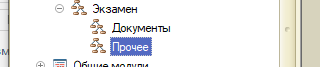
**БИЛЕТ 4**

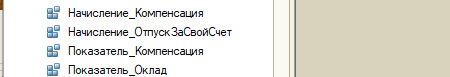
**РЕШЕНИЕ:**

**I КОНФИГУРАТОР.**

1. **Снимаем с поддержки, создаем подсистемы:**



1. **Делаем 4 константы (подсистема Экзамен-Прочее):**



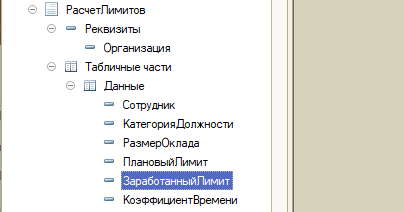
**Начисление\_Компенсация** (ПланВидовРасчетаСсылка.Начисления)

**Показатель\_Компенсация** (СправочникСсылка.ПоказателиРасчетаЗарплаты)

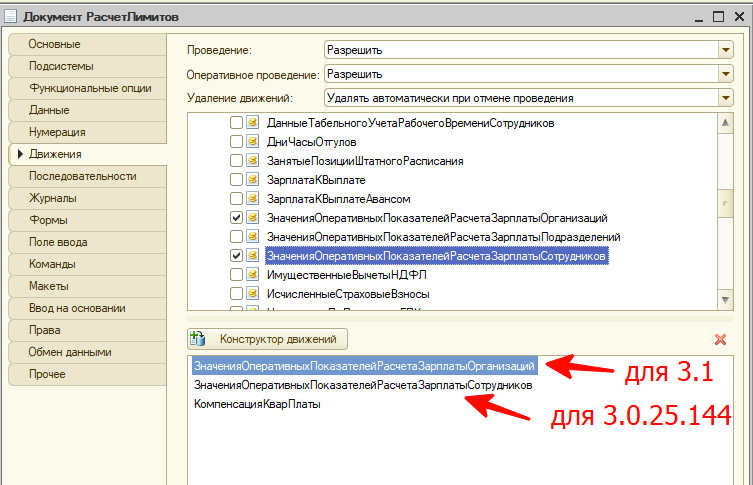
**Начисление\_ОтпускЗаСвойСчет** (ПланВидовРасчетаСсылка.Начисления)

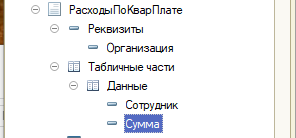
**Показатель\_Оклад** (СправочникСсылка.ПоказателиРасчетаЗарплаты)

1. **Создаем два документа РасчетЛимитов и РасходыПоКварПлате (подсистема Экзамен-документы) и регистр накопления «КомпенсацияКварПлаты»:**

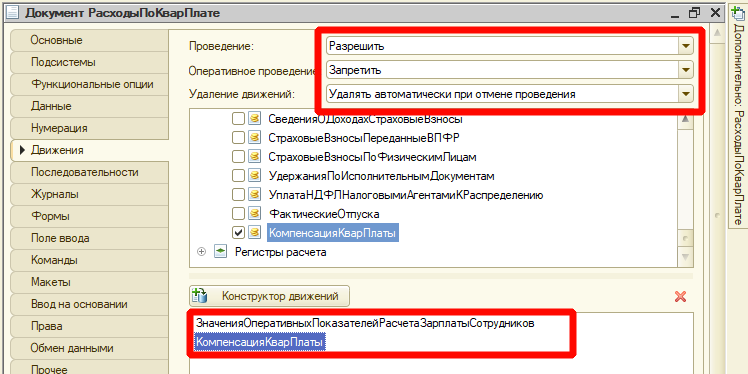


Реквизиты(1) – **Организация** (СправочникСсылка.Организации), Табл.часть – **Данные** (6) – **Сотрудник** (СправочникСсылка.Сотрудники), **КатегорияДолжности** (СправочникСсылка.ТарифныеСетки), **РазмерОклада**, **ПлановыйЛимит**, **ЗаработанныйЛимит** (Число 10.2), **КоэффициентВремени** (Число 3.2)

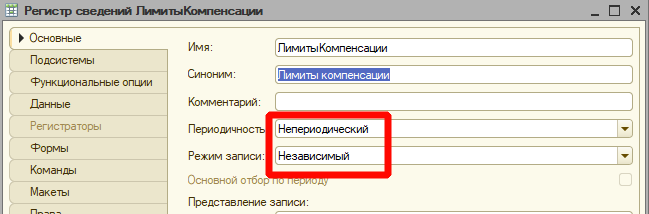


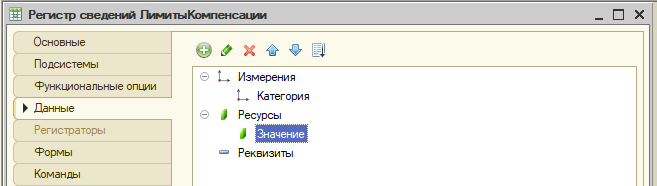


Реквизиты шапки: **Организация** (СправочникСсылка.Организации), табличная часть – **Данные** – **Сотрудник** – (СправочникСсылка.Сотрудники), **Сумма** – Число – 10,2



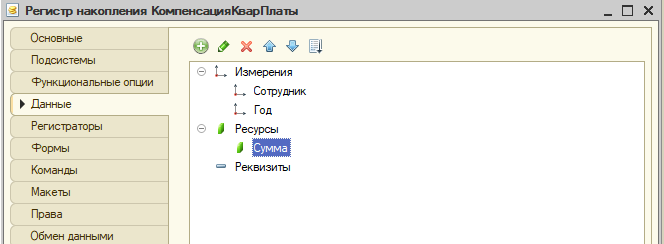
1. **Добавляем регистр сведений ЛимитыКомпенсации (Подсистема-экзамен-прочее):**





Измерения(1) – **Категория** (СправочникСсылка.ТарифныеСетки), Ресурсы(1) – **Значение** (Число 10.2)

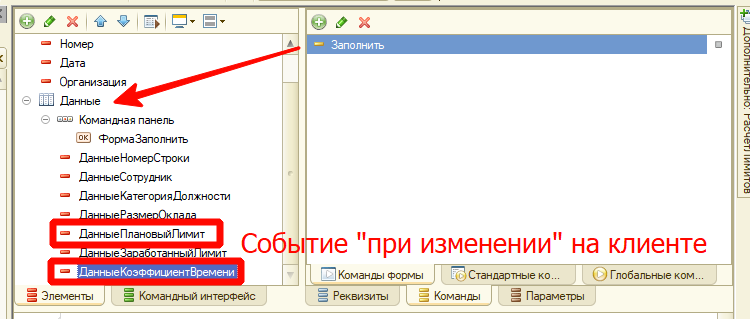
1. **Редактируем созданный регистр КомпенсацияКварПлаты (подсистема-экзамен-прочее):**



Измерения(2) – **Сотрудник** (СправочникСсылка.Сотрудники) и **Год** (Дата), Ресурсы(1) – **Сумма** (Число 10.2). **Регистраторы** – вновь введенные документы - **РасчетЛимитов** и **РасходыПоКварПлате**

1. **Работаем с документом РасчетЛимитов. Создаем форму документа.**

**Создаем и перетаскиваем на форму клиент-серверную команду «Заполнить» а для ДанныеПлановыйЛимит и ДанныеКоэффициентВремени создаем событие при изменении на клиенте:**



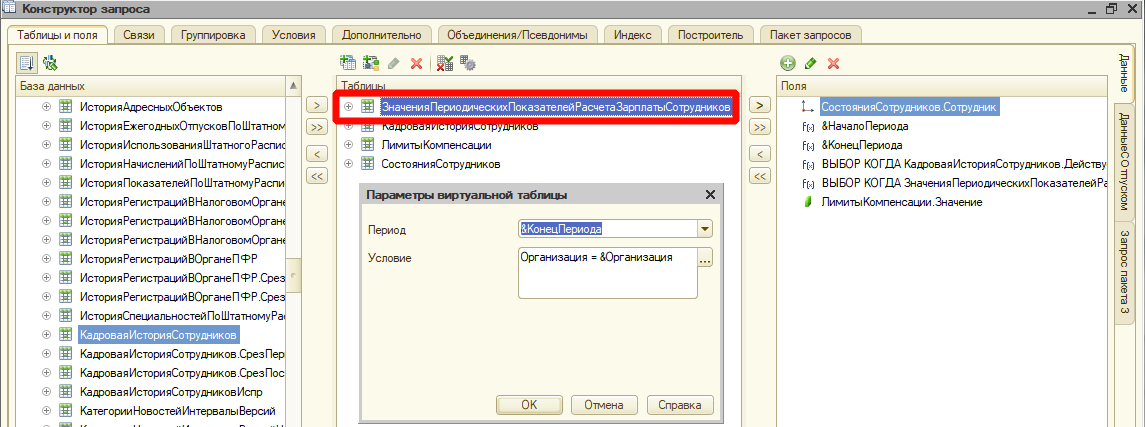
**МОДУЛЬ ФОРМЫ документа РасчетЛимитов:**

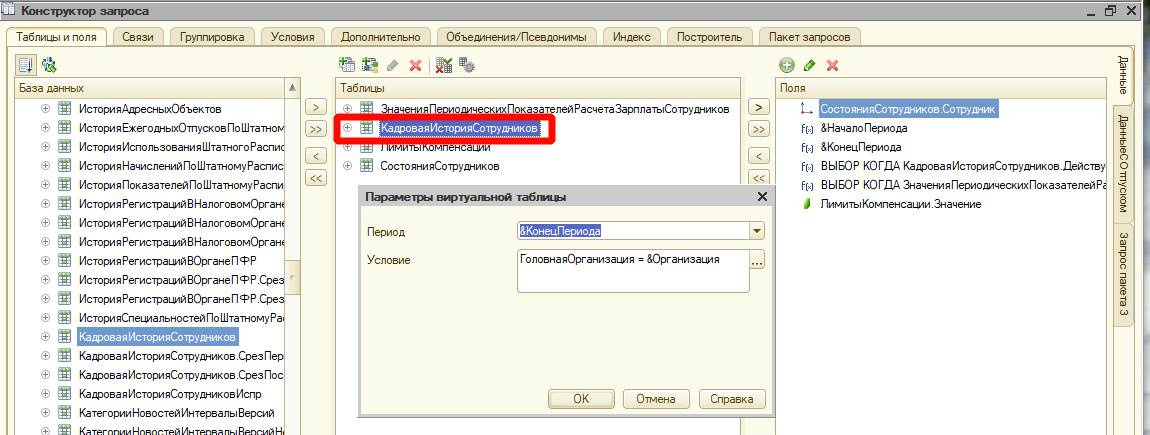
**ЗАПРОС:**

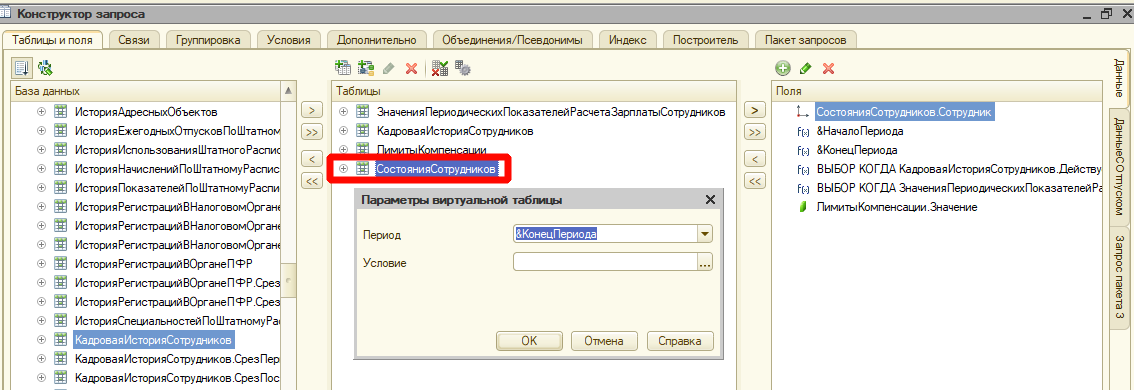
**ПЕРВЫЙ ПАКЕТ:**

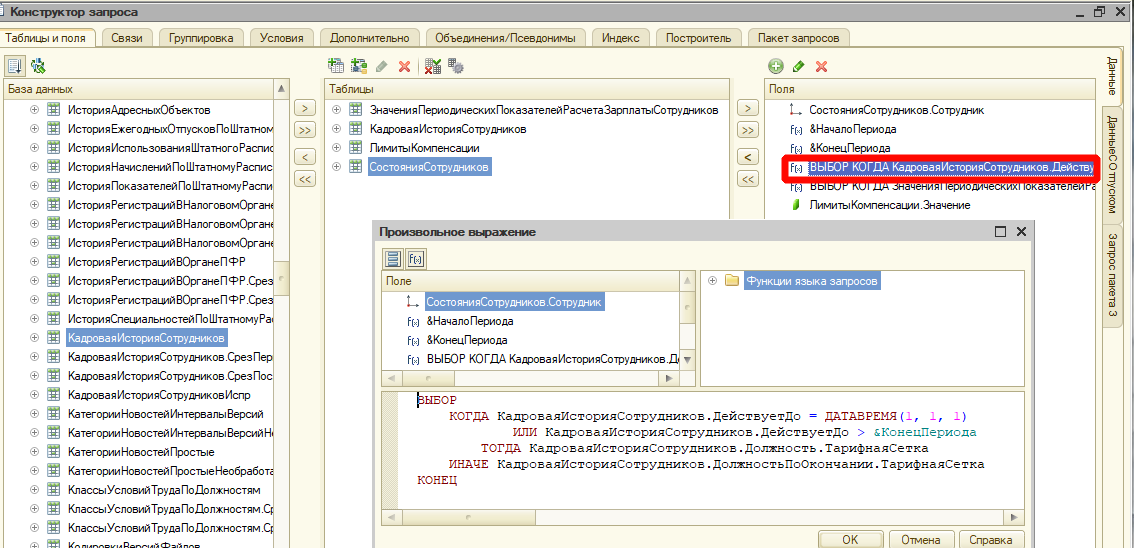
**ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудниковСрезПоследних** – переименовать в ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников, **СостоянияСотрудниковСрезПоследних** переименовать в СостоянияСотрудников

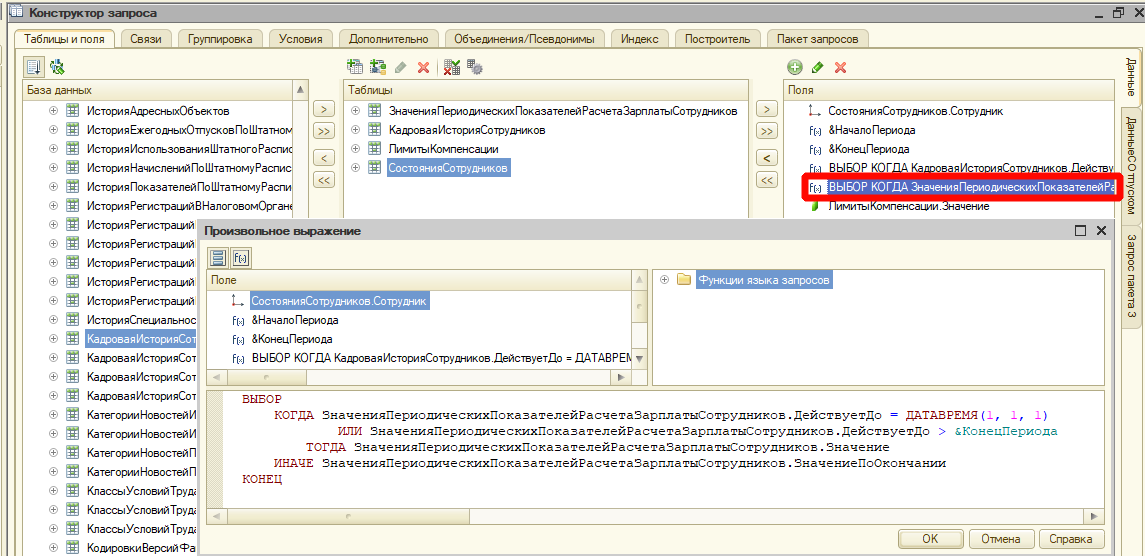
Тоже **КадроваяИсторияСотрудниковСрезПоследних** переименовать.

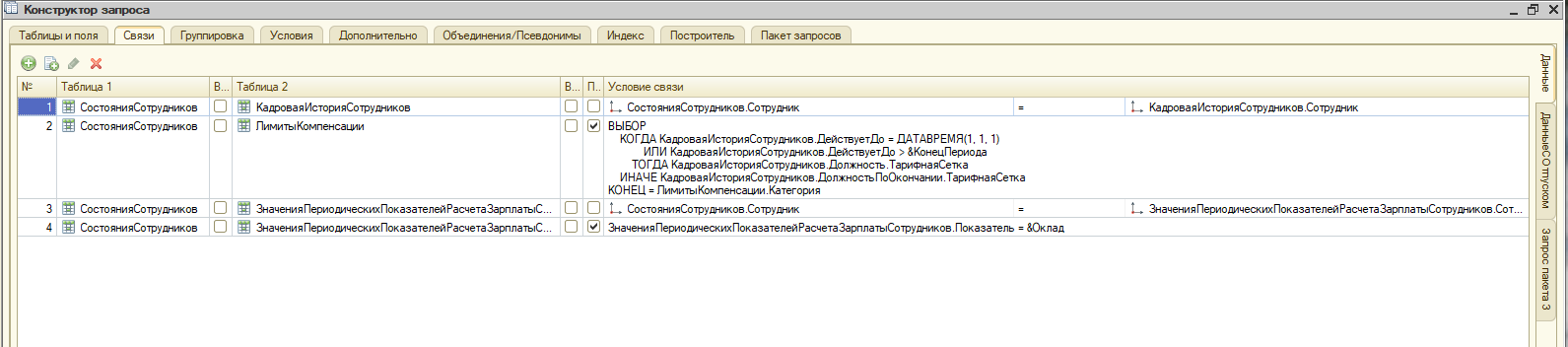


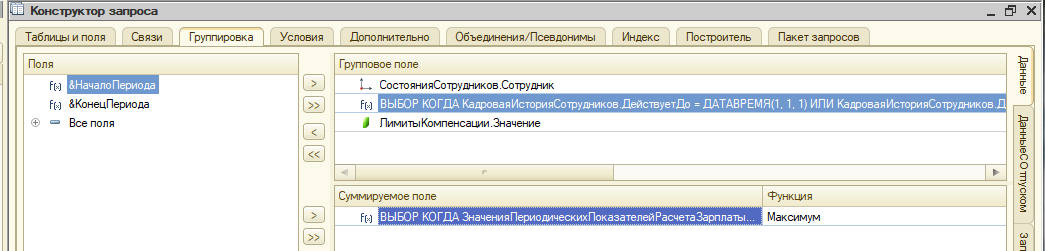




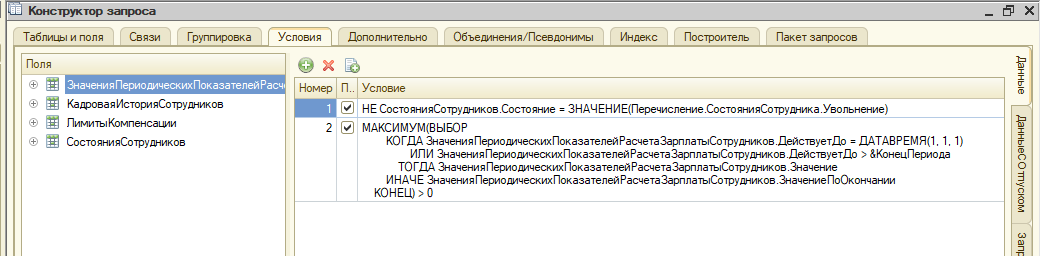


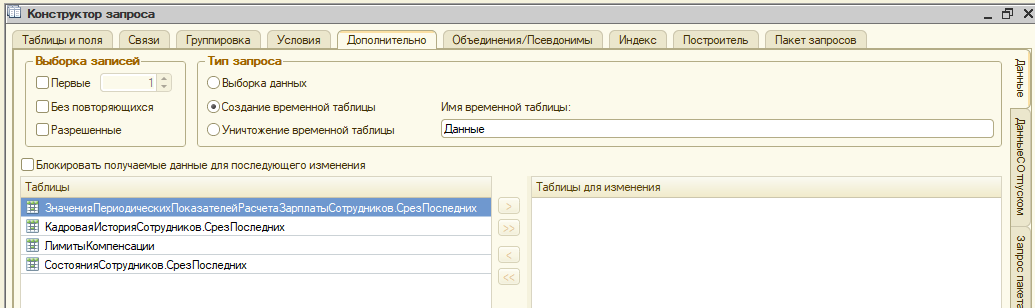


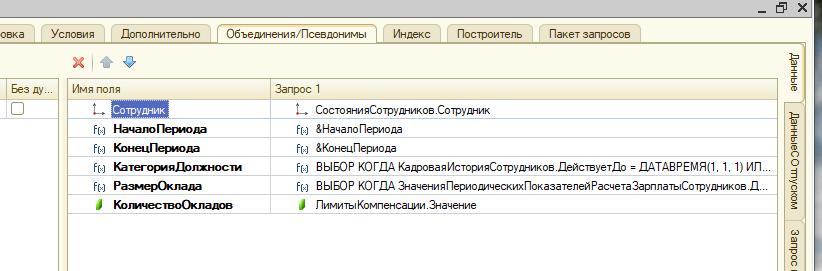




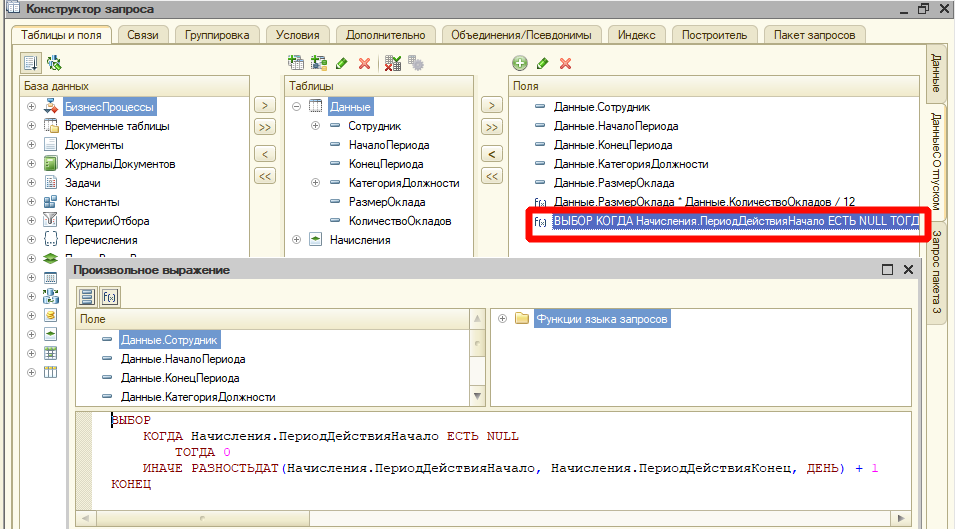
**Если условие «увольнение» не водится – дописывать вручную:**

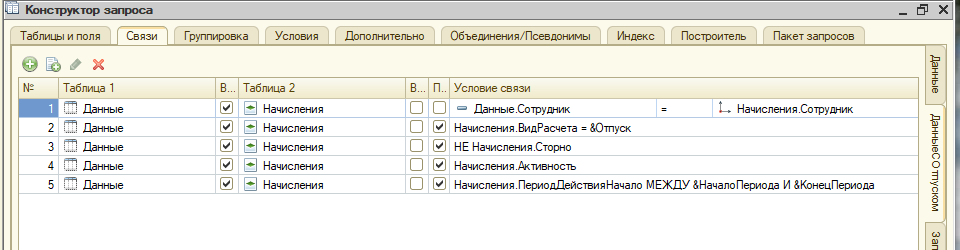


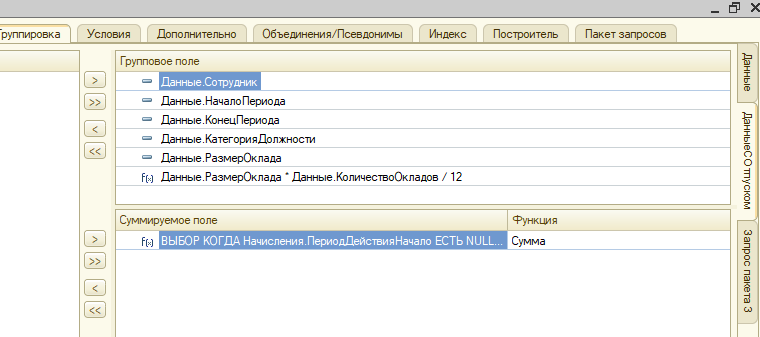


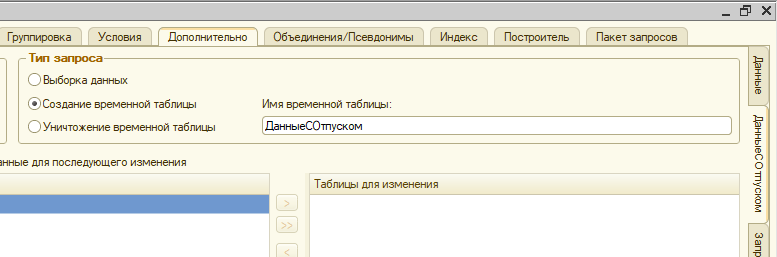


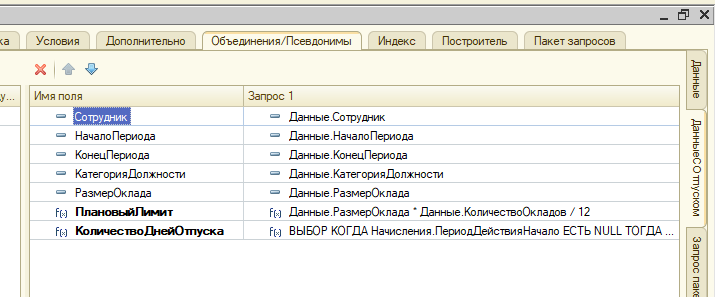
**ВТОРОЙ ПАКЕТ:**



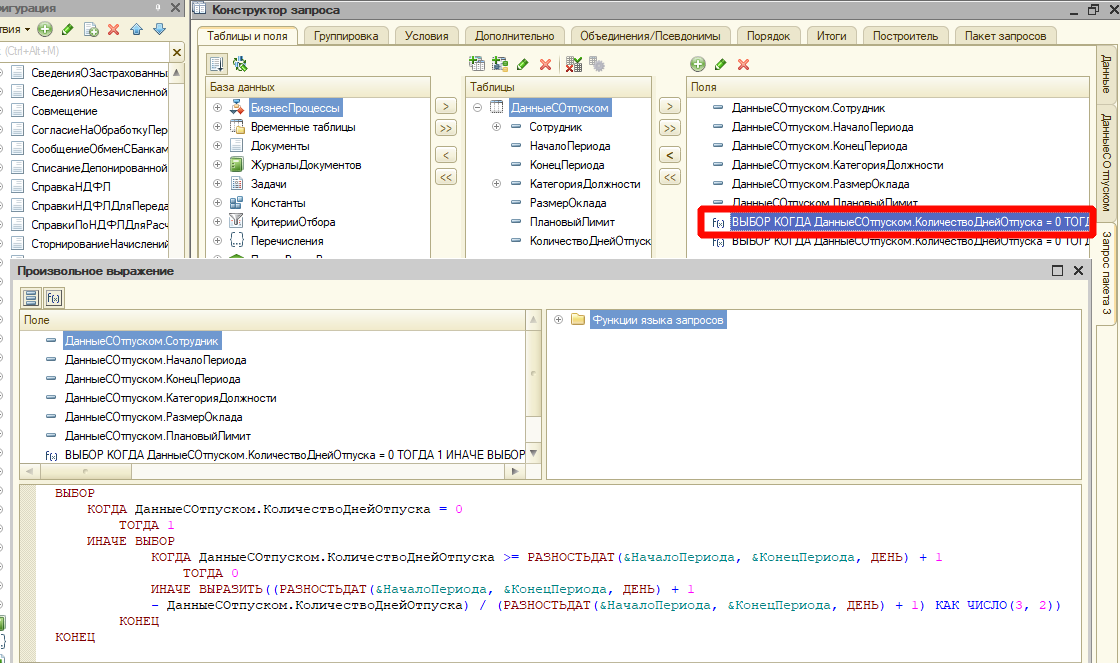


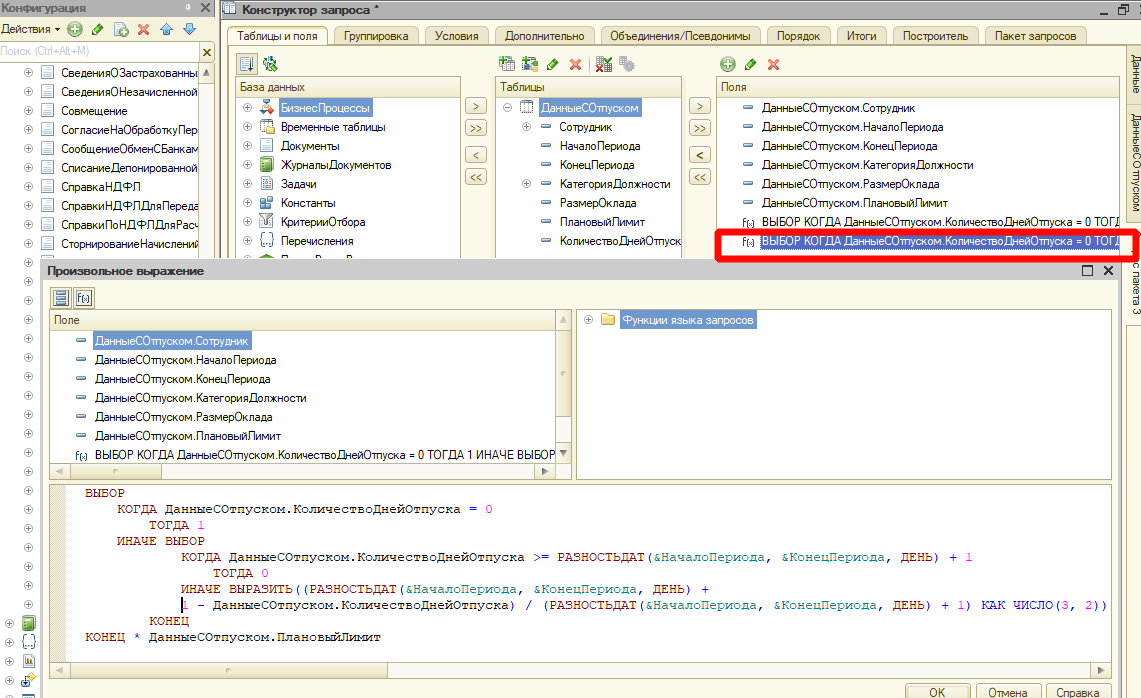


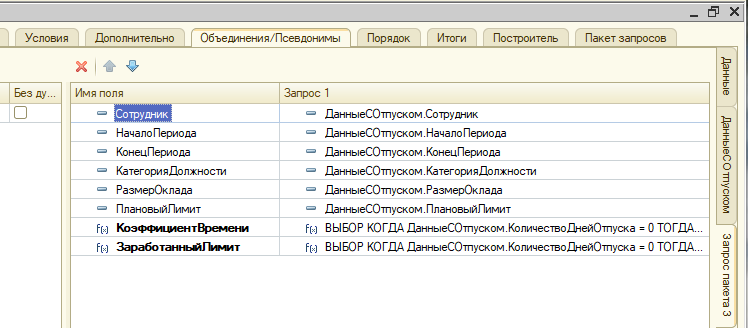




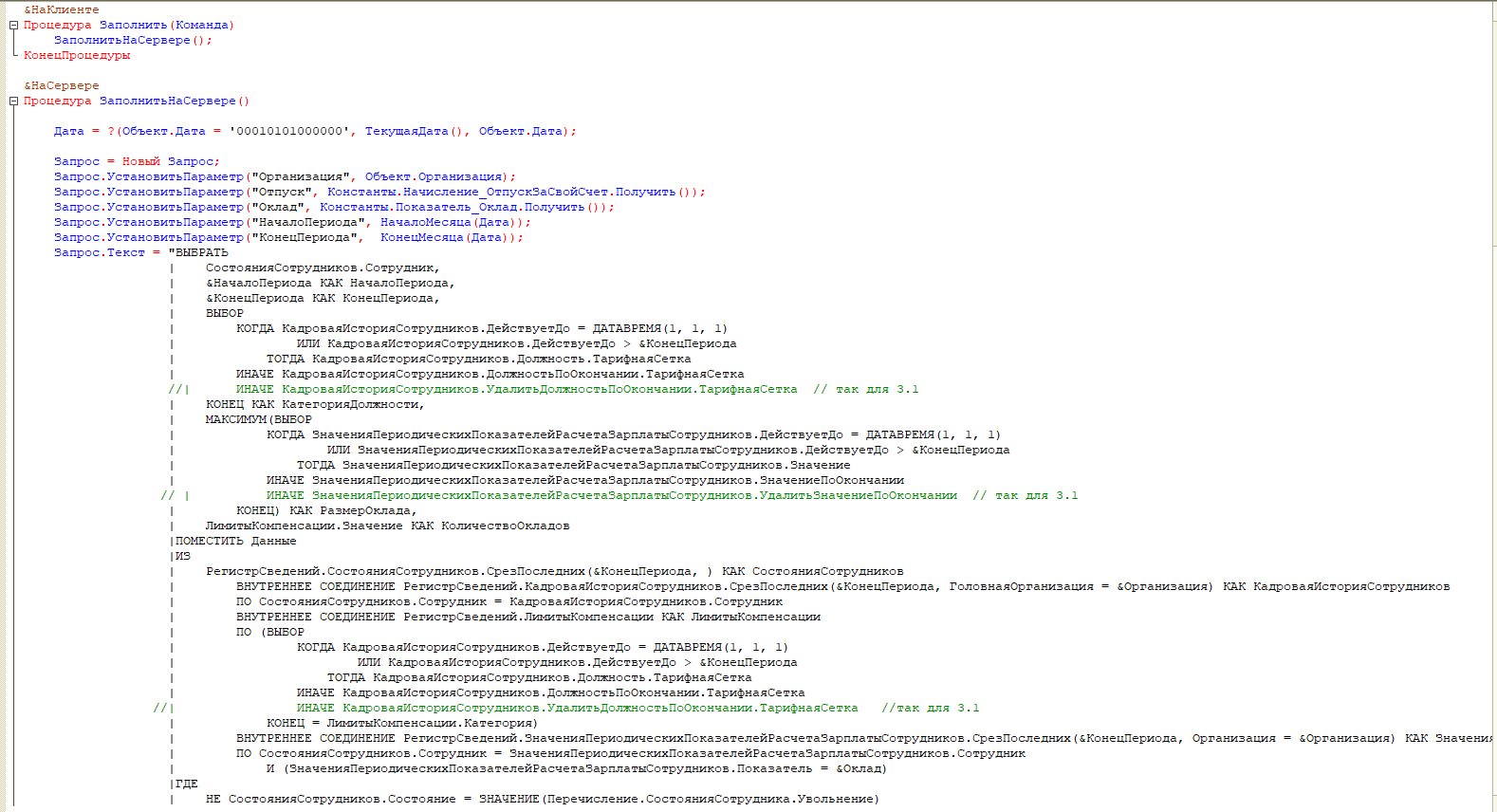
**ТРЕТИЙ ПАКЕТ:**





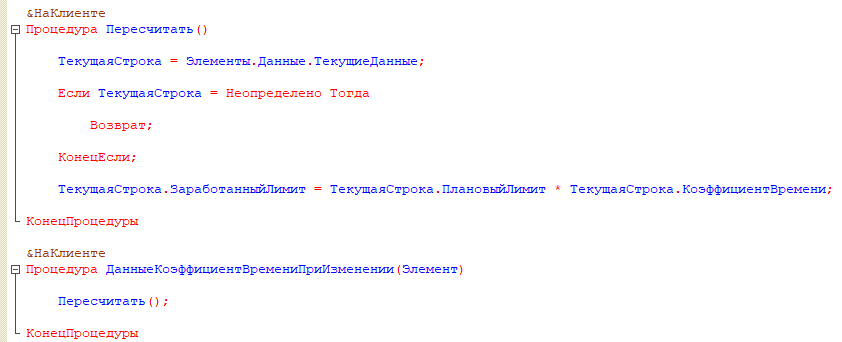


**ПОЛНОСТЬЮ МОДУЛЬ ФОРМЫ РасчетЛимитов:**









&НаКлиенте

Процедура Заполнить(Команда)

ЗаполнитьНаСервере();

КонецПроцедуры

&НаСервере

Процедура ЗаполнитьНаСервере()

Дата = ?(Объект.Дата = '00010101000000', ТекущаяДата(), Объект.Дата);

Запрос = Новый Запрос;

Запрос.УстановитьПараметр("Организация", Объект.Организация);

Запрос.УстановитьПараметр("Отпуск", Константы.Начисление\_ОтпускЗаСвойСчет.Получить());

Запрос.УстановитьПараметр("Оклад", Константы.Показатель\_Оклад.Получить());

Запрос.УстановитьПараметр("НачалоПериода", НачалоМесяца(Дата));

Запрос.УстановитьПараметр("КонецПериода", КонецМесяца(Дата));

Запрос.Текст = "ВЫБРАТЬ

| СостоянияСотрудников.Сотрудник,

| &НачалоПериода КАК НачалоПериода,

| &КонецПериода КАК КонецПериода,

| ВЫБОР

| КОГДА КадроваяИсторияСотрудников.ДействуетДо = ДАТАВРЕМЯ(1, 1, 1)

| ИЛИ КадроваяИсторияСотрудников.ДействуетДо > &КонецПериода

| ТОГДА КадроваяИсторияСотрудников.Должность.ТарифнаяСетка

| ИНАЧЕ КадроваяИсторияСотрудников.ДолжностьПоОкончании.ТарифнаяСетка

//| ИНАЧЕ КадроваяИсторияСотрудников.УдалитьДолжностьПоОкончании.ТарифнаяСетка // так для 3.1

| КОНЕЦ КАК КатегорияДолжности,

| МАКСИМУМ(ВЫБОР

| КОГДА ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.ДействуетДо = ДАТАВРЕМЯ(1, 1, 1)

| ИЛИ ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.ДействуетДо > &КонецПериода

| ТОГДА ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.Значение

| ИНАЧЕ ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.ЗначениеПоОкончании

// | ИНАЧЕ ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.УдалитьЗначениеПоОкончании // так для 3.1

| КОНЕЦ) КАК РазмерОклада,

| ЛимитыКомпенсации.Значение КАК КоличествоОкладов

|ПОМЕСТИТЬ Данные

|ИЗ

| РегистрСведений.СостоянияСотрудников.СрезПоследних(&КонецПериода, ) КАК СостоянияСотрудников

| ВНУТРЕННЕЕ СОЕДИНЕНИЕ РегистрСведений.КадроваяИсторияСотрудников.СрезПоследних(&КонецПериода, ГоловнаяОрганизация = &Организация) КАК КадроваяИсторияСотрудников

| ПО СостоянияСотрудников.Сотрудник = КадроваяИсторияСотрудников.Сотрудник

| ВНУТРЕННЕЕ СОЕДИНЕНИЕ РегистрСведений.ЛимитыКомпенсации КАК ЛимитыКомпенсации

| ПО (ВЫБОР

| КОГДА КадроваяИсторияСотрудников.ДействуетДо = ДАТАВРЕМЯ(1, 1, 1)

| ИЛИ КадроваяИсторияСотрудников.ДействуетДо > &КонецПериода

| ТОГДА КадроваяИсторияСотрудников.Должность.ТарифнаяСетка

| ИНАЧЕ КадроваяИсторияСотрудников.ДолжностьПоОкончании.ТарифнаяСетка

//| ИНАЧЕ КадроваяИсторияСотрудников.УдалитьДолжностьПоОкончании.ТарифнаяСетка //так для 3.1

| КОНЕЦ = ЛимитыКомпенсации.Категория)

| ВНУТРЕННЕЕ СОЕДИНЕНИЕ РегистрСведений.ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.СрезПоследних(&КонецПериода, Организация = &Организация) КАК ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников

| ПО СостоянияСотрудников.Сотрудник = ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.Сотрудник

| И (ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.Показатель = &Оклад)

|ГДЕ

| НЕ СостоянияСотрудников.Состояние = ЗНАЧЕНИЕ(Перечисление.СостоянияСотрудника.Увольнение)

|

|СГРУППИРОВАТЬ ПО

| СостоянияСотрудников.Сотрудник,

| ВЫБОР

| КОГДА КадроваяИсторияСотрудников.ДействуетДо = ДАТАВРЕМЯ(1, 1, 1)

| ИЛИ КадроваяИсторияСотрудников.ДействуетДо > &КонецПериода

| ТОГДА КадроваяИсторияСотрудников.Должность.ТарифнаяСетка

| ИНАЧЕ КадроваяИсторияСотрудников.ДолжностьПоОкончании.ТарифнаяСетка

//| ИНАЧЕ КадроваяИсторияСотрудников.УдалитьДолжностьПоОкончании.ТарифнаяСетка //так для 3.1

| КОНЕЦ,

| ЛимитыКомпенсации.Значение

|

|ИМЕЮЩИЕ

| МАКСИМУМ(ВЫБОР

| КОГДА ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.ДействуетДо = ДАТАВРЕМЯ(1, 1, 1)

| ИЛИ ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.ДействуетДо > &КонецПериода

| ТОГДА ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.Значение

| ИНАЧЕ ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.ЗначениеПоОкончании

//| ИНАЧЕ ЗначенияПериодическихПоказателейРасчетаЗарплатыСотрудников.УдалитьЗначениеПоОкончании //так для 3.1

| КОНЕЦ) > 0

|;

|

|////////////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| Данные.Сотрудник,

| Данные.НачалоПериода,

| Данные.КонецПериода,

| Данные.КатегорияДолжности,

| Данные.РазмерОклада,

| Данные.РазмерОклада \* Данные.КоличествоОкладов / 12 КАК ПлановыйЛимит,

| СУММА(ВЫБОР

| КОГДА Начисления.ПериодДействияНачало ЕСТЬ NULL

| ТОГДА 0

| ИНАЧЕ РАЗНОСТЬДАТ(Начисления.ПериодДействияНачало, Начисления.ПериодДействияКонец, ДЕНЬ) + 1

| КОНЕЦ) КАК КоличествоДнейОтпуска

|ПОМЕСТИТЬ ДанныеСОтпуском

|ИЗ

| Данные КАК Данные

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрРасчета.Начисления КАК Начисления

| ПО Данные.Сотрудник = Начисления.Сотрудник

| И (Начисления.ВидРасчета = &Отпуск)

| И (НЕ Начисления.Сторно)

| И (Начисления.Активность)

| И (Начисления.ПериодДействияНачало МЕЖДУ &НачалоПериода И &КонецПериода)

|

|СГРУППИРОВАТЬ ПО

| Данные.Сотрудник,

| Данные.НачалоПериода,

| Данные.КонецПериода,

| Данные.КатегорияДолжности,

| Данные.РазмерОклада,

| Данные.РазмерОклада \* Данные.КоличествоОкладов / 12

|;

|

|////////////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| ДанныеСОтпуском.Сотрудник,

| ДанныеСОтпуском.НачалоПериода,

| ДанныеСОтпуском.КонецПериода,

| ДанныеСОтпуском.КатегорияДолжности,

| ДанныеСОтпуском.РазмерОклада,

| ДанныеСОтпуском.ПлановыйЛимит,

| ВЫБОР

| КОГДА ДанныеСОтпуском.КоличествоДнейОтпуска = 0

| ТОГДА 1

| ИНАЧЕ ВЫБОР

| КОГДА ДанныеСОтпуском.КоличествоДнейОтпуска >= РАЗНОСТЬДАТ(&НачалоПериода, &КонецПериода, ДЕНЬ) + 1

| ТОГДА 0

| ИНАЧЕ ВЫРАЗИТЬ((РАЗНОСТЬДАТ(&НачалоПериода, &КонецПериода, ДЕНЬ) + 1 - ДанныеСОтпуском.КоличествоДнейОтпуска) / (РАЗНОСТЬДАТ(&НачалоПериода, &КонецПериода, ДЕНЬ) + 1) КАК ЧИСЛО(3, 2))

| КОНЕЦ

| КОНЕЦ КАК КоэффициентВремени,

| ВЫБОР

| КОГДА ДанныеСОтпуском.КоличествоДнейОтпуска = 0

| ТОГДА 1

| ИНАЧЕ ВЫБОР

| КОГДА ДанныеСОтпуском.КоличествоДнейОтпуска >= РАЗНОСТЬДАТ(&НачалоПериода, &КонецПериода, ДЕНЬ) + 1

| ТОГДА 0

| ИНАЧЕ ВЫРАЗИТЬ((РАЗНОСТЬДАТ(&НачалоПериода, &КонецПериода, ДЕНЬ) + 1 - ДанныеСОтпуском.КоличествоДнейОтпуска) / (РАЗНОСТЬДАТ(&НачалоПериода, &КонецПериода, ДЕНЬ) + 1) КАК ЧИСЛО(3, 2))

| КОНЕЦ

| КОНЕЦ \* ДанныеСОтпуском.ПлановыйЛимит КАК ЗаработанныйЛимит

|ИЗ

| ДанныеСОтпуском КАК ДанныеСОтпуском";

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

Объект.Данные.Очистить();

Пока Выборка.Следующий() Цикл

СтрокаТаблицыДанные = Объект.Данные.Добавить();

ЗаполнитьЗначенияСвойств(СтрокаТаблицыДанные, Выборка);

КонецЦикла;

КонецПроцедуры

&НаКлиенте

Процедура ДанныеПлановыйЛимитПриИзменении(Элемент)

Пересчитать();

КонецПроцедуры

&НаКлиенте

Процедура Пересчитать()

ТекущаяСтрока = Элементы.Данные.ТекущиеДанные;

Если ТекущаяСтрока = Неопределено Тогда

Возврат;

КонецЕсли;

ТекущаяСтрока.ЗаработанныйЛимит = ТекущаяСтрока.ПлановыйЛимит \*

ТекущаяСтрока.КоэффициентВремени;

КонецПроцедуры

&НаКлиенте

Процедура ДанныеКоэффициентВремениПриИзменении(Элемент)

Пересчитать();

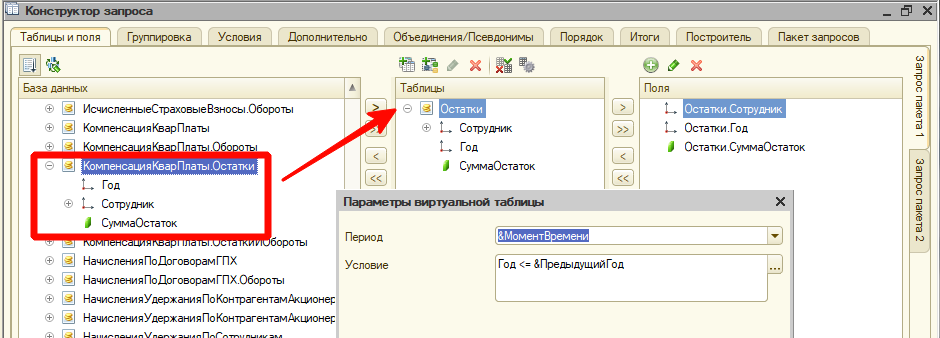
КонецПроцедуры

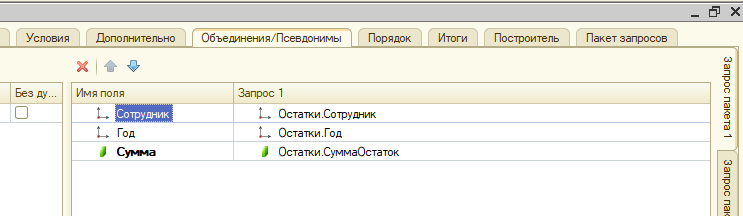
**МОДУЛЬ ОБЪЕКТА документа РасчетЛимитов:**

**Сначала делаем движение конструктором:**

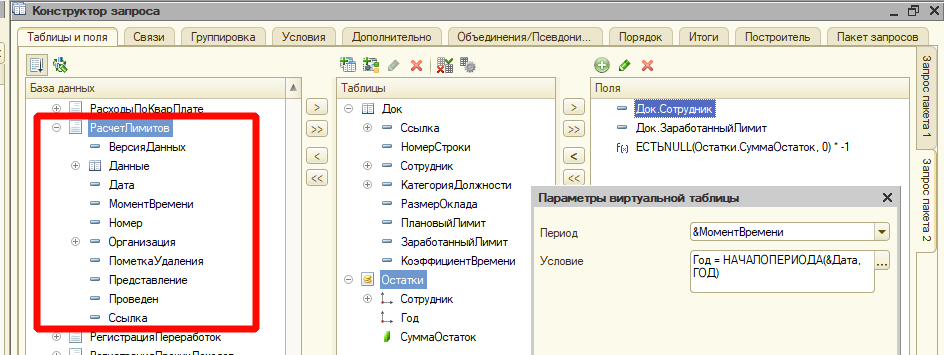
**ЗАПРОС:**

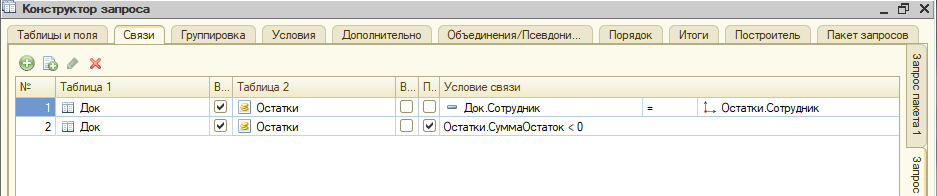
**ПЕРВЫЙ ПАКЕТ:**

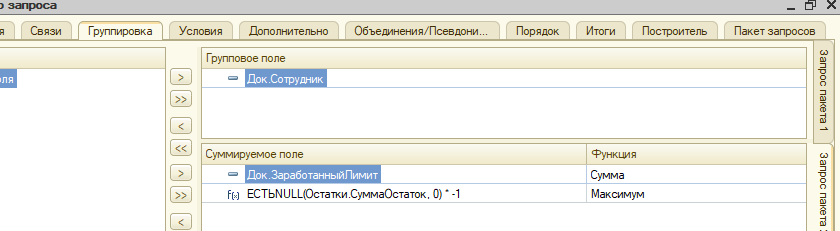


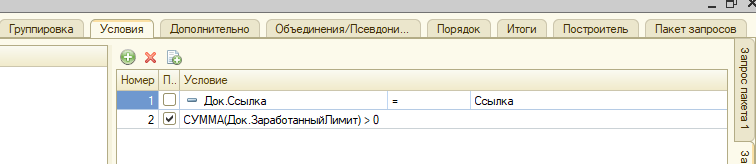


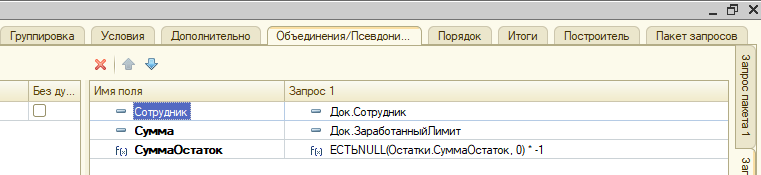
**ВТОРОЙ ПАКЕТ:**





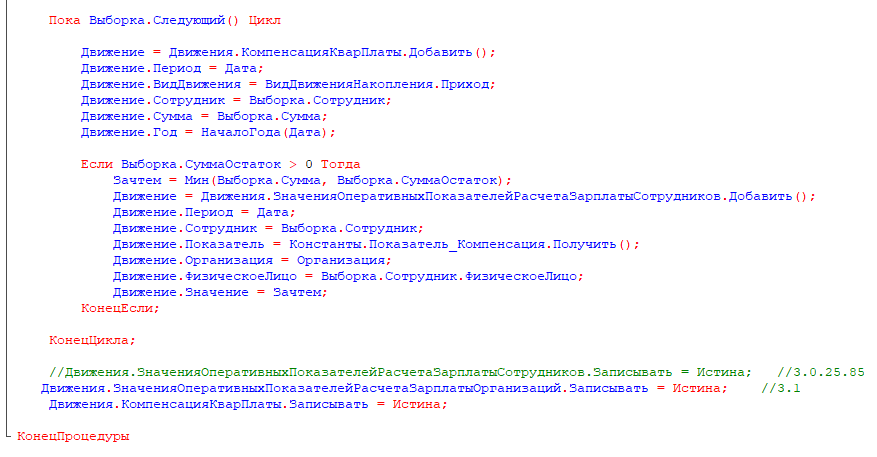






**ПОЛНОСТЬЮ МОДУЛЬ РасчетЛимитов полностью:**





Процедура ОбработкаПроведения(Отказ, Режим)

Движения.КомпенсацияКварПлаты.Записать();

Запрос = Новый Запрос;

Запрос.УстановитьПараметр("Ссылка", Ссылка);

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Запрос.УстановитьПараметр("Дата", Дата);

Запрос.УстановитьПараметр("ПредыдущийГод", НачалоГода(ДобавитьМесяц(Дата, -12)));

Запрос.Текст = "ВЫБРАТЬ

| Остатки.Сотрудник,

| Остатки.Год,

| Остатки.СуммаОстаток КАК Сумма

|ИЗ

| РегистрНакопления.КомпенсацияКварПлаты.Остатки(&МоментВремени, Год <= &ПредыдущийГод) КАК Остатки

|;

|

|////////////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| Док.Сотрудник,

| СУММА(Док.ЗаработанныйЛимит) КАК Сумма,

| МАКСИМУМ(ЕСТЬNULL(Остатки.СуммаОстаток, 0) \* -1) КАК СуммаОстаток

|ИЗ

| Документ.РасчетЛимитов.Данные КАК Док

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.КомпенсацияКварПлаты.Остатки(&МоментВремени, Год = НАЧАЛОПЕРИОДА(&Дата, ГОД)) КАК Остатки

| ПО Док.Сотрудник = Остатки.Сотрудник

| И (Остатки.СуммаОстаток < 0)

|ГДЕ

| Док.Ссылка = &Ссылка

|

|СГРУППИРОВАТЬ ПО

| Док.Сотрудник

|

|ИМЕЮЩИЕ

| СУММА(Док.ЗаработанныйЛимит) > 0";

Результат = Запрос.ВыполнитьПакет();

Выборка = Результат[1].Выбрать();

ВыборкаКорректировкаПрошлыхПериодов = Результат[0].Выбрать();

Пока ВыборкаКорректировкаПрошлыхПериодов.Следующий() Цикл

Движение = Движения.КомпенсацияКварПлаты.Добавить();

Движение.Период = Дата;

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Сотрудник = ВыборкаКорректировкаПрошлыхПериодов.Сотрудник;

Движение.Сумма = ВыборкаКорректировкаПрошлыхПериодов.Сумма;

Движение.Год = ВыборкаКорректировкаПрошлыхПериодов.Год;

КонецЦикла;

Пока Выборка.Следующий() Цикл

Движение = Движения.КомпенсацияКварПлаты.Добавить();

Движение.Период = Дата;

Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

Движение.Сотрудник = Выборка.Сотрудник;

Движение.Сумма = Выборка.Сумма;

Движение.Год = НачалоГода(Дата);

Если Выборка.СуммаОстаток > 0 Тогда

Зачтем = Мин(Выборка.Сумма, Выборка.СуммаОстаток);

Движение = Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников.Добавить();

Движение.Период = Дата;

Движение.Сотрудник = Выборка.Сотрудник;

Движение.Показатель = Константы.Показатель\_Компенсация.Получить();

Движение.Организация = Организация;

Движение.ФизическоеЛицо = Выборка.Сотрудник.ФизическоеЛицо;

Движение.Значение = Зачтем;

КонецЕсли;

КонецЦикла;

//Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников.Записывать = Истина; //3.0.25.85

Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыОрганизаций.Записывать = Истина; //3.1

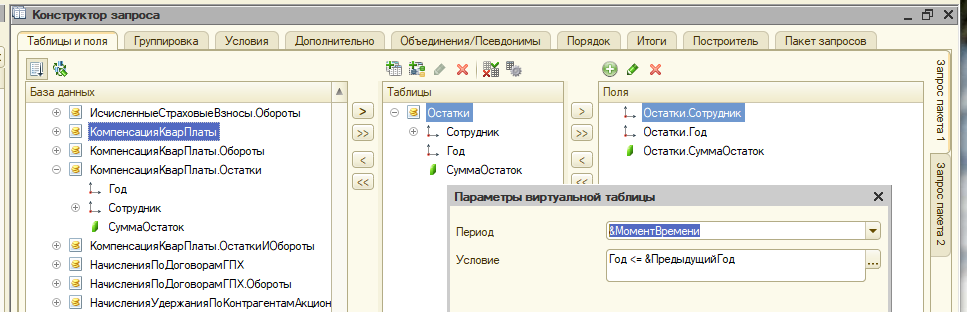
Движения.КомпенсацияКварПлаты.Записывать = Истина;

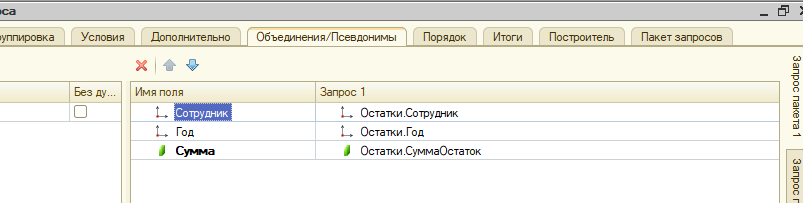
КонецПроцедуры

1. **МОДУЛЬ ОБЪЕКТА документа РасходыПоКварПлате:**

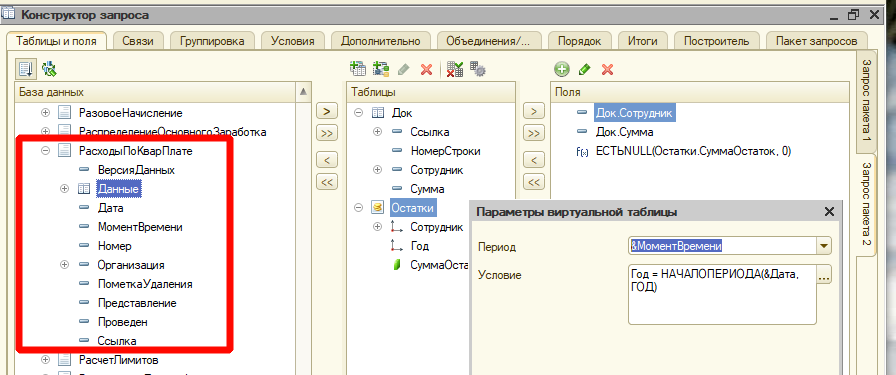
**ЗАПРОС**

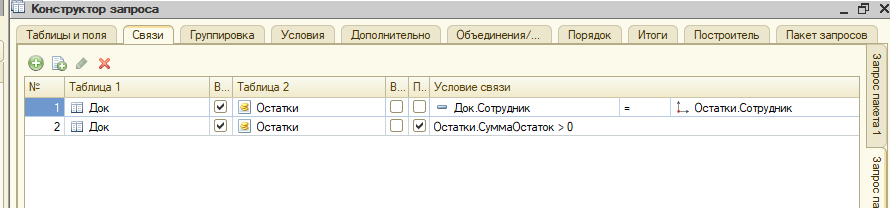
**ПЕРВЫЙ ПАКЕТ:**

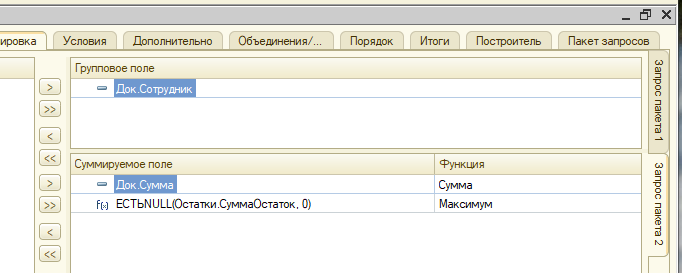


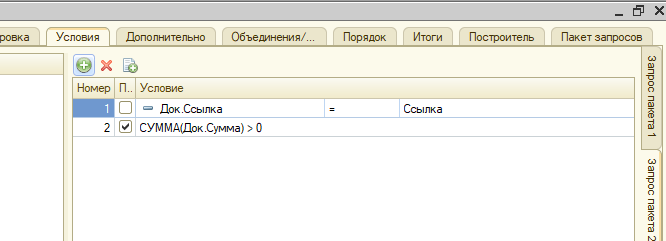


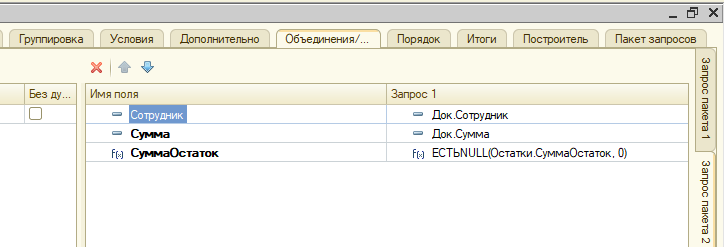
**ВТОРОЙ ПАКЕТ:**





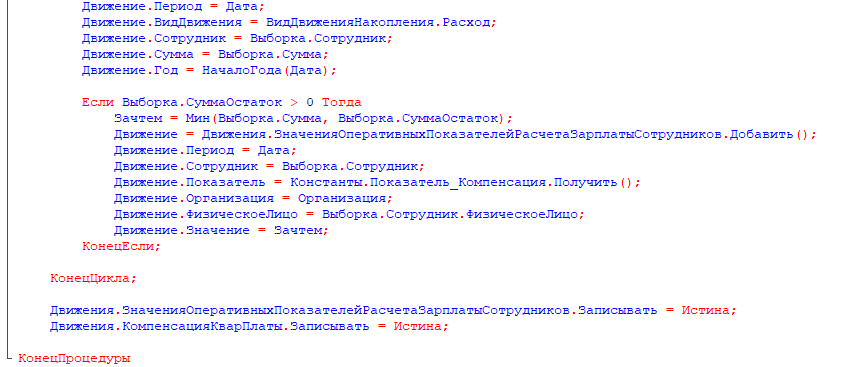






**Полностью модуль. После запроса код можно скопировать с документа РасчетЛимитов:**





Процедура ОбработкаПроведения(Отказ, Режим)

Движения.КомпенсацияКварПлаты.Записать();

Запрос = Новый Запрос;

Запрос.УстановитьПараметр("Ссылка", Ссылка);

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Запрос.УстановитьПараметр("Дата", Дата);

Запрос.УстановитьПараметр("КонецПредыдущегоГода", Новый Граница(КонецГода(ДобавитьМесяц(Дата, -12)),

ВидГраницы.Включая));

Запрос.УстановитьПараметр("ПредыдущийГод", НачалоГода(ДобавитьМесяц(Дата, -12)));

Запрос.Текст = "ВЫБРАТЬ

| Остатки.Сотрудник,

| Остатки.Год,

| Остатки.СуммаОстаток КАК Сумма

|ИЗ

| РегистрНакопления.КомпенсацияКварПлаты.Остатки(&МоментВремени, Год <= &ПредыдущийГод) КАК Остатки

|;

|

|////////////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| Док.Сотрудник,

| СУММА(Док.Сумма) КАК Сумма,

| МАКСИМУМ(ЕСТЬNULL(Остатки.СуммаОстаток, 0)) КАК СуммаОстаток

|ИЗ

| Документ.РасходыПоКварПлате.Данные КАК Док

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.КомпенсацияКварПлаты.Остатки(&МоментВремени, Год = НАЧАЛОПЕРИОДА(&Дата, ГОД)) КАК Остатки

| ПО Док.Сотрудник = Остатки.Сотрудник

| И (Остатки.СуммаОстаток > 0)

|ГДЕ

| Док.Ссылка = &Ссылка

|

|СГРУППИРОВАТЬ ПО

| Док.Сотрудник

|

|ИМЕЮЩИЕ

| СУММА(Док.Сумма) > 0";

Результат = Запрос.ВыполнитьПакет();

Выборка = Результат[1].Выбрать();

ВыборкаКорректировкаПрошлыхПериодов = Результат[0].Выбрать();

Пока ВыборкаКорректировкаПрошлыхПериодов.Следующий() Цикл

Движение = Движения.КомпенсацияКварПлаты.Добавить();

Движение.Период = Дата;

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Сотрудник = ВыборкаКорректировкаПрошлыхПериодов.Сотрудник;

Движение.Сумма = ВыборкаКорректировкаПрошлыхПериодов.Сумма;

Движение.Год = ВыборкаКорректировкаПрошлыхПериодов.Год;

КонецЦикла;

Пока Выборка.Следующий() Цикл

Движение = Движения.КомпенсацияКварПлаты.Добавить();

Движение.Период = Дата;

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Сотрудник = Выборка.Сотрудник;

Движение.Сумма = Выборка.Сумма;

Движение.Год = НачалоГода(Дата);

Если Выборка.СуммаОстаток > 0 Тогда

Зачтем = Мин(Выборка.Сумма, Выборка.СуммаОстаток);

Движение = Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников.Добавить();

Движение.Период = Дата;

Движение.Сотрудник = Выборка.Сотрудник;

Движение.Показатель = Константы.Показатель\_Компенсация.Получить();

Движение.Организация = Организация;

Движение.ФизическоеЛицо = Выборка.Сотрудник.ФизическоеЛицо;

Движение.Значение = Зачтем;

КонецЕсли;

КонецЦикла;

Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников.Записывать = Истина;

Движения.КомпенсацияКварПлаты.Записывать = Истина;

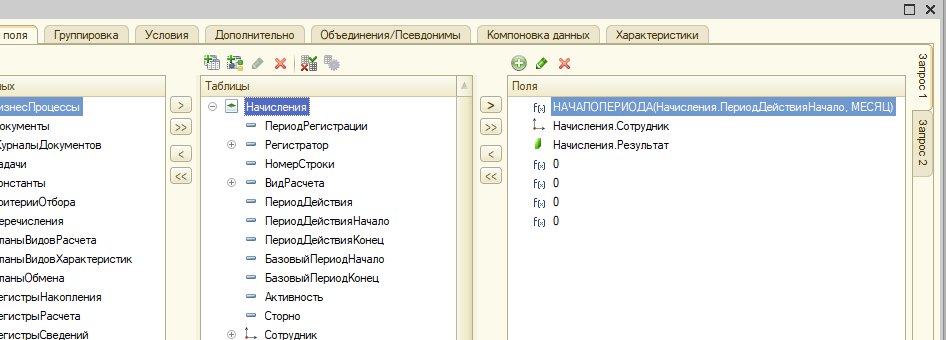
КонецПроцедуры

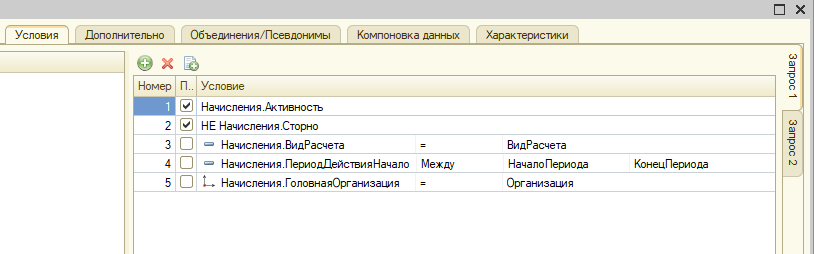
1. **Создаем отчет ВозмещениеСоциальныхРасходов (подсистема - экзамен):**

**ЗАПРОС:**

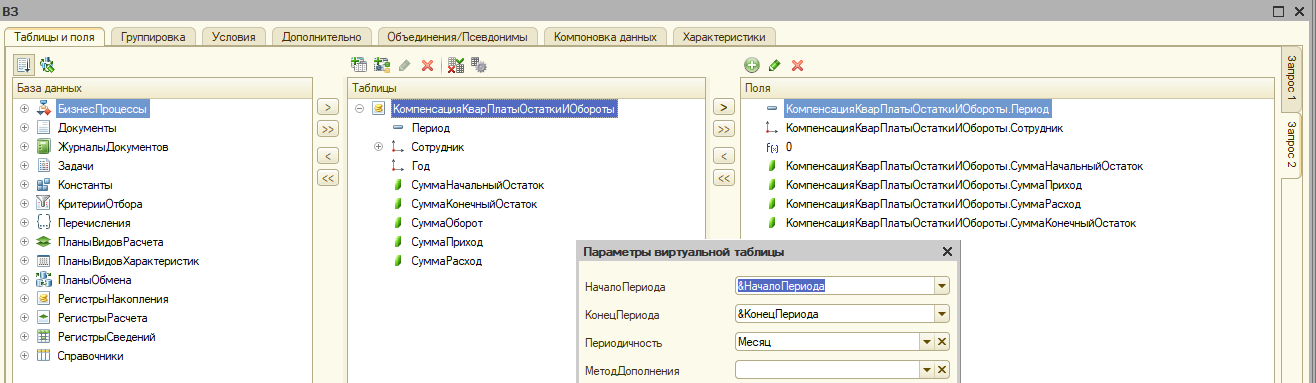
**НАЧАЛО ВЛОЖЕННОГО ЗАПРОСА ВЗ:**

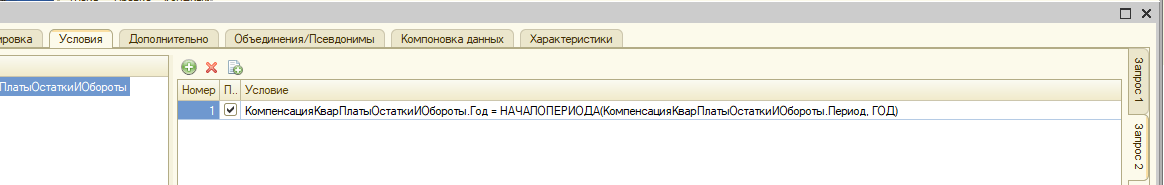
**ПЕРВЫЙ ПАКЕТ:**



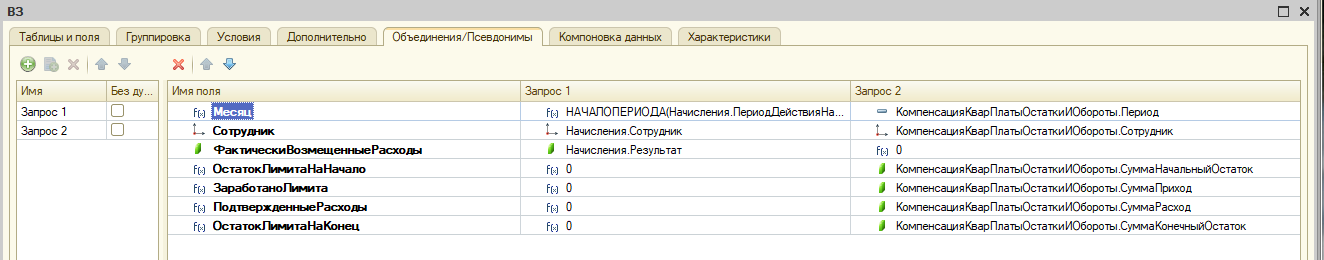


**ВТОРОЙ ПАКЕТ:**

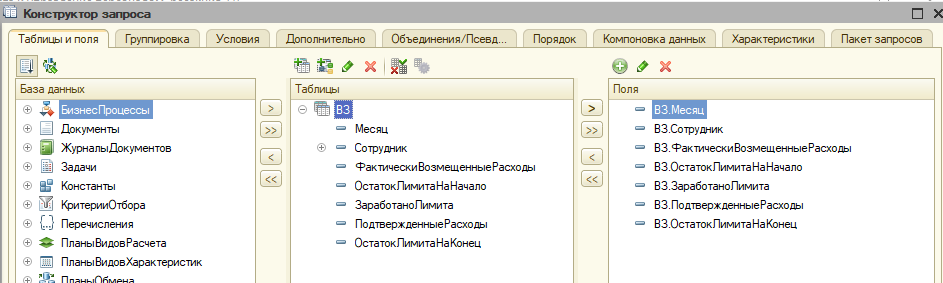


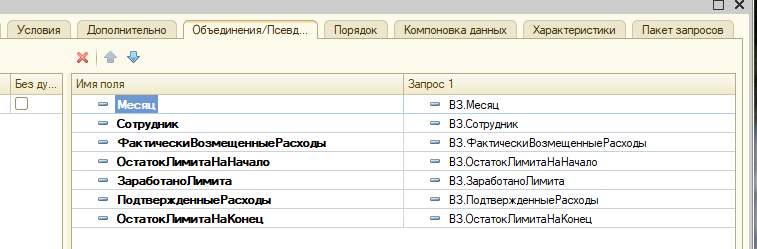


**ОБЪЕДИНЕНИЕ:**



**ОКОНЧАНИЕ ВЛОЖЕННОГО ЗАПРОСА ВЗ:**





**ТЕКСТ ЗАПРОСА отчета ВозмещениеСоциальныхРасходов:**

ВЫБРАТЬ

ВЗ.Месяц КАК Месяц,

ВЗ.Сотрудник КАК Сотрудник,

ВЗ.ФактическиВозмещенныеРасходы КАК ФактическиВозмещенныеРасходы,

ВЗ.ОстатокЛимитаНаНачало КАК ОстатокЛимитаНаНачало,

ВЗ.ЗаработаноЛимита КАК ЗаработаноЛимита,

ВЗ.ПодтвержденныеРасходы КАК ПодтвержденныеРасходы,

ВЗ.ОстатокЛимитаНаКонец КАК ОстатокЛимитаНаКонец

ИЗ

(ВЫБРАТЬ

НАЧАЛОПЕРИОДА(Начисления.ПериодДействияНачало, МЕСЯЦ) КАК Месяц,

Начисления.Сотрудник КАК Сотрудник,

Начисления.Результат КАК ФактическиВозмещенныеРасходы,

0 КАК ОстатокЛимитаНаНачало,

0 КАК ЗаработаноЛимита,

0 КАК ПодтвержденныеРасходы,

0 КАК ОстатокЛимитаНаКонец

ИЗ

РегистрРасчета.Начисления КАК Начисления

ГДЕ

Начисления.Активность

И НЕ Начисления.Сторно

И Начисления.ВидРасчета = &ВидРасчета

И Начисления.ПериодДействияНачало МЕЖДУ &НачалоПериода И &КонецПериода

И Начисления.ГоловнаяОрганизация = &Организация

ОБЪЕДИНИТЬ ВСЕ

ВЫБРАТЬ

КомпенсацияКварПлатыОстаткиИОбороты.Период,

КомпенсацияКварПлатыОстаткиИОбороты.Сотрудник,

0,

КомпенсацияКварПлатыОстаткиИОбороты.СуммаНачальныйОстаток,

КомпенсацияКварПлатыОстаткиИОбороты.СуммаПриход,

КомпенсацияКварПлатыОстаткиИОбороты.СуммаРасход,

КомпенсацияКварПлатыОстаткиИОбороты.СуммаКонечныйОстаток

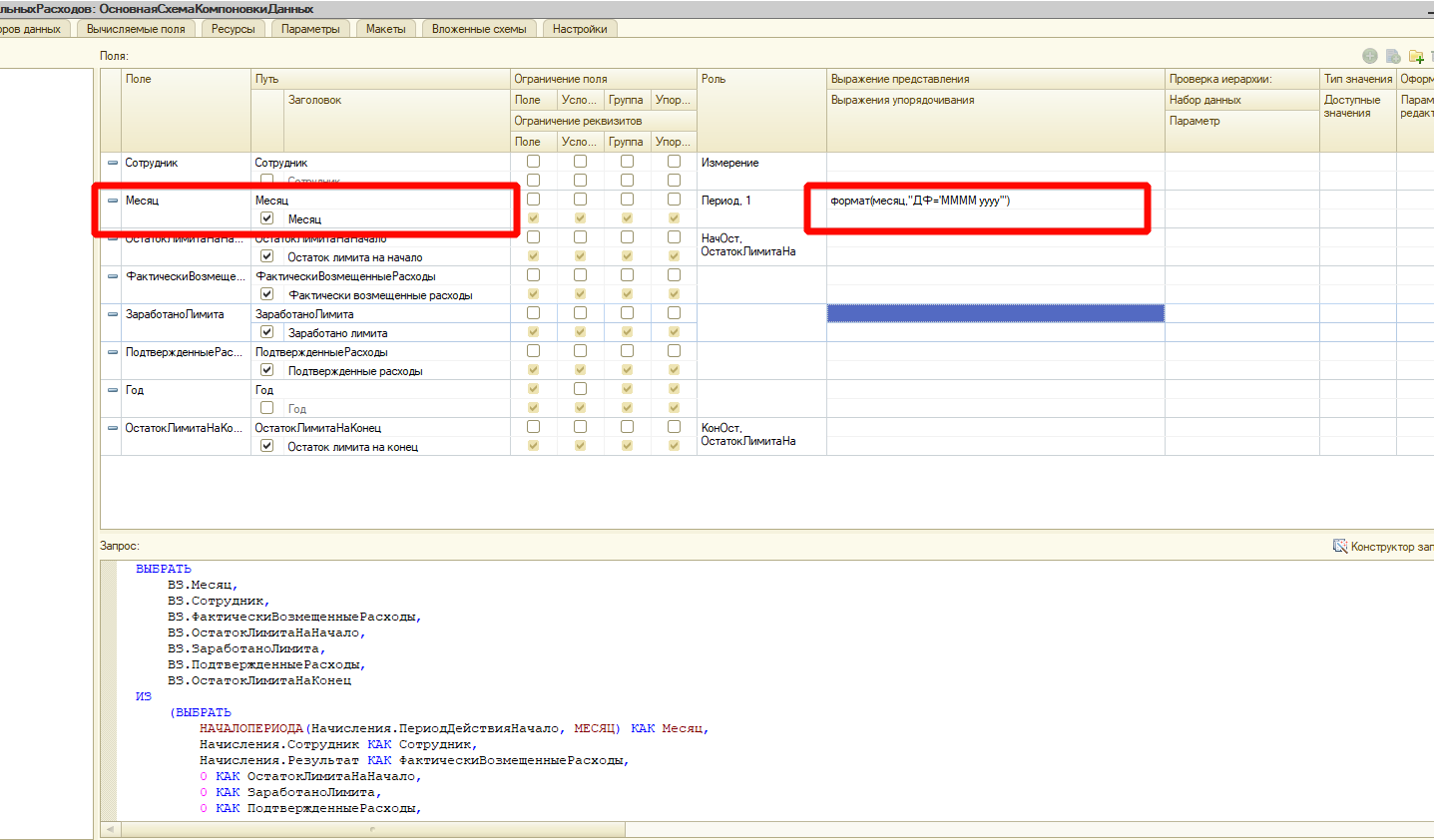
ИЗ

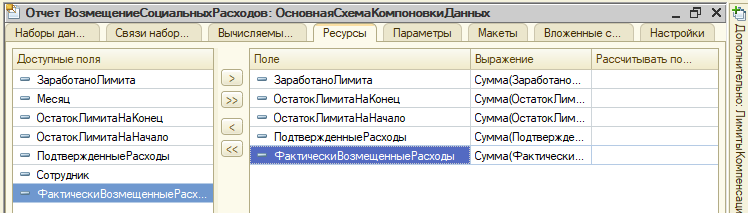
РегистрНакопления.КомпенсацияКварПлаты.ОстаткиИОбороты(&НачалоПериода, &КонецПериода, Месяц, , ) КАК КомпенсацияКварПлатыОстаткиИОбороты

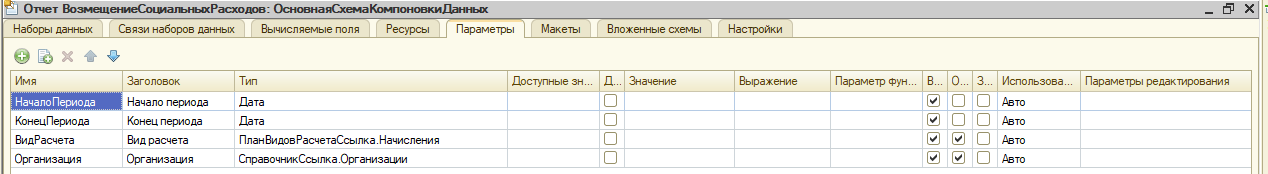
ГДЕ

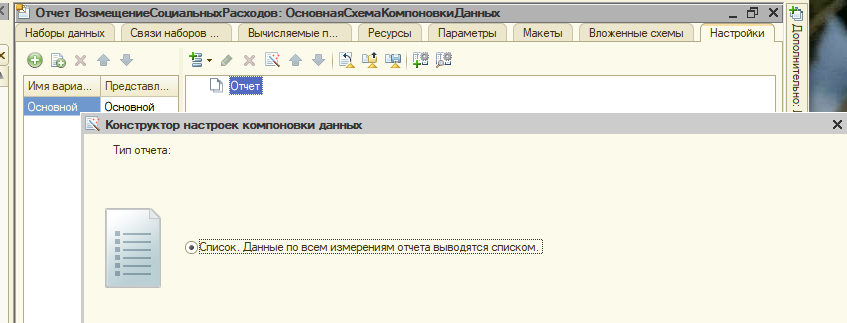
КомпенсацияКварПлатыОстаткиИОбороты.Год = НАЧАЛОПЕРИОДА(КомпенсацияКварПлатыОстаткиИОбороты.Период, ГОД)) КАК ВЗ

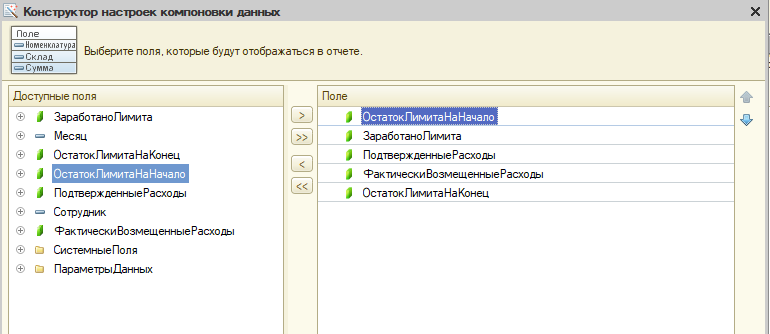
**СКД:**

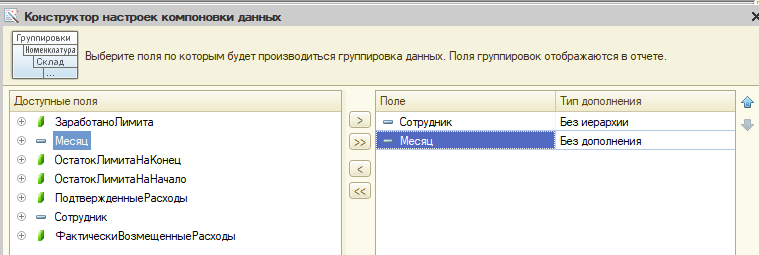




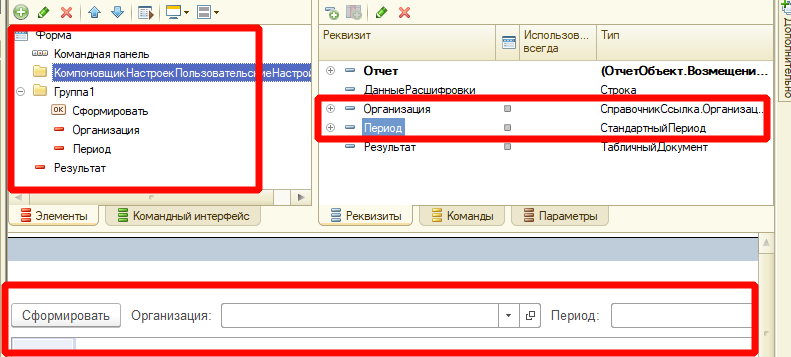




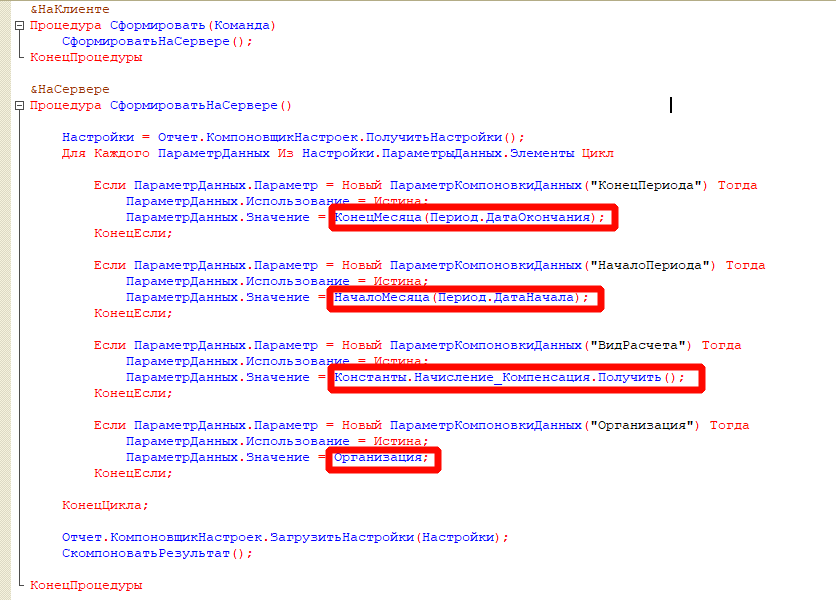




**Создаем форму отчета. Создаем команду Сформировать (клиент и сервер), реквизиты Организация и Период, удаляем лишнее. Форма должна получиться такая:**



**Модуль формы отчета ВозмещениеСоциальныхРасходов:**



&НаКлиенте

Процедура Сформировать(Команда)

СформироватьНаСервере();

КонецПроцедуры

&НаСервере

Процедура СформироватьНаСервере()

Настройки = Отчет.КомпоновщикНастроек.ПолучитьНастройки();

Для Каждого ПараметрДанных Из Настройки.ПараметрыДанных.Элементы Цикл

Если ПараметрДанных.Параметр = Новый ПараметрКомпоновкиДанных("КонецПериода") Тогда

ПараметрДанных.Использование = Истина;

ПараметрДанных.Значение = КонецМесяца(Период.ДатаОкончания);

КонецЕсли;

Если ПараметрДанных.Параметр = Новый ПараметрКомпоновкиДанных("НачалоПериода") Тогда

ПараметрДанных.Использование = Истина;

ПараметрДанных.Значение = НачалоМесяца(Период.ДатаНачала);

КонецЕсли;

Если ПараметрДанных.Параметр = Новый ПараметрКомпоновкиДанных("ВидРасчета") Тогда

ПараметрДанных.Использование = Истина;

ПараметрДанных.Значение = Константы.Начисление\_Компенсация.Получить();

КонецЕсли;

Если ПараметрДанных.Параметр = Новый ПараметрКомпоновкиДанных("Организация") Тогда

ПараметрДанных.Использование = Истина;

ПараметрДанных.Значение = Организация;

КонецЕсли;

КонецЦикла;

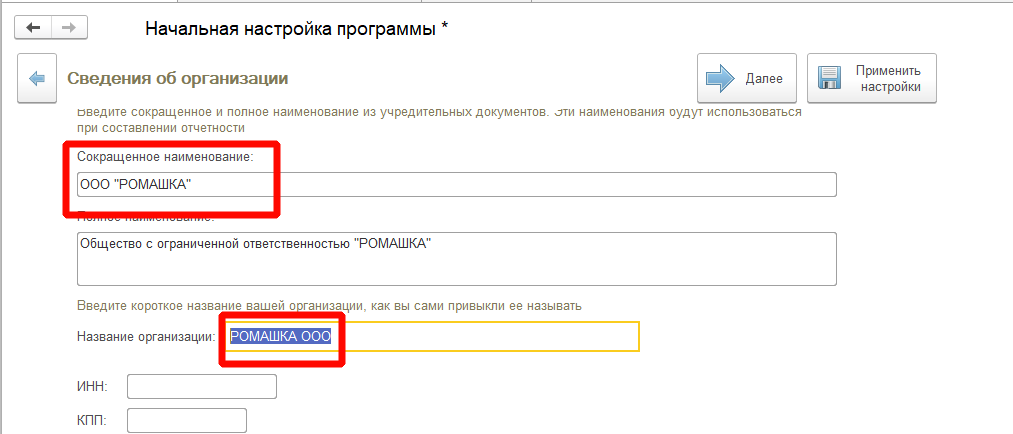
Отчет.КомпоновщикНастроек.ЗагрузитьНастройки(Настройки);

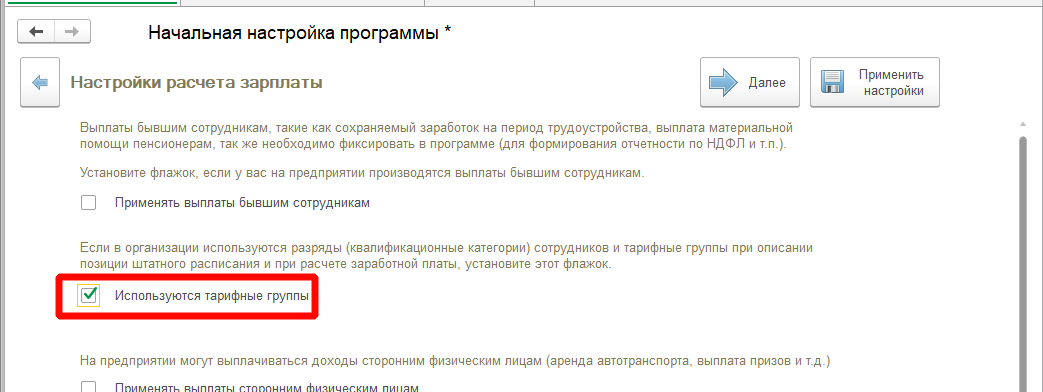
СкомпоноватьРезультат();

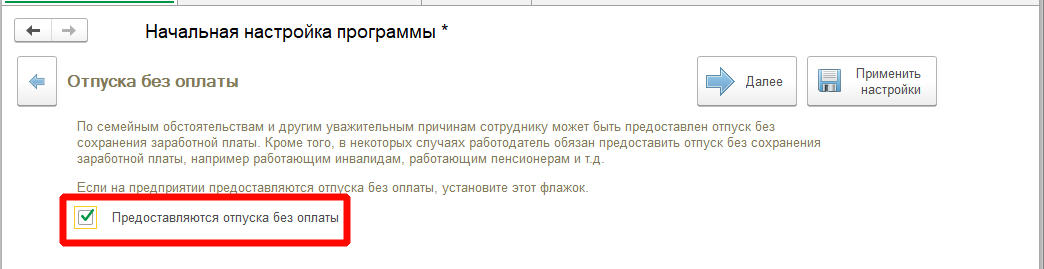
КонецПроцедуры

**III Ввод контрольного примера и проверка решения:**

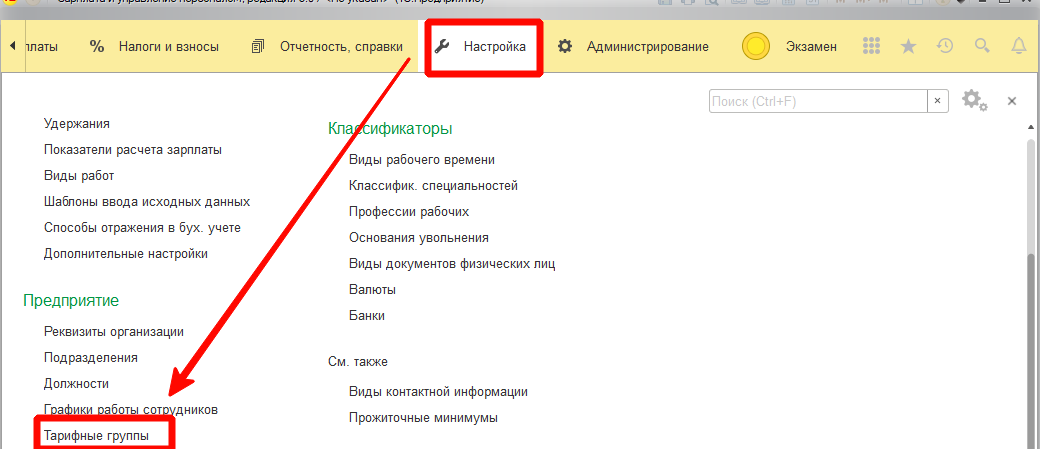
1. **Делаем начальную настройку программы, затем заполняем график на 2016 год:**

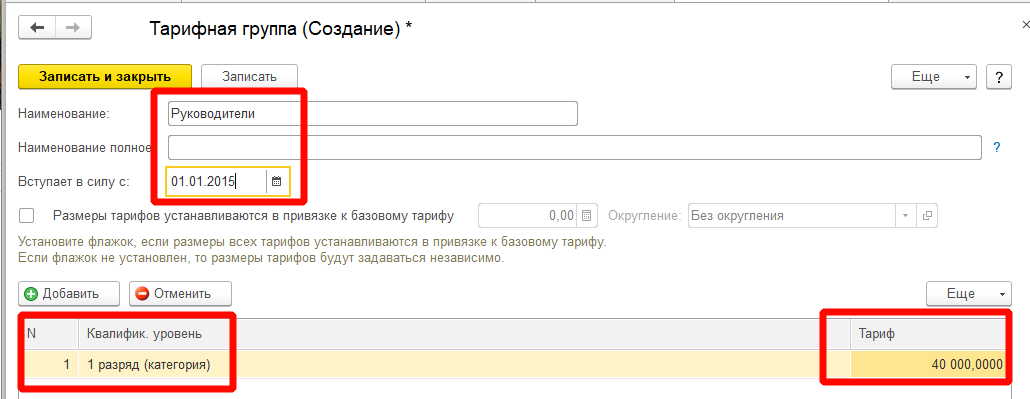


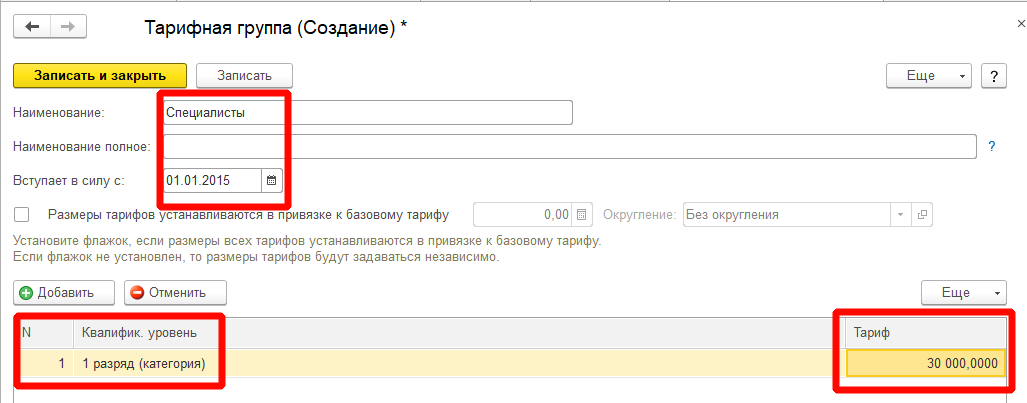


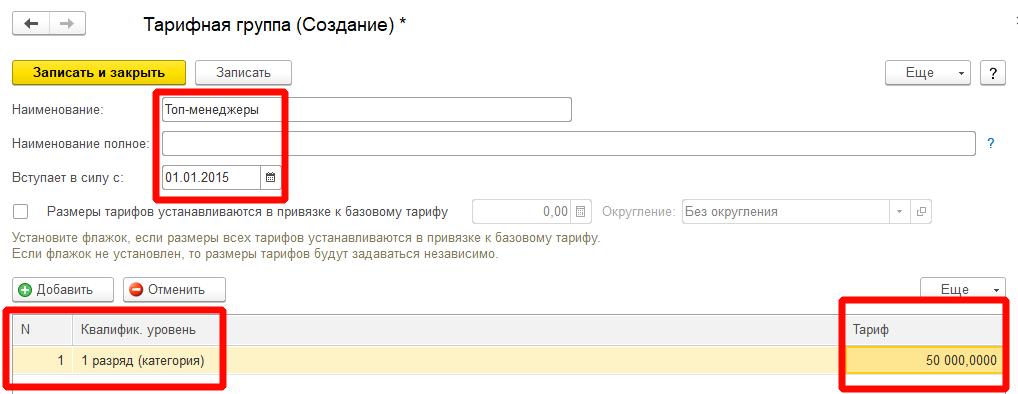


1. **Создаем тарифные группы (Руководители, Специалисты, Топ-менеджеры):**

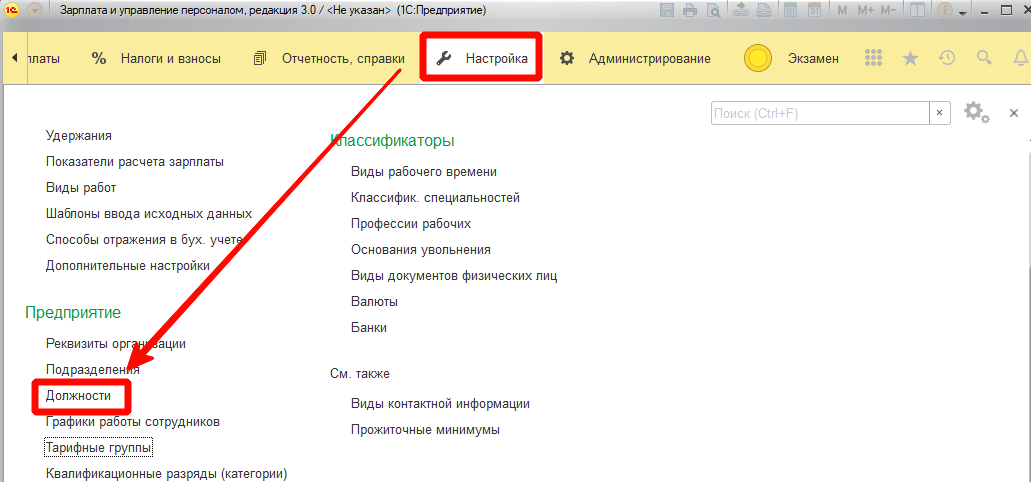


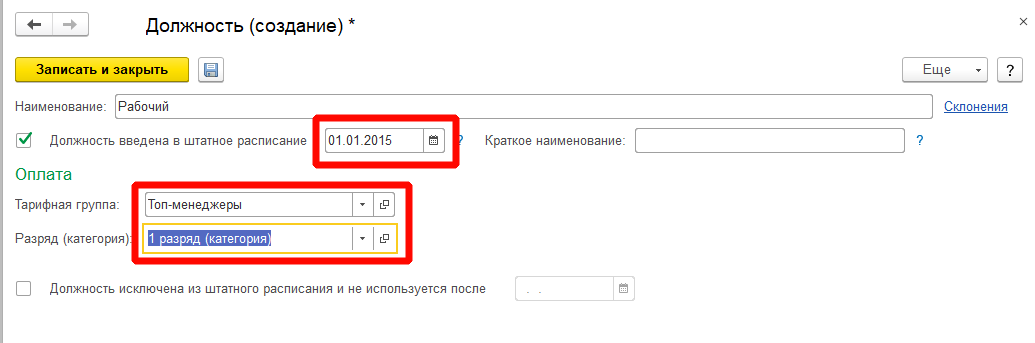


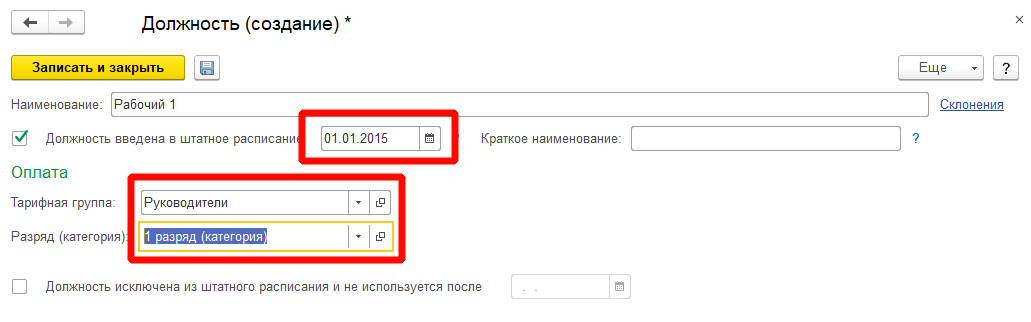




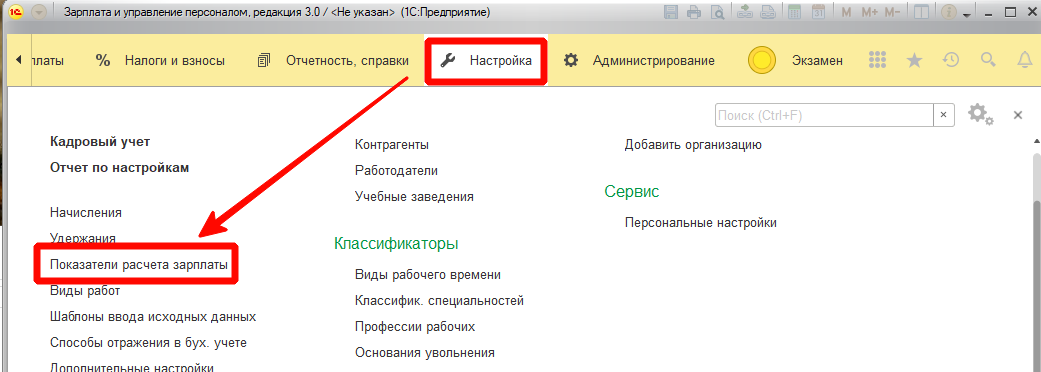
1. **Создаем должности:**

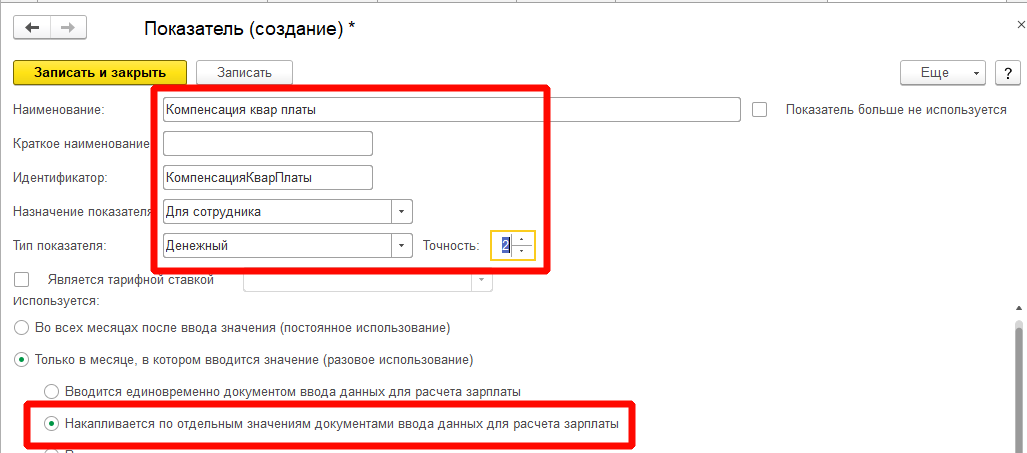




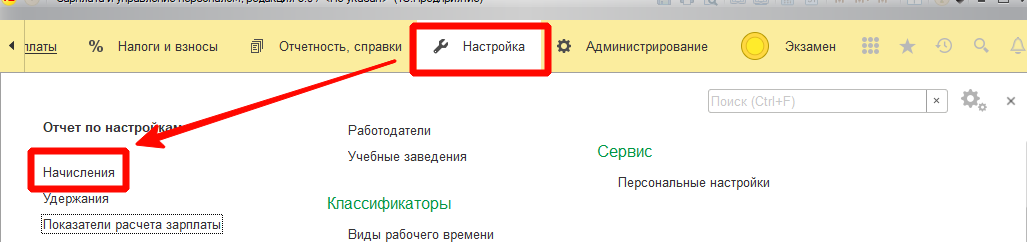


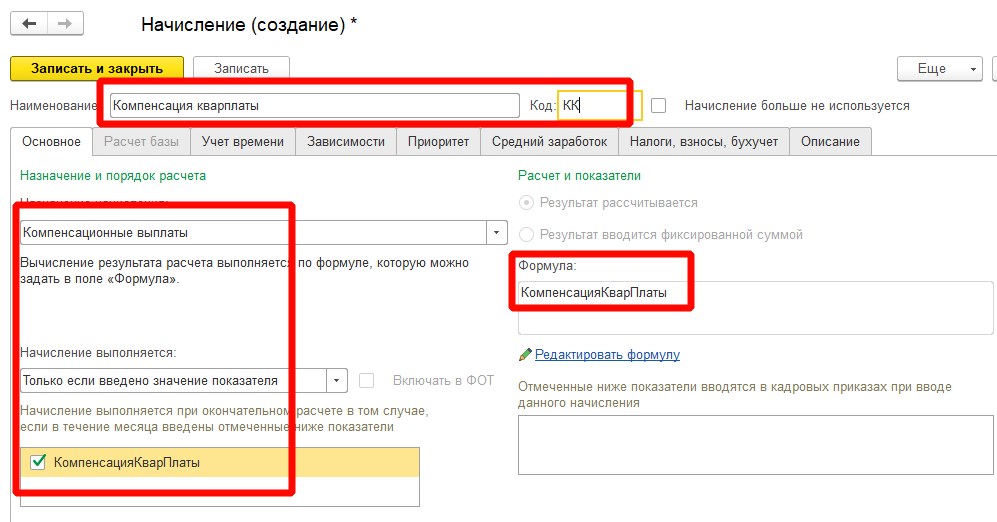
1. **Создаем показатели расчета зарплаты:**



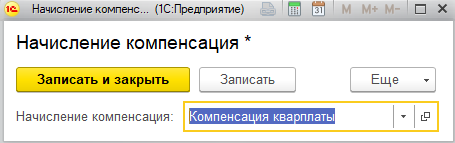


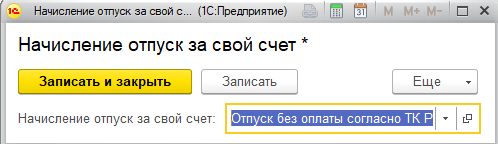
1. **Создаем начисление:**

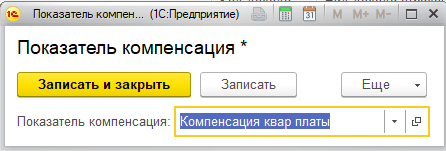


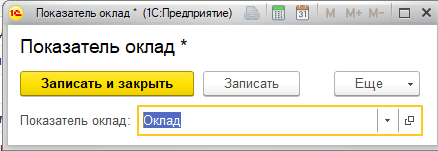


1. **Заполняем константы:**

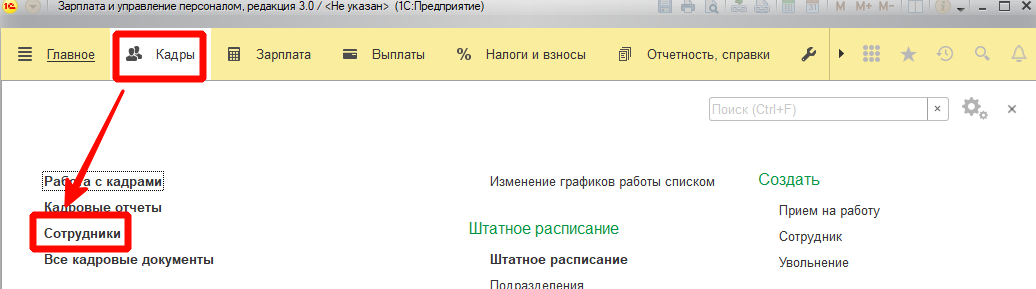


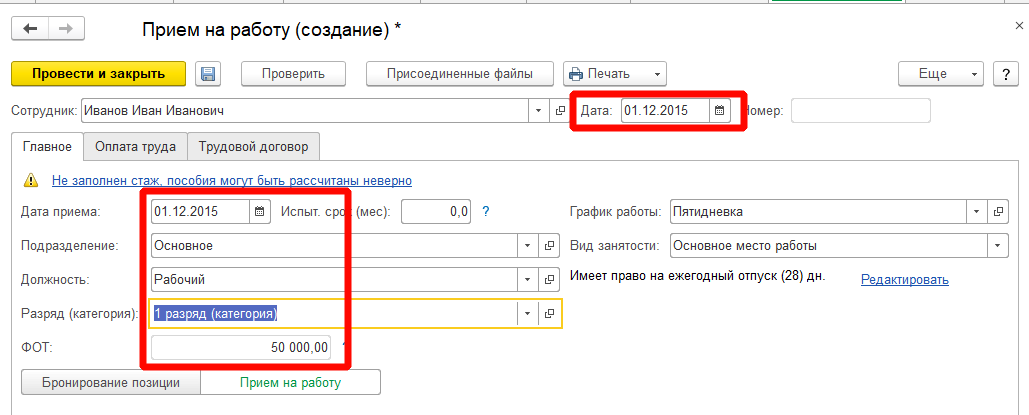


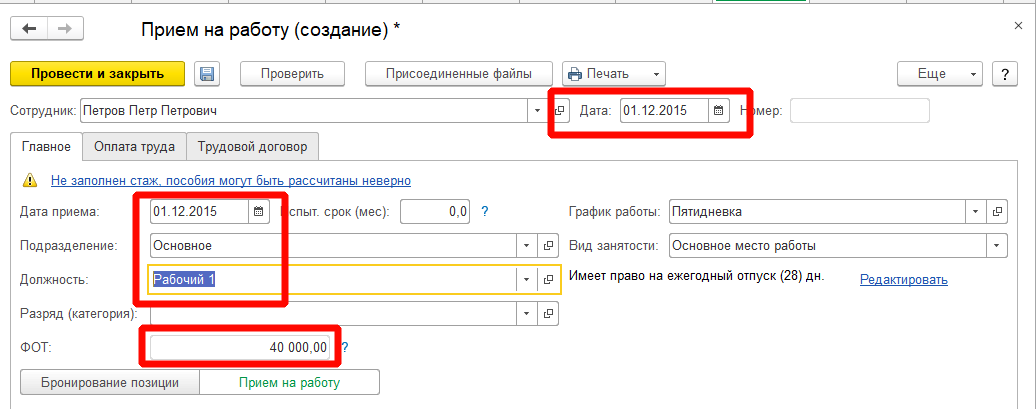




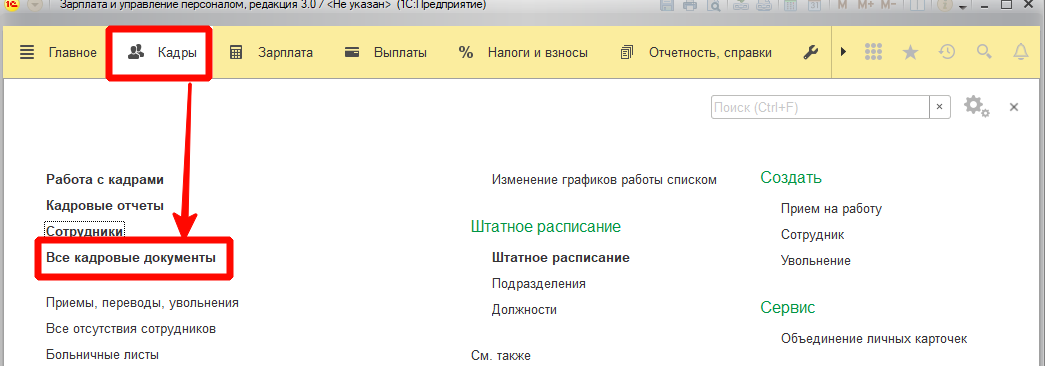
1. **Принимаем на работу Иванова и Петрова**

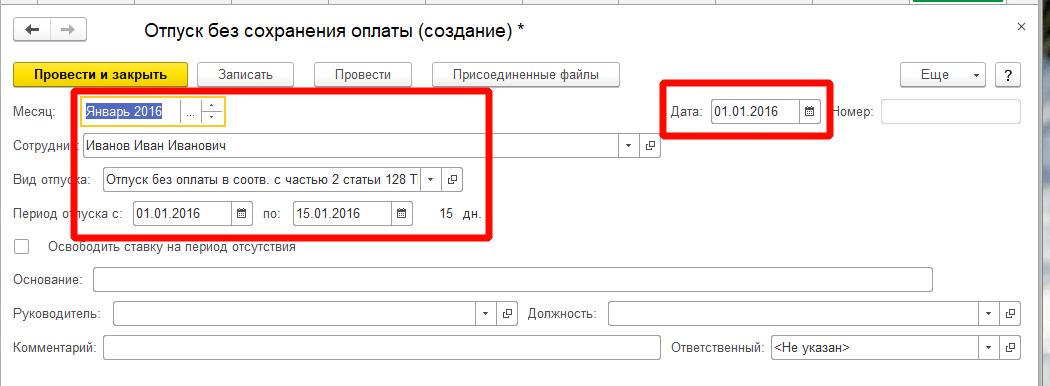




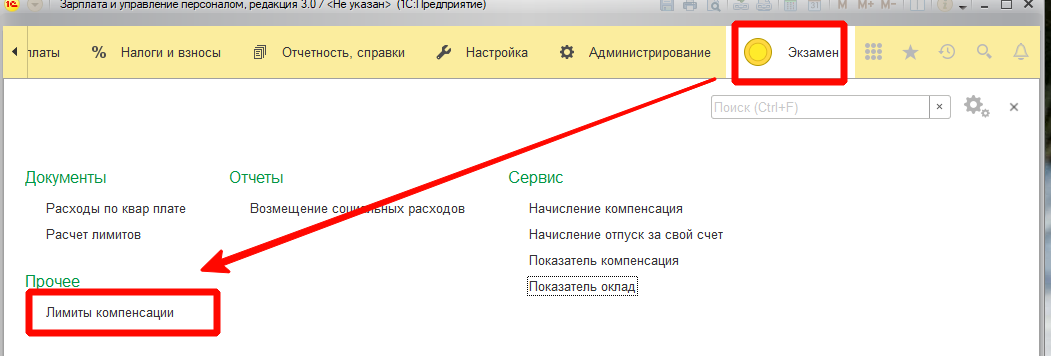


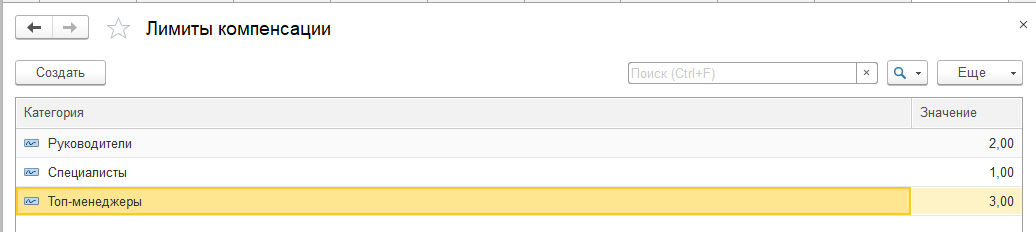
1. **Вводим отпуск без сохранения:**



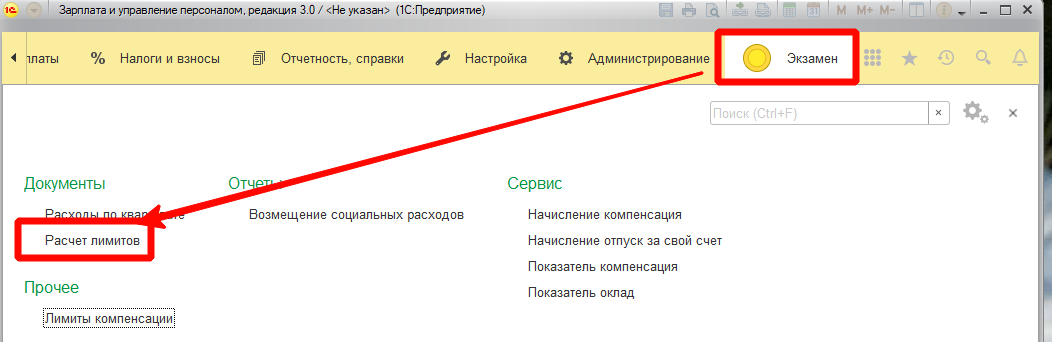


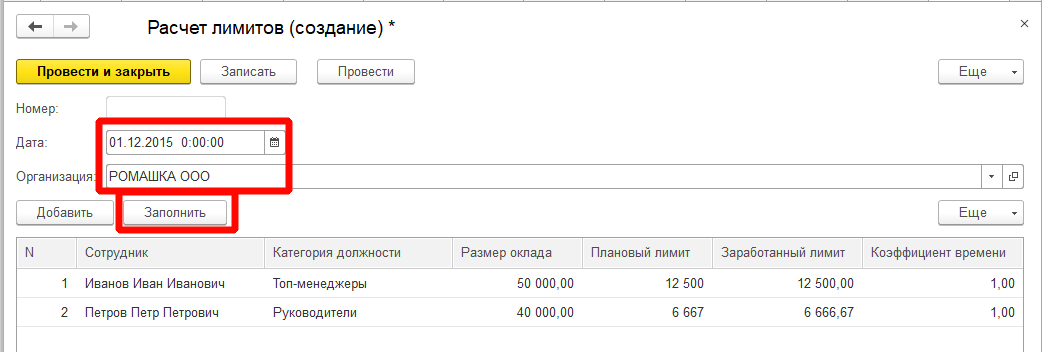
1. **Создаем лимиты компенсаций:**

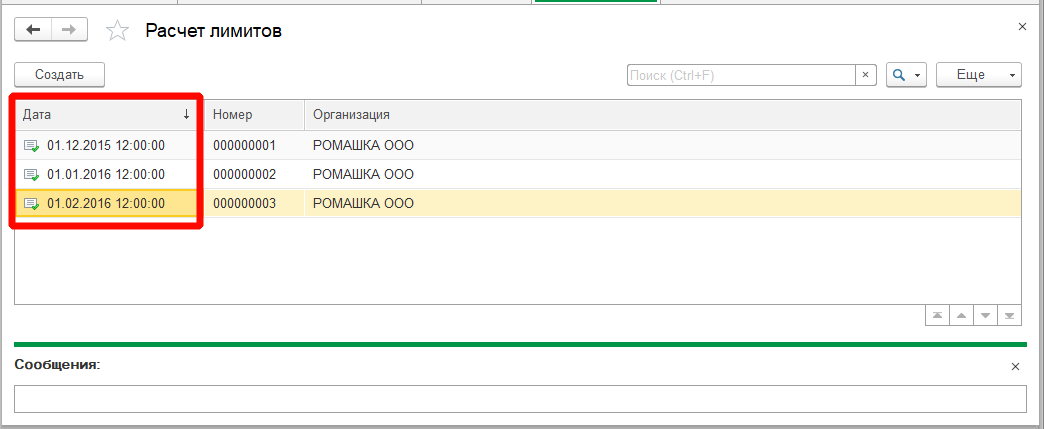




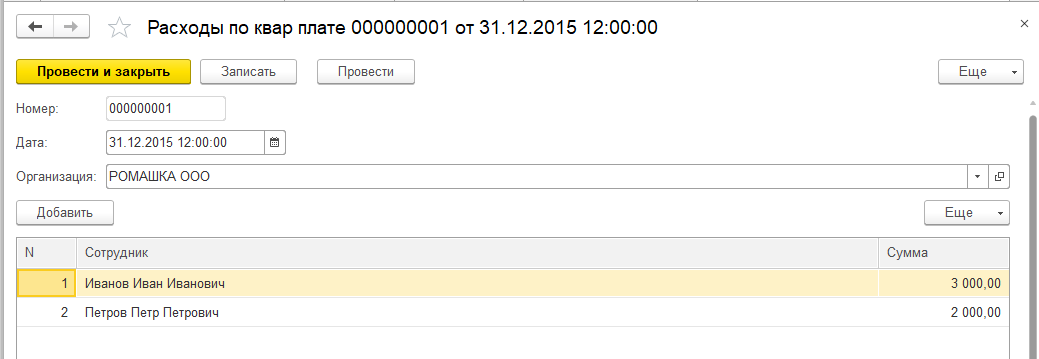
1. **Вводим три документа Расчет лимитов. Документы заполняются автоматически при нажатии «Заполнить»:**

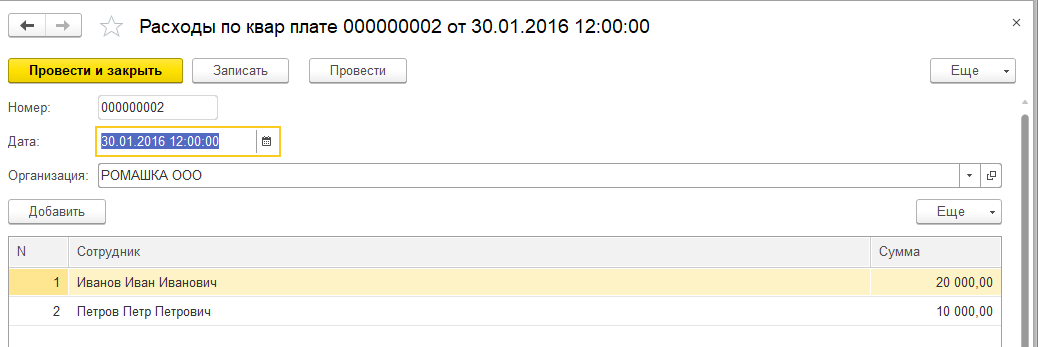


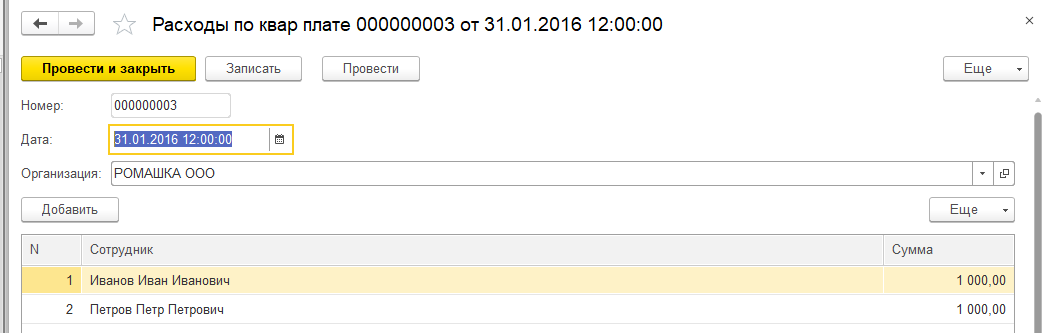


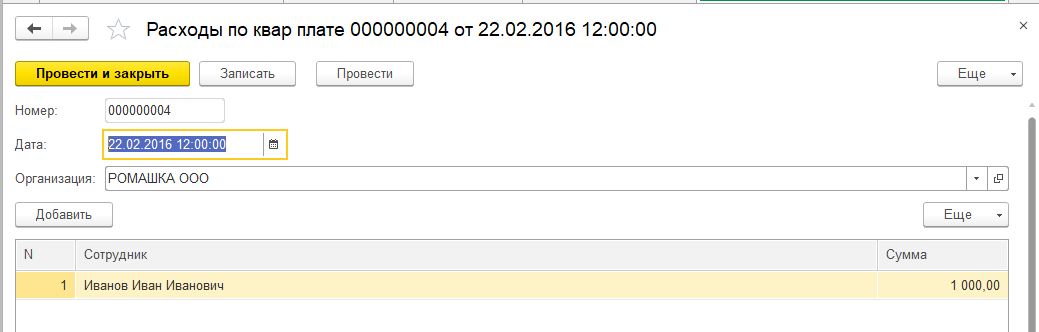


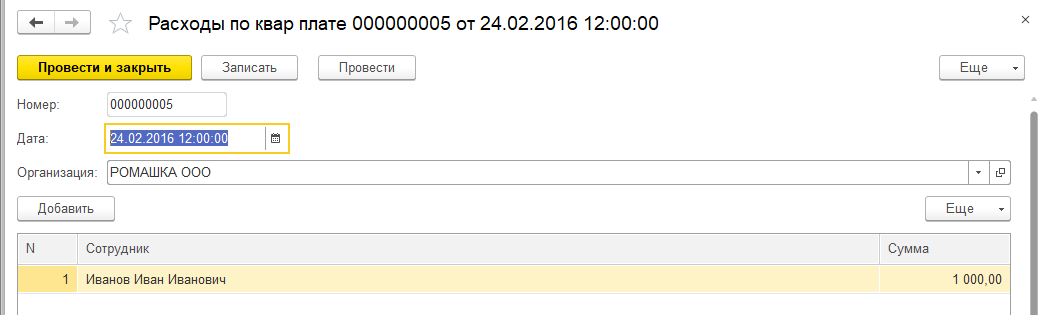
1. **Вводим пять документов РасходыПоКварПлате:**





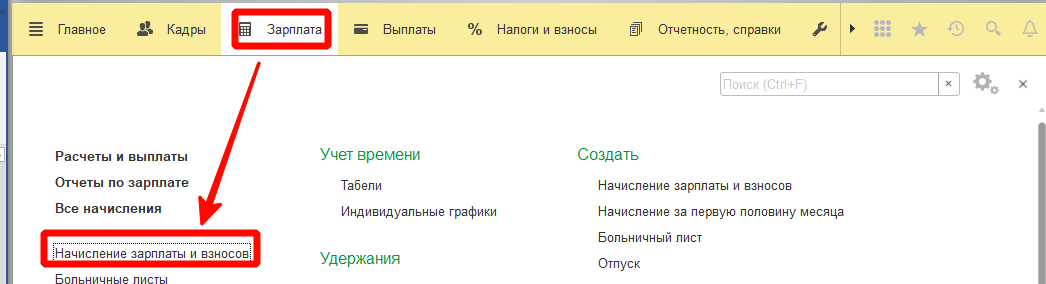


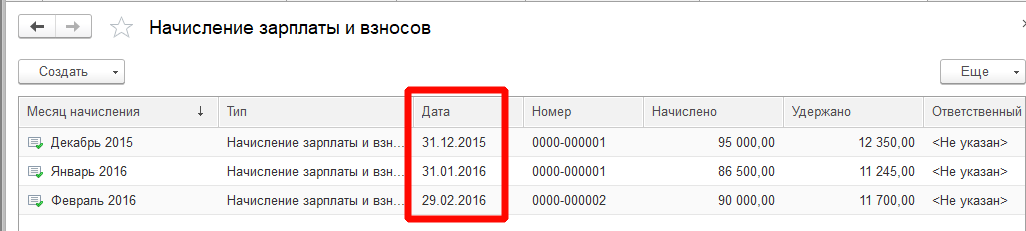






1. **Начисляем ЗП за декабрь 2015, январь, февраль 2016:**





1. **Формируем отчет ВозмещениеСоциальныхРасходов. Если все верно, то увидим примерно это:**



**IV ПОЯСНЕНИЕ К РЕШЕНИЮ**

Для реализации решения созданы 4 константы для хранения показателей расчета зарплаты и начислений.

Создан документ РасчетЛимитов в соответствии с заданием и документ РасходыПоКварПлате для учета и регистр КомпенсацияКварПлаты, регистр сведений ЛимитыКомпенсации для регистрации фактических расходов и учета начисления для целей налогообложения (движения в регистр ЗначенияОперативныхПоказателейРассчетаЗарплатыСотрудников) и обложения страховыми взносами.

Создан отчет ВозмещениеСоциальныхРасходов в соответствии с заданием. Отчет проверен на работоспособность.

Все объекты конфигурации, созданные в процессе решения задачи помещены в подсистему Экзамен.

В базе данных приведен контрольный пример начисления за 3 месяца по 2-м сотрудникам.

Таким образом все требуемые в задании механизмы реализованы.