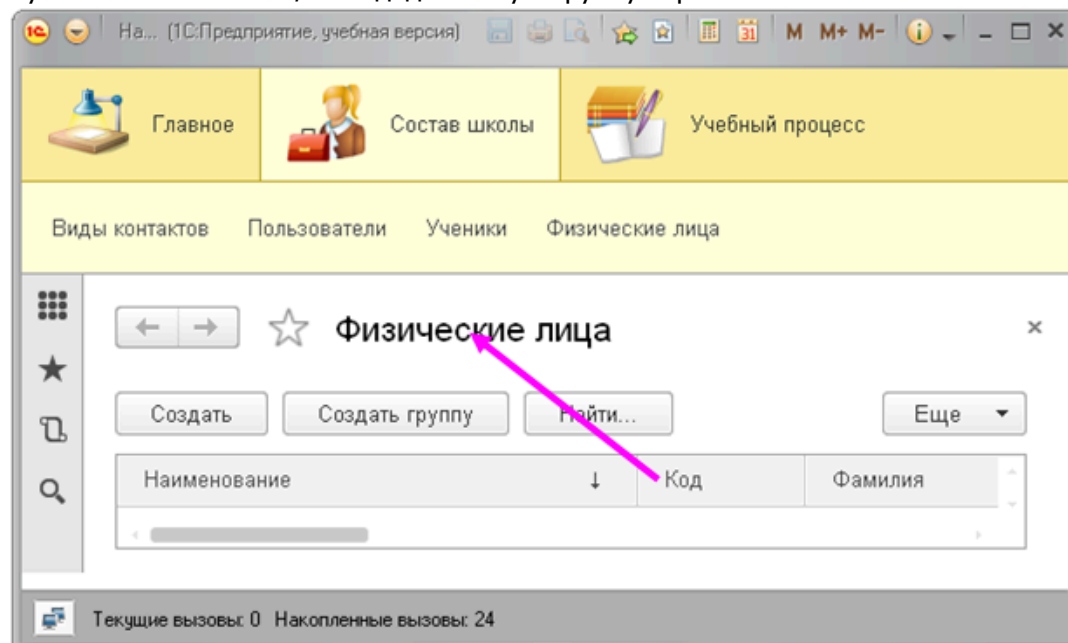
Укажем имя реквизита *ВидКонтакта*, тип – *СправочникСсылка.ВидыКонтактов* :



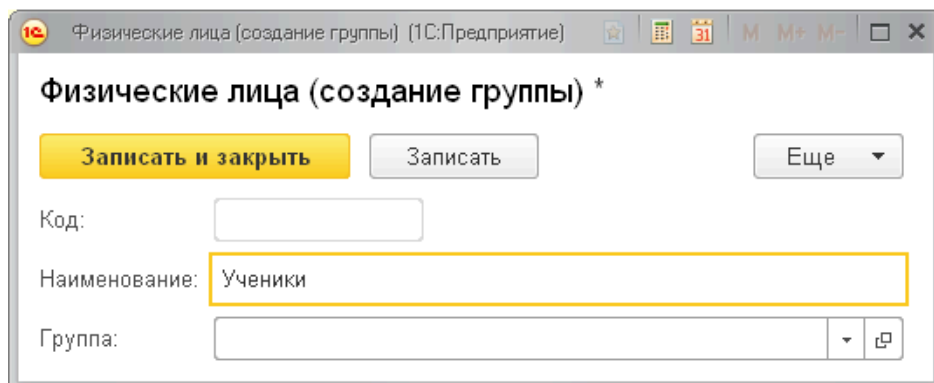
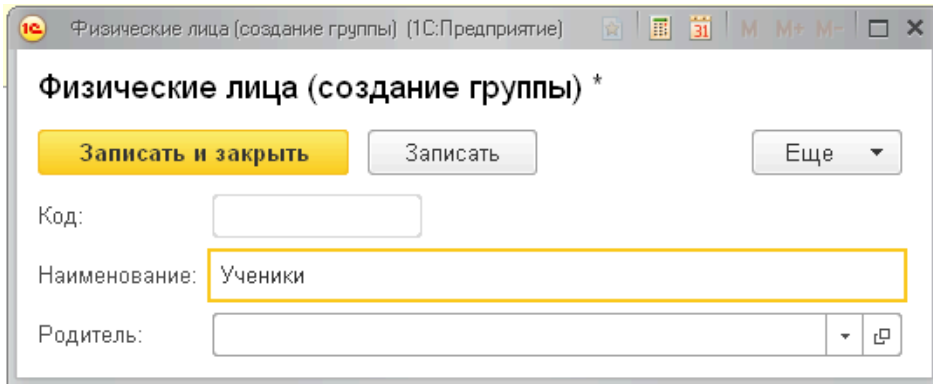
7. Второй реквизит табличной части назовем *Значение*, тип – *Строка*, длина – 512 символов:



8. Теперь проверим, что у нас получилось. Для этого запустим программу в режиме «1С:Предприятие», перейдем в раздел *Состав школы*, выберем на панели навигации в правой части окна программы пункт *Физические лица*. Создадим новую группу справочника:

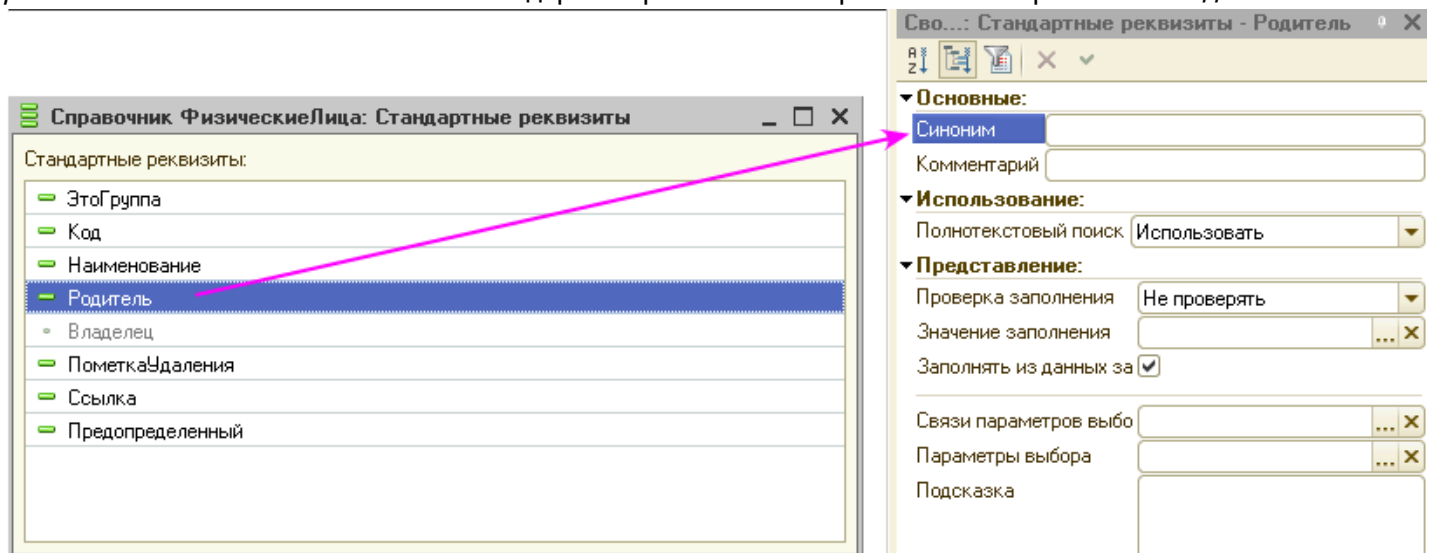


Дадим ей наименование *Ученики* и сохраним кнопкой *Записать и закрыть*:



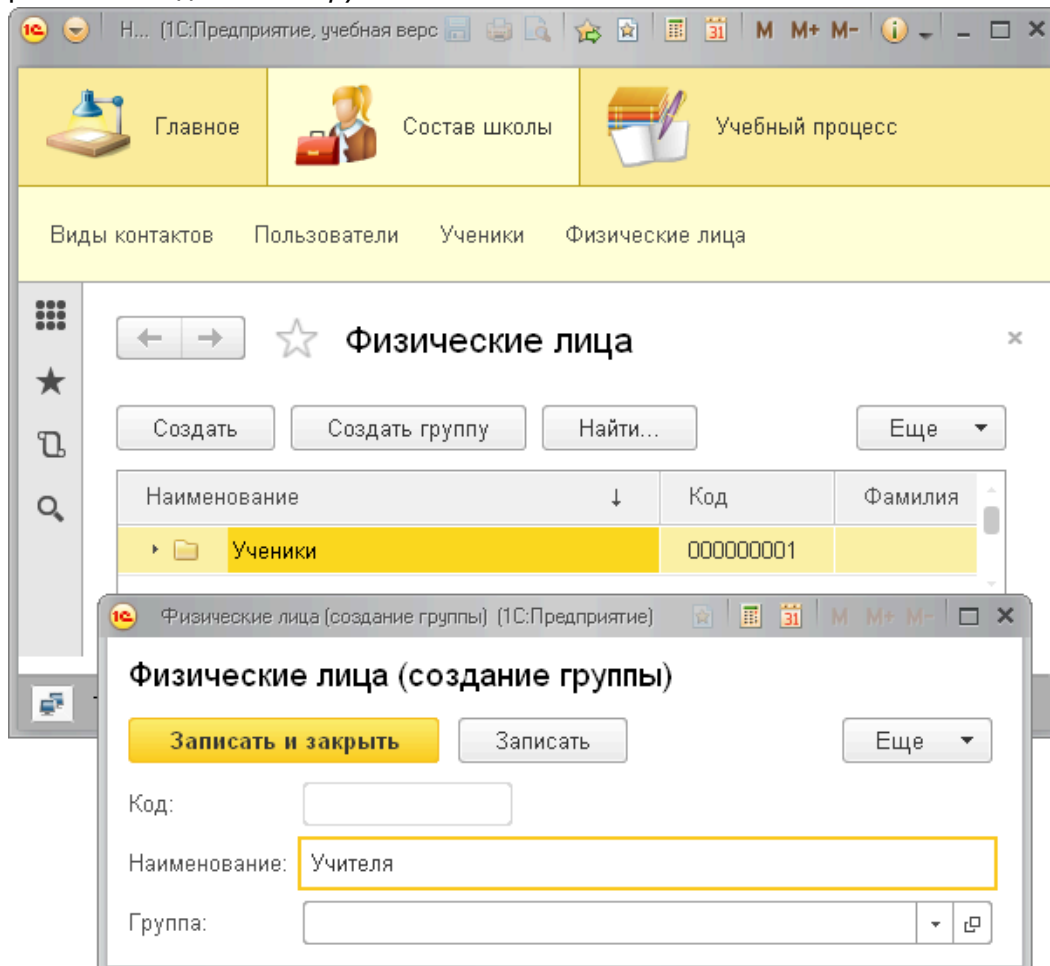
9. На форме создания группы мы увидели поле под названием *Родитель*. Это поле содержит вышестоящую группу справочника *Физические лица*. В данный момент оно пустое, поскольку мы создаем группу самого верхнего уровня.

Поскольку наименование поля *Родитель* в контексте данного справочника может иметь двойной смысл, нам необходимо переименовать его. Для этого нам необходимо в конфигураторе, в окне редактирования справочника на закладке *Данные* открыть список стандартных реквизитов. Нажмем кнопку *Стандартные реквизиты* и в появившемся списке стандартных реквизитов откроем свойства реквизита *Родитель*:



В качестве синонима укажем значение *Группа*, которое бы мы хотели видеть в форме. Запустим программу в режиме отладки (F5) При добавлении новой группы увидим как изменилось представление

реквизита *Родитель* на *Группа*:

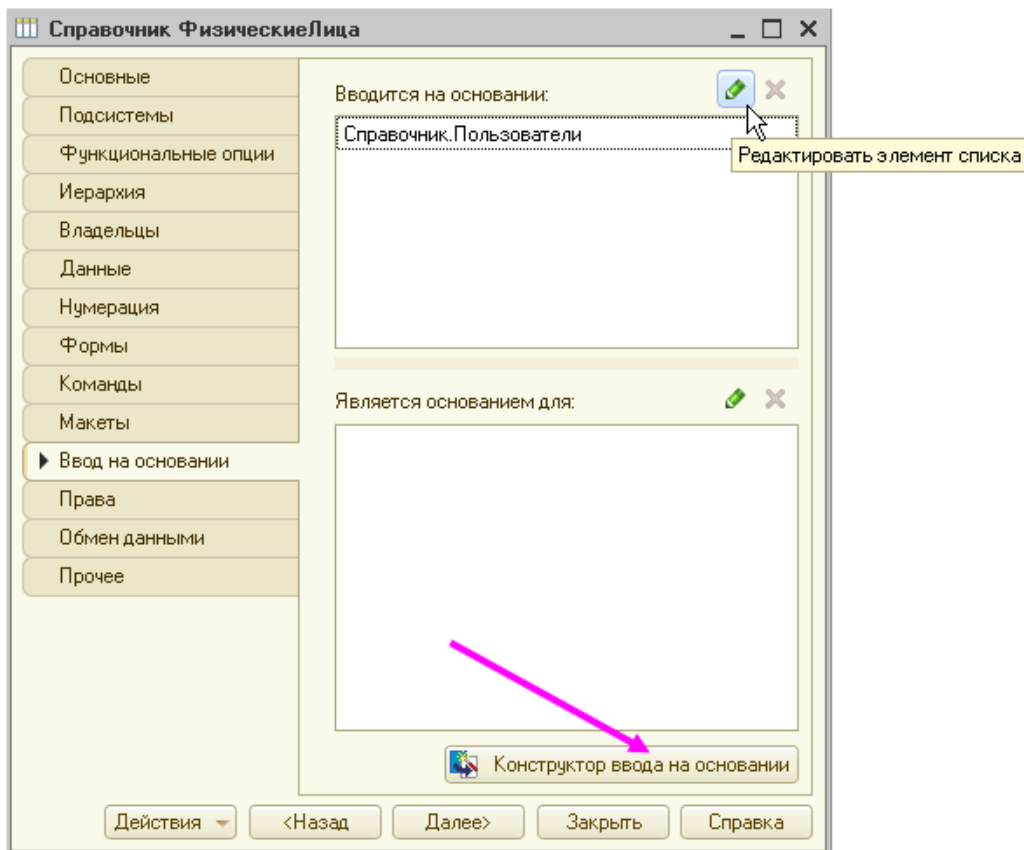


Справочник *Физические лица* создан, можно приступать к его заполнению. Но у нас в программе уже есть сведения о некоторых людях в справочнике *Пользователи*. Система «1С:Предприятие 8» позволяет реализовать заполнение объектов при их вводе на основании других объектов информационной базы.

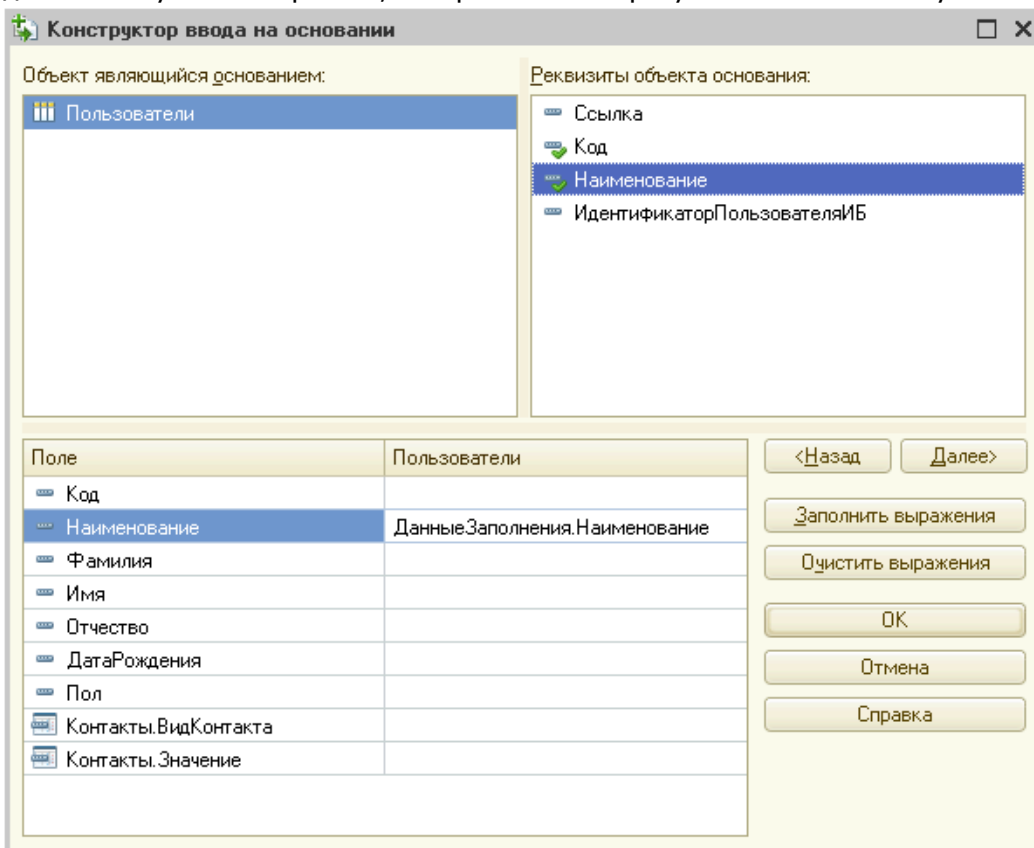
Задание 5

Настройте для справочника *Физические лица* ввод данных на основании справочника *Пользователи*.

1. В окне редактирования справочника *Физические лица* перейдем на закладку *Ввод на основании*, в верхнем окне добавим справочник *Пользователи* и нажмем кнопку *Конструктор ввода на основании* :



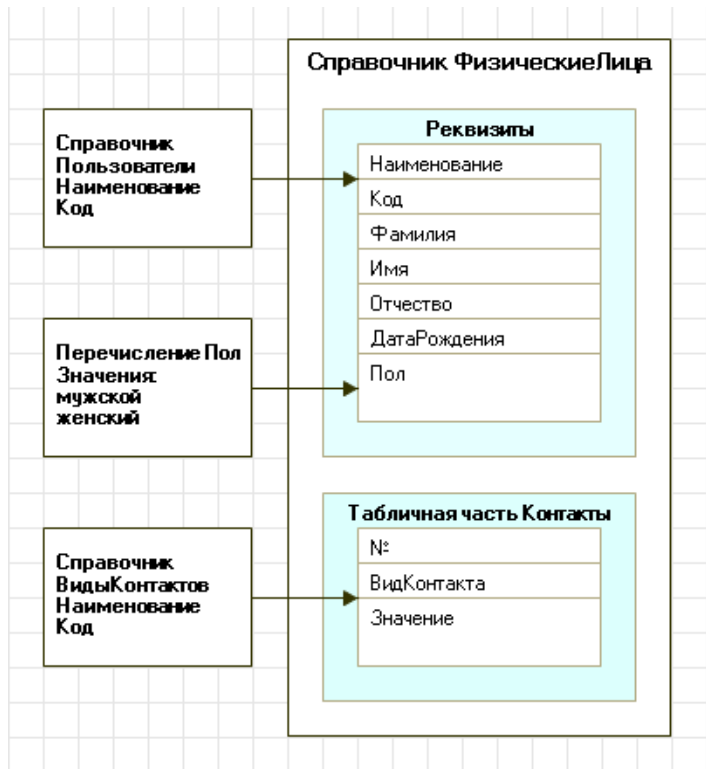
2. В открывшемся окне конструктора ввода на основании выберем в нижнем и верхнем окнах поле Наименование, затем дважды щелкнем левой клавишей мыши по верхнему правому окну. В результате должна получиться картинка, изображенная на рисунке. Нажмем кнопку **OK** :



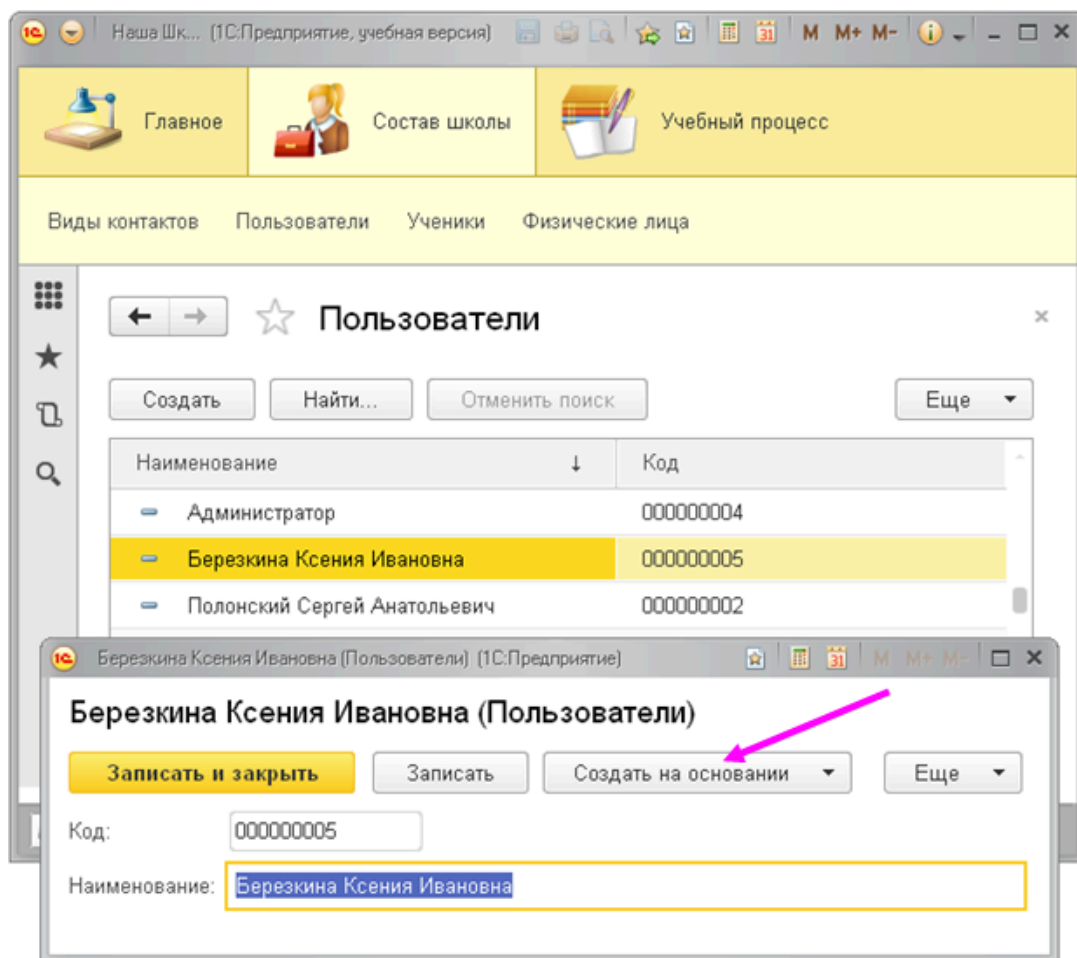
Конструктор создаст в модуле объекта текст:

```
□ Процедура ОбработкаЗаполнения(ДанныеЗаполнения, СтандартнаяОбработка)
  ///({_КОНСТРУКТОР_ВВОД_НА_ОСНОВАНИИ
  // Данный фрагмент построен конструктором.
  // При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!!
  Если ТипЗнч(ДанныеЗаполнения) = Тип("СправочникСсылка.Пользователи") Тогда
    // Заполнение шапки
    Наименование = ДанныеЗаполнения.Наименование;
  КонецЕсли;
  ///})_КОНСТРУКТОР_ВВОД_НА_ОСНОВАНИИ
КонецПроцедуры
```

Данная процедура позволяет связать справочники *Физические лица* и *Пользователи* :



3. Для проверки запустим программу. Перейдем на список пользователей. Выберем кого-либо из пользователей и нажмем кнопку *Создать на основании* :



В появившемся окне должно быть заполнено поле *Наименование*, заполним недостающие реквизиты, как показано на рисунке, и сохраним кнопкой *Записать и закрыть* :

Физическое лицо (создание)

Записать и закрыть Записать Еще ▾

Код:

Наименование: Березкина Ксения Ивановна

Группа: ▾

Фамилия:

Имя:

Отчество:

Дата рождения:

Пол: ▾

Добавить Еще ▾

N	Вид контакта	Значение
---	--------------	----------

Введем данные

Физическое лицо (создание) (1С:Предприятие)

Физическое лицо (создание) *

Записать и закрыть Записать Еще ▾

Код:

Наименование:

Группа: ▾

Фамилия:

Имя:

Отчество:

Дата рождения:

Пол: ▾

Добавить Еще ▾

N	Вид контакта	Значение
1	e-mail	berezk@mail.ru
2	Телефон	89125674545

В результате в справочник Физические лица будет добавлена еще одна запись:

Наша Школа(Поль... (1С:Предприятие, учебная версия)

Главное Состав школы Учебный процесс

Виды контактов Пользователи Ученики **Физические лица**

Физические лица

Создать Создать группу Найти... Отменить поиск Еще ▾

Наименование	Код	Фамилия	Имя
Ученики	000000001		
Березкина Ксения Ивановна	000000006	Березкина	Ксения
Полонский Сергей Анатольевич	000000004	Полонский	Сергей
Смирнова Евгения Владимировна	000000002	Смирнова	Евгения
Федотов Сергей Иванович	000000003	Федотов	Сергей

Текущие вызовы: 0 Накопленные вызовы: 106