**Решение задачи 2.20 (подобна задаче 2.18, 2.19)**

1. Создаем отчет, например, с именем «экзамен\_АБСКлассификацияПродажПоВесу», копируем в него «Форма Отчета» И «Форма Настроек Отчета» из «Общие Формы».
2. Создаем запрос Основной Схемы Компоновки Данных со следующей идеологией:

--Получаем Партнера по соединению таблиц «Справочник.ВнешниеПользователи» и «Справочник.Партнеры» И объединения «Справочник.ВнешниеПользователи» и «Справочник.КонтактныеЛицаПартнеров»

--Из РН «ВыручкаИСебестоимостьПродажОбороты» Получаем таблицу Регистраторов движений

--Из ТЧ «Товары» документов реализации получаем выборку необходимых данных

*ВЫБРАТЬ*

*Партнеры.Ссылка КАК Партнер*

*ПОМЕСТИТЬ ИсточникПартнер*

*ИЗ*

*Справочник.ВнешниеПользователи КАК ВнешниеПользователи*

*ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Партнеры КАК Партнеры*

*ПО ВнешниеПользователи.ОбъектАвторизации = Партнеры.Ссылка*

*ГДЕ*

*ВнешниеПользователи.Ссылка = &АвторизованныйПользователь*

*ОБЪЕДИНИТЬ*

*ВЫБРАТЬ*

*КонтактныеЛицаПартнеров.Владелец*

*ИЗ*

*Справочник.ВнешниеПользователи КАК ВнешниеПользователи*

*ЛЕВОЕ СОЕДИНЕНИЕ Справочник.КонтактныеЛицаПартнеров КАК КонтактныеЛицаПартнеров*

*ПО ВнешниеПользователи.ОбъектАвторизации = КонтактныеЛицаПартнеров.Ссылка*

*ГДЕ*

*ВнешниеПользователи.Ссылка = &АвторизованныйПользователь*

*;*

*////////////////////////////////////////////////////////////////////////////////*

*ВЫБРАТЬ*

*ВыручкаИСебестоимостьПродажОбороты.Регистратор КАК Накладная*

*ПОМЕСТИТЬ ИсточникДокументов*

*ИЗ*

*ИсточникПартнер КАК ИсточникПартнер*

*ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.ВыручкаИСебестоимостьПродаж.Обороты(, , Регистратор, ) КАК ВыручкаИСебестоимостьПродажОбороты*

*ПО ИсточникПартнер.Партнер = ВыручкаИСебестоимостьПродажОбороты.АналитикаУчетаПоПартнерам.Партнер*

*;*

*////////////////////////////////////////////////////////////////////////////////*

*ВЫБРАТЬ*

*РеализацияТоваровУслугТовары.Номенклатура КАК Номенклатура,*

*РеализацияТоваровУслугТовары.Характеристика КАК Характеристика,*

*РеализацияТоваровУслугТовары.Упаковка КАК Упаковка,*

*СУММА(ВЫБОР*

*КОГДА РеализацияТоваровУслугТовары.Упаковка = ЗНАЧЕНИЕ(Справочник.УпаковкиЕдиницыИзмерения.ПустаяСсылка)*

*ТОГДА РеализацияТоваровУслугТовары.Количество*

*ИНАЧЕ РеализацияТоваровУслугТовары.КоличествоУпаковок*

*КОНЕЦ) КАК Количество*

*ПОМЕСТИТЬ ИсточникНоменклатуры*

*ИЗ*

*ИсточникДокументов КАК ИсточникДокументов*

*ЛЕВОЕ СОЕДИНЕНИЕ Документ.РеализацияТоваровУслуг.Товары КАК РеализацияТоваровУслугТовары*

*ПО ИсточникДокументов.Накладная = РеализацияТоваровУслугТовары.Ссылка*

*СГРУППИРОВАТЬ ПО*

*РеализацияТоваровУслугТовары.Номенклатура,*

*РеализацияТоваровУслугТовары.Характеристика,*

*РеализацияТоваровУслугТовары.Упаковка*

*;*

*////////////////////////////////////////////////////////////////////////////////*

*ВЫБРАТЬ*

*ИсточникНоменклатуры.Номенклатура КАК Номенклатура,*

*ИсточникНоменклатуры.Характеристика КАК Характеристика,*

*ИсточникНоменклатуры.Количество КАК Количество,*

*ВЫБОР*

*КОГДА ИсточникНоменклатуры.Упаковка = ЗНАЧЕНИЕ(Справочник.УпаковкиЕдиницыИзмерения.ПустаяСсылка)*

*ТОГДА ИсточникНоменклатуры.Номенклатура.ЕдиницаИзмерения*

*ИНАЧЕ ИсточникНоменклатуры.Упаковка*

*КОНЕЦ КАК Упаковка,*

*ВЫБОР*

*КОГДА ИсточникНоменклатуры.Упаковка = ЗНАЧЕНИЕ(Справочник.УпаковкиЕдиницыИзмерения.ПустаяСсылка)*

*ТОГДА ВЫБОР*

*КОГДА ИсточникНоменклатуры.Номенклатура.ВесИспользовать*

*ТОГДА ЕСТЬNULL(ИсточникНоменклатуры.Количество, 0) \* ЕСТЬNULL(ИсточникНоменклатуры.Номенклатура.ВесЧислитель, 0) / ЕСТЬNULL(ИсточникНоменклатуры.Номенклатура.ВесЗнаменатель, 1)*

*ИНАЧЕ 0*

*КОНЕЦ*

*ИНАЧЕ ЕСТЬNULL(ИсточникНоменклатуры.Количество, 0) \* &ТекстЗапросаВесНоменклатуры*

*КОНЕЦ КАК Вес*

*ИЗ*

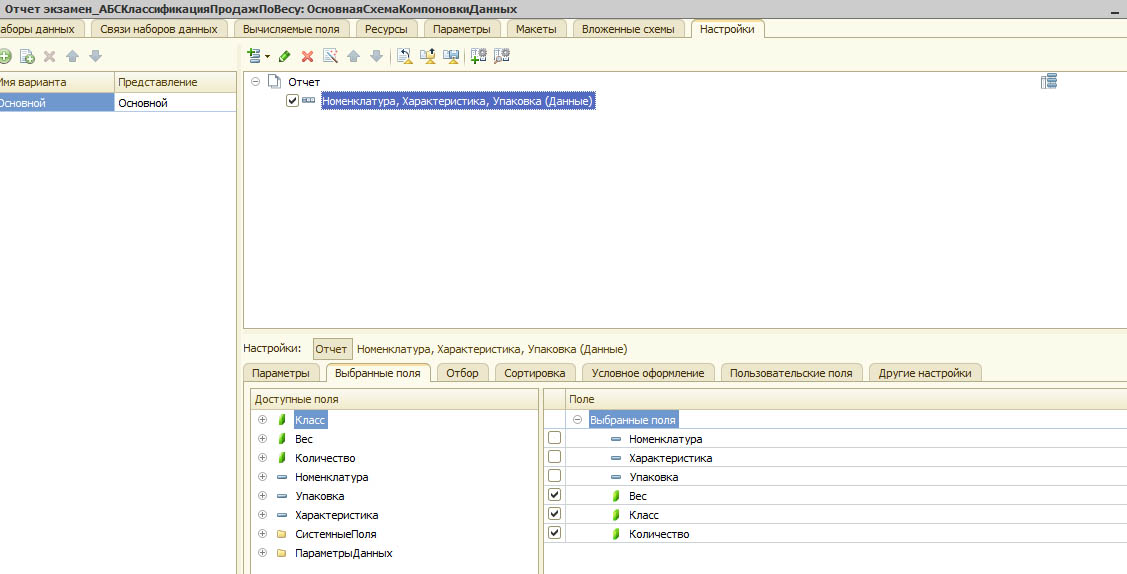
*ИсточникНоменклатуры КАК ИсточникНоменклатуры*

*ЛЕВОЕ СОЕДИНЕНИЕ Справочник.УпаковкиЕдиницыИзмерения КАК УпаковкиЕдиницыИзмерения*

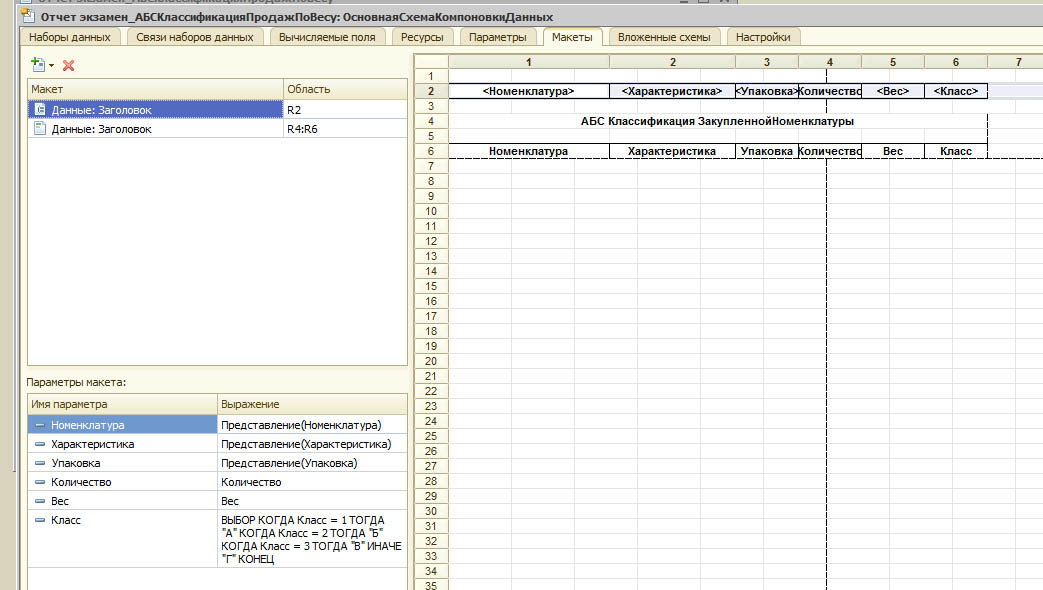
*ПО ИсточникНоменклатуры.Упаковка = УпаковкиЕдиницыИзмерения.Ссылка*

3. Создаем Вычислимое Поле «Класс» тип «Число» , в ресурсах объявляем Класс = КлассификацияABC("Сумма(Вес)", 4, "60, 80, 95")

4. Настройки:



5. Макет



6. Вводим параметр «Период» (Параметры НачалоПериода = &Период.ДатаНачала и КонецПериода = &Период.ДатаОкончания –ограничиваем доступ)

7. В «Модуль Формы» Процедура при открытии: Проверяем наличие параметра «Период», При наличии незаполненного параметра – устанавливаем его в соответствии с заданием.

Затем – заполняем соответствующий элемент формы, чтоб установленное значение было видно пользователю.

Тоже самое делаем с параметром «&АвторизованныйПользователь»

*//ЭКЗАМЕН*

*ПараметрПериод = Отчет.КомпоновщикНастроек.Настройки.ПараметрыДанных.Элементы.Найти("Период");*

*Если ПараметрПериод <> Неопределено*

*Тогда*

*Период = Новый СтандартныйПериод;*

*Период.ДатаОкончания = КонецДня(ТекущаяДата());*

*Период.ДатаНачала = НачалоДня(ДобавитьМесяц(Период.ДатаОкончания, -12));*

*ЗначениеПараметраПериода = Отчет.КомпоновщикНастроек.ПользовательскиеНастройки.Элементы.Найти(ПараметрПериод.ИдентификаторПользовательскойНастройки);*

*Если ЗначениеПараметраПериода = Неопределено*

*Тогда*

*ПараметрПериод.Значение = Период;*

*Иначе*

*ЗначениеПараметраПериода.Значение = Период;*

*КонецЕсли;*

*КонецЕсли;*

*Если ЗначениеЗаполнено(ЗначениеПараметраПериода.ИдентификаторПользовательскойНастройки)*

*Тогда*

*ЭлементПараметрПериод = Отчет.КомпоновщикНастроек.ПользовательскиеНастройки.Элементы.Найти(ЗначениеПараметраПериода.ИдентификаторПользовательскойНастройки);*

*Если ТипЗнч(ЭлементПараметрПериод) = Тип("ЗначениеПараметраНастроекКомпоновкиДанных")*

*Тогда*

*ЭлементПараметрПериод.Значение = ЗначениеПараметраПериода.Значение;*

*КонецЕсли;*

*КонецЕсли;*

*ПараметрАвторизованныйПользователь = Отчет.КомпоновщикНастроек.Настройки.ПараметрыДанных.Элементы.Найти("АвторизованныйПользователь");*

*ТекущийПользователь = ПолучитьТекущегоПользователя();*

*Если ТипЗнч(ТекущийПользователь) = Тип("СправочникСсылка.ВнешниеПользователи") Тогда*

*ЗначениеПараметраАвторизованныйПользователь = Отчет.КомпоновщикНастроек.ПользовательскиеНастройки.Элементы.Найти(ПараметрАвторизованныйПользователь.ИдентификаторПользовательскойНастройки);*

*Если ЗначениеПараметраАвторизованныйПользователь = Неопределено*

*Тогда*

*ПараметрАвторизованныйПользователь.Значение = ТекущийПользователь;*

*Иначе*

*ЗначениеПараметраАвторизованныйПользователь.Значение = ТекущийПользователь;*

*КонецЕсли;*

*Если ЗначениеЗаполнено(ЗначениеПараметраАвторизованныйПользователь.ИдентификаторПользовательскойНастройки) Тогда*

*ЭлементТекущийПользователь = Отчет.КомпоновщикНастроек.ПользовательскиеНастройки.Элементы.Найти(ЗначениеПараметраАвторизованныйПользователь.ИдентификаторПользовательскойНастройки);*

*Если ТипЗнч(ЭлементТекущийПользователь) = Тип("ЗначениеПараметраНастроекКомпоновкиДанных")*

*Тогда*

*ЭлементТекущийПользователь.Значение = ТекущийПользователь;*

*КонецЕсли;*

*КонецЕсли;*

*КонецЕсли;*

*//ЭКЗАМЕН*

*КонецПроцедуры*

*&НаСервереБезКонтекста*

*Функция ПолучитьТекущегоПользователя()*

*Возврат Пользователи.АвторизованныйПользователь();*

*КонецФункции*

8. Осталось только внести измненение в текст запроса СКД для подмены параметра «&ТекстЗапросаВесНоменклатуры» на полученный по функции Модуля Менеджера «Справочники.УпаковкиЕдиницыИзмерения».

Создаем «МодульОбъекта» нашего отчета, добавляем процедуру (Там же на всякий случай, проверяем параметр «&АвторизованныйПользователь»

*Процедура ПриКомпоновкеРезультата(ДокументРезультат, ДанныеРасшифровки, СтандартнаяОбработка)*

*Если ПолучитьФункциональнуюОпцию("ИспользоватьУпаковкиНоменклатуры")*

*Тогда*

*ТекстЗапросаСКД = СхемаКомпоновкиДанных.НаборыДанных[0].Запрос;*

*ТекстЗамены = Справочники.УпаковкиЕдиницыИзмерения.ТекстЗапросаВесУпаковки("ИсточникНоменклатуры.Упаковка", "ИсточникНоменклатуры.Номенклатура");*

*ТекстЗапросаСКД = СтрЗаменить(ТекстЗапросаСКД, "&ТекстЗапросаВесНоменклатуры", ТекстЗамены);*

*СхемаКомпоновкиДанных.НаборыДанных[0].Запрос = ТекстЗапросаСКД;*

*КонецЕсли;*

*ПараметрАвторизованныйПользователь = КомпоновкаДанныхКлиентСервер.ПолучитьПараметр(КомпоновщикНастроек, "АвторизованныйПользователь");*

*Если ПараметрАвторизованныйПользователь <> НеОпределено*

*Тогда*

*Если НЕ ЗначениеЗаполнено(ПараметрАвторизованныйПользователь.Значение)*

*Тогда*

*ТекущийПользователь = Пользователи.АвторизованныйПользователь();*

*Если ТипЗнч(ТекущийПользователь) = Тип("СправочникСсылка.ВнешниеПользователи")*

*Тогда*

*КомпоновкаДанныхКлиентСервер.УстановитьПараметр(КомпоновщикНастроек, "АвторизованныйПользователь", ТекущийПользователь, Истина);*

*КонецЕсли;*

*КонецЕсли;*

*КонецЕсли;*

*КонецПроцедуры*