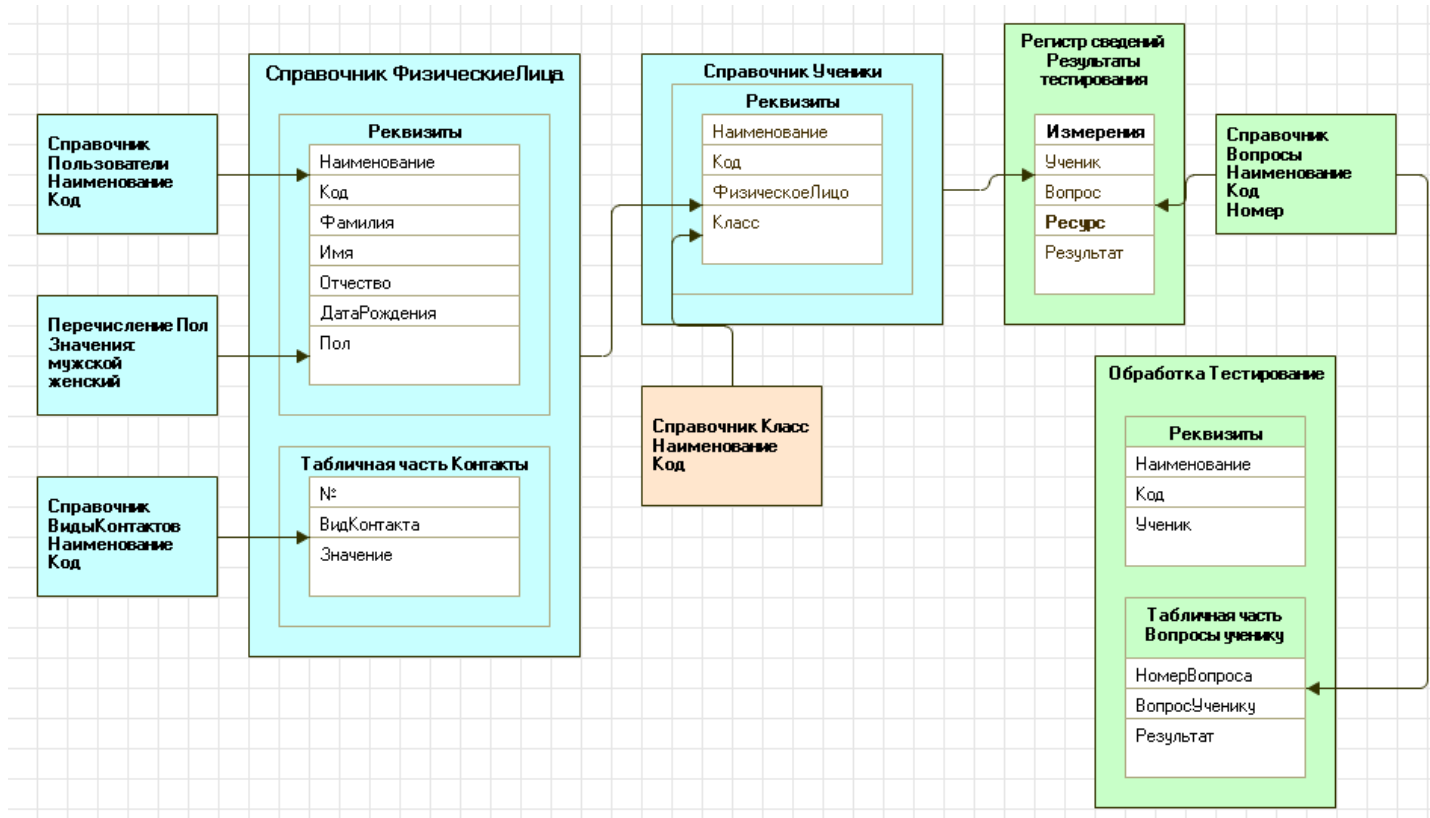


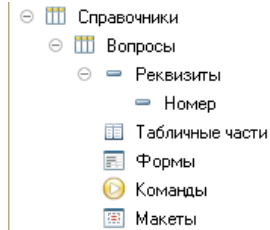
Необходимо создать обработку для тестирования учащихся школы, а также разработать механизм для хранения и обработки результатов тестирования. Для этого создать в конфигурации объекты: справочник *Вопросы*, регистр сведений *Результаты тестирования*, обработку для проведения тестирования. Представим объекты на схеме, выделим цветом объекты, принадлежащие одной подсистеме:



Задание 1

Создайте справочник *Вопросы* для хранения вопросов тестирования.

Добавим справочник *Вопросы*, относящийся к подсистеме *Учебный процесс* с реквизитом *Номер* – тип *Число*, длина – 3 знака.



Задание 2

Создайте регистр сведений *РезультатыТестирования* для хранения результатов тестирования.

1. В дереве конфигурации найдем ветвь *Регистры сведений*. Добавим регистр сведений *РезультатыТестирования* :

2. На закладке *Подсистемы* укажем подсистему *Учебный процесс*. На закладке *Данные* добавим:

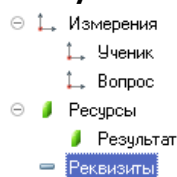
Измерения:

Ученик – тип *СправочникСсылка.Ученики*, установить свойство *Ведущее* и *Запрет незаполненных значений*.

Вопрос – тип *СправочникСсылка.Вопросы*, установить свойство *Ведущее* и *Запрет незаполненных значений*.

Ресурс:

Результат – тип *Булево*.



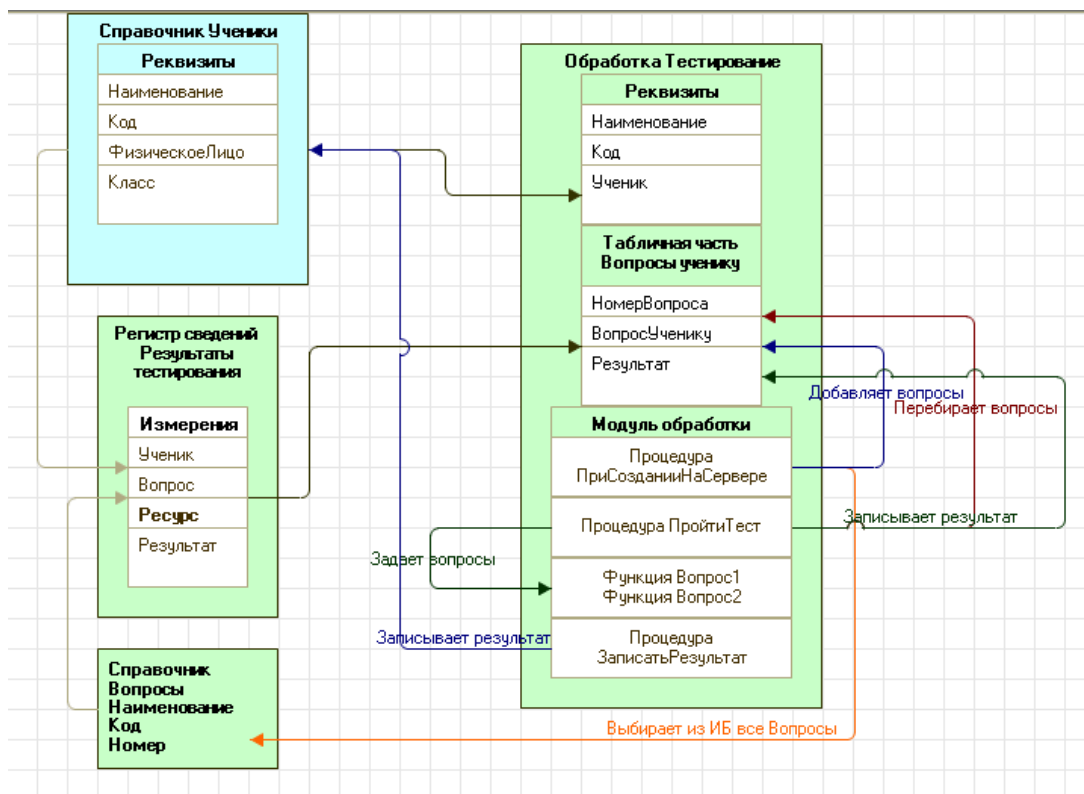
Свойство *Ведущее* означает, что запись регистра сведений имеет смысл, только пока существует значение измерения. При удалении какого-либо ученика из информационной базы будут удалены все записи в регистре сведений с результатами этого ученика.

Также из формы элемента справочника ученика можно будет открыть список результатов тестирования с отбором по этому ученику:

Задание 3

Создайте обработку, с помощью которой можно будет проводить тестирование и сохранять результат в регистре сведений.

В обработке *Тестирование* создадим реквизиты, табличную часть, в модуле обработки – несколько процедур. Покажем их назначение на схеме:



1. Для того чтобы создать новую обработку, в дереве конфигурации выделим ветвь *Обработки*, в выпадающем меню выберем пункт *Добавить*.

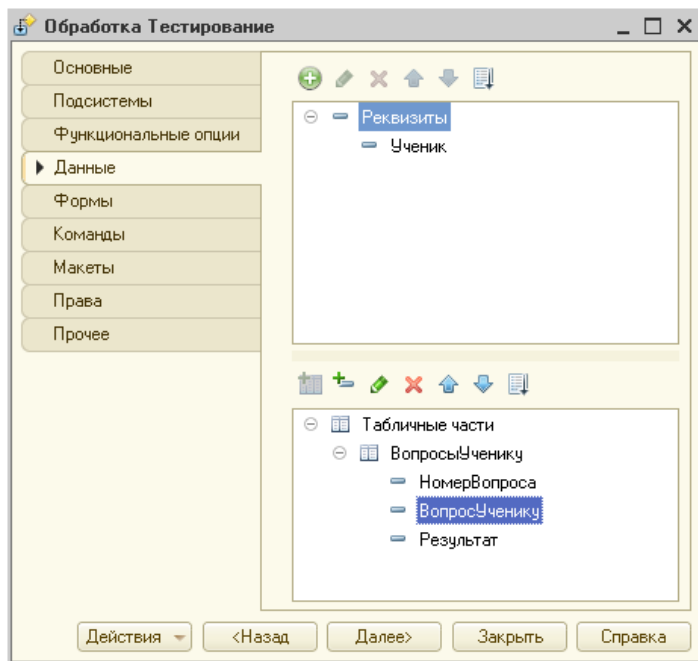
2. Введем имя обработки – *Тестирование* :

3. На закладке *Подсистемы* отнесем эту обработку к разделу *Учебный процесс*.

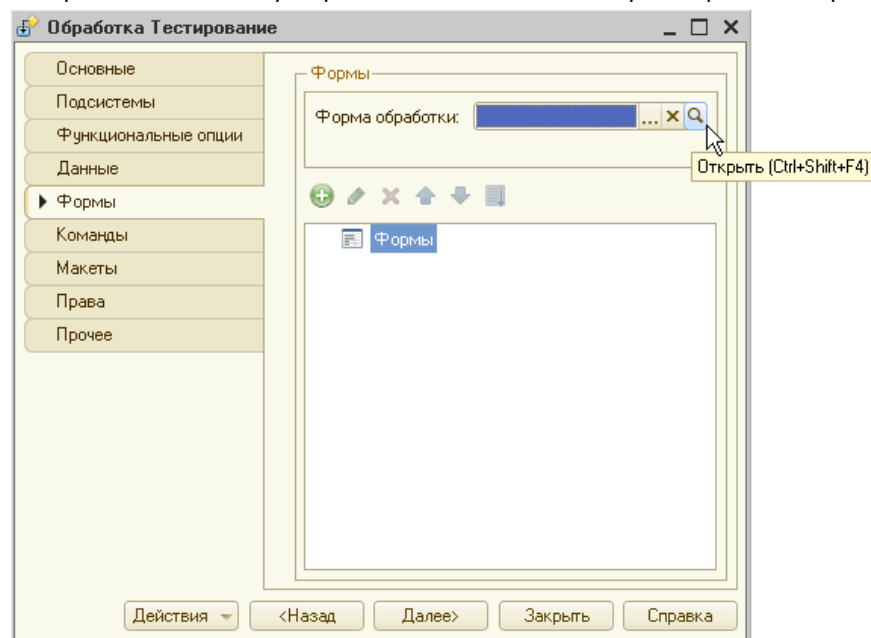
4. На закладке *Данные* создадим новый реквизит *Ученик*, тип – *СправочникСсылка.Ученики*.

5. Также создадим табличную часть *ВопросыУченику* с реквизитами:

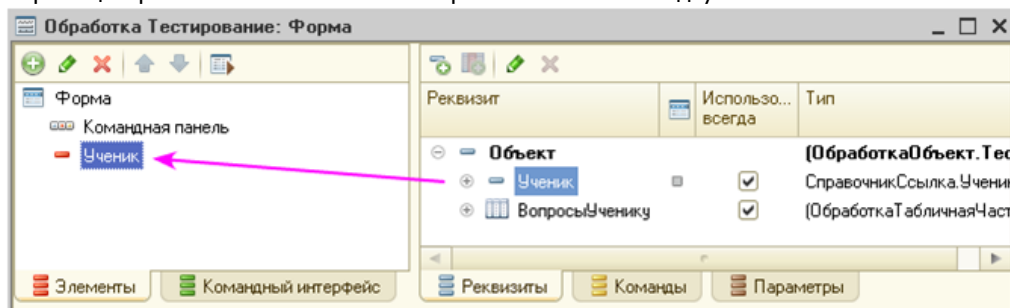
- *НомерВопроса*, тип – *Число*, длина – 3 цифры;
- *ВопросУченику*, тип – *СправочникСсылка.Вопросы*;
- *Результат*, тип – *Булево*.



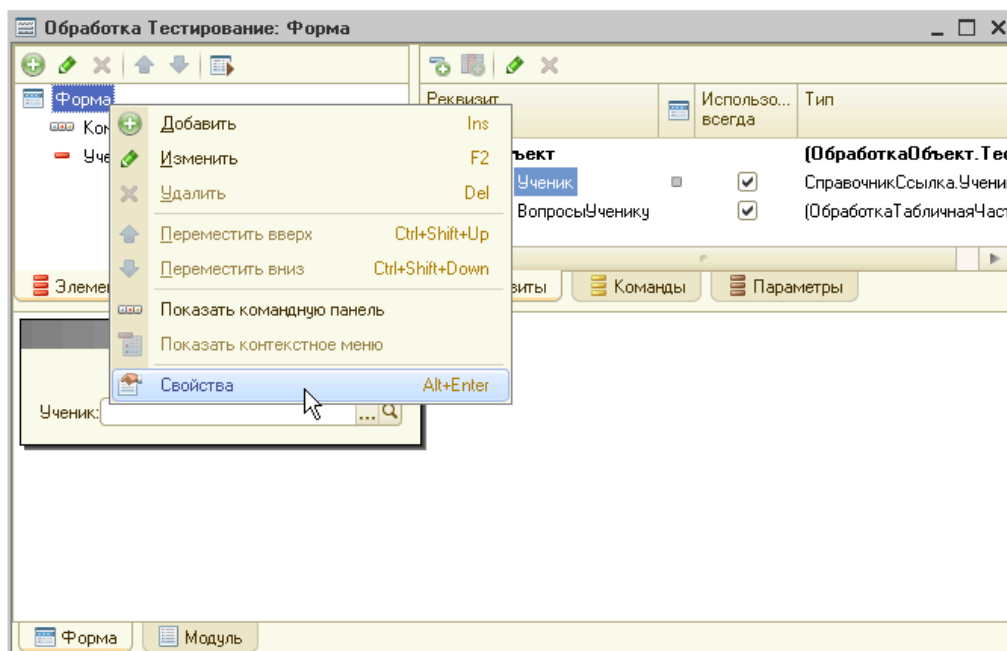
6. Перейдем на закладку *Формы* и с помощью кнопки просмотра поля *Форма обработки* создадим новую форму:



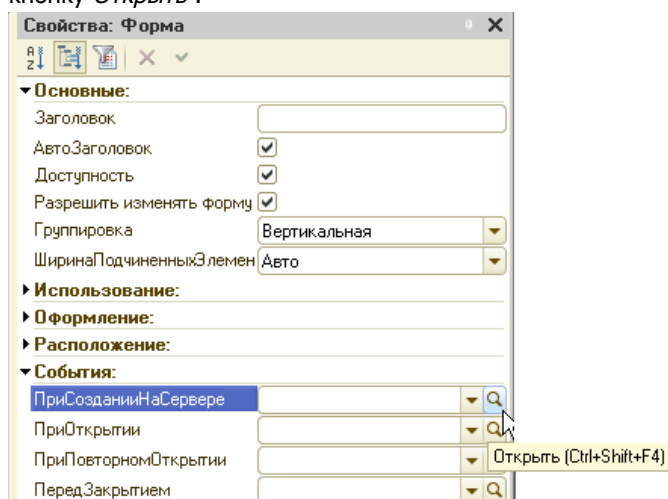
7. В открывшемся окне редактирования формы раскроем в правом верхнем окне на закладке *Реquisиты* *Объект* и перетащим реquisит *Ученик* в левое верхнее окно на закладку *Элементы* :



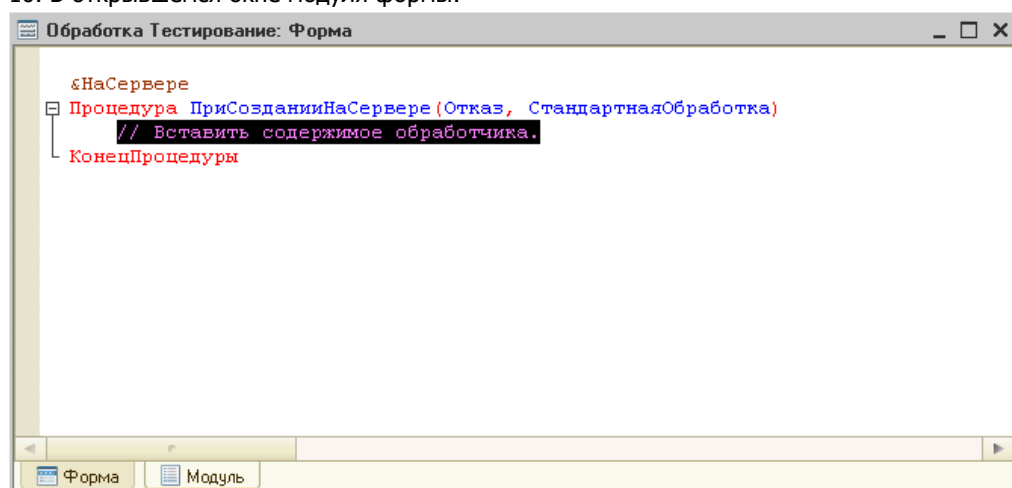
8. Затем откроем свойства формы, нажав правой клавишей мыши на верхней строке окна *Элементы* и выбрав пункт *Свойства* :



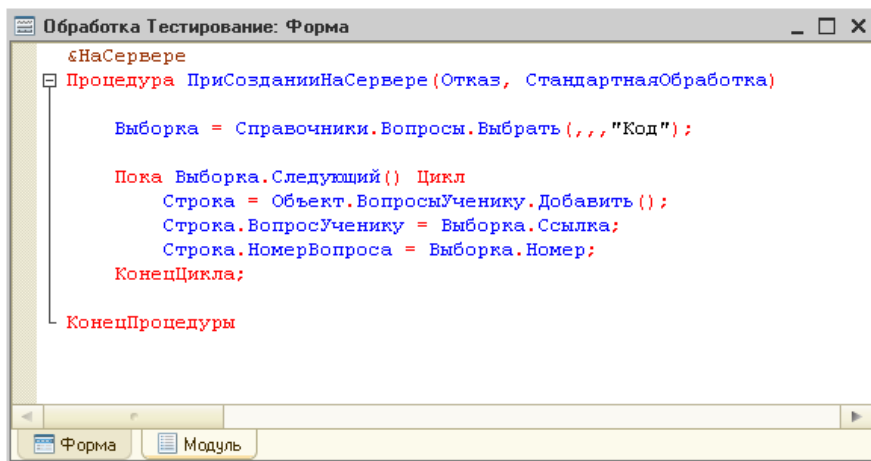
9. В нижней части окна свойств в группе свойств *События* найдем событие *ПриСозданииНаСервере* и нажмем на кнопку *Открыть* :



10. В открывшемся окне модуля формы:

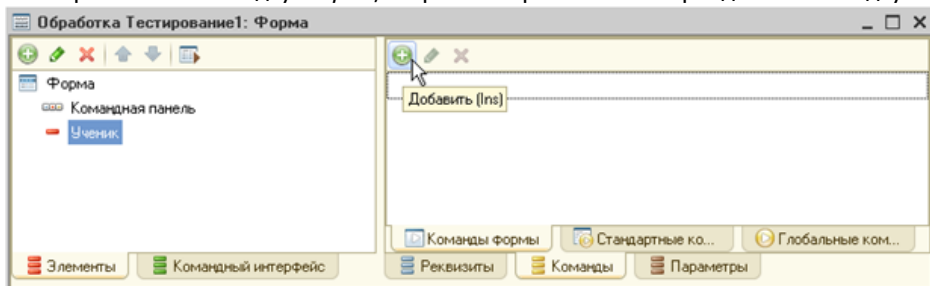


Напишем текст процедуры, которая будет выбирать из информационной базы все элементы справочника *Вопросы* и добавлять данные в табличную часть:

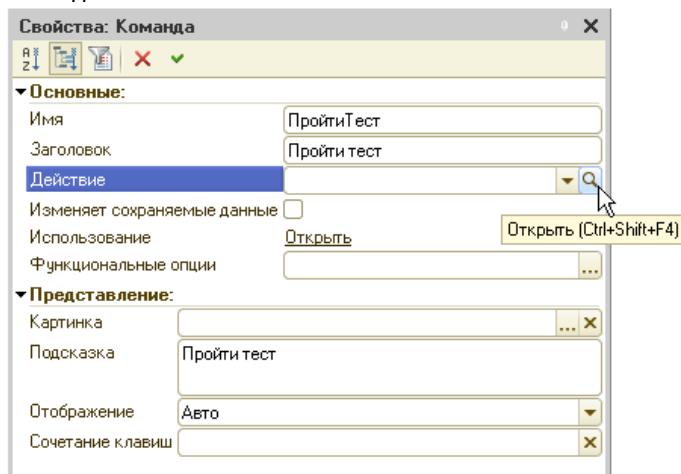


Скачать листинг процедуры

11. Вернемся на закладку *Форма*, в правом верхнем окне перейдем на закладку *Команды* и добавим новую команду:



12. Назовем эту команду *ПройтиТест* и нажатием на кнопку просмотра свойства *Действие* создадим обработчик этой команды:



Напишем текст обработчика команды:

```

    &НаКлиенте
    Процедура ПройтиТест(Команда)

        Для каждого Строка Из Объект.ВопросыУченику Цикл
            // Для каждого вопроса выполняем функцию,
            //результат которой сохраняем в строке ТЧ

            Если Строка.НомерВопроса = 1 Тогда

                Результат = Вопрос1();

            ИначеЕсли Строка.НомерВопроса = 2 Тогда

                Результат = Вопрос2();

            КонецЕсли;

            Строка.Результат = Результат;

            Если Результат Тогда
                ТекстСообщения = "Ответ правильный!";
            Иначе
                ТекстСообщения = "Ответ не правильный";
            КонецЕсли;

            Сообщение = Новый СообщениеПользователю();
            Сообщение.Текст = ТекстСообщения;

            Сообщение.Сообщить();

        КонецЦикла;

        ЗаписатьРезультаты();

    КонецПроцедуры

```

Скачать листинг процедуры

Данная процедура в цикле перебирает все вопросы, которые имеются в справочнике *Вопросы*, и в зависимости от номера вопроса вызывает на выполнение несуществующие пока функции *Вопрос1* и *Вопрос2*.

Что должны делать эти функции? Они должны задавать вопрос ученику в форме обработки, анализировать ответ, возвращать результат *Истина*, если ответ правильный, *Ложь* – если неверный.

13. Для проверки работоспособности обработки вместо настоящих функций задающих вопросы мы сделаем «заглушки». Первая функция всегда будет возвращать *Истина*, вторая – *Ложь*.

В дальнейшем мы будем добавлять отработку и другие вопросы, т.е. добавлять функции *Вопрос3*, *Вопрос4* и т.д. Пока для проверки достаточно двух вопросов:

```

    &НаКлиенте
    Функция Вопрос1()

        Возврат Истина;

    КонецФункции // Вопрос1()

    &НаКлиенте
    Функция Вопрос2()

        Возврат Ложь;

    КонецФункции // Вопрос2()

```

Поместим этот текст в модуле формы, ниже текста процедуры.

Для чего нужны и что означают директивы компиляции &НаКлиенте и &НаСервере?

14. Наша обработка *Тестирование* должна выполнять следующие действия:

- прочитать содержимое справочника *Вопросы*;
- задать соответствующие вопросы ученику;
- записать результаты в регистр сведений.

Для уменьшения количества вызовов сервера мы организовали хранение вопросов и результатов ответа на них в табличной части обработки. Это позволило прочитать все вопросы из справочника за одно обращение к серверу, затем в цикле без

обращения к серверу задать эти вопросы ученику. После завершения цикла мы запишем результаты, сохраненные в табличной части, в регистр сведений. Для этого создадим в модуле формы серверную процедуру:

```
НаСервере
Процедура ЗаписатьРезультаты()
    //Создаем набор записей регистра сведений
    Набор = РегистрыСведений.РезультатыТестирования.СоздатьНаборЗаписей();

    Набор.Очистить();
    Набор.Отбор.Ученик.Значение = Объект.Ученик;
    Набор.Отбор.Ученик.Использование = Истина;

    Для каждого Строка Из Объект.ВопросыУченику Цикл

        // Добавляем в набор одну запись
        Запись = Набор.Добавить();
        Запись.Ученик = Объект.Ученик;
        Запись.Вопрос = Строка.ВопросУченику;
        Запись.Результат = Строка.Результат;

    КонецЦикла;

    // Записываем набор записей в базу данных
    Набор.Записать();

КонецПроцедуры
```

[Скачать листинг процедуры](#)

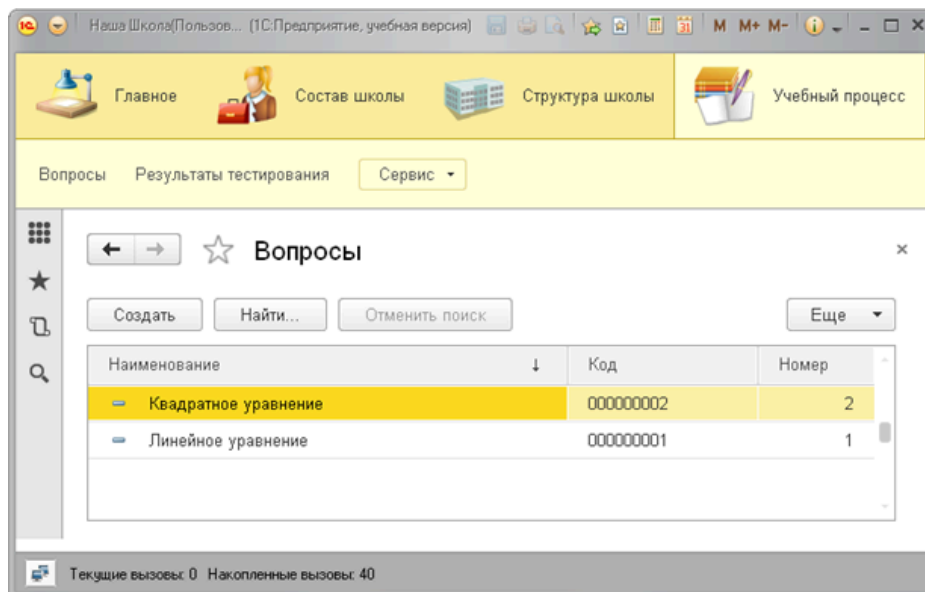
Задание 4

Проверьте работу полученного механизма.

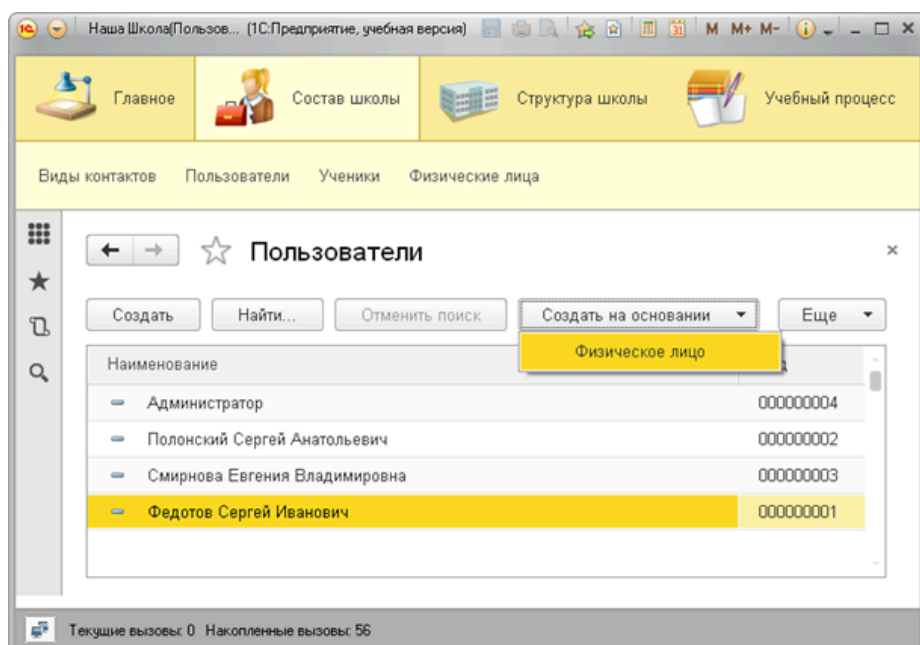
Запустим программу в режиме «1С:Предприятие». Перейдем в раздел *Учебный процесс*, с помощью команды панели навигации откроем справочник *Вопросы*.

Добавим в справочник *Вопросы* добавим наименования двух вопросов:

- Линейное уравнение, номер вопроса – 1;
- Квадратное уравнение, номер вопроса – 2.



Затем перейдем в раздел *Состав школы*, откроем справочник *Пользователи* и введем на основании пользователей два элемента справочника *Физические лица* :



Заполним недостающие реквизиты:

Физическое лицо (создание) *

Записать и закрыть Записать Еще

Код:

Наименование:

Группа: ...

Фамилия:

Имя:

Отчество:

Дата рождения:

Пол: ...

Добавить Еще

	Вид контакта	Значение
1	Адрес	Севастопольская ул., д. 51, корп. 2, кв. 201

Откроем справочник *Ученики* и добавим двух учеников, при этом *Наименование* вручную не пишем, а выбираем из справочника реквизит *Физическое лицо*. Наименование должно заполниться автоматически:

Полонский Серге... (1С:Предприятие)
Федотов Сергей ... (1С:Предприятие)

Полонский Сергей Анатолье (Ученики) *

Федотов Сергей Иванович (Ученики)

Главное Результаты тестирования
Главное Результаты тестирования

Записать и закрыть

Записать

Еще ▾

Записать и закрыть

Записать

Еще ▾

Код: 000000002

Код: 000000003

Наименование: Полонский Сергей Анатольевич

Наименование: Федотов Сергей Иванович

Группа:

...

Группа:

...

Физическое лицо: Полонский Сергей Анатольевич

...

Физическое лицо: Федотов Сергей Иванович

...

Класса: 11 6

...

Класса: 11 6

...

В разделе Учебный процесс запустим на выполнение обработку *Тестирование* :

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

Главное

Состав школы

Структура школы

Учебный процесс

Настройка навигации

Настройка действий

✕

Вопросы

Результаты тестирования

Сервис

Тестирование

Текущие вызовы: 0

Накопленные вызовы: 40

Поочередно выбирая каждого ученика, нажмем кнопку *Пройти тест* :

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

Главное

Состав школы

Структура школы

Учебный процесс

Настройка навигации

Настройка действий

✕

Вопросы

Результаты тестирования

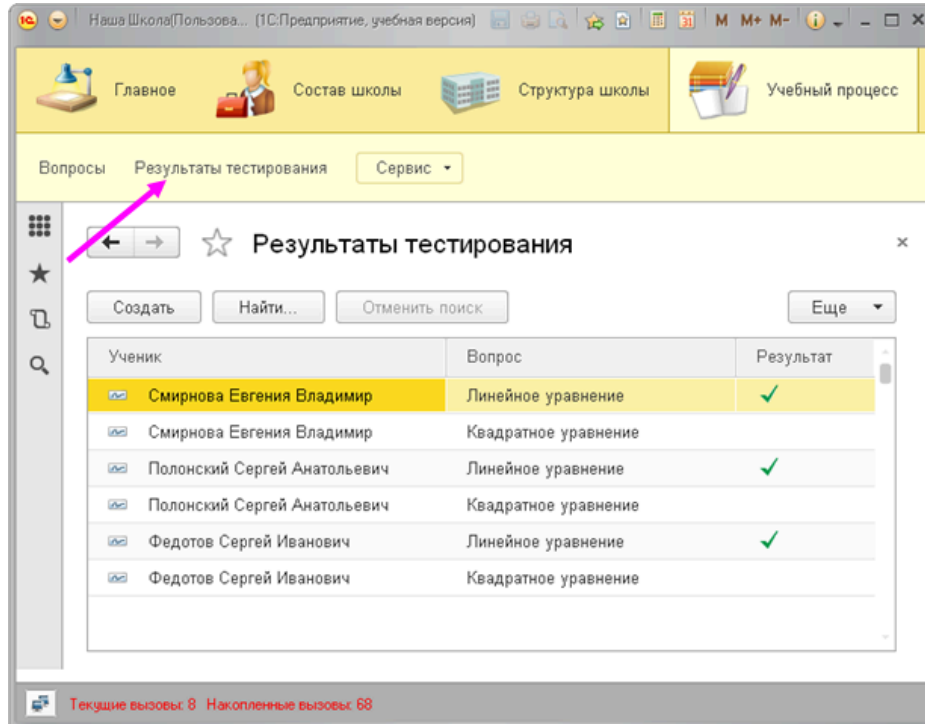
Сервис

Тестирование

Текущие вызовы: 0

Накопленные вызовы: 40

Откроем регистр сведений *Результаты тестирования*, сравним результат с рисунком:



Если Ваша информационная база, по каким-то причинам некорректно работает, то сохраните файл с **выгрузкой** и загрузите информационную базу: в режиме configurатора откройте в главном меню *Администрирование – Загрузить информационную базу* – укажите путь к сохраненному файлу.