

## 5.2 ЗАПРОСЫ

1. Сделайте форму с таблицей значений на форме.

- У таблицы будут следующие колонки:
  - *Ссылка* (заголовок ПКО, тип *Ссылка на документ Приходно-кассовый ордер*),
  - *Контрагент* (тип *Ссылка на справочник Контрагенты*) и
  - *Сумма*.
- Также на форме будет поле ввода *Касса* (реквизит с типом *Ссылка на справочник Кассы*).
- При выборе кассы перезаполняйте таблицу документами только с выбранной кассой.

```

&НаСервере
Процедура 31_КассаПриИзменениеНаСервере ()
    31_ДокументыПКО.Очистить ();

    Запрос = Новый Запрос;
    Запрос.Текст = "ВЫБРАТЬ
        | ПриходноКассовыйОрдер.Ссылка КАК Ссылка,
        | ПриходноКассовыйОрдер.Контрагент КАК Контрагент,
        | ПриходноКассовыйОрдер.Сумма КАК Сумма
    | ИЗ
    | Документ.ПриходноКассовыйОрдер КАК ПриходноКассовыйОрдер
    | ГДЕ
    | ПриходноКассовыйОрдер.Касса = &Касса";
    Запрос.УстановитьПараметр ("Касса", 31_Касса);

    Выборка = Запрос.Выполнить ().Выбрать ();

    Пока Выборка.Следующий () Цикл
        НоваяСтрока = 31_ДокументыПКО.Добавить ();
        ЗаполнитьЗначенияСвойств (НоваяСтрока, Выборка);
    КонецЦикла;

КонецПроцедуры

&НаКлиенте
Процедура 31_КассаПриИзменении (Элемент)
    Если 31_Касса.Пустая () Тогда
        Возврат;
    КонецЕсли;

    31_КассаПриИзменениеНаСервере ();
КонецПроцедуры

```

2. При записи элемента справочника *Контрагенты* (используйте метод *ПередЗаписью* модуля объекта справочника), если *ИНН* контрагента заполнен, проверяйте, есть ли контрагент с таким ИНН в базе, если есть, то не записывайте.

- Создайте отдельную процедуру, в которой выполняйте запрос, и используйте её в методе *ПередЗаписью* модуля справочника.

The screenshot shows a software interface with a title bar "Справочник Контрагенты: Модуль объекта". Below it, there are two code snippets:

```
Функция ЕстьКонтрагентСИНН(ИНН)
    Запрос = Новый Запрос;
    Запрос.Текст = "ВЫБРАТЬ
        | Контрагенты.Ссылка КАК Ссылка
        | ИЗ
        | Справочник.Контрагенты КАК Контрагенты
        | ГДЕ
        | Контрагенты.ИНН = &ИНН";
    Запрос.УстановитьПараметр ("ИНН", ИНН);
    Результат = Запрос.Выполнить();
    Если Результат.Пустой() Тогда
        Возврат Ложь;
    Иначе
        Возврат Истина;
    КонецЕсли;
КонецФункции

Процедура ПередЗаписью(Отказ)
    Если ЗначениеЗаполнено (ИНН) Тогда
        ЕстьКонтрагент = ЕстьКонтрагентСИНН (ИНН);

        Если ЕстьКонтрагент Тогда
            Отказ = Истина;
            Сообщить (СтрШаблон ("Контрагент с ИНН %1 уже есть в базе", ИНН));
        КонецЕсли;
    КонецЕсли;
КонецПроцедуры
```

3. Разместите на форме два элемента: список (тип значения Ссылка на справочник *Контрагенты*) и таблицу значений (колонки: *Документ* (составной тип Ссылка на документы *Поступление товара* и *Продажа товаров*), *Сумма*, *Контрагент*).

- Реализуйте возможность добавлять в список и удалять из поля списка элементы справочника *Контрагенты*.
- После выполнения команды *Заполнить*, заполняйте таблицу значений документами *Поступление товара* и *Продажа товаров* с теми контрагентами, которые выбраны в списке.
- Проверяйте перед заполнением, что список контрагентов не пустой.

4. На форме создайте таблицу значений *Приход средств по кассе* (колонки – *Касса*, *Сумма*).

- Выведите общую сумму по всем документам *Приходно-кассовый ордер* в разрезе касс. Берите в учёт только проведённые документы.

5. Выведите в таблицу на форме все документы *Расходно-кассовый ордер* (колонки *РКО*, *Касса*, *Сумма*) с отбором по нужной кассе, упорядоченные по дате.

The screenshot shows the 1C Development Environment interface. The top part displays the object browser with various objects like '35\_Заполнить' and '35\_Касса'. Below it is the code editor window. The code is written in 1C script and defines two procedures: '35\_ЗаполнитьНаСервере()' and '35\_Заполнить(Команда)'. The first procedure fills the database with data from the server, while the second handles client-side filling based on a command. Both procedures use the '35\_РасходноКассовыеОрдера' document type.

```

&НаСервере
Процедура 35_ЗаполнитьНаСервере()
{
    35_РасходноКассовыеОрдера.Очистить();

    Запрос = Новый Запрос;
    Запрос.Текст = "ВЫБРАТЬ
        | РасходноКассовыйОрдер.Ссылка КАК РКО,
        | РасходноКассовыйОрдер.Касса КАК Касса,
        | РасходноКассовыйОрдер.Сумма КАК Сумма
    |ИЗ
    | Документ.РасходноКассовыйОрдер КАК РасходноКассовыйОрдер
    |ГДЕ
    | РасходноКассовыйОрдер.Касса = &Касса
    |
    |УПОРЯДОЧИТЬ ПО
    | РасходноКассовыйОрдер.Дата";
    Запрос.УстановитьПараметр("Касса", 35_Касса);

    Выборка = Запрос.Выполнить().Выбрать();

    Пока Выборка.Следующий() Цикл
        НоваяСтрока = 35_РасходноКассовыеОрдера.Добавить();
        ЗаполнитьЗначенияСвойств(НоваяСтрока, Выборка);
    КонецЦикла;

КонецПроцедуры

&НаКлиенте
Процедура 35_Заполнить(Команда)
{
    Если Не ЗначениеЗаполнено(35_Касса) Тогда
        Сообщить("Выберете кассу");
        Возврат;
    КонецЕсли;

    35_ЗаполнитьНаСервере();
}
КонецПроцедуры

```

6. Создайте таблицу значений на форме, в которой будут поля *Контрагент* и *Приходно- кассовый ордер*,
- выведите список контрагентов (только элементы) и документы *ПКО*, которые есть у данных контрагентов.
  - Если документов нет, то выводите сообщение «Документы ПКО не оформлены».

7. На форме подсчитывайте, сколько у Вас имеется контрагентов юр. лиц и сколько контрагентов физ. лиц. Данные выводите в поля надписи на форме.
  
8. Выведите в таблицу значений данные из журнала документов *Финансовые Документы*. Выведите графы *Ссылка* (заголовок Документ), *Касса*, *Контрагент*, *Сумма*, также ещё должна быть колонка *Название* (строка), в той строке, где выходит документ *Приходно- кассовый ордер*, должен быть текст «*ПКО*», а где расходно-кассовый ордер – «*РКО*».