

Федеральное государственное образовательное бюджетное учреждение высшего образования "Финансовый университет при Правительстве Российской Федерации"

Федеральное государственное автономное образовательное учреждение высшего образования "Московский физико-технический институт (национальный исследовательский университет)"

Федеральное государственное бюджетное образовательное учреждение высшего образования "Российский экономический университет имени Г.В. Плеханова"

Федеральное государственное бюджетное образовательное учреждение высшего образования "Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)"

Федеральное государственное бюджетное образовательное учреждение высшего образования "Московский государственный технологический университет "СТАНКИН"

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования "Московский технический университет связи и информатики"

Федеральное учебно-методическое объединение по укрупненной группе специальностей и направлений подготовки высшего образования "Информатика и вычислительная техника"

Фирма "1С"

Мероприятие проходит при поддержке Федерального агентства связи (Россвязь)

**XX международная научно-практическая конференция  
"Новые информационные технологии в образовании"  
(Технологии 1С: Перспективные решения для построения карьеры,  
цифровизации организаций и непрерывного обучения)  
4–5 февраля 2020 года**

**Мастер-класс:  
Разработка системы взаимодействия в 1С, на примере создания чат-бота**

**Раздаточный материал**

**г. Москва, 2020**

ПРАВО ТИРАЖИРОВАНИЯ ДОКУМЕНТАЦИИ ПРИНАДЛЕЖИТ ФИРМЕ "1С"

Получив настоящие материалы для обучения от фирмы "1С",

Вы тем самым даете согласие не допускать

их копирования без письменного

разрешения фирмы "1С".

© ООО "1С-Софт"

Москва, ул. Селезневская, д. 21

Телефон: (495) 688-90-02

Автор методических материалов: 1С:Учебный центр №1

## Оглавление

Введение .....	4
Подготовительный этап .....	4
Регистрация отзыва .....	12
Регистрация в системе взаимодействия .....	18
Бизнес-процессы .....	24
Информирование .....	28
Заключение .....	48

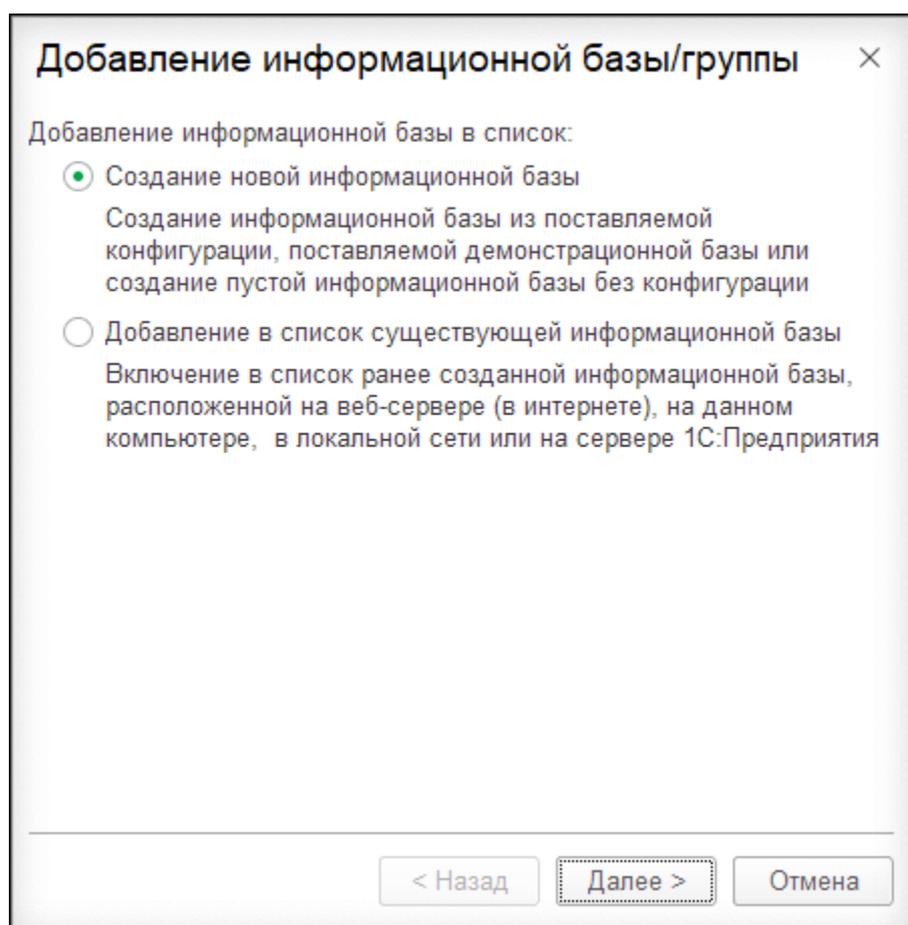
## Введение

На этом мастер-классе я покажу совсем нестандартное использование системы 1С:Предприятие. Я создам приложение-бот для выбора и оценки увеселительных заведений студентами.

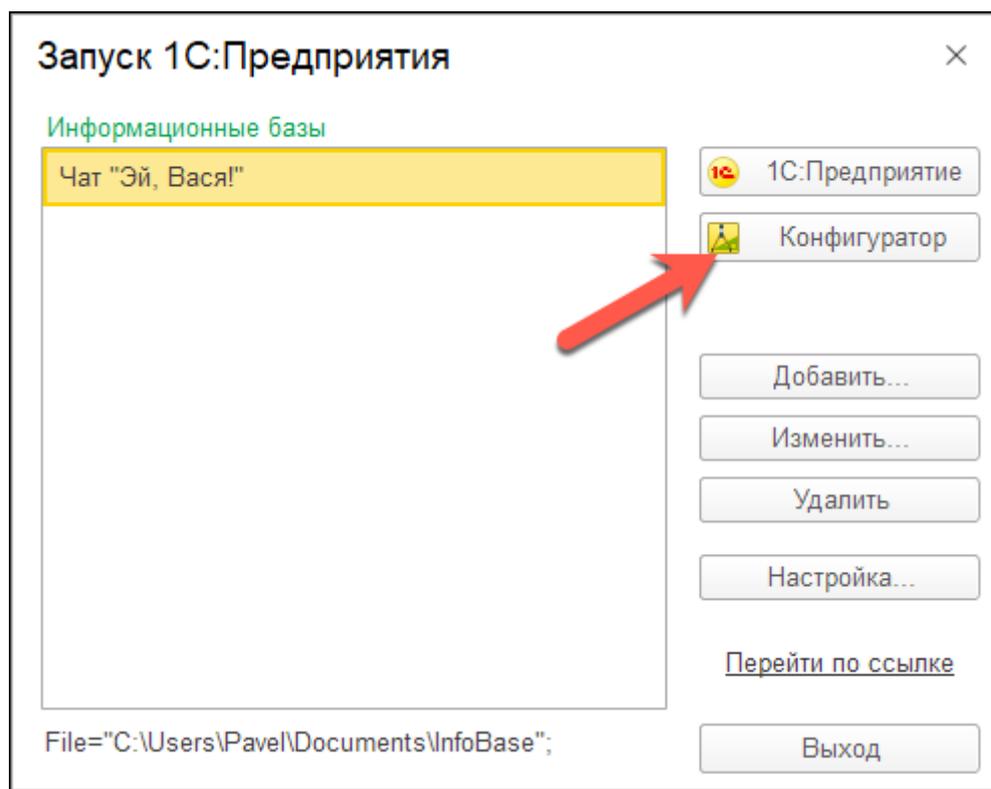
Задача звучит так: Студенты периодически собираются хорошо провести время после занятий и выбирают куда-бы сходить. Критериев оценки может быть бесконечно много, но мы возьмем три: средняя оценка посетителей (таких же студентов), средний счет, расстояние от УЗ. Разработаем что-то вроде «Алисы» от Яндекса. Назовем все это «Эй, Вася!»

## Подготовительный этап

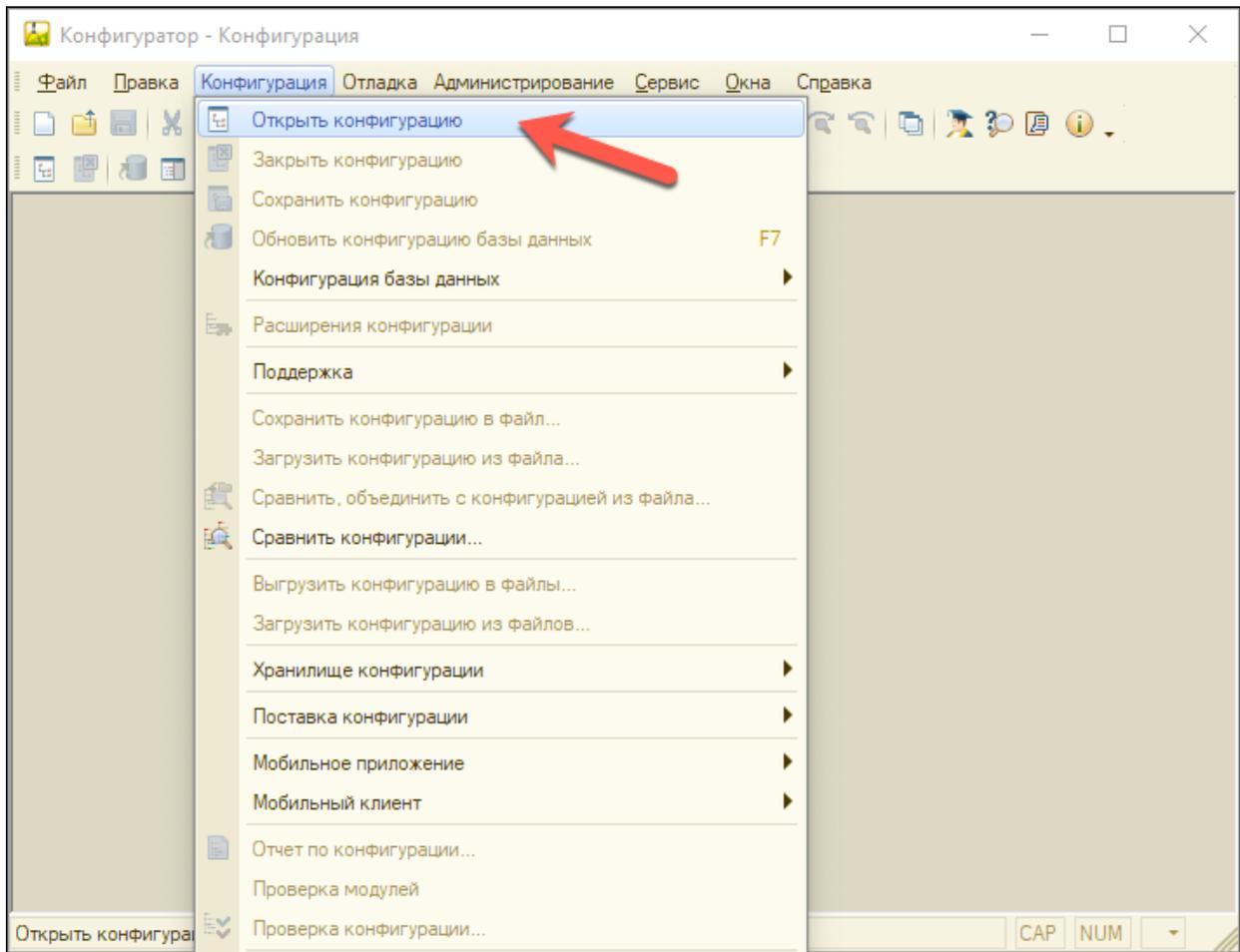
Так как мы делаем автономную систему, начнем с нуля. Создадим новую информационную базу.



Базу создадим в файловом варианте, назовем все это «Чат «Эй, Вася!»».

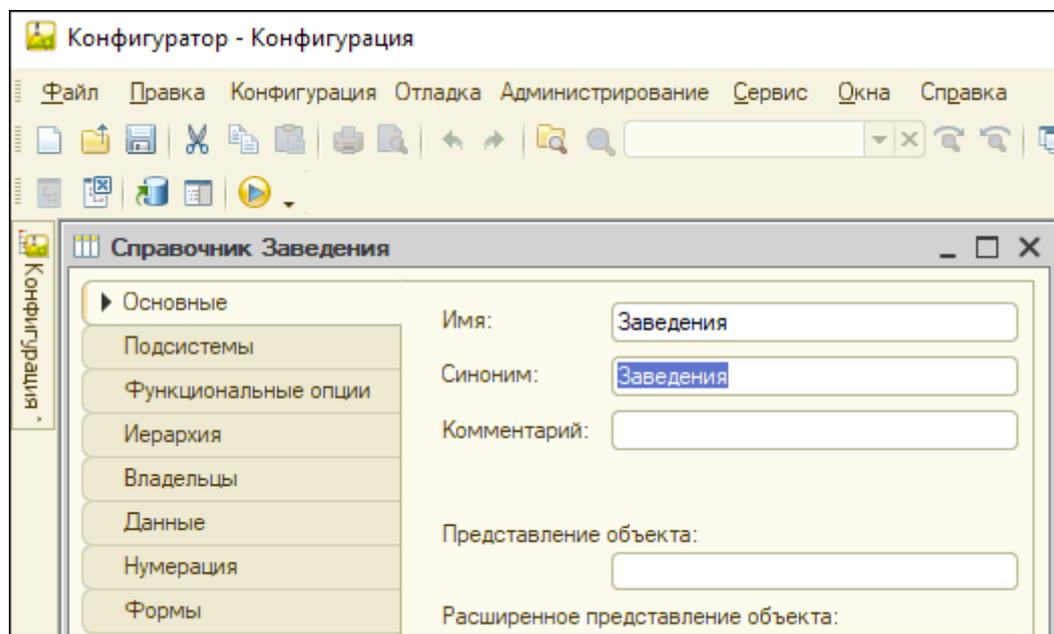
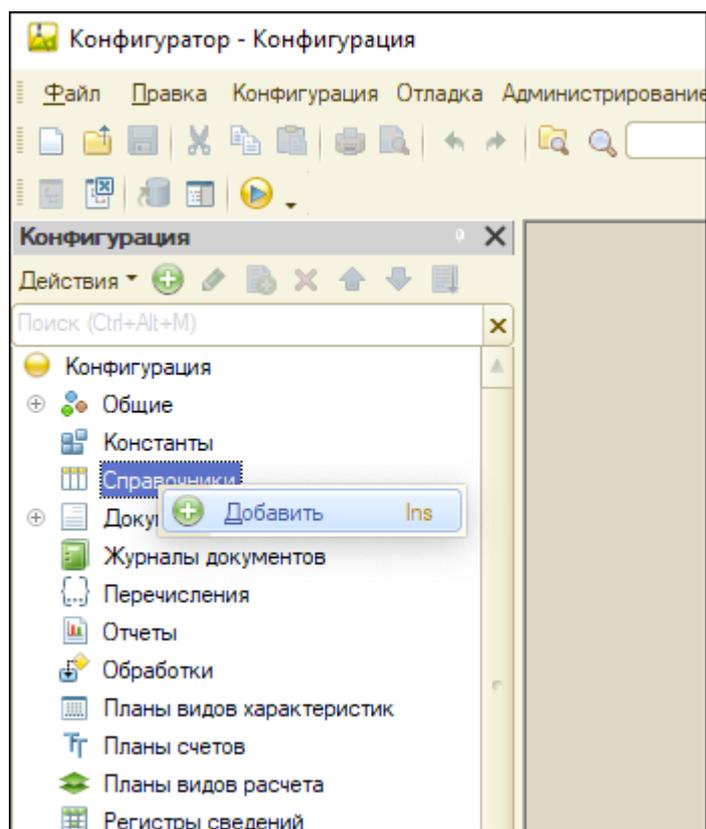


Откроем Конфигуратор и создадим объекты, которые будут хранить информацию о заведениях и рейтингах.

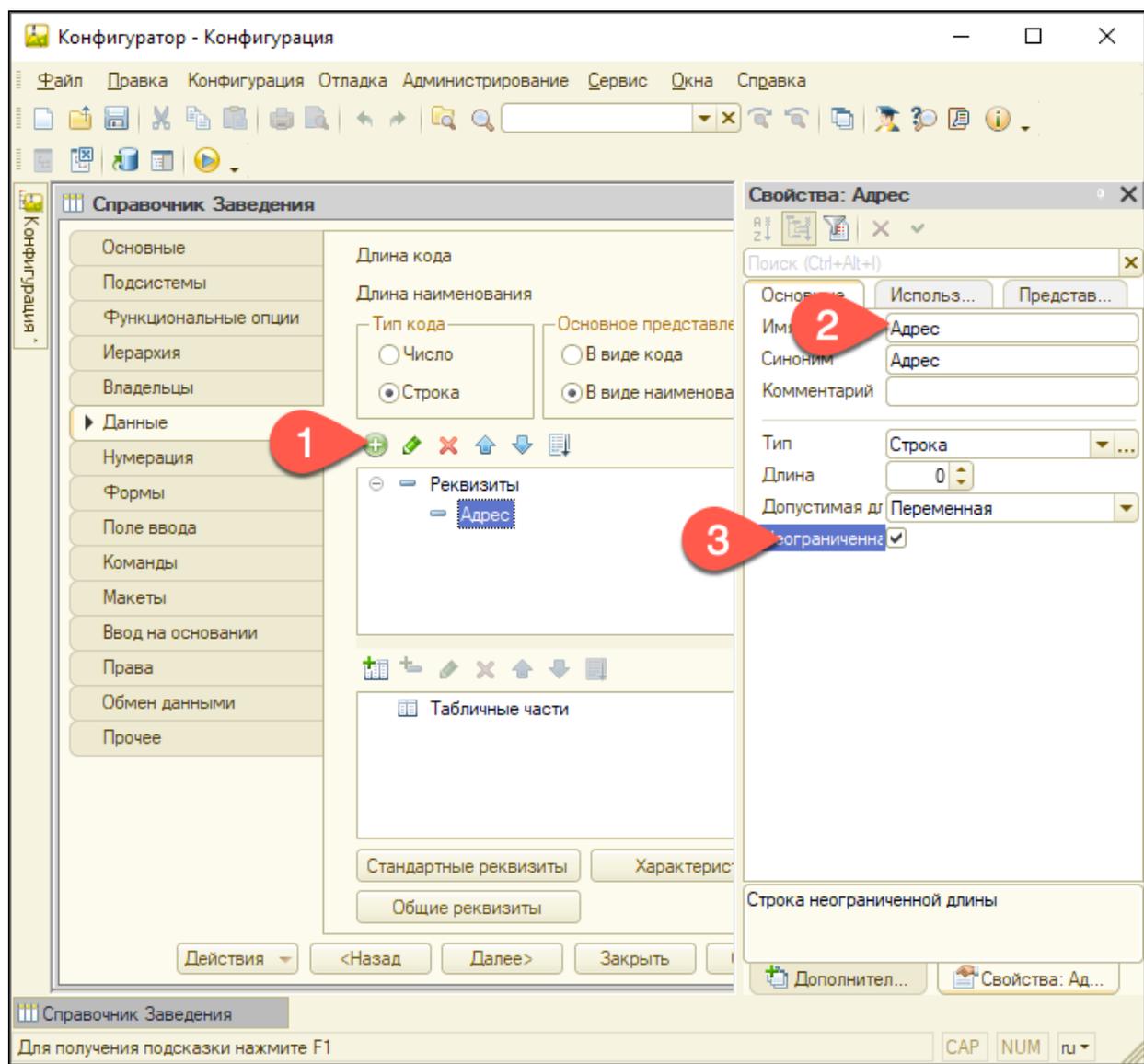


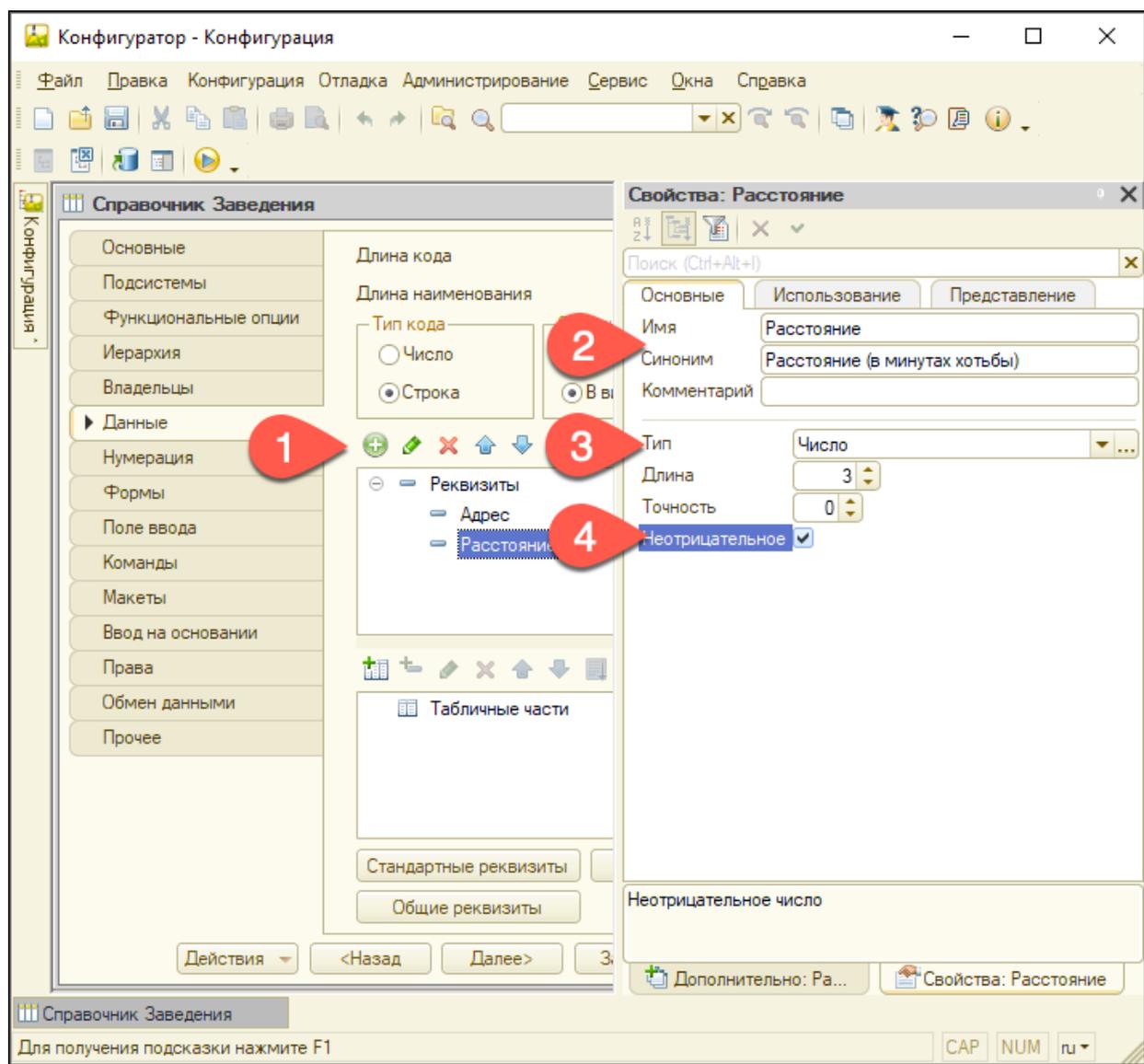
Откроем конфигурацию.

Добавим справочник «Заведения»



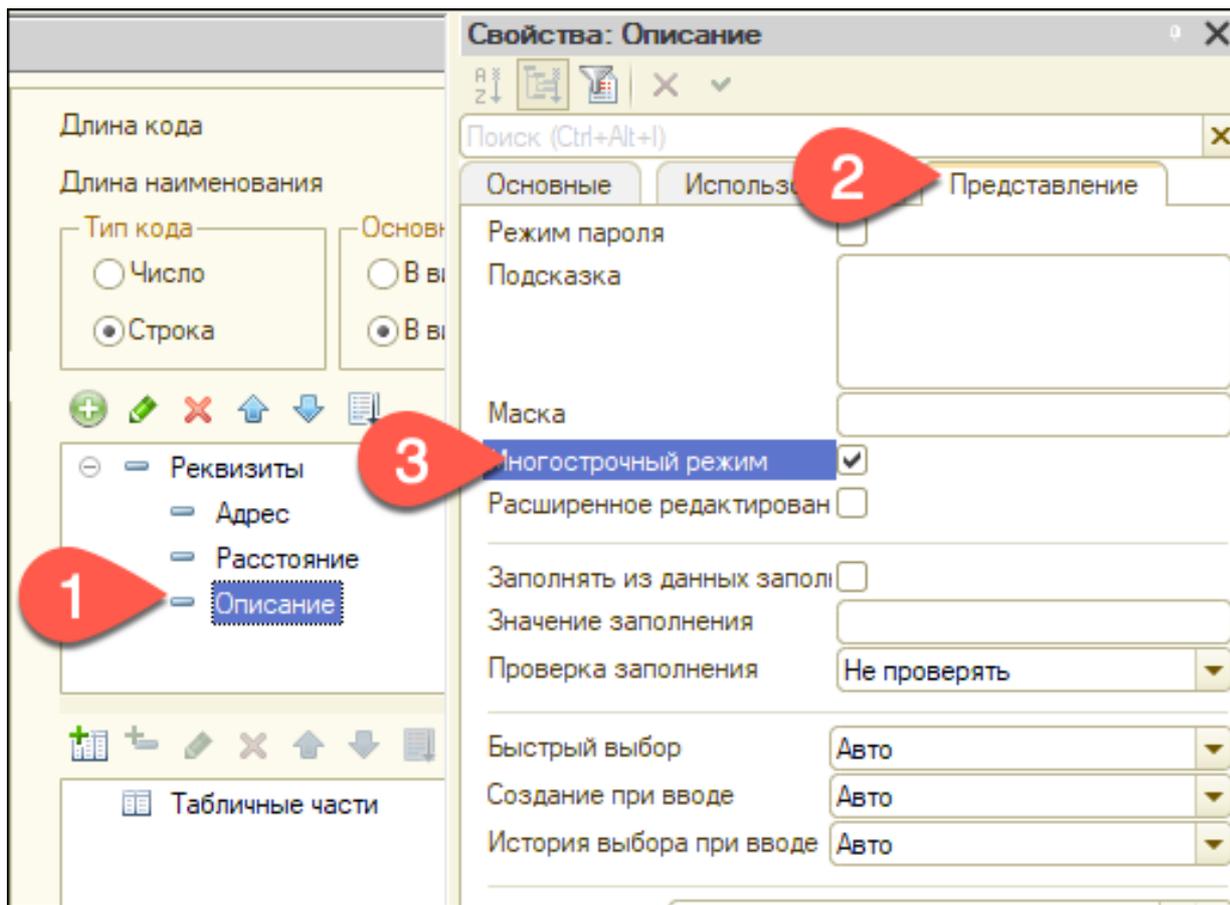
Сохраним информацию о адресе заведения, и расстоянии от УЗ.



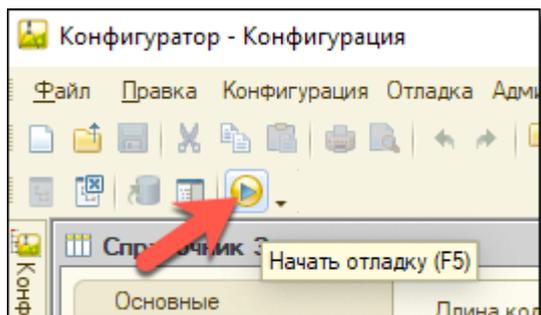


И общее описание заведения.

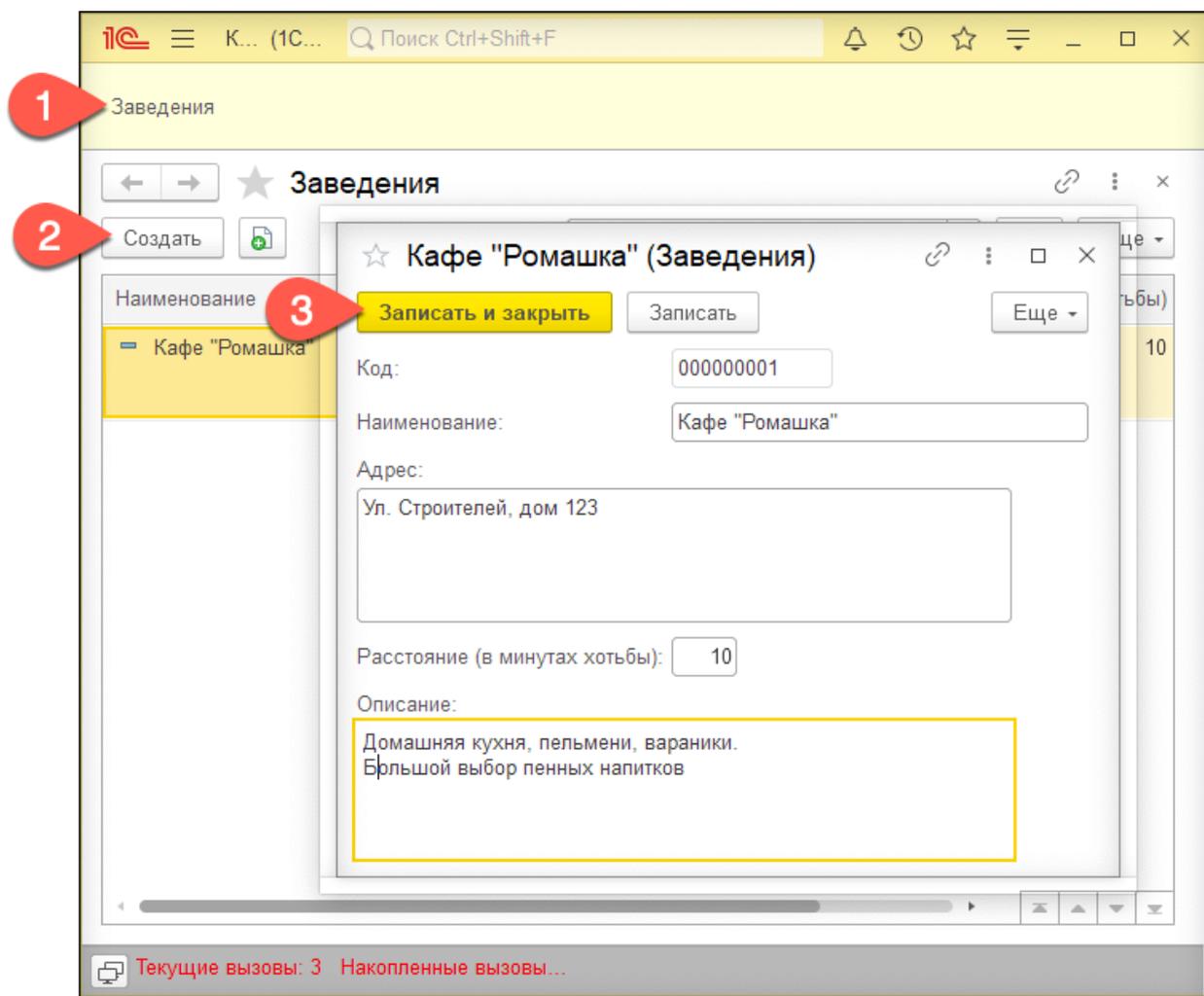




Протестируем в режиме 1С:Предприятие как выглядит форма для ввода данных о заведениях.



Запустим отладку и откроем форму элемента справочника.

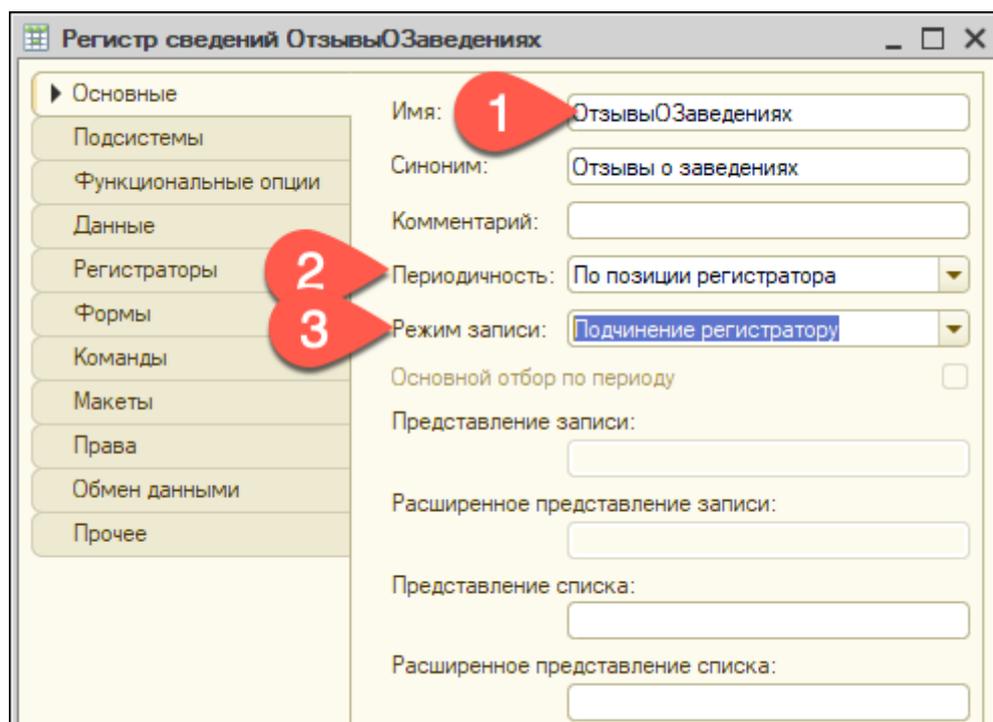


Форму рисовать не будем, стандартная вполне нам подходит.

## Регистрация отзыва

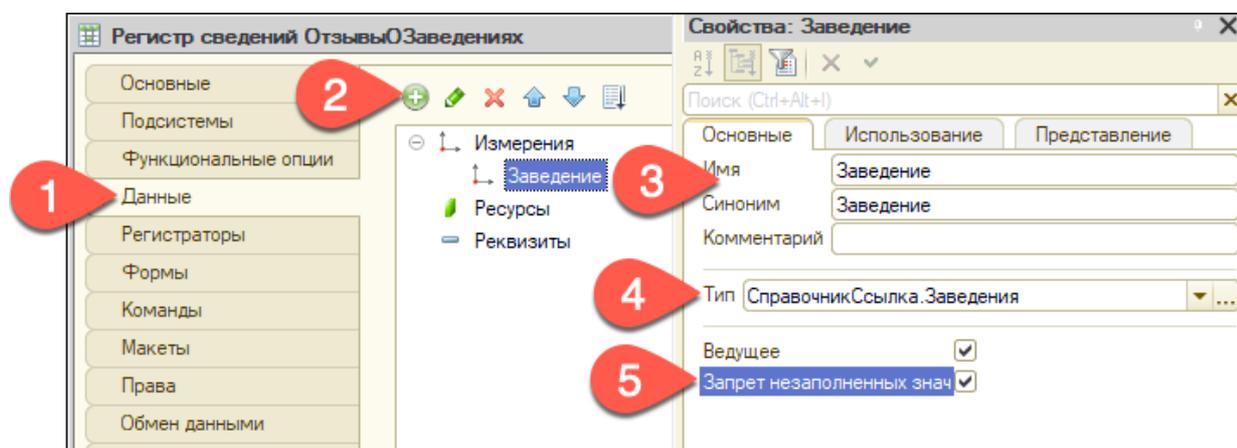
Для регистрации отзывов будем использовать регистры сведений и документы.

Создадим регистр сведений «Отзывы о заведениях».

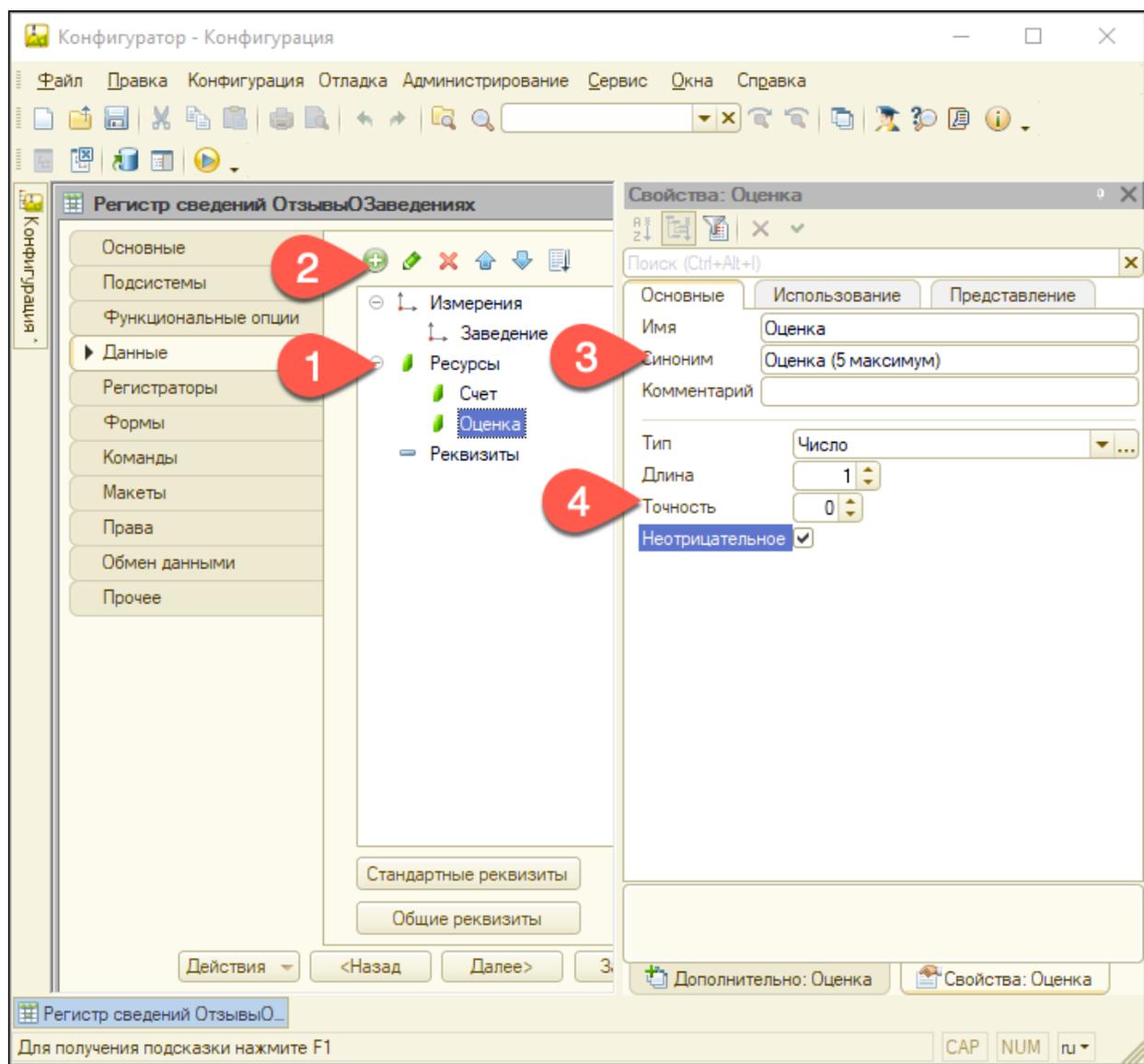


Так как регистрировать отзывы мы хотим событийно, то установим подчинение регистратору.

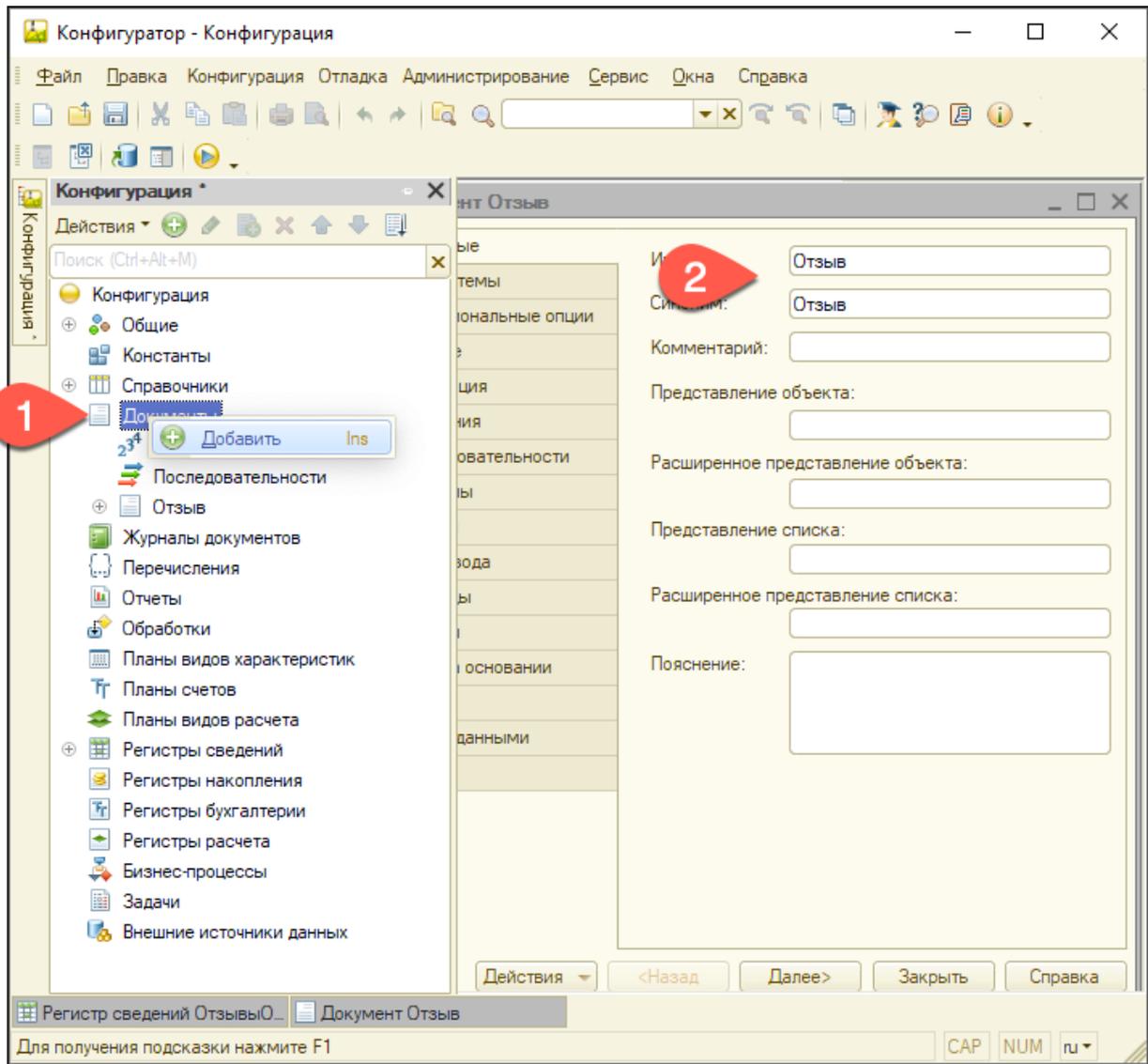
На вкладке «Данные» добавим измерение – Заведение.



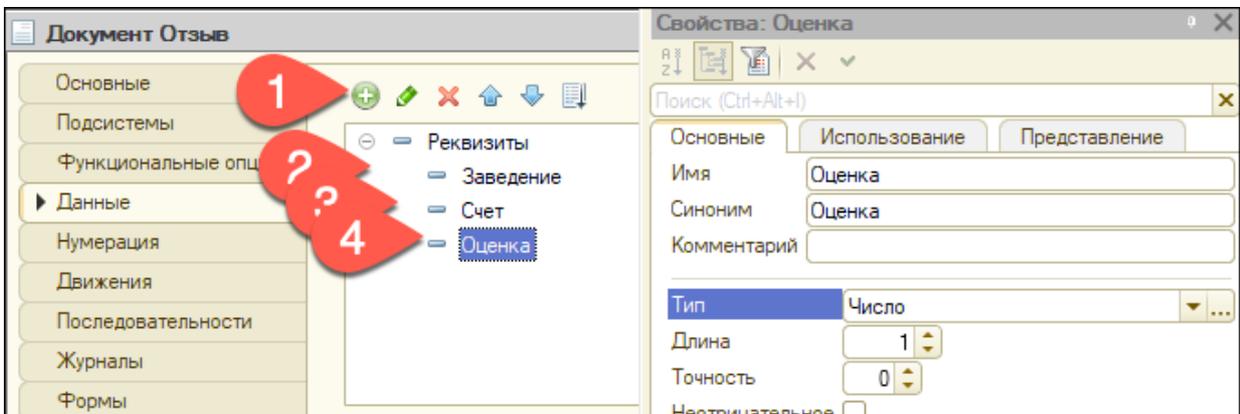
И два ресурса: Счет – это число длина 10, и оценка – так же число, но длина 1.



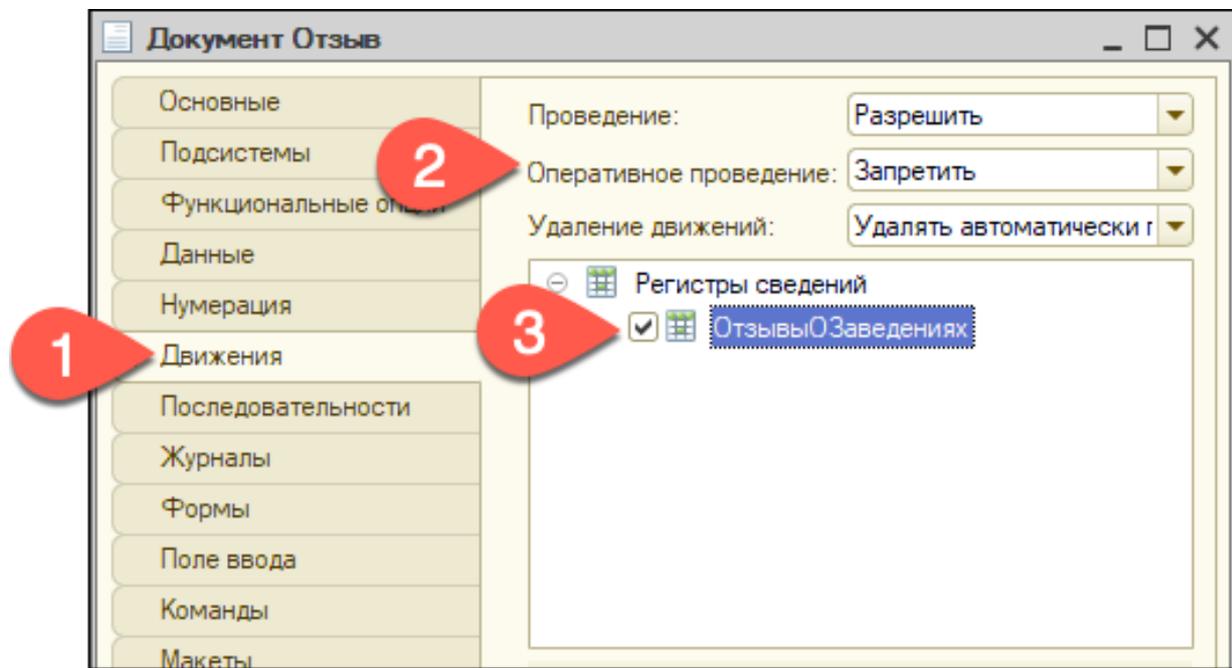
Когда объекты для хранения информации готовы создадим документ для регистрации отзыва.



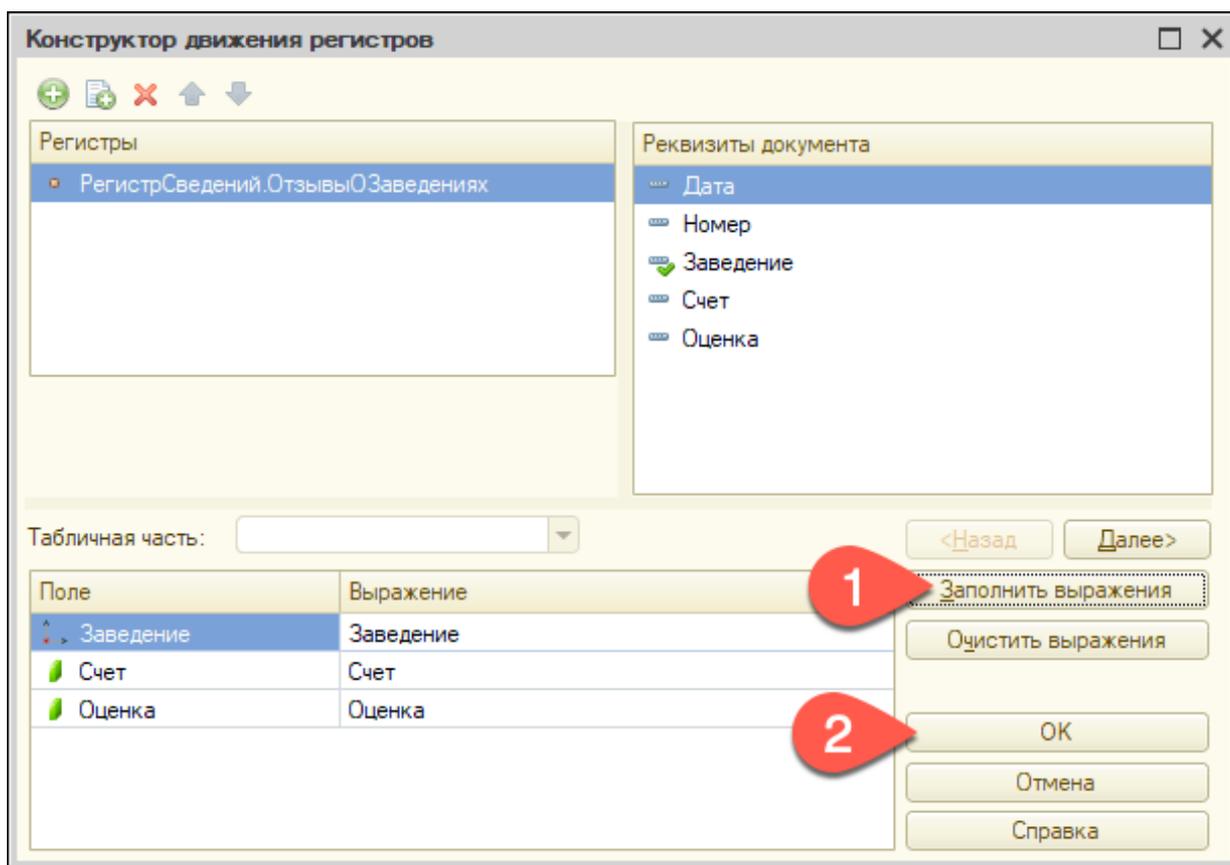
Добавим реквизиты. По аналогии с регистром: Заведение, Счет, Оценка.



На вкладке «Движения» опишем по каким регистрам может совершать записи наш документ. Регистр у нас один – не промахнетесь 😊



И запустим конструктор движений.



В результате сформируется код проведения документа.

```
□ Процедура ОбработкаПроведения(Отказ, Режим)
  {{{_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
  // Данный фрагмент построен конструктором.
  // При повторном использовании конструктора, внесенные

  // регистр ОтзывыОЗаведениях
  Движения.ОтзывыОЗаведениях.Записывать = Истина;
  Движение = Движения.ОтзывыОЗаведениях.Добавить ();
  Движение.Период = Дата;
  Движение.Заведение = Заведение;
  Движение.Счет = Счет;
  Движение.Оценка = Оценка;

  }}}_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
  КонецПроцедуры
```

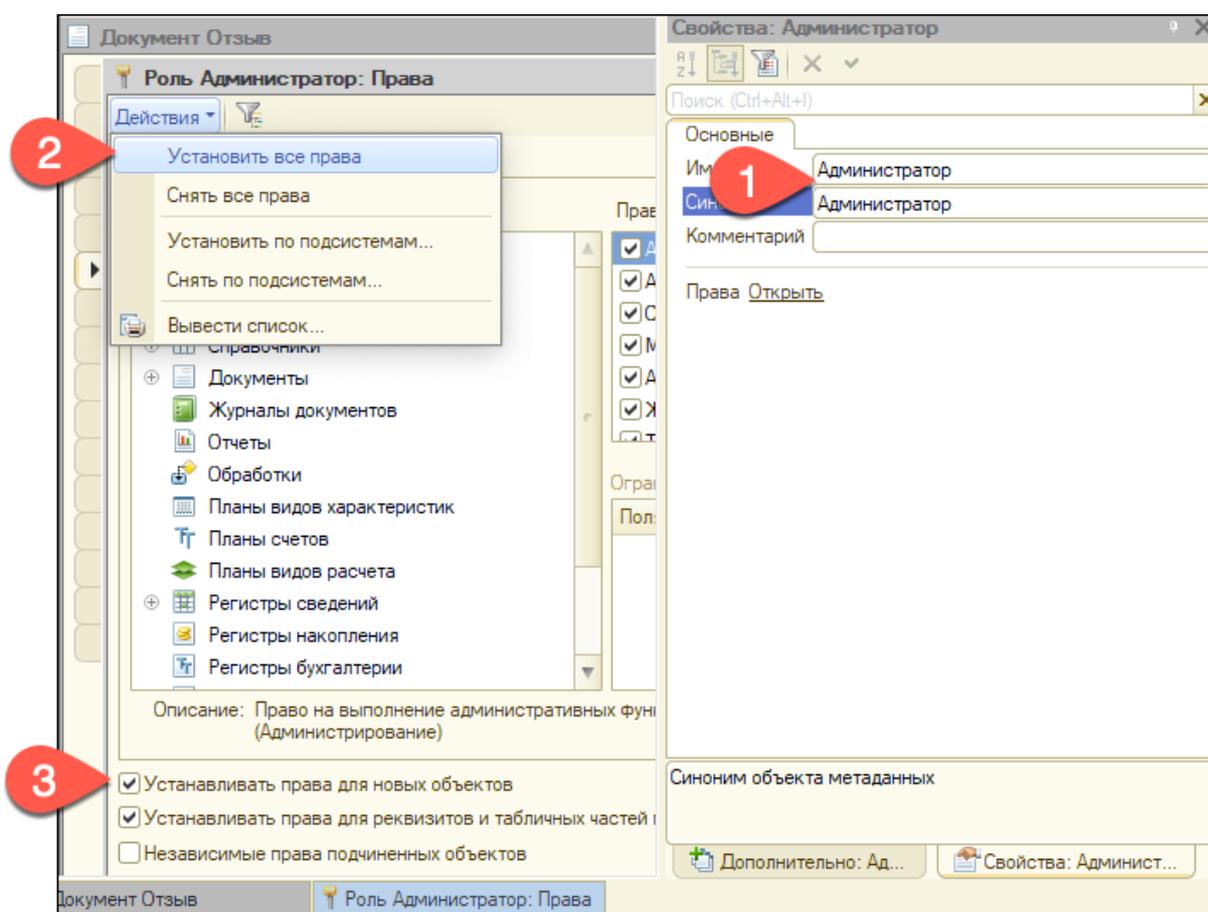
Механизм об оценке заведений готов. Пора приступить к созданию бота.

## Регистрация в системе взаимодействия

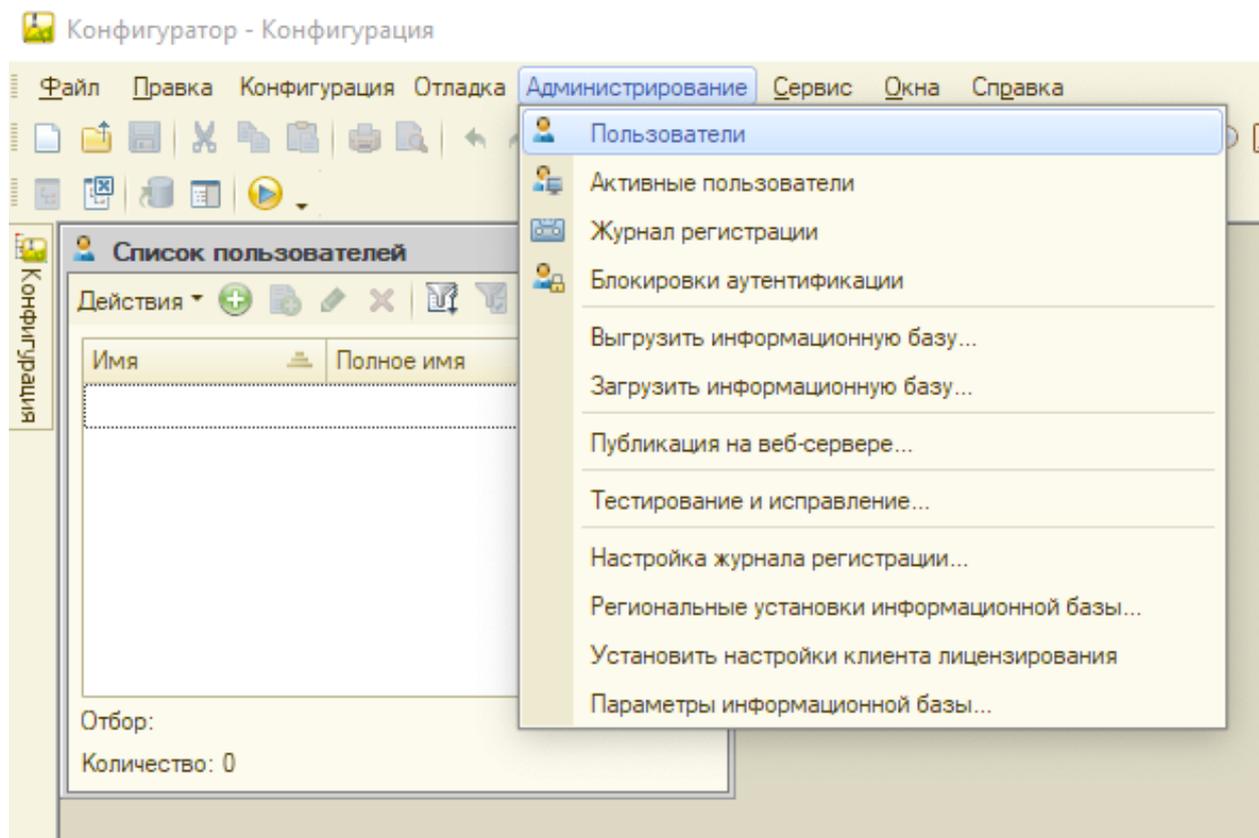
Приступим к созданию бота.

Нам нужно создать «Васю» и того, кто с ним будет общаться.

Создадим роль «Администратор», без этого мы не сможем создавать пользователей.



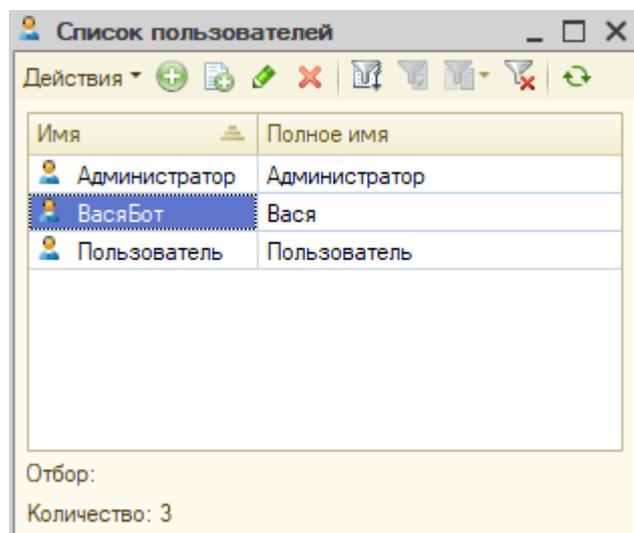
Обязательно сохраняем конфигурацию базы данных (F7) и создаем пользователей.



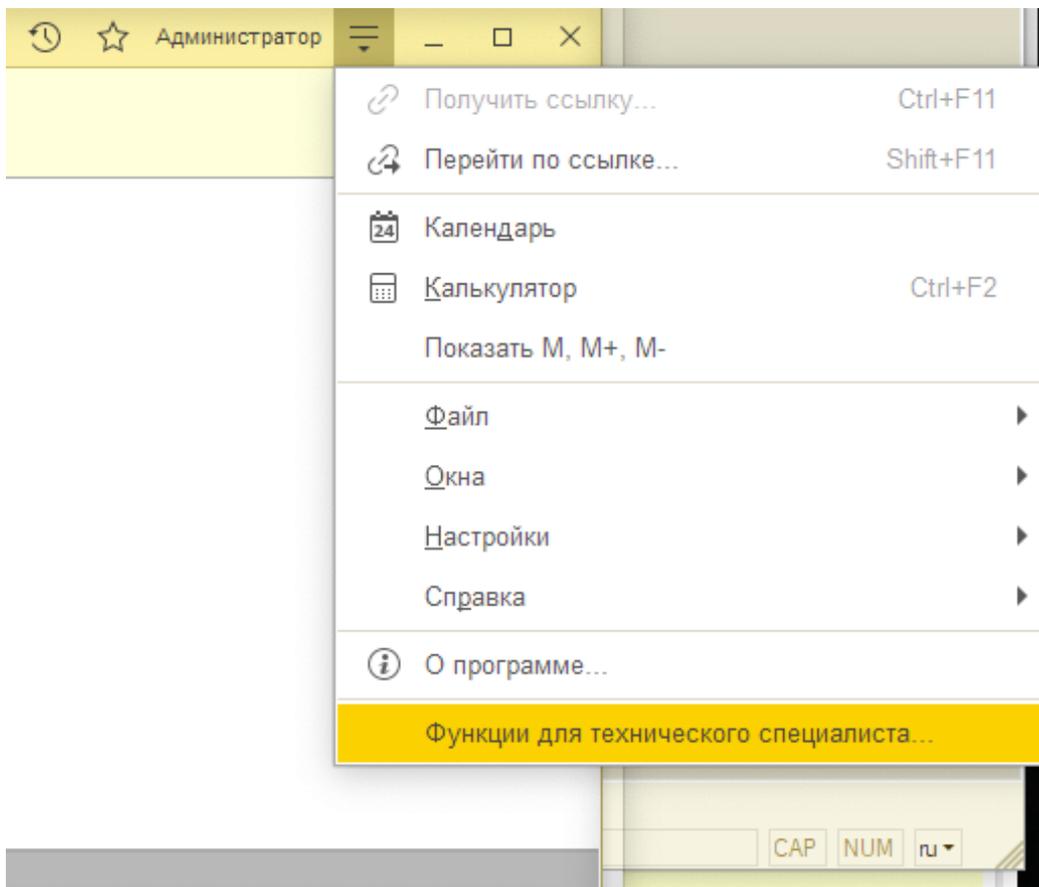
Нам нужны пользователи:

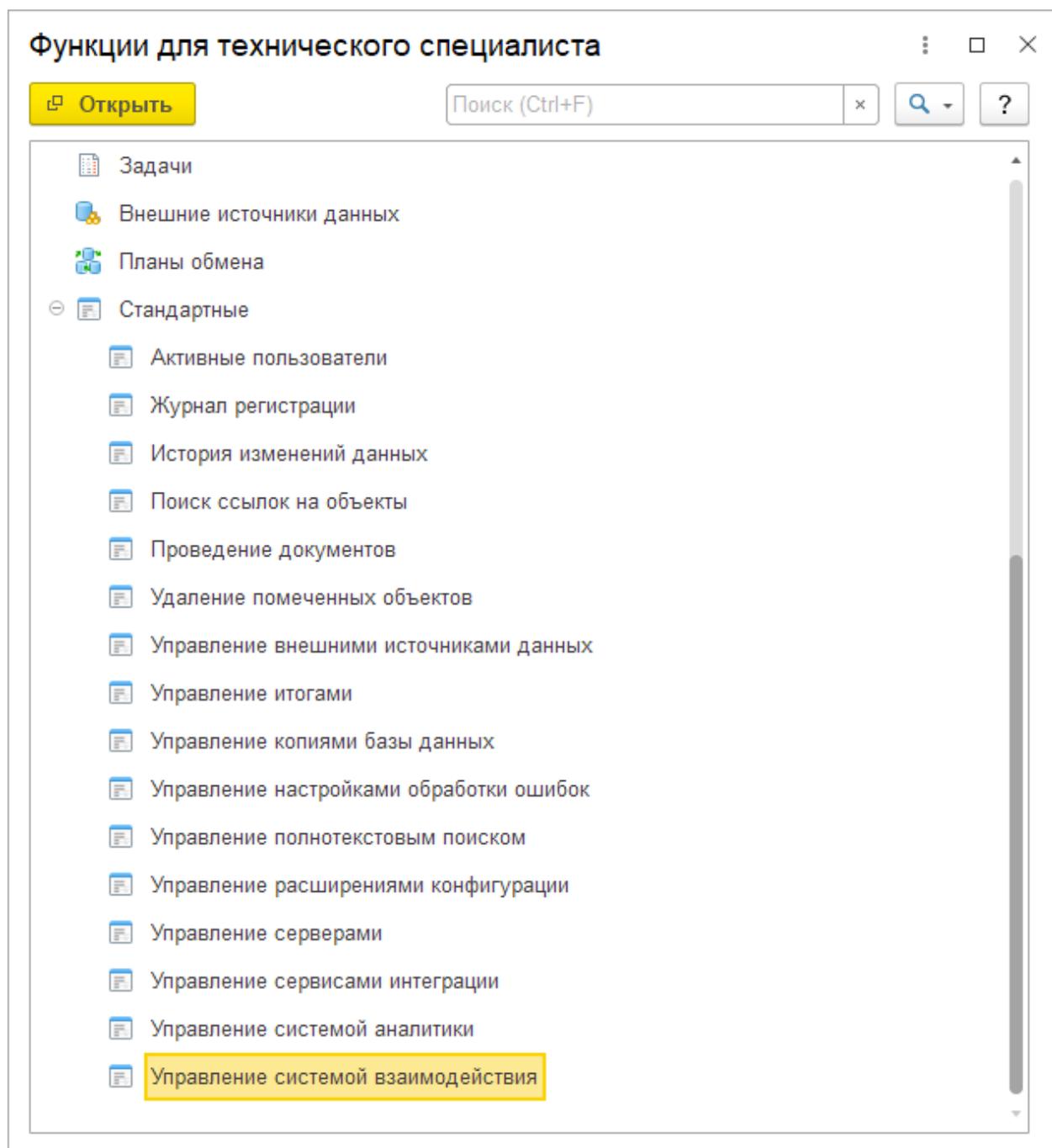
- Администратор
- Бот
- Пользователь.

В правах у них различий нет. Все Администраторы.



Запустим систему откроем регистрацию системы взаимодействия.





Заведения   Отзыв

### Регистрация абонента сервиса и приложения

Адрес электронной почты абонента:

Приложение будет связано с этим адресом.  
В дальнейшем на этот адрес будут приходить коды подтверждения

Наименование приложения:

Код регистрации:  Получить код

Зарегистрировать

Текущие вызовы: 1   Накопленные вызовы: 65

### Код подтверждения регистрации информационной базы

1С:Диалог <info@1cdialog.com>  
24.01.2020 15:29

Кому: [Чистов Павел Анатольевич <pchi@1c.ru>](mailto:pchi@1c.ru)

---

 1С:Диалог ООО «1С»

---

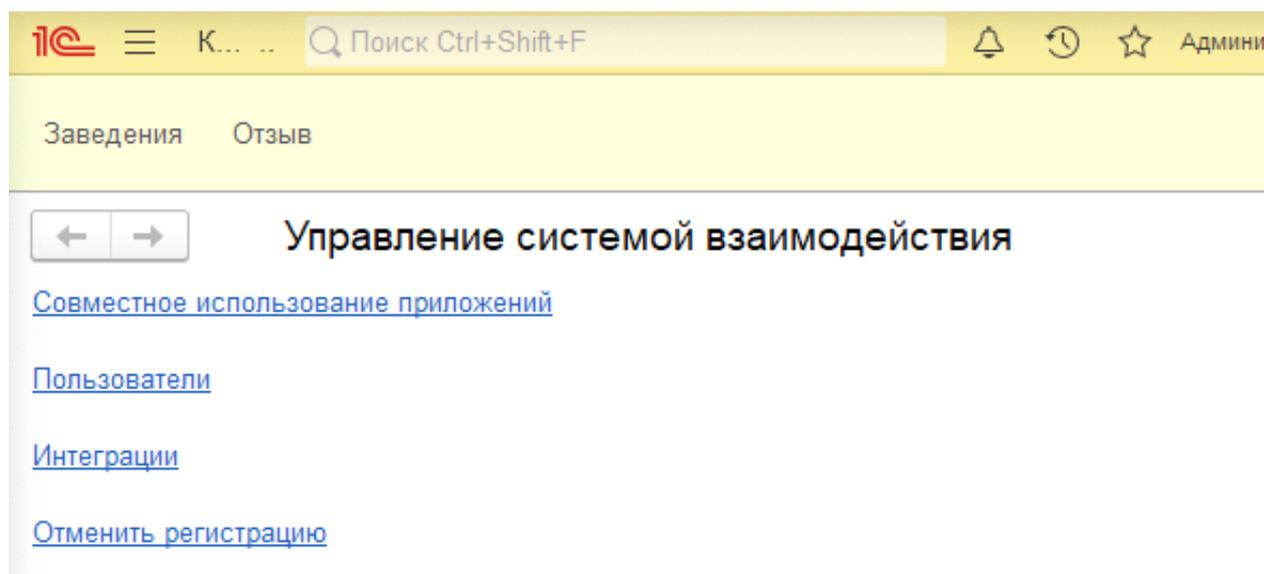
Уважаемый абонент,

для подтверждения подключения информационной базы к сервису 1С:Диалог используйте код:

59 i2

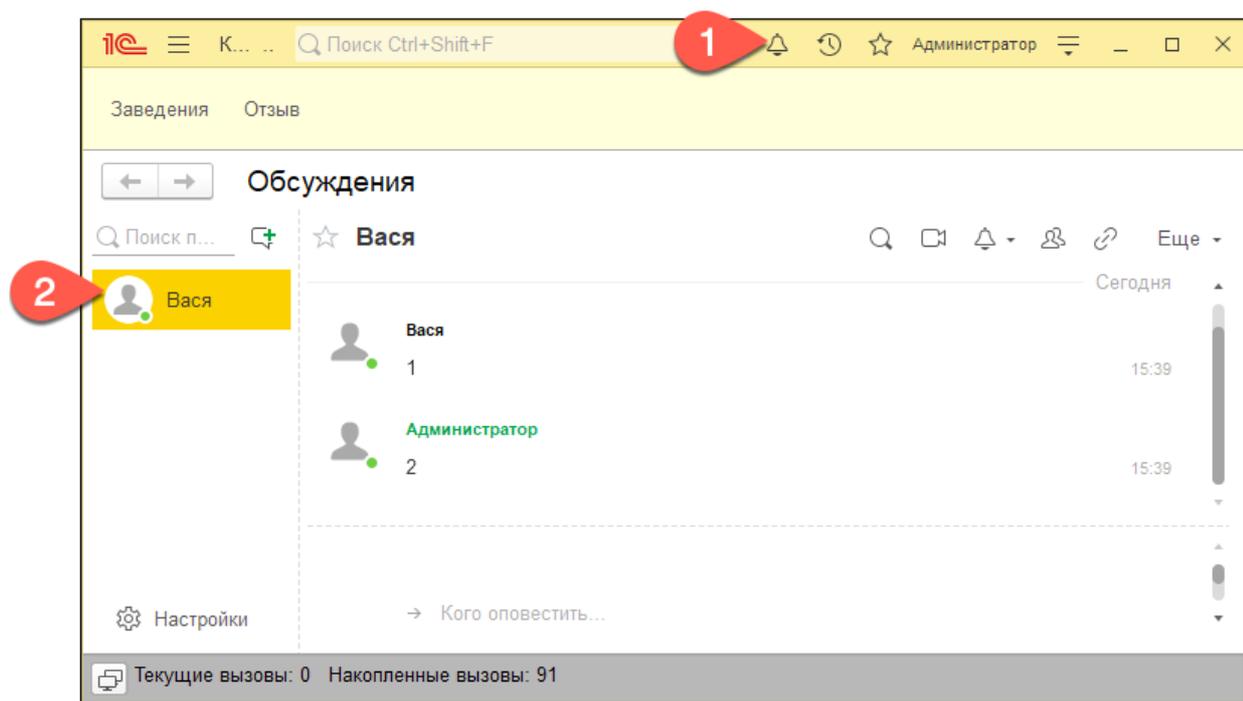
Используйте данный адрес электронной почты при подключении других информационных баз вашей организации к сервису.

По вопросам работы сервиса обращайтесь на линию консультаций фирмы «1С» по телефону (495) 688-10-01 (общий многоканальный) и электронной почте v8@1c.ru.



Регистрация завершена.

Теперь через сервис 1С:Диалог мы можем общаться между пользователями базы.



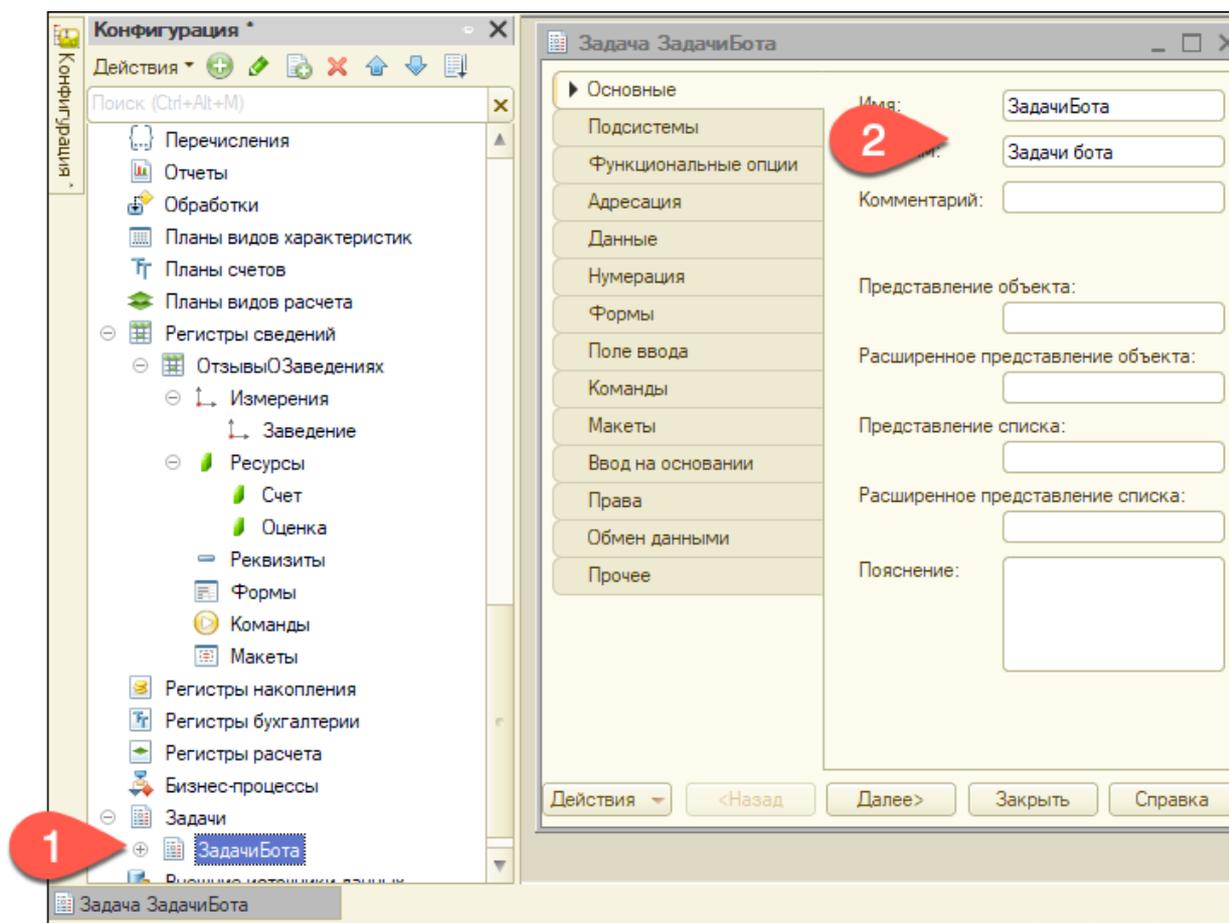
Осталось прикрутить бота.

## Бизнес-процессы

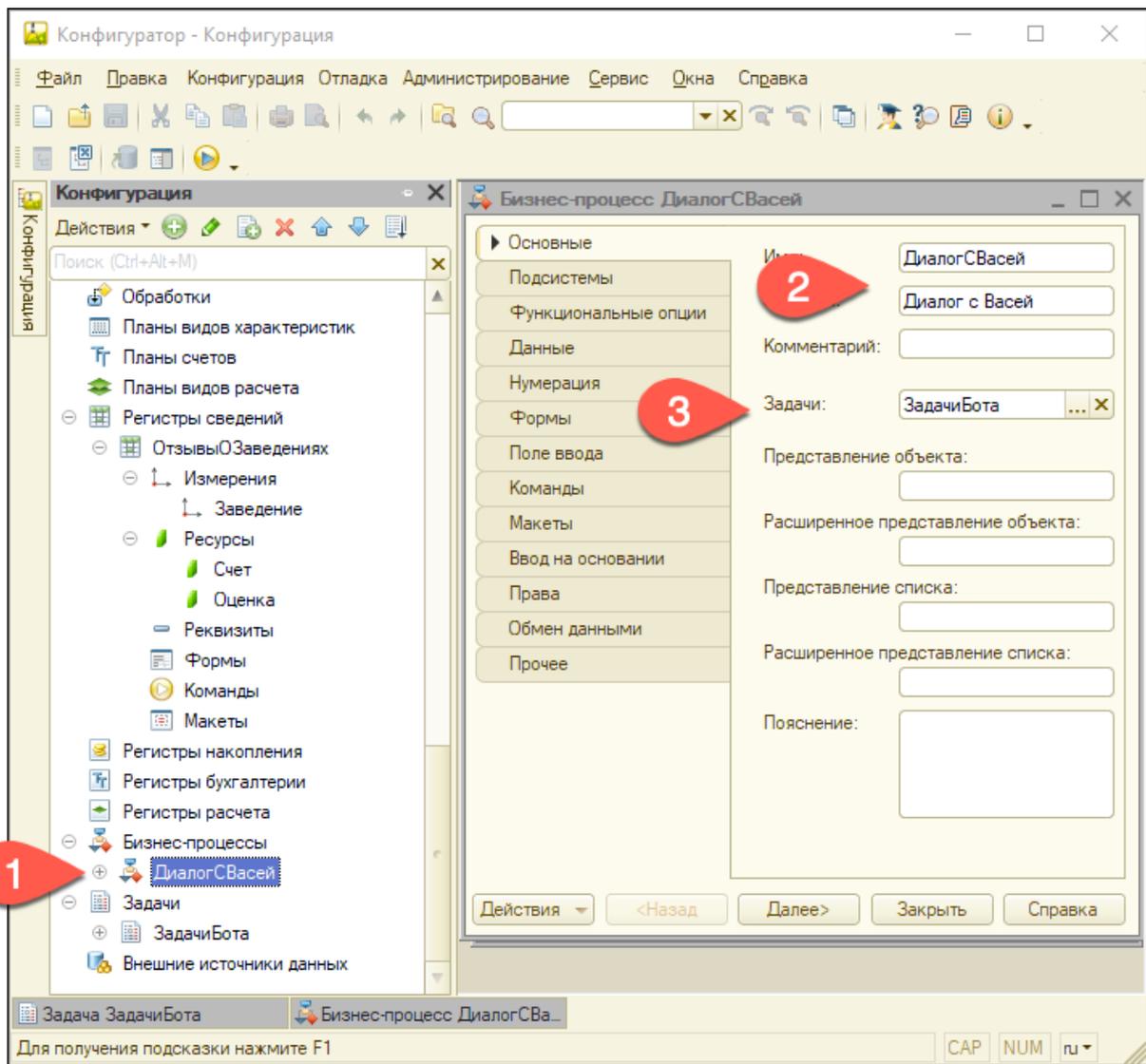
Бота будем прикручивать нестандартно, с помощью бизнес-процессов. Всю логику будем рисовать руками.

Для работы бизнес-процессов нужен объект «Задачи» в которых бизнес-процессы будут создаваться по точкам действия записи, и при выполнении которых будут двигаться дальше.

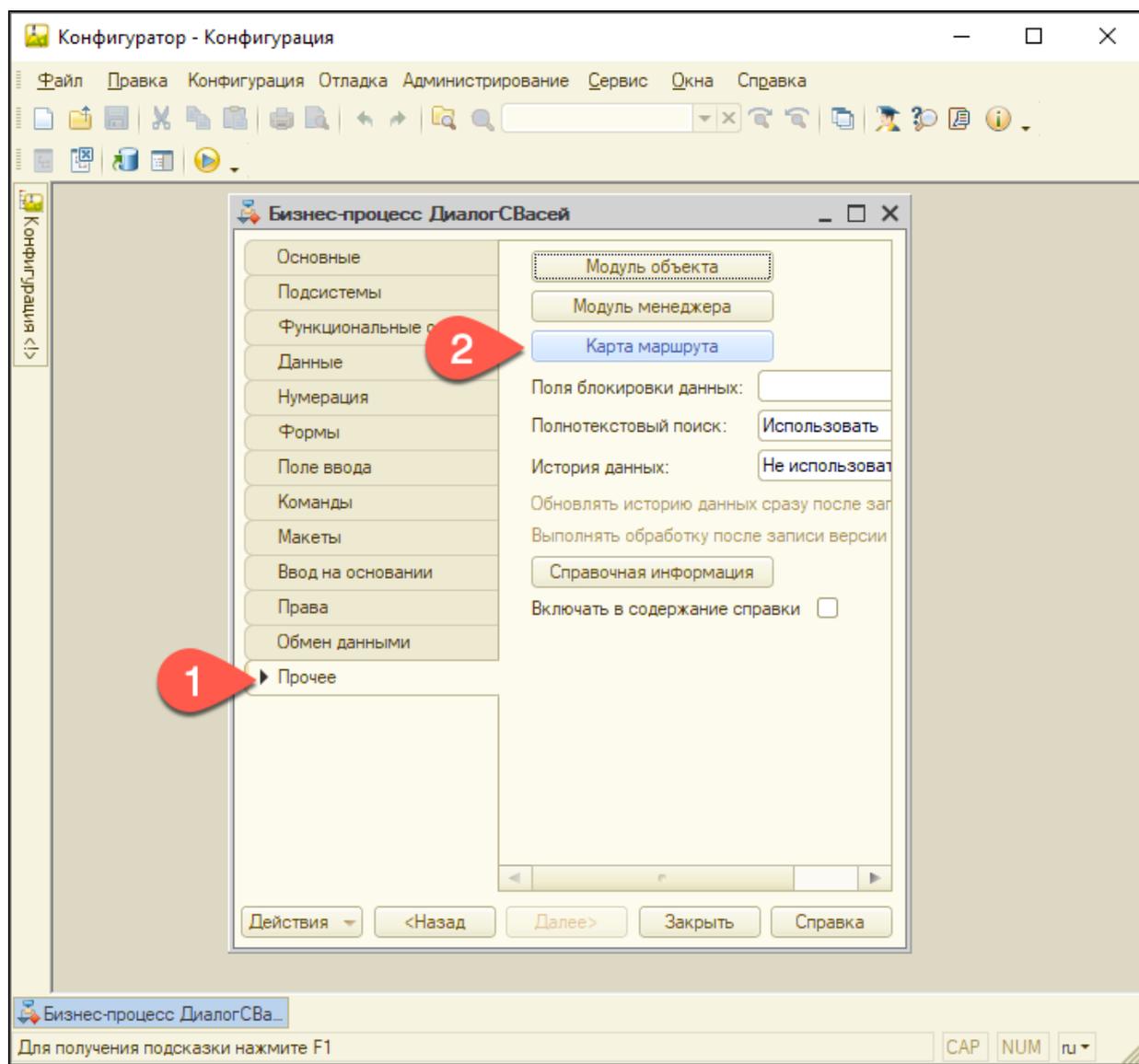
Создадим такой объект.



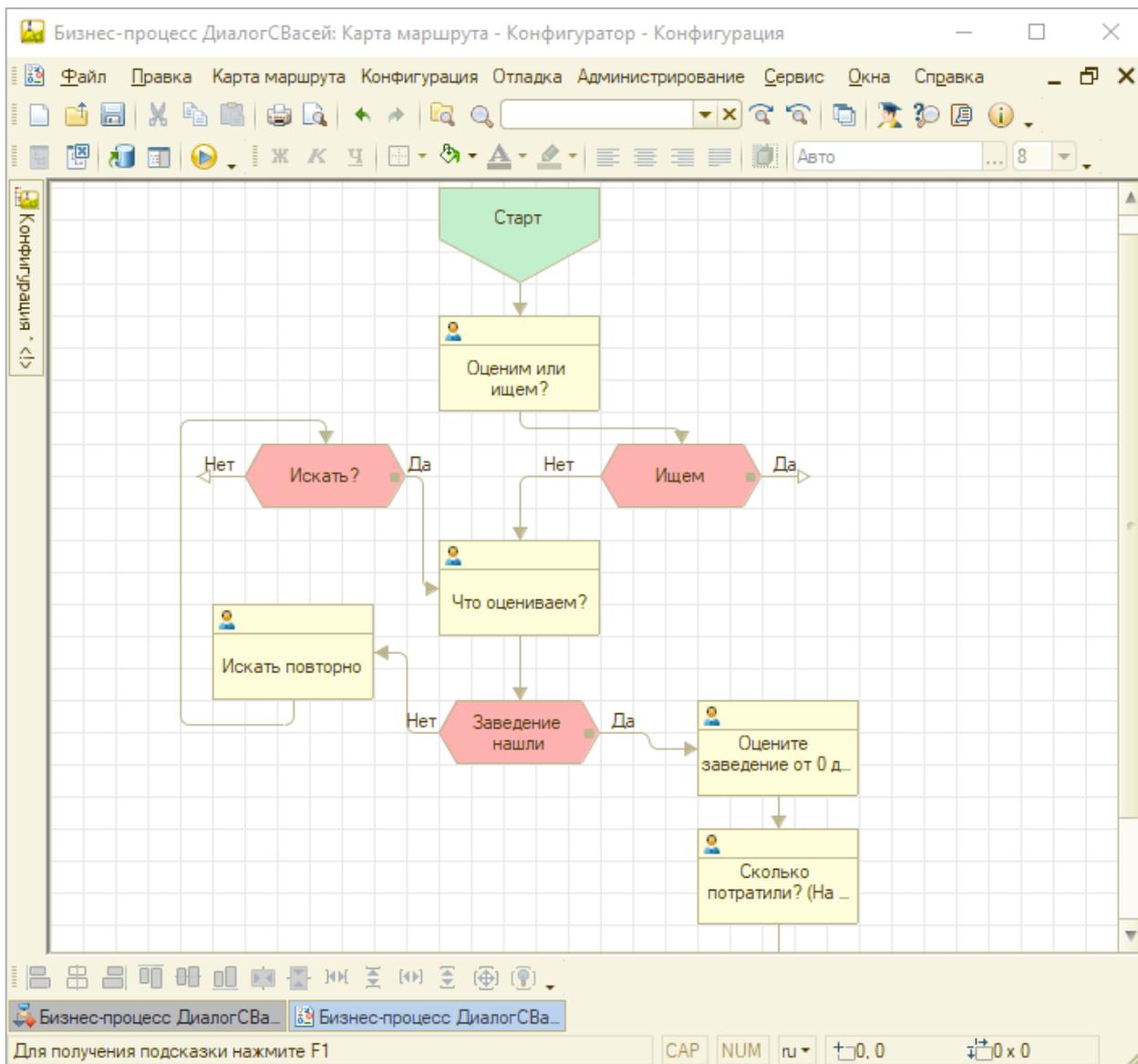
Добавим Бизнес-Процесс «ДиалогСВасей», и свяжем его с задачами.



И начнем описывать логику, в виде графической схемы.



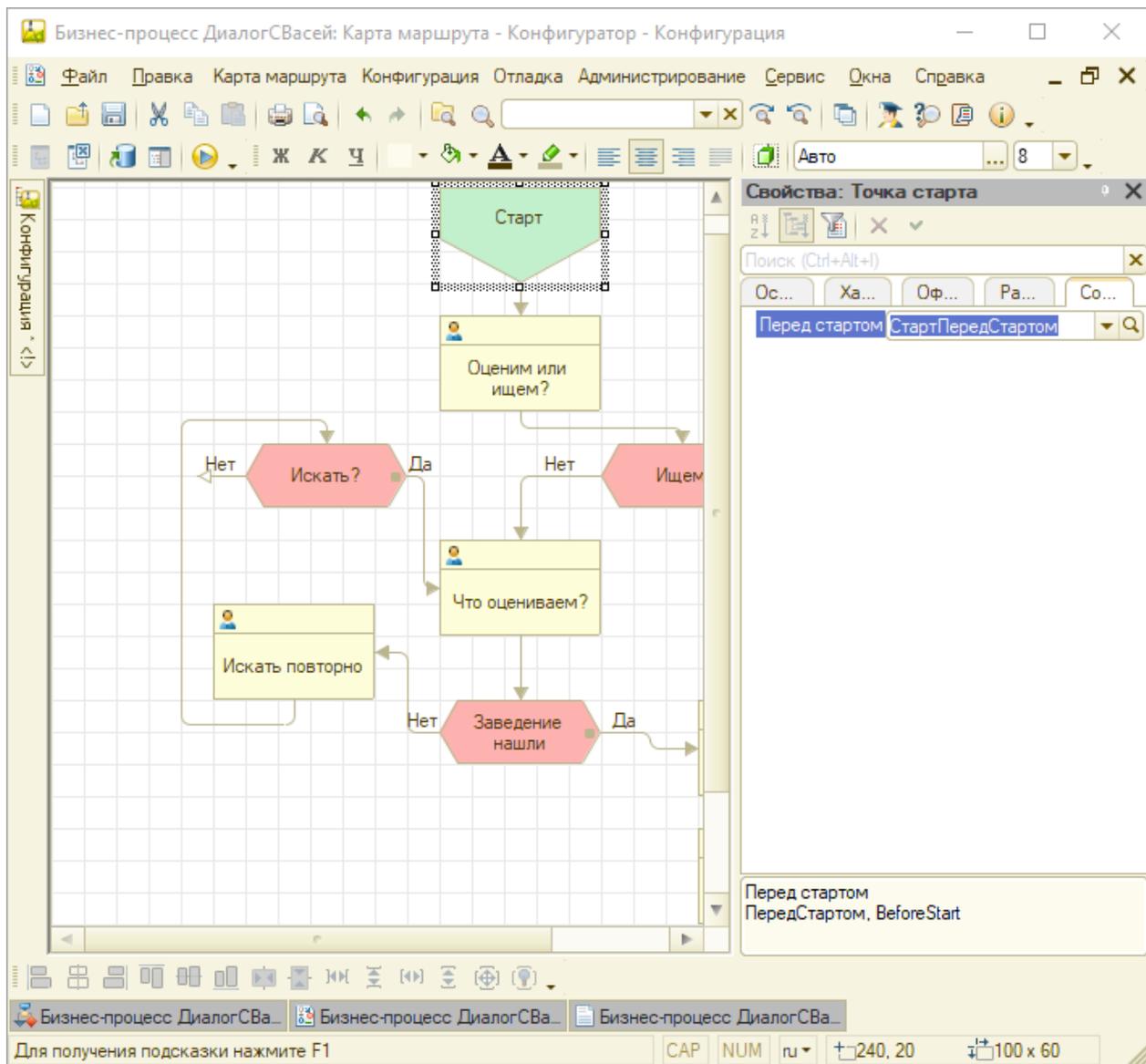
Приведем пример бизнес-процесса. Дополнять и адаптировать его можно бесконечно.



Вся работа с системой взаимодействия сводится к трем действиям: Информирование, Принятие решения, Выбор.

Рассмотрим пример.

## Информирование



Создадим общий модель и опишем обработчики для трех событий: отправка сообщения, обработка действий и обработка формирования команд.

Процедура ПриНачалеРаботыСистемы() Экспорт

Если НЕ СистемаВзаимодействия.ИнформационнаяБазаЗарегистрирована() Тогда  
Возврат;  
КонецЕсли;

Обработчик = Новый ОписаниеОповещения("ОбработкаОтправкиСообщения",  
ЭтотОбъект);  
СистемаВзаимодействия.ПодключитьОбработчикПослеОтправкиСообщения(Обработчик);

Обработчик2 = Новый ОписаниеОповещения("ОбработкаДействияСообщения",  
ЭтотОбъект);  
СистемаВзаимодействия.ПодключитьОбработчикДействияСообщения(Обработчик2);

Обработчик3 = Новый ОписаниеОповещения("ОбработкаФормированияКоманд",  
ЭтотОбъект);  
СистемаВзаимодействия.ПодключитьОбработчикФормированияКоманд(Обработчик3  
);

КонецПроцедуры

Процедура ОбработкаДействияСообщения(Сообщение, Действие, ДопПараметры) Экспорт

Если ТипЗнч(Действие) = Тип("СправочникСсылка.Пользователи") Тогда

КонецЕсли;

Если Действие = "Connect" Тогда

КонецЕсли;

Если Действие = "EMail" Тогда

КонецЕсли;

КонецПроцедуры

Процедура ОбработкаОтправкиСообщения(Сообщение, Обсуждение, ДопПараметры)  
Экспорт

КадровыйПомощник.ПолучитьСписокСотрудников(Сообщение.Текст,  
Обсуждение.Идентификатор);

КонецПроцедуры

Процедура ОбработкаФормированияКоманд(Параметры, Команды, КомандаПоУмолчанию,  
ДопПараметры) Экспорт

ДанныеСообщения = Параметры.Сообщение.Данные;

Если ТипЗнч(ДанныеСообщения) = Тип("Структура") И

ДанныеСообщения.Свойство("Печатать") Тогда

ОбработчикКоманды = Новый

ОписаниеОповещения("ОбработкаВыполненияКоманды", ЭтотОбъект,  
ДанныеСообщения.Сотрудник);

```
НоваяКоманда = Новый  
ОписаниеКомандыСистемыВзаимодействия(ОбработчикКоманды, "Печать");  
НоваяКоманда.Доступность = ДанныеСообщения.Печатать;  
Команды.Добавить(НоваяКоманда);  
КонецЕсли;
```

КонецПроцедуры

Процедура ОбработкаВыполненияКоманды(ДопПараметры) Экспорт

```
КадровыйПомощник.ПечатьИнформацииОСотруднике(ДопПараметры);
```

КонецПроцедуры

Процедуру общего модуля ПриНачалеРаботыСистемы вызовем из соответствующего обработчика в модуле приложения.

Процедура ПриНачалеРаботыСистемы()

```
ОбщийМодульКлиент.ПриНачалеРаботыСистемы();
```

КонецПроцедуры

Создадим серверный общий модуль опишем там инициализацию системы взаимодействия и стартанем бизнес-процесс.

Процедура Инициализация() Экспорт

```
Если НЕ СистемаВзаимодействия.ИнформационнаяБазаЗарегистрирована() Тогда  
Возврат;  
КонецЕсли;
```

```
УстановитьПривилегированныйРежим(Истина);
```

```
// Создаем пользователя информационной базы Помощник
```

```
ПользовательИБПомощник =
```

```
ПользователиИнформационнойБазы.НайтиПоИмени("КадровыйПомощник");
```

```
Если ПользовательИБПомощник = Неопределено Тогда
```

```
ПользовательИБПомощник =
```

```
ПользователиИнформационнойБазы.СоздатьПользователя();
```

```
ПользовательИБПомощник.Имя = "КадровыйПомощник";
```

```
ПользовательИБПомощник.ПолноеИмя = "КадровыйПомощник";
```

```
ПользовательИБПомощник.ПоказыватьВСпискеВыбора = Ложь;
```

```
ПользовательИБПомощник.Роли.Добавить(Метаданные.Роли.КадровыйПомощник);
```

```
ПользовательИБПомощник.Записать();
```

```
    ПользовательСВПомощник =
СистемаВзаимодействия.СоздатьПользователя(ПользовательИБПомощник);
    ПользовательСВПомощник.Записать();

    КонецЕсли;

    ИдентификаторПользователяСВПомощник =

        СистемаВзаимодействия.ПолучитьИдентификаторПользователя(ПользовательИБПомощник.УникальныйИдентификатор);

        // Создаем пользователей системы взаимодействия
        Для Каждого ПользовательИБ ИЗ
ПользователиИнформационнойБазы.ПолучитьПользователей() Цикл

            Попытка

                ИдентификаторПользователяСВ =

                    СистемаВзаимодействия.ПолучитьИдентификаторПользователя(ПользовательИБ.УникальныйИдентификатор);
                    ПользовательСВ =

                        СистемаВзаимодействия.ПолучитьПользователя(ИдентификаторПользователяСВ);

            Исключение

                ПользовательСВ =

                    СистемаВзаимодействия.СоздатьПользователя(ПользовательИБ);
                    ПользовательСВ.Записать();

            КонецПопытки;

            ИдентификаторПользователяСВСотрудник = ПользовательСВ.Идентификатор;
            Если ИдентификаторПользователяСВСотрудник <>
ИдентификаторПользователяСВПомощник Тогда

                КлючОбсуждения = Строка(ИдентификаторПользователяСВПомощник)
+" - "+ Строка(ИдентификаторПользователяСВСотрудник);
                Обсуждение =

                    СистемаВзаимодействия.ПолучитьОбсуждение(КлючОбсуждения);

                Если Обсуждение = Неопределено Тогда
                    Обсуждение = СистемаВзаимодействия.СоздатьОбсуждение();
                    Обсуждение.Групповое = Ложь;
```

Обсуждение.Ключ = КлючОбсуждения;

Обсуждение.Участники.Добавить(ИдентификаторПользователяСВПомощник);

Обсуждение.Участники.Добавить(ИдентификаторПользователяСВСотрудник);

Обсуждение.Записать();

КонецЕсли;

Сообщение =

СистемаВзаимодействия.СоздатьСообщение(Обсуждение.Идентификатор);

Сообщение.Автор = ИдентификаторПользователяСВПомощник;

Текст = "Приветствую! Я бот Вася :)" + Символы.ПС +

"Чтобы получить информацию о заведениях, просто введите его название.";

Сообщение.Текст = Новый ФорматированнаяСтрока(Текст);

Сообщение.Записать();

КонецЕсли;

КонецЦикла;

КонецПроцедуры

Процедура НачатьЧат() Экспорт

БП = БизнесПроцессы.ДиалогСВасей.СоздатьБизнесПроцесс();

БП.Записать();

БП.Старт();

КонецПроцедуры

Дополним в модуле приложения процедуру, укажем вызов.

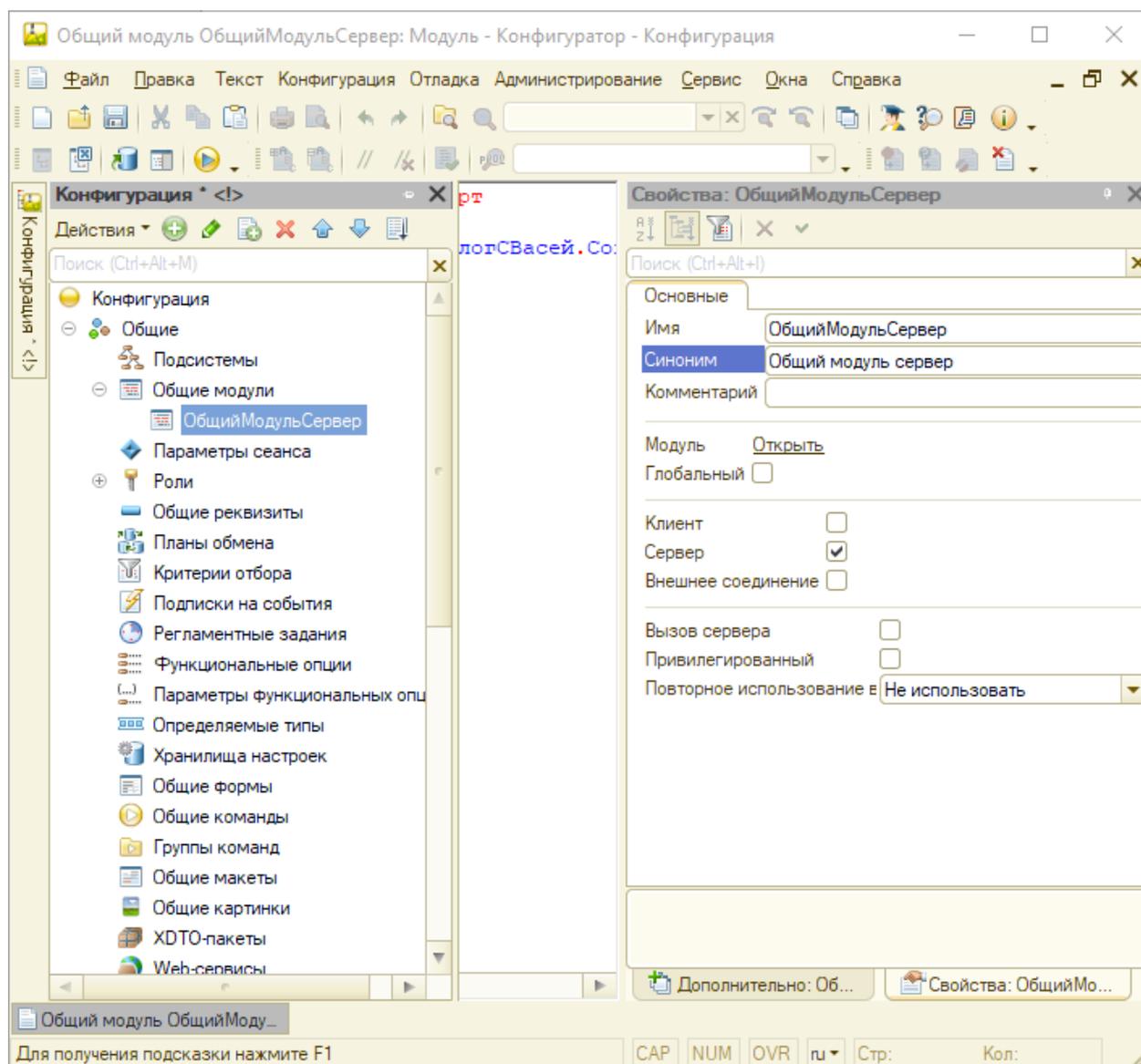
Процедура ПриНачалеРаботыСистемы()

ОбщийМодульСервер.ИнициализацияСВ();

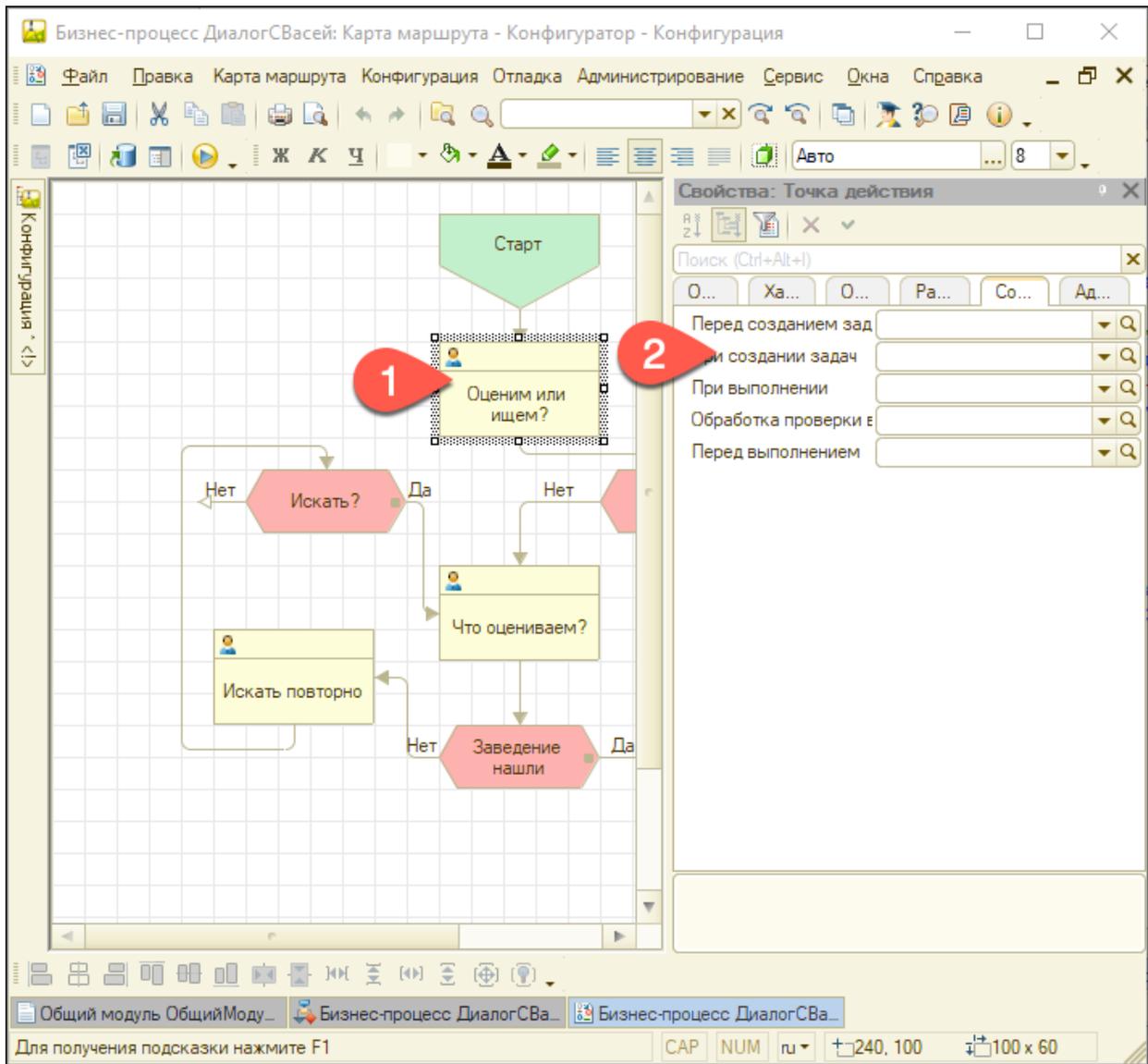
ОбщийМодульКлиент.ПриНачалеРаботыСистемы();

ОбщийМодульСервер.НачатьЧат();

КонецПроцедуры



Теперь нам нужно методично описать каждую точку действия и точку условия в бизнес-процесса. Вернемся к карте маршрута и опишем обработчик события первой точки действия «Оценим или ищем»



В обработчике событий сформируем для пользователя ответ.

Процедура ПолучитьЗаведения(ТекстЗапроса, ИдентификаторОбсуждения) Экспорт

```
УстановитьПривилегированныйРежим(Истина);
Сообщение =
СистемаВзаимодействия.СоздатьСообщение(ИдентификаторОбсуждения);
Заведение = СокрЛП(ТекстЗапроса);
```

```
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
| Ссылка КАК Ссылка,
| Код КАК Код,
| Наименование КАК Наименование
```

```
|ИЗ
|   Справочник.Заведения КАК Пользователи
|ГДЕ
|   Наименование ПОДОБНО ""%" + &ЧастьНаименования + ""%"
|   И ПометкаУдаления = ЛОЖЬ
|
|УПОРЯДОЧИТЬ ПО
|   Наименование";
```

Запрос.УстановитьПараметр("ЧастьНаименования", ФИОСотрудника);

РезультатЗапроса = Запрос.Выполнить();

Если РезультатЗапроса.Пустой() Тогда

Сообщение.Автор = ПолучитьИдентификаторСВПомощника();

Сообщение.Текст = Новый ФорматированнаяСтрока("По вашему запросу заведений не найдены :(");

Сообщение.Записать();

Возврат;

КонецЕсли;

Выборка = РезультатЗапроса.Выбрать();

Пока Выборка.Следующий() Цикл

Если ПолучитьИдентификаторСВСотрудника(Выборка.Ссылка) <>

СистемаВзаимодействия.ИдентификаторТекущегоПользователя()

Тогда

Сообщение.Действия.Добавить(Выборка.Ссылка,

Выборка.Наименование);

КонецЕсли;

КонецЦикла;

Сообщение.Автор = ПолучитьИдентификаторСВПомощника();

Сообщение.Текст = "Выберите нужное заведение";

Сообщение.Записать();

КонецПроцедуры

Далее нужно обработать выбор пользователем заведений. Для этого опишем процедуру ОбработкаДействияСообщения.

Процедура ОбработкаДействияСообщения(Сообщение, Действие, ДопПараметры) Экспорт

Если ТипЗнч(Действие) = Тип("СправочникСсылка.Заведения") Тогда

ОбщийМодульКлиент.ПолучитьИнформациюОЗаведении(Действие,

Сообщение.Обсуждение);

КонецЕсли;

КонецПроцедуры

В серверном общем модуле опишем вопрос для пользователя, оцениваем или выбираем

Процедура ПолучитьИнформациюОЗаведении(СсылкаНаЗаведение,  
ИдентификаторОбсуждения) Экспорт

УстановитьПривилегированныйРежим(Истина);

Сообщение =

СистемаВзаимодействия.СоздатьСообщение(ИдентификаторОбсуждения);

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Код КАК Код,

| Наименование КАК Наименование,

| ПРЕДСТАВЛЕНИЕ(Адрес) КАК Адрес,

| ПРЕДСТАВЛЕНИЕ(Телефон) КАК Телефон

|ИЗ

| Справочник.Заведения

|ГДЕ

| Ссылка = &Ссылка

| ИПометкаУдаления = ЛОЖЬ";

Запрос.УстановитьПараметр("Ссылка", СсылкаНаЗаведение);

РезультатЗапроса = Запрос.Выполнить();

Если РезультатЗапроса.Пустой() Тогда

Сообщение.Автор = ПолучитьИдентификаторСВПомощника();

Сообщение.Текст = Новый ФорматированнаяСтрока("По вашему запросу информация не найдена :(");

Сообщение.Записать();

Возврат;

КонецЕсли;

Выборка = РезультатЗапроса.Выбрать();

Выборка.Следующий();

Текст =

"Сотрудник: " + Выборка.Наименование + Символы.ПС +

"Подразделение: " + Выборка.Адрес + Символы.ПС +

"Должность: " + Выборка.Телефон;

Сообщение.Автор = ПолучитьИдентификаторСВПомощника();

Сообщение.Текст = Новый ФорматированнаяСтрока(Текст);

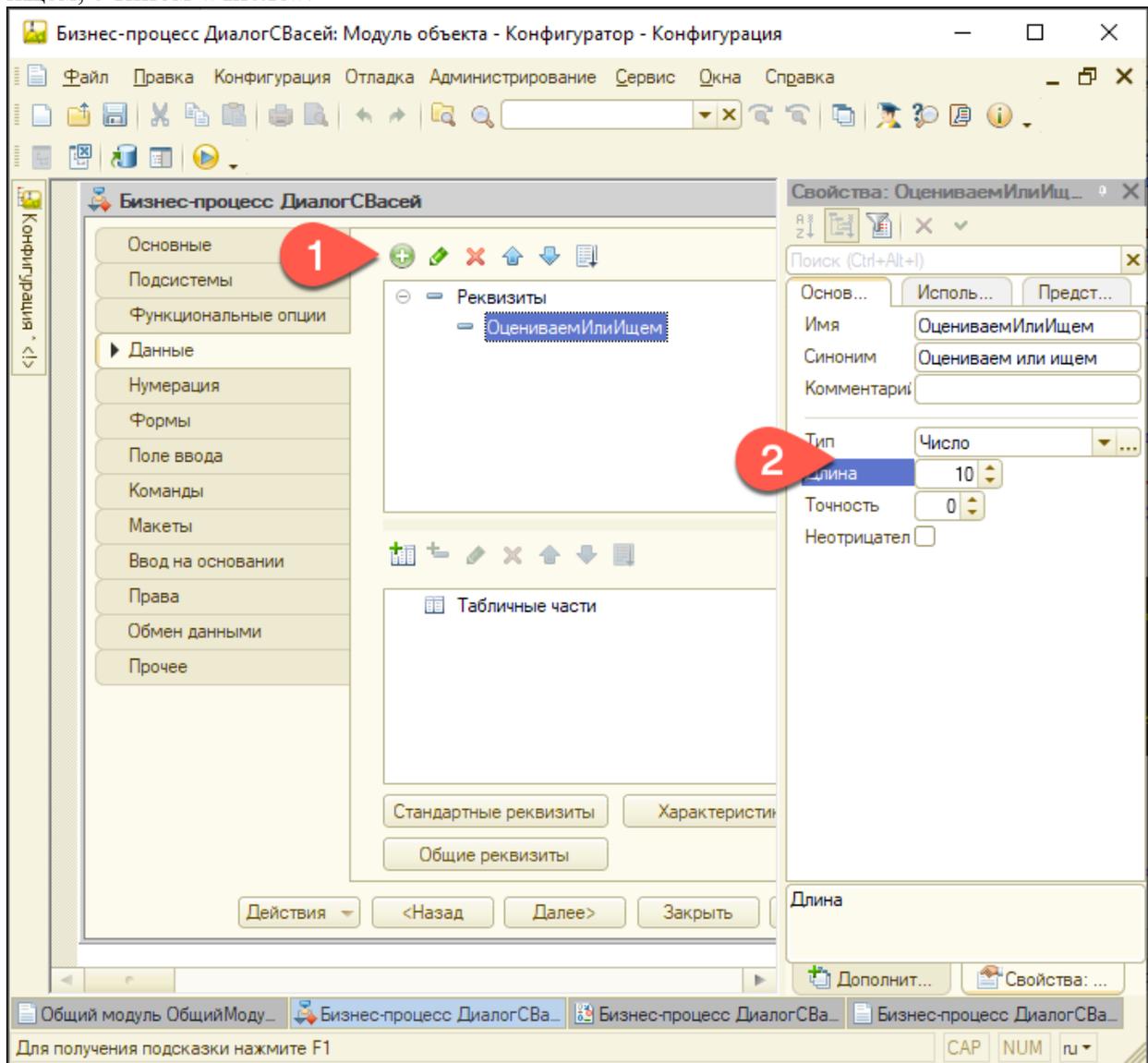
Сообщение.Действия.Добавить("rate", "Оцениваем");

Сообщение.Действия.Добавить("search", "Ищем");

Сообщение.Записать();

КонецПроцедуры

Для сохранения ответа пользователя добавим в бизнес-процесс реквизит «Оцениваем или ищем, с типом «Число».



Дополним процедуру ОбработкаДействияСообщение следующим условием.

Если Действие = "rate" Тогда

КонецЕсли;

Если Действие = "search" Тогда

КонецЕсли;

Далее принцип работы будет следующий: проверяем действие, которое выбрал пользователь, находим бизнес-процесс, записываем в реквизит результат действия и выполняем задачу.

После ее выполнения описываем проверку точки условия.

Конструктором запроса с обработкой результата найдем последний бизнес-процесс и запишем реквизит.

В итоге получится следующий текст модуля.

```
Запрос = Новый Запрос;
```

```
Запрос.Текст =
```

```
"ВЫБРАТЬ ПЕРВЫЕ 1
```

```
|     ДиалогСВасей.Ссылка КАК Ссылка
```

```
|ИЗ
```

```
|     БизнесПроцесс.ДиалогСВасей КАК ДиалогСВасей
```

```
|
```

```
|УПОРЯДОЧИТЬ ПО
```

```
|     ДиалогСВасей.Дата УБЫВ";
```

```
РезультатЗапроса = Запрос.Выполнить();
```

```
ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();
```

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

БП = Выборка.Ссылка.ПолучитьОбъект();

КонецЦикла;

Если Действие = "rate" Тогда

БП.ЗаписываемИлиИщем = 1;

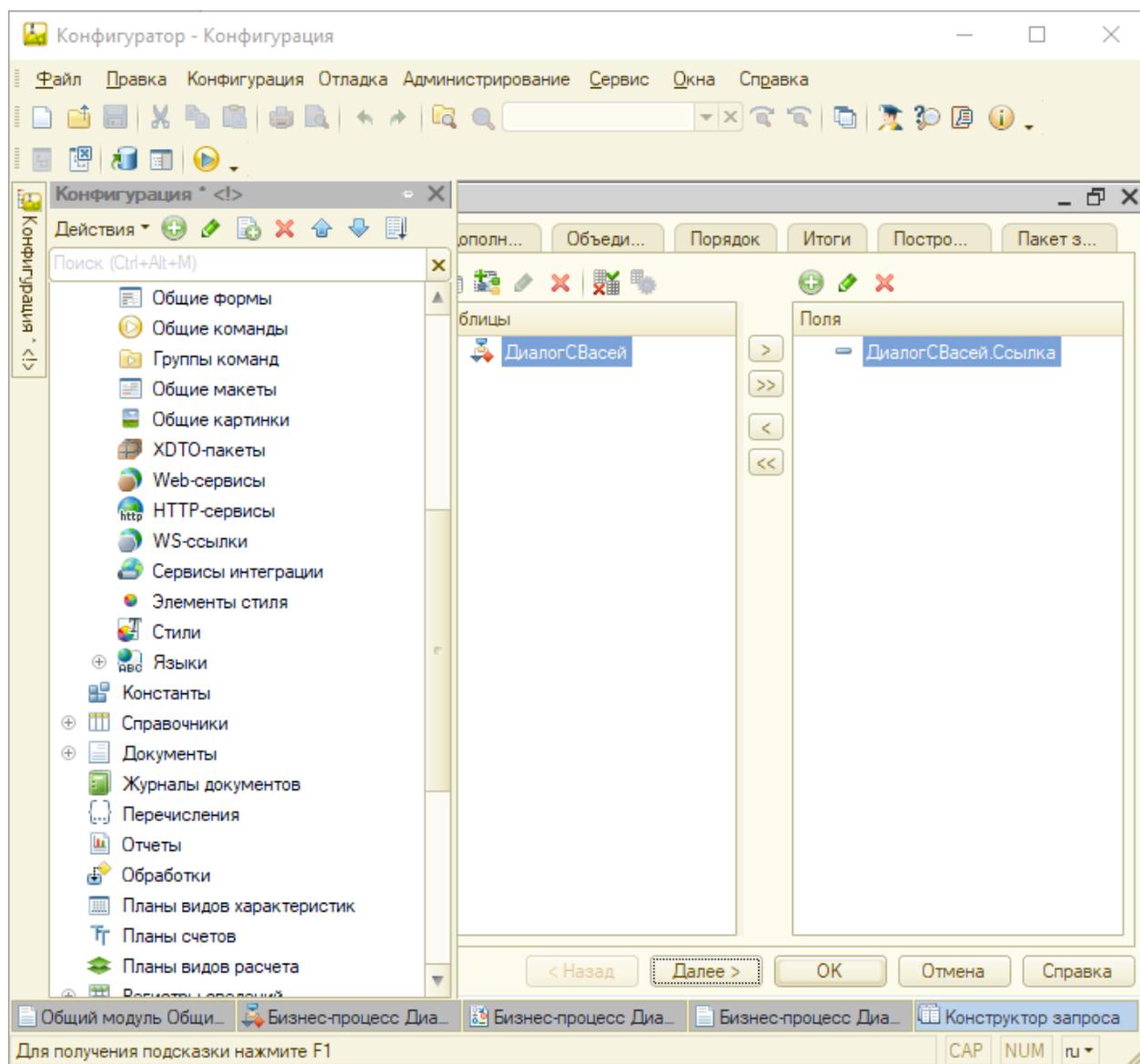
КонецЕсли;

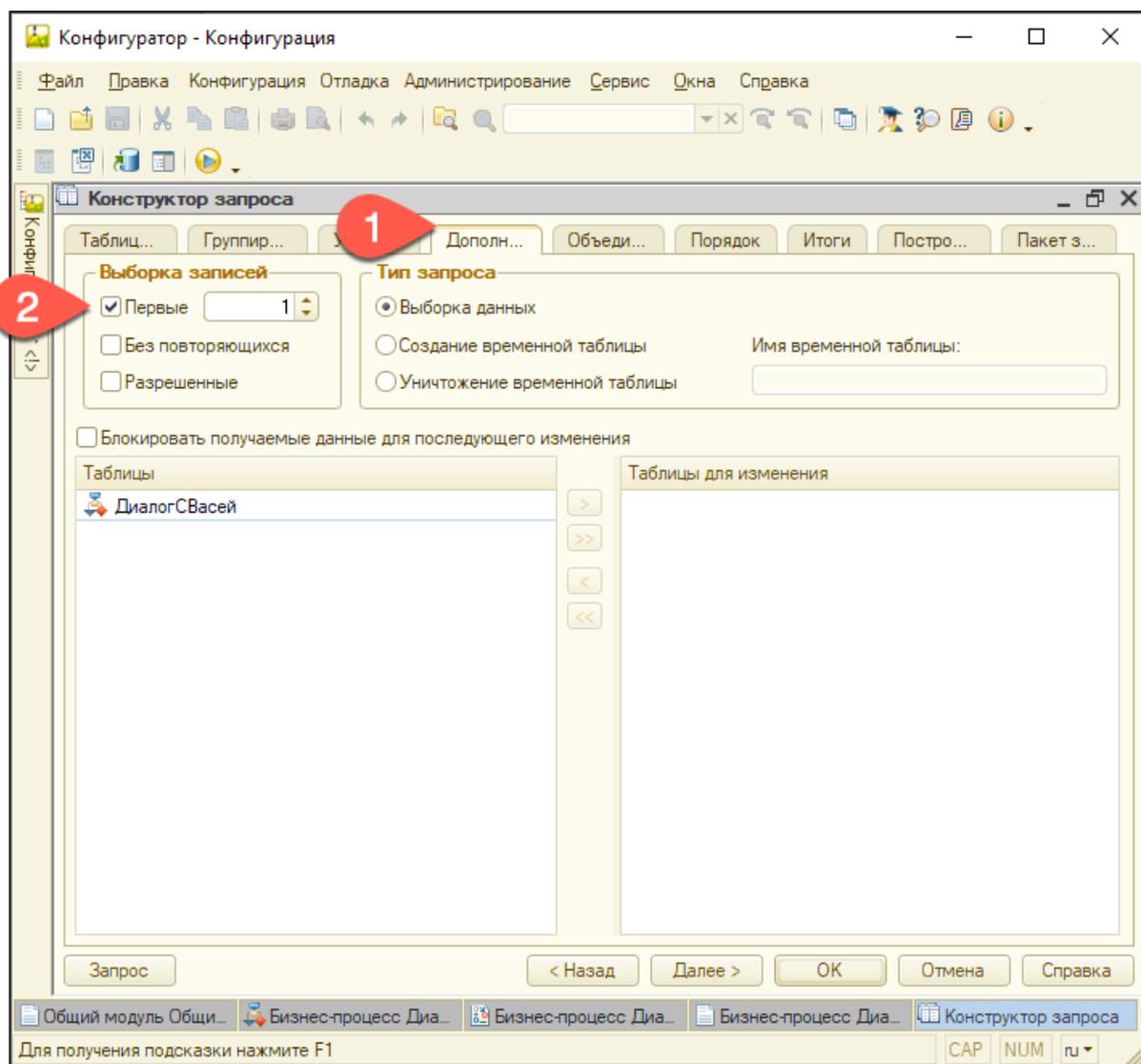
Если Действие = "search" Тогда

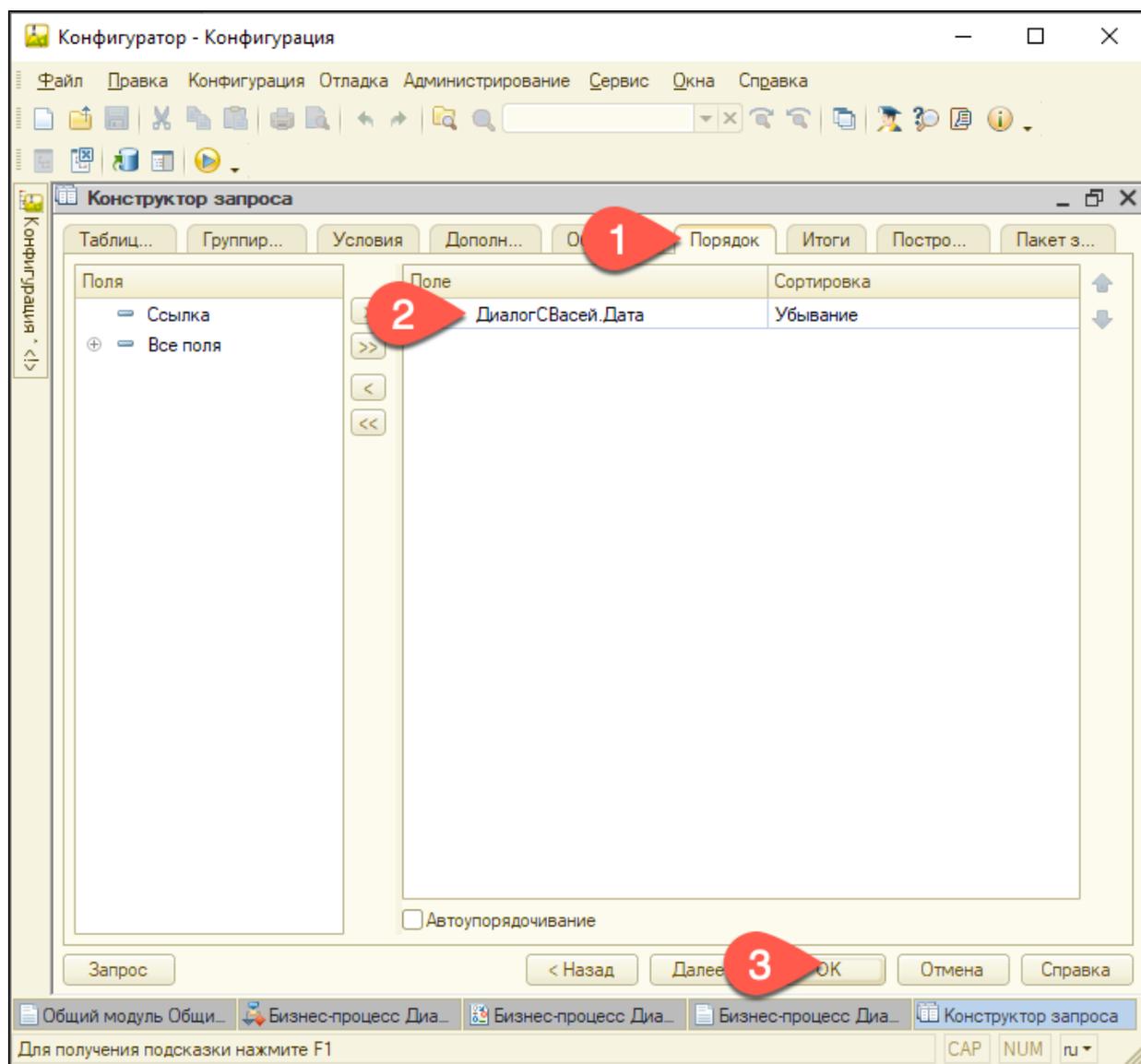
БП.ЗаписываемИлиИщем = 2;

КонецЕсли;

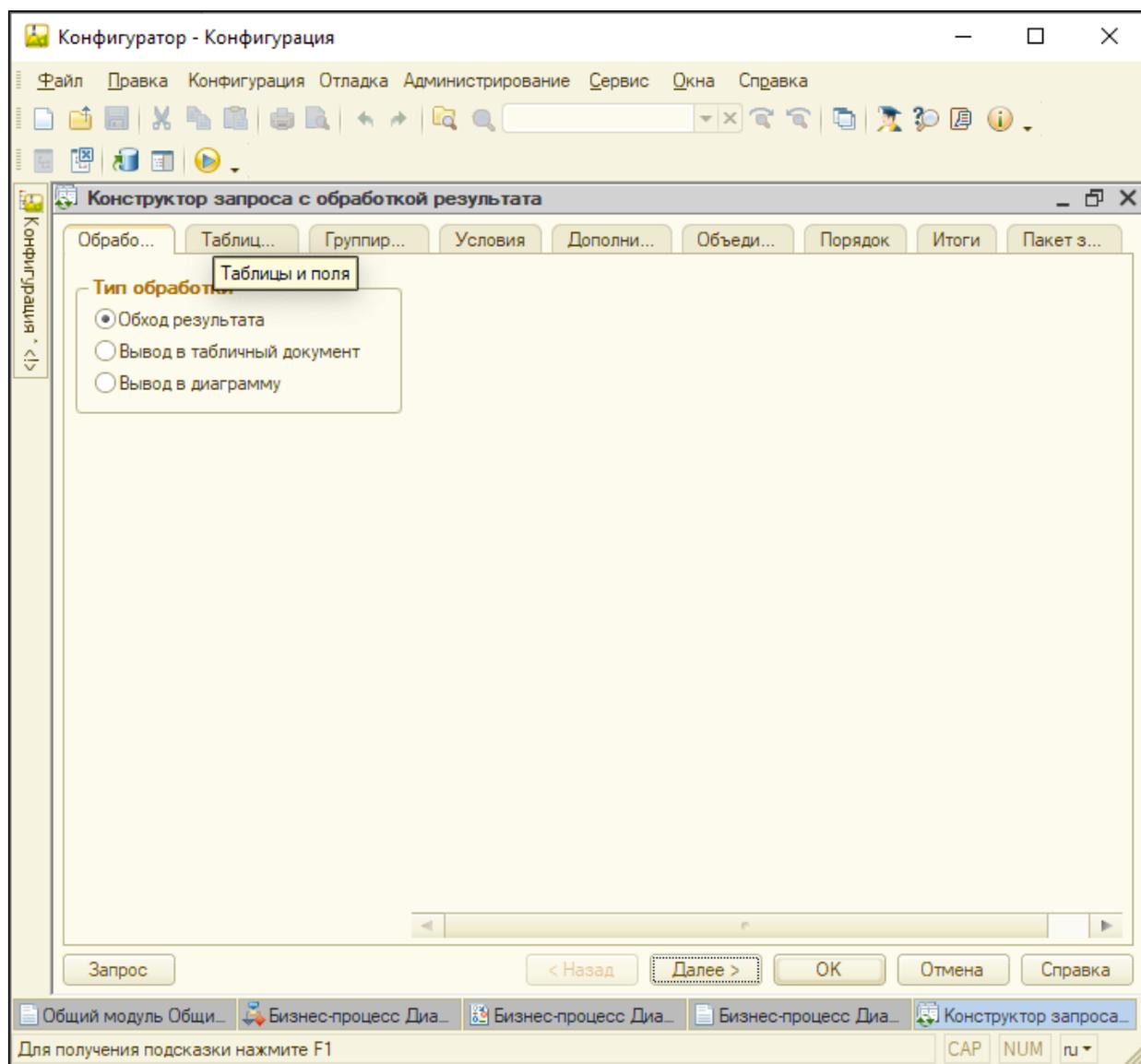
В итоге получится следующий текст модуля.

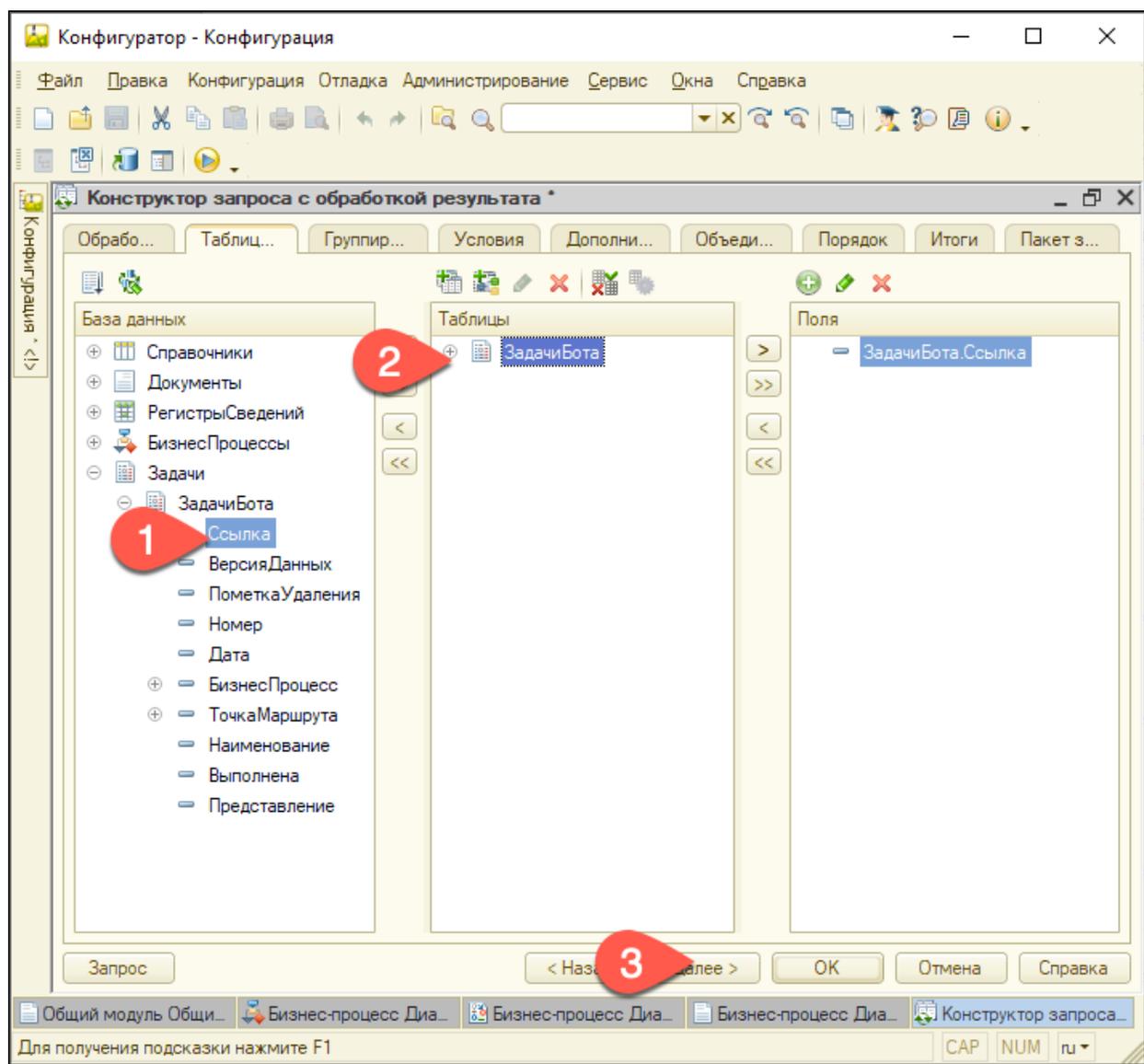


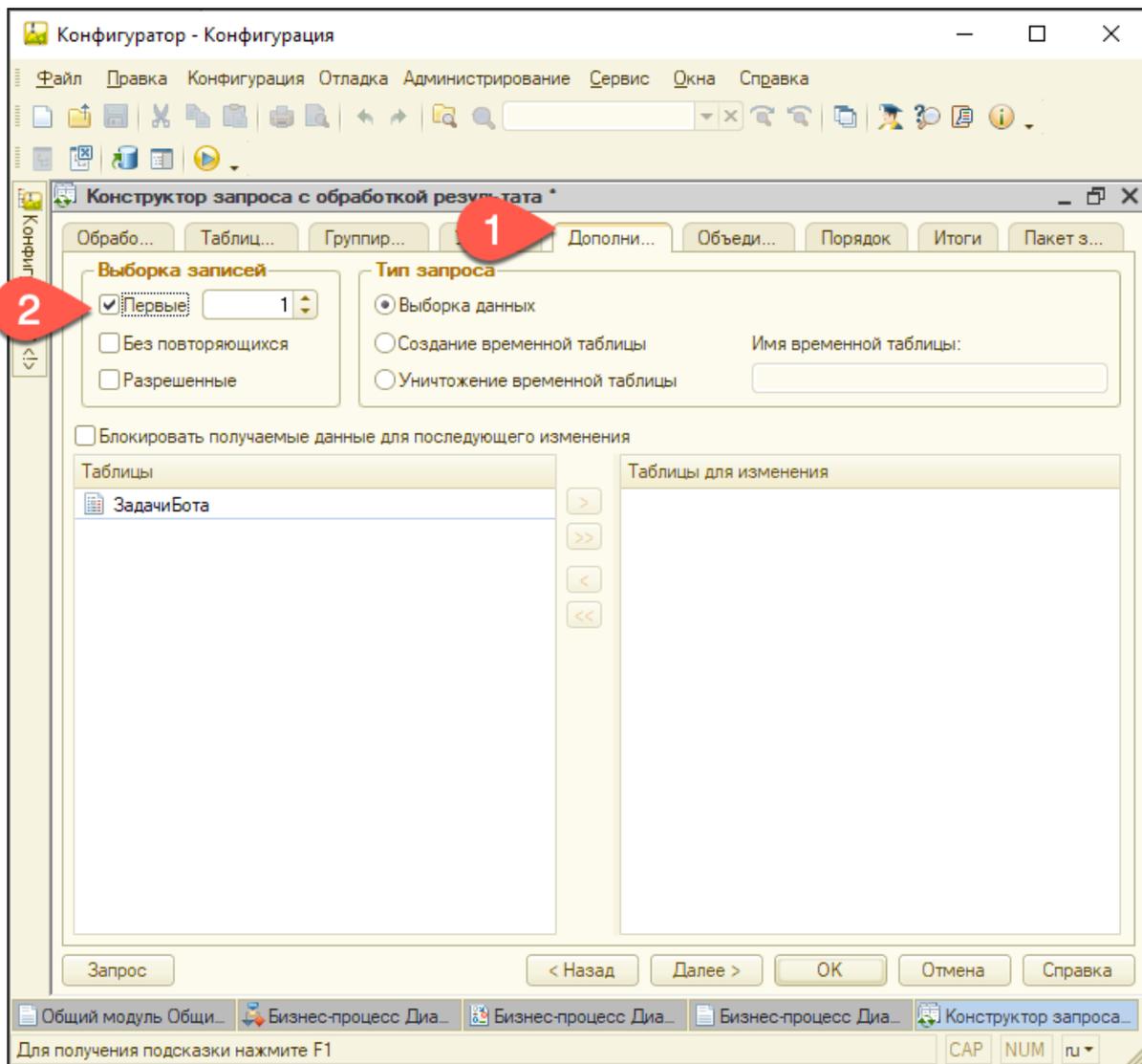


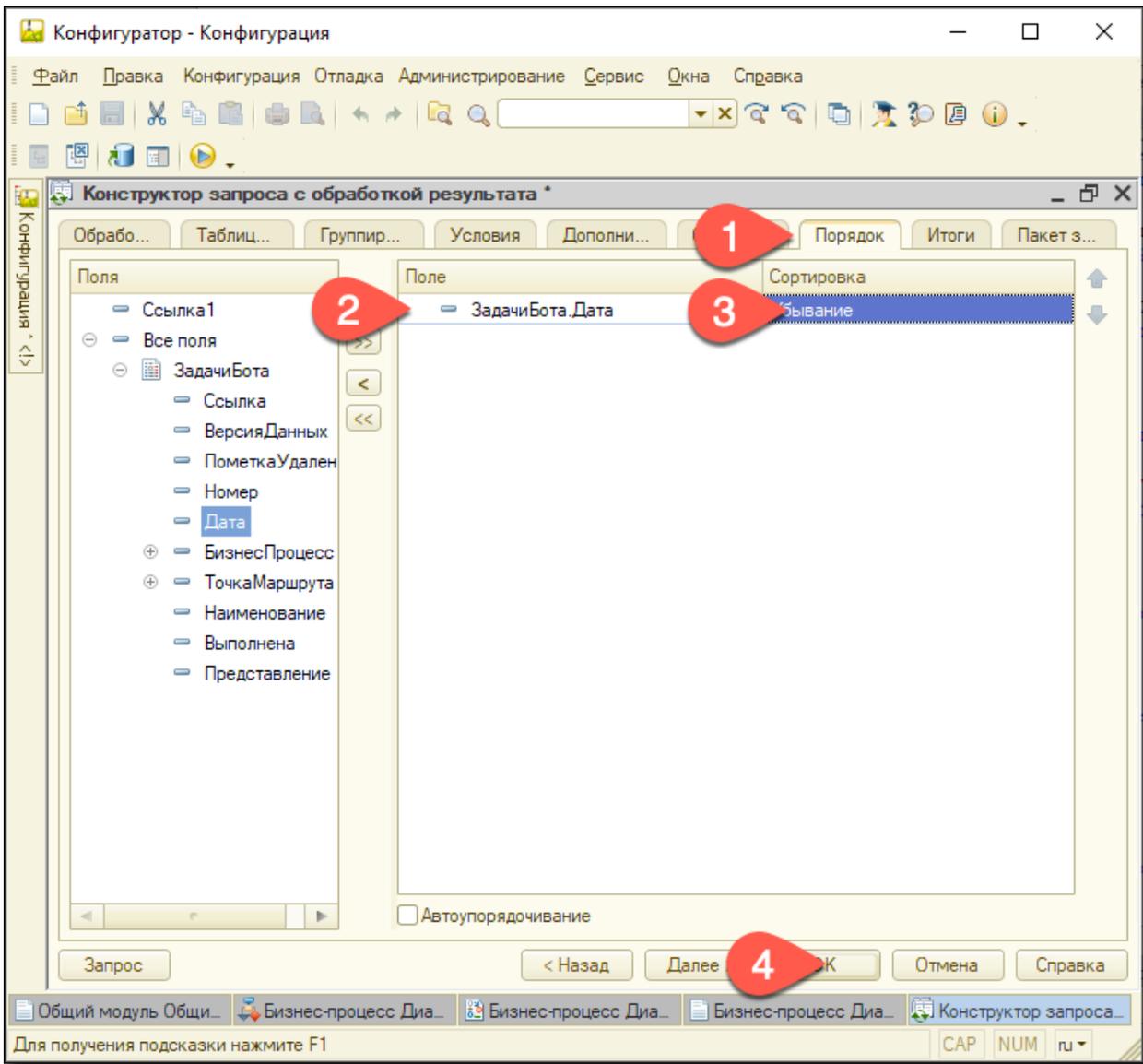


Еще одним запросом найдем последнюю задачу и выполним ее.









Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ ПЕРВЫЕ 1

|       ЗадачиБота.Ссылка КАК Ссылка

|ИЗ

|       Задача.ЗадачиБота КАК ЗадачиБота

|

|УПОРЯДОЧИТЬ ПО

|       ЗадачиБота.Дата УБЫВ";

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

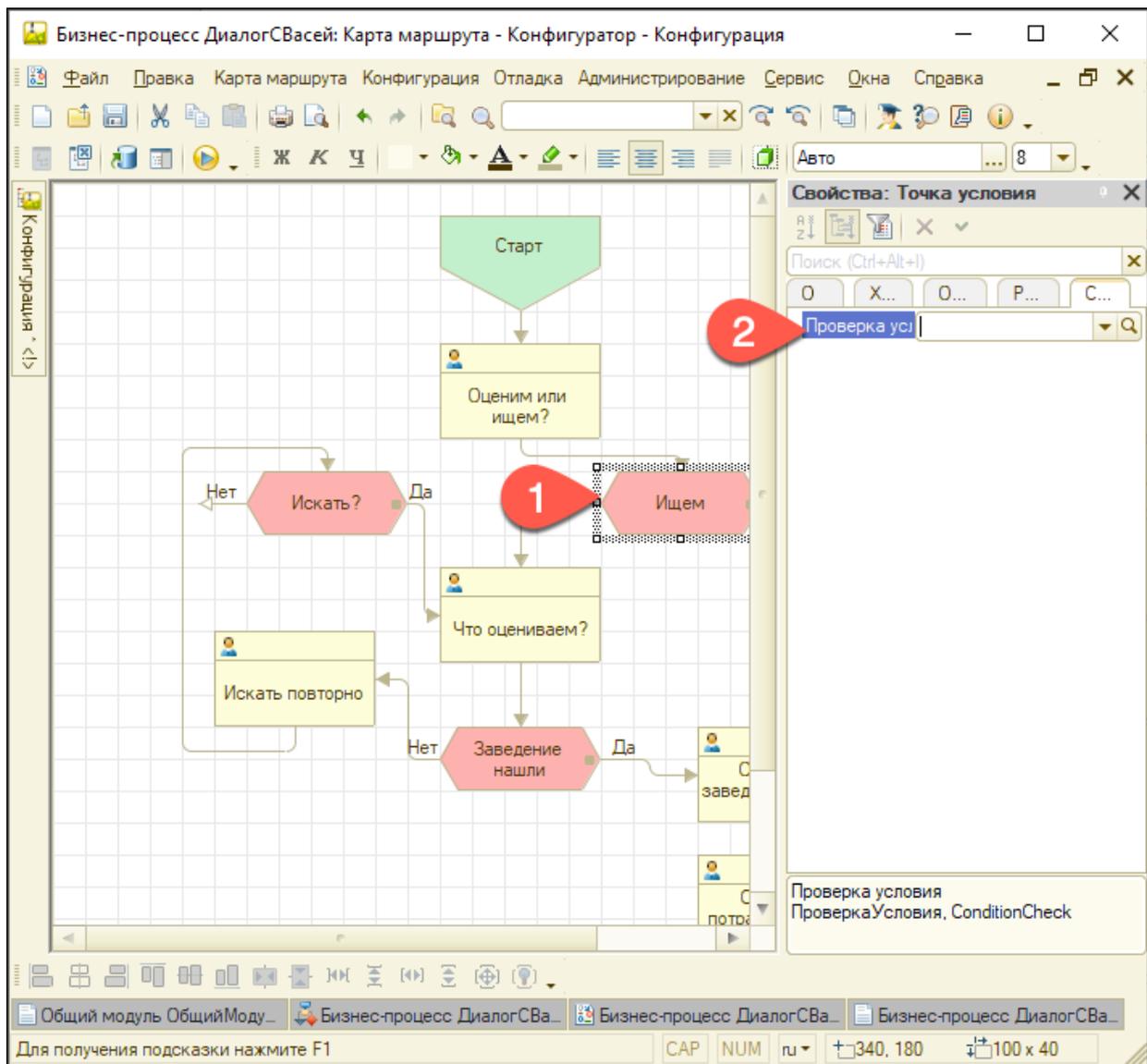
Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

Задача = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьЗадачу();

Задача.ВыполнитьЗадачу();

КонецЦикла;

Последнее что осталось сделать, описать проверку условия в карте маршрута:



Процедура ИщемПроверкаУсловия(ТочкаМаршрутаБизнесПроцесса, Результат)

Результат = ?(ОцениваемИлиИщем = 1, Истина, Ложь);

КонецПроцедуры

Остальные точки действия и точки условия опишем по аналогии.

## Заключение

Как мы увидели сочетание визуального программирования и системы взаимодействия позволяет быстро описывать логику, визуально контролировать ход беседы с ботом, а при необходимости довольно быстро изменять алгоритм поведения.