**БИЛЕТ 6**

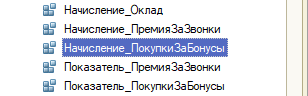
**РЕШЕНИЕ:**

**I КОНФИГУРАТОР.**

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

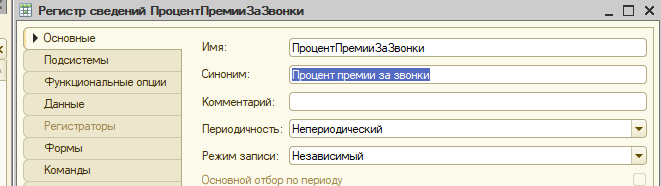


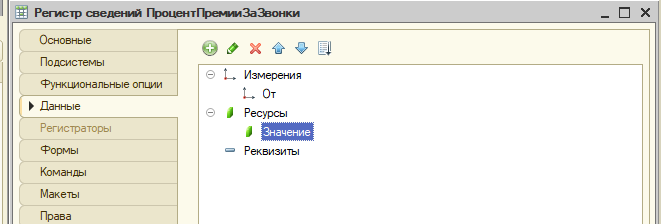
1. **Делаем 5 констант (подсистема Экзамен-Прочее):**



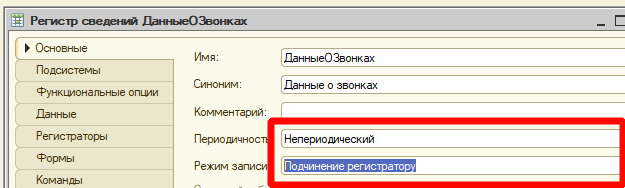
**Начисление\_Оклад**, **Начисление\_ПремияЗаЗвонки**, **Начисление\_ПокупкиЗаБонусы** - ПланВидовРасчетаСсылка.Начисления; **Показатель\_ПремияЗаЗвонки** и **Показатель\_ПокупкиЗаБонусы** - СправочникСсылка.ПоказателиРасчетаЗарплаты

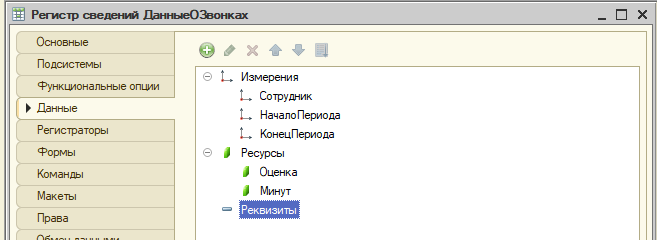
1. **Создаем три документа ДанныеОЗвонках, НачислениеБонусов и СписаниеБонусов (подсистема Экзамен-документы).**
2. **Создаем два регистра сведений: ПроцентПремииЗаЗвонки и ДанныеОЗвонках (экзамен - прочее):**



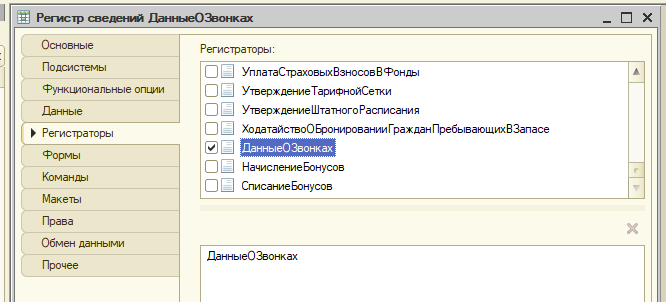


Измерения(1) – **От** (Число 10.2). Ресурсы(1) – **Значение** (Число 10.2)

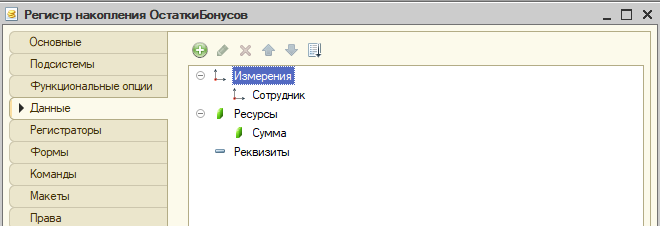




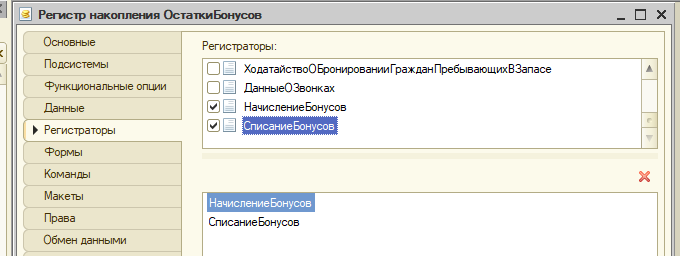
Измерения(3) – **Сотрудник** (СправочникСсылка.Сотрудники), **НачалоПериода** и **КонецПериода** (Дата, состав Дата и Время). Ресурсы(2) – **Оценка** и **Минут**(Число 10.2)



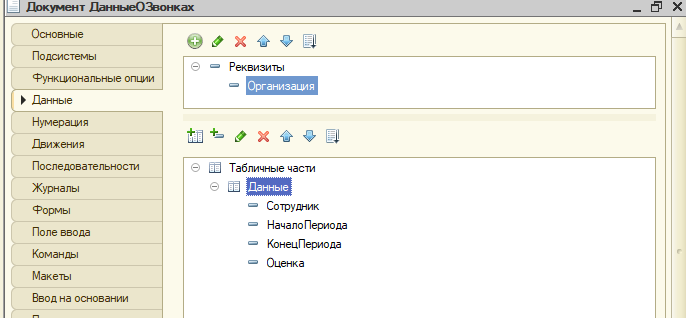
1. **Создаем регистр накопления (остатки) ОстаткиБонусов:**



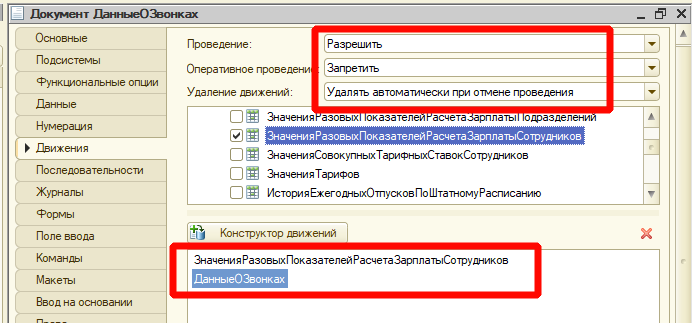
Измерения (1) – Сотрудник (СправочникСсылка.Сотрудники). Ресурсы(1) – Сумма (Число 10.2)



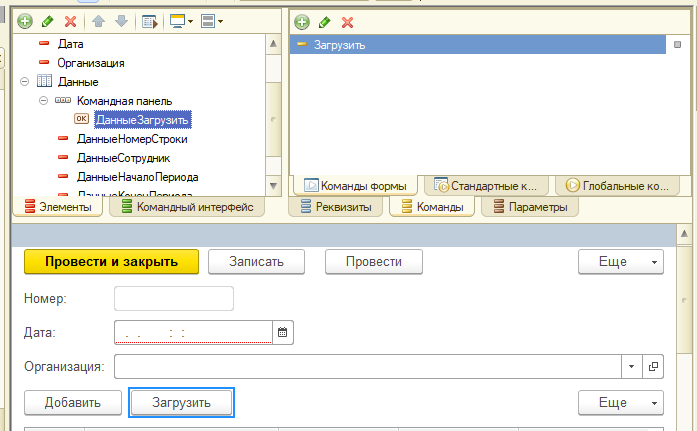
1. **Редактируем документ ДанныеОЗвонках:**



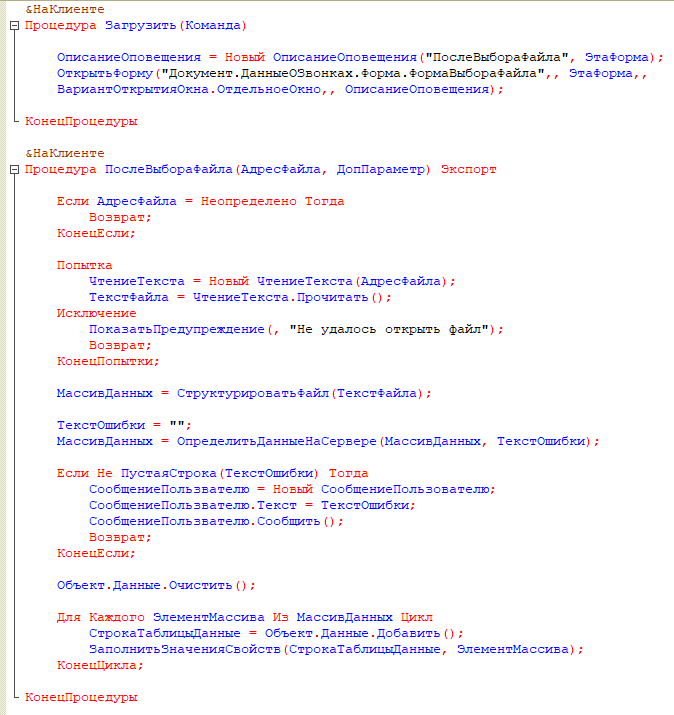
Реквизиты(1) – **Организация** (СправочникСсылка.Организации), табл. часть – **Данные** – **Сотрудник** (СправочникСсылка.Сотрудники), **НачалоПериода** и **КонецПериода** (Дата, состав Дата и время), **Оценка** (Число 10.2)

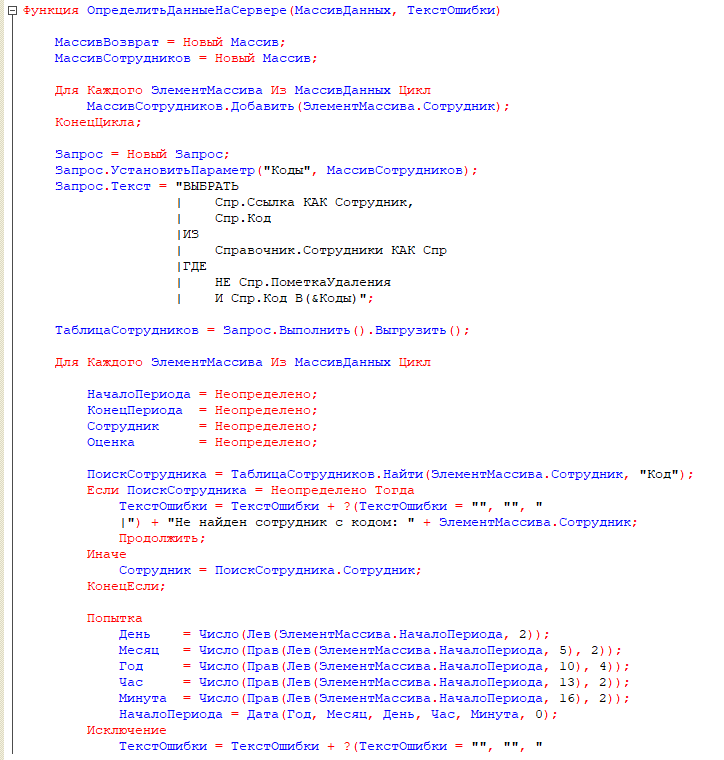


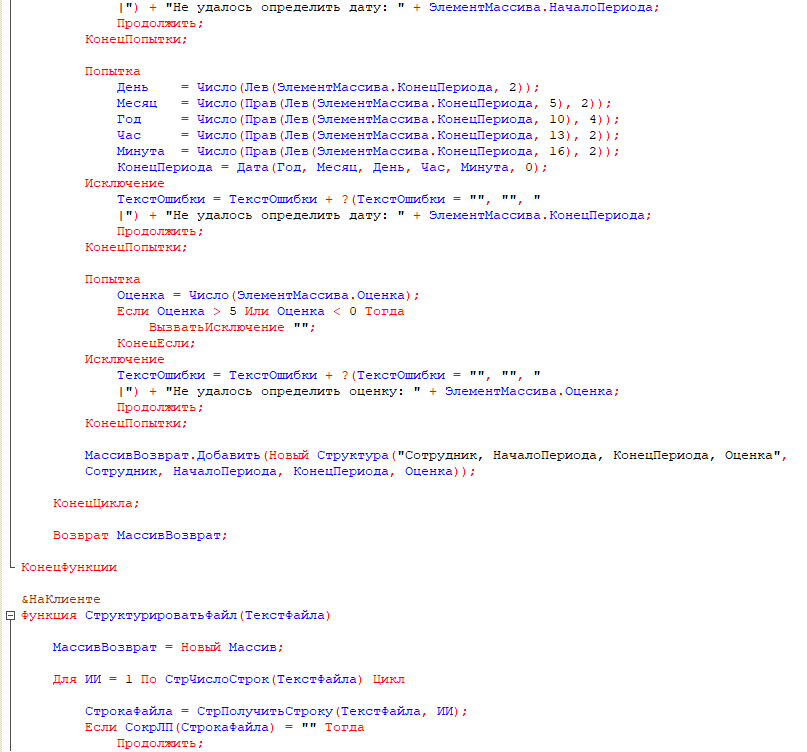
**Создаем форму, создаем на форме кнопку и действие на клиенте:**



**Модуль формы:**









&НаКлиенте

Процедура Загрузить(Команда)

ОписаниеОповещения = Новый ОписаниеОповещения("ПослеВыбораФайла", ЭтаФорма);

ОткрытьФорму("Документ.ДанныеОЗвонках.Форма.ФормаВыбораФайла",, ЭтаФорма,,

ВариантОткрытияОкна.ОтдельноеОкно,, ОписаниеОповещения);

КонецПроцедуры

&НаКлиенте

Процедура ПослеВыбораФайла(АдресФайла, ДопПараметр) Экспорт

Если АдресФайла = Неопределено Тогда

Возврат;

КонецЕсли;

Попытка

ЧтениеТекста = Новый ЧтениеТекста(АдресФайла);

ТекстФайла = ЧтениеТекста.Прочитать();

Исключение

ПоказатьПредупреждение(, "Не удалось открыть файл");

Возврат;

КонецПопытки;

МассивДанных = СтруктурироватьФайл(ТекстФайла);

ТекстОшибки = "";

МассивДанных = ОпределитьДанныеНаСервере(МассивДанных, ТекстОшибки);

Если Не ПустаяСтрока(ТекстОшибки) Тогда

СообщениеПользвателю = Новый СообщениеПользователю;

СообщениеПользвателю.Текст = ТекстОшибки;

СообщениеПользвателю.Сообщить();

Возврат;

КонецЕсли;

Объект.Данные.Очистить();

Для Каждого ЭлементМассива Из МассивДанных Цикл

СтрокаТаблицыДанные = Объект.Данные.Добавить();

ЗаполнитьЗначенияСвойств(СтрокаТаблицыДанные, ЭлементМассива);

КонецЦикла;

КонецПроцедуры

Функция ОпределитьДанныеНаСервере(МассивДанных, ТекстОшибки)

МассивВозврат = Новый Массив;

МассивСотрудников = Новый Массив;

Для Каждого ЭлементМассива Из МассивДанных Цикл

МассивСотрудников.Добавить(ЭлементМассива.Сотрудник);

КонецЦикла;

Запрос = Новый Запрос;

Запрос.УстановитьПараметр("Коды", МассивСотрудников);

Запрос.Текст = "ВЫБРАТЬ

| Спр.Ссылка КАК Сотрудник,

| Спр.Код

|ИЗ

| Справочник.Сотрудники КАК Спр

|ГДЕ

| НЕ Спр.ПометкаУдаления

| И Спр.Код В(&Коды)";

ТаблицаСотрудников = Запрос.Выполнить().Выгрузить();

Для Каждого ЭлементМассива Из МассивДанных Цикл

НачалоПериода = Неопределено;

КонецПериода = Неопределено;

Сотрудник = Неопределено;

Оценка = Неопределено;

ПоискСотрудника = ТаблицаСотрудников.Найти(ЭлементМассива.Сотрудник, "Код");

Если ПоискСотрудника = Неопределено Тогда

ТекстОшибки = ТекстОшибки + ?(ТекстОшибки = "", "", "

|") + "Не найден сотрудник с кодом: " + ЭлементМассива.Сотрудник;

Продолжить;

Иначе

Сотрудник = ПоискСотрудника.Сотрудник;

КонецЕсли;

Попытка

День = Число(Лев(ЭлементМассива.НачалоПериода, 2));

Месяц = Число(Прав(Лев(ЭлементМассива.НачалоПериода, 5), 2));

Год = Число(Прав(Лев(ЭлементМассива.НачалоПериода, 10), 4));

Час = Число(Прав(Лев(ЭлементМассива.НачалоПериода, 13), 2));

Минута = Число(Прав(Лев(ЭлементМассива.НачалоПериода, 16), 2));

НачалоПериода = Дата(Год, Месяц, День, Час, Минута, 0);

Исключение

ТекстОшибки = ТекстОшибки + ?(ТекстОшибки = "", "", "

|") + "Не удалось определить дату: " + ЭлементМассива.НачалоПериода;

Продолжить;

КонецПопытки;

Попытка

День = Число(Лев(ЭлементМассива.КонецПериода, 2));

Месяц = Число(Прав(Лев(ЭлементМассива.КонецПериода, 5), 2));

Год = Число(Прав(Лев(ЭлементМассива.КонецПериода, 10), 4));

Час = Число(Прав(Лев(ЭлементМассива.КонецПериода, 13), 2));

Минута = Число(Прав(Лев(ЭлементМассива.КонецПериода, 16), 2));

КонецПериода = Дата(Год, Месяц, День, Час, Минута, 0);

Исключение

ТекстОшибки = ТекстОшибки + ?(ТекстОшибки = "", "", "

|") + "Не удалось определить дату: " + ЭлементМассива.КонецПериода;

Продолжить;

КонецПопытки;

Попытка

Оценка = Число(ЭлементМассива.Оценка);

Если Оценка > 5 Или Оценка < 0 Тогда

ВызватьИсключение "";

КонецЕсли;

Исключение

ТекстОшибки = ТекстОшибки + ?(ТекстОшибки = "", "", "

|") + "Не удалось определить оценку: " + ЭлементМассива.Оценка;

Продолжить;

КонецПопытки;

МассивВозврат.Добавить(Новый Структура("Сотрудник, НачалоПериода, КонецПериода, Оценка",

Сотрудник, НачалоПериода, КонецПериода, Оценка));

КонецЦикла;

Возврат МассивВозврат;

КонецФункции

&НаКлиенте

Функция СтруктурироватьФайл(ТекстФайла)

МассивВозврат = Новый Массив;

Для ИИ = 1 По СтрЧислоСтрок(ТекстФайла) Цикл

СтрокаФайла = СтрПолучитьСтроку(ТекстФайла, ИИ);

Если СокрЛП(СтрокаФайла) = "" Тогда

Продолжить;

КонецЕсли;

НачалоПериода = "";

КонецПериода = "";

Сотрудник = "";

Оценка = "";

Разделитель = СтрНайти(СтрокаФайла, ";");

Если Разделитель > 0 Тогда

НачалоПериода = СокрЛП(Лев(СтрокаФайла, Разделитель - 1));

СтрокаФайла = Прав(СтрокаФайла, СтрДлина(СтрокаФайла) - Разделитель);

Иначе

НачалоПериода = СокрЛП(СтрокаФайла);

МассивВозврат.Добавить(Новый Структура("НачалоПериода, КонецПериода, Сотрудник, Оценка",

НачалоПериода, КонецПериода, Сотрудник, Оценка));

Продолжить;

КонецЕсли;

Разделитель = СтрНайти(СтрокаФайла, ";");

Если Разделитель > 0 Тогда

КонецПериода = СокрЛП(Лев(СтрокаФайла, Разделитель - 1));

СтрокаФайла = Прав(СтрокаФайла, СтрДлина(СтрокаФайла) - Разделитель);

Иначе

КонецПериода = СокрЛП(СтрокаФайла);

МассивВозврат.Добавить(Новый Структура("НачалоПериода, КонецПериода, Сотрудник, Оценка",

НачалоПериода, КонецПериода, Сотрудник, Оценка));

Продолжить;

КонецЕсли;

Разделитель = СтрНайти(СтрокаФайла, ";");

Если Разделитель > 0 Тогда

Сотрудник = СокрЛП(Лев(СтрокаФайла, Разделитель - 1));

СтрокаФайла = Прав(СтрокаФайла, СтрДлина(СтрокаФайла) - Разделитель);

Иначе

Сотрудник = СокрЛП(СтрокаФайла);

МассивВозврат.Добавить(Новый Структура("НачалоПериода, КонецПериода, Сотрудник, Оценка",

НачалоПериода, КонецПериода, Сотрудник, Оценка));

Продолжить;

КонецЕсли;

Оценка = СокрЛП(СтрокаФайла);

МассивВозврат.Добавить(Новый Структура("НачалоПериода, КонецПериода, Сотрудник, Оценка",

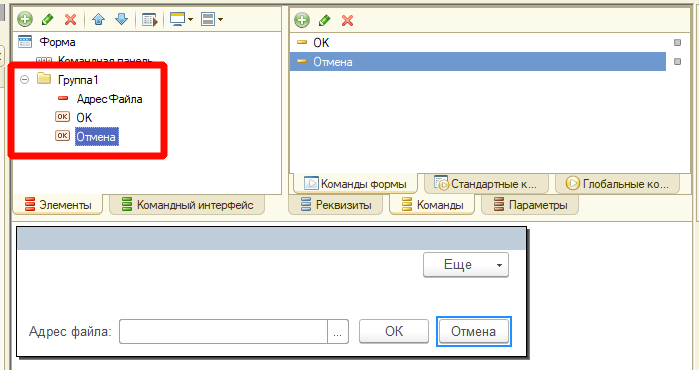
НачалоПериода, КонецПериода, Сотрудник, Оценка));

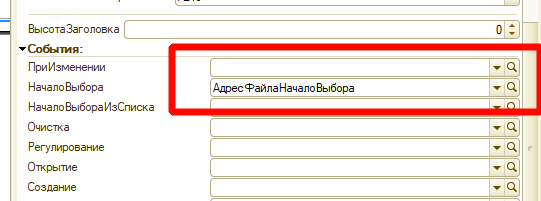
КонецЦикла;

Возврат МассивВозврат;

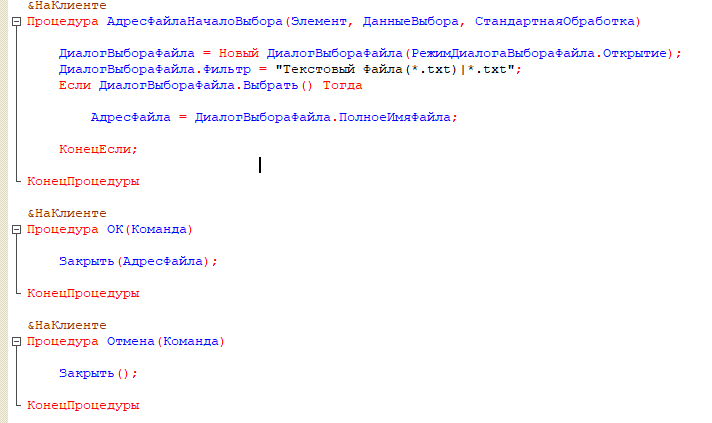
КонецФункции

1. **Создаем форму выбора файла. Кнопки ОК и Отмена (на клиенте):**





**Модуль формы выбора:**



&НаКлиенте

Процедура АдресФайлаНачалоВыбора(Элемент, ДанныеВыбора, СтандартнаяОбработка)

ДиалогВыбораФайла = Новый ДиалогВыбораФайла(РежимДиалогаВыбораФайла.Открытие);

ДиалогВыбораФайла.Фильтр = "Текстовый Файла(\*.txt)|\*.txt";

Если ДиалогВыбораФайла.Выбрать() Тогда

АдресФайла = ДиалогВыбораФайла.ПолноеИмяФайла;

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура ОК(Команда)

Закрыть(АдресФайла);

КонецПроцедуры

&НаКлиенте

Процедура Отмена(Команда)

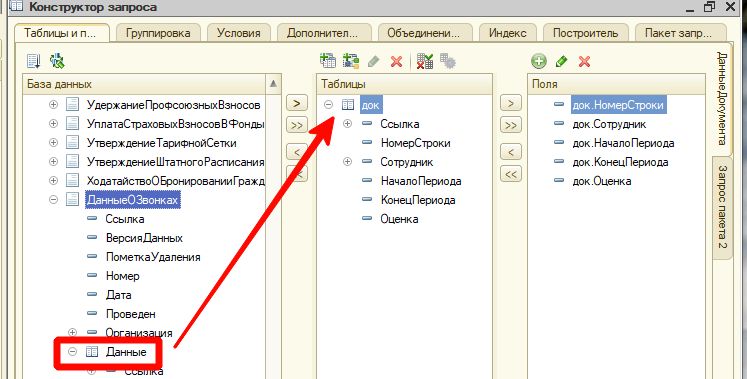
Закрыть();

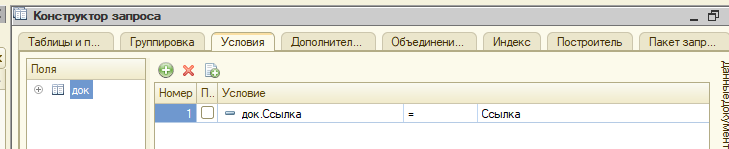
КонецПроцедуры

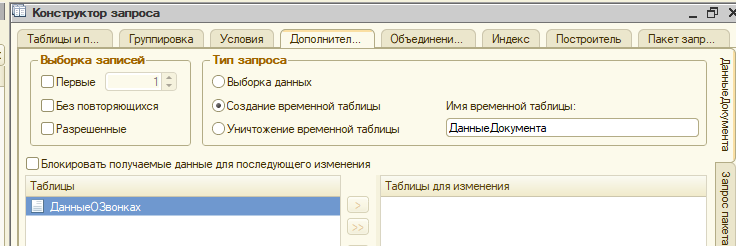
1. **Модуль документа ДанныеОЗвонках:**

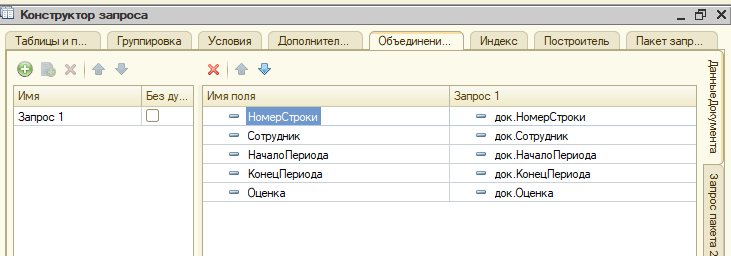
**ЗАПРОС:**

**ПЕРВЫЙ ПАКЕТ:**





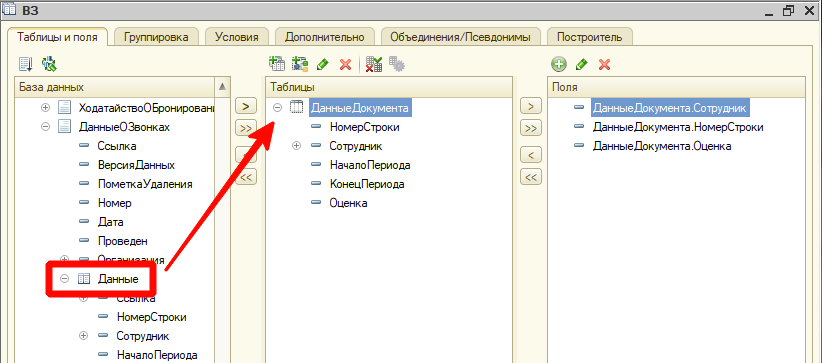


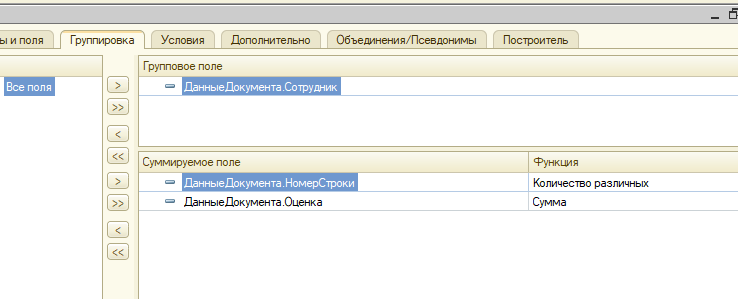


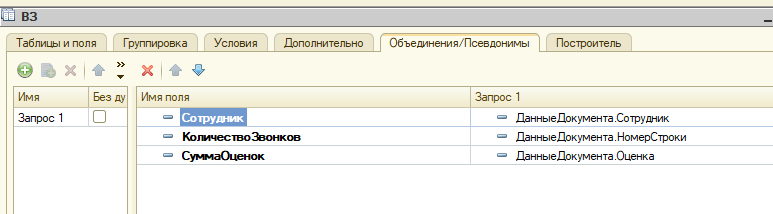
**ВТОРОЙ ПАКЕТ:**

**НАЧАЛО ВЛОЖЕННОГО ЗАПРОСА ВЗ**

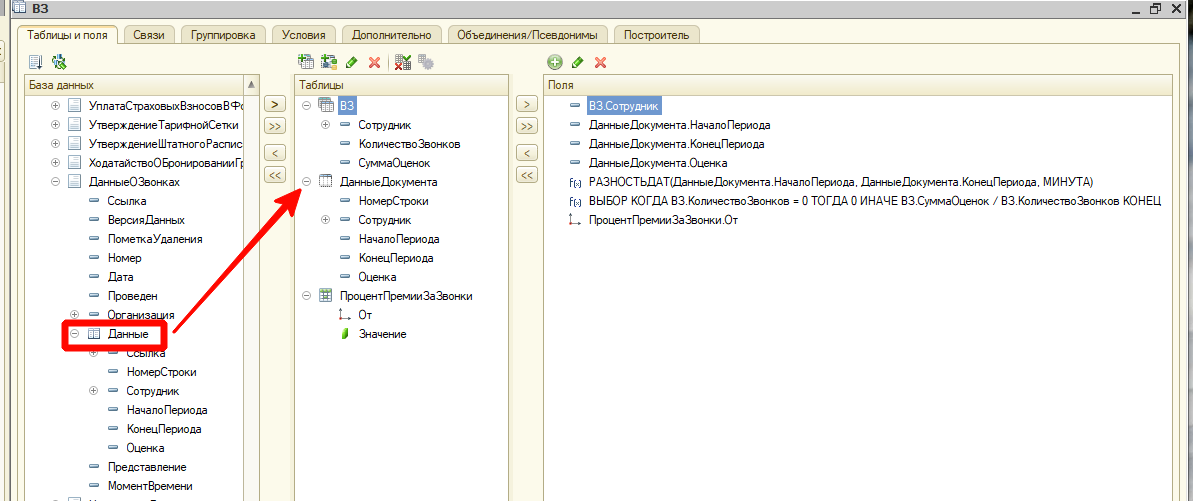
**НАЧАЛО ВЛОЖЕННОГО ЗАПРОСА ВЗ ВО ВЛОЖЕННЫЙ ЗАПРОС ВЗ**

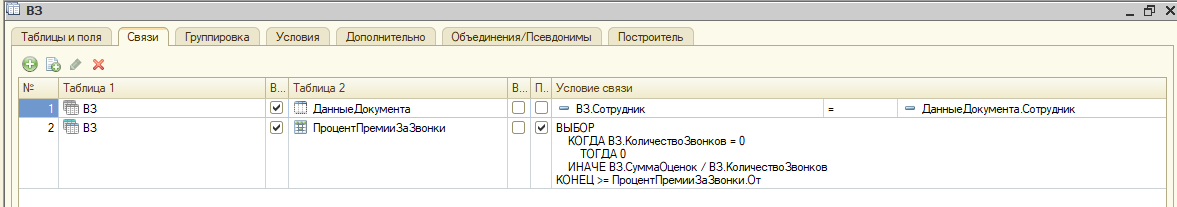


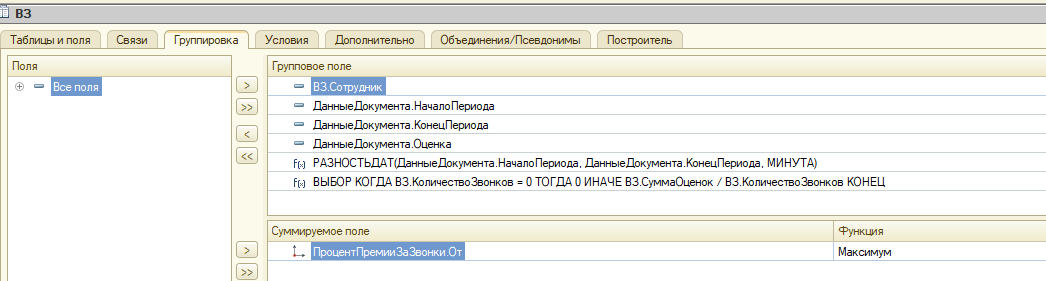


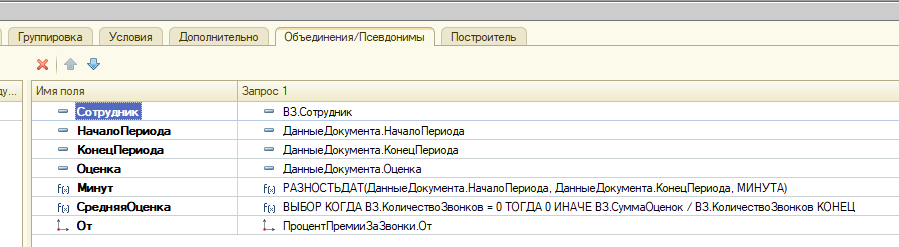


**ОКОНЧАНИЕ ВЛОЖЕННОГО ЗАПРОСА ВЗ ВО ВЛОЖЕННЫЙ ЗАПРОС ВЗ**

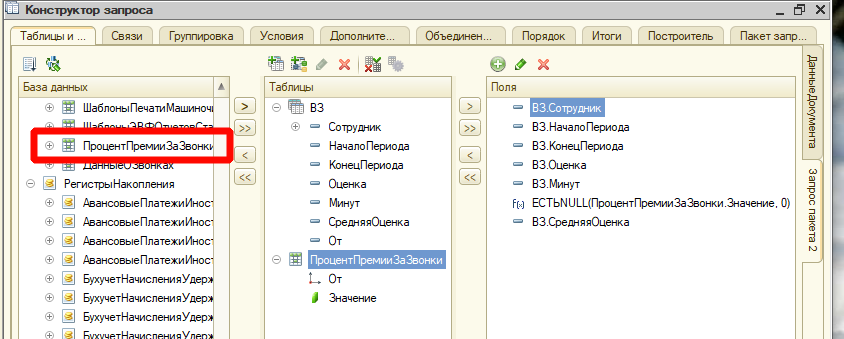


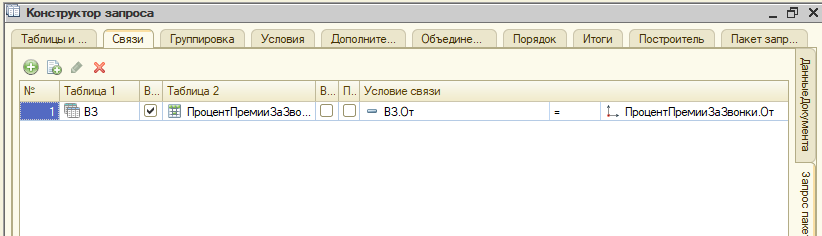


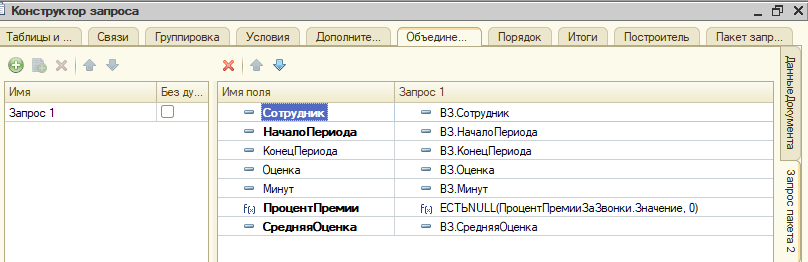


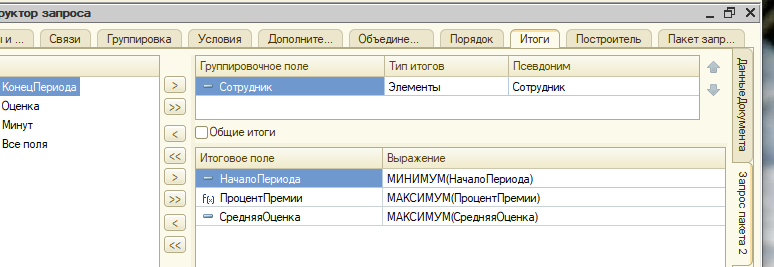


**ОКОНЧАНИЕ ВЛОЖЕННОГО ЗАПРОСА ВЗ**

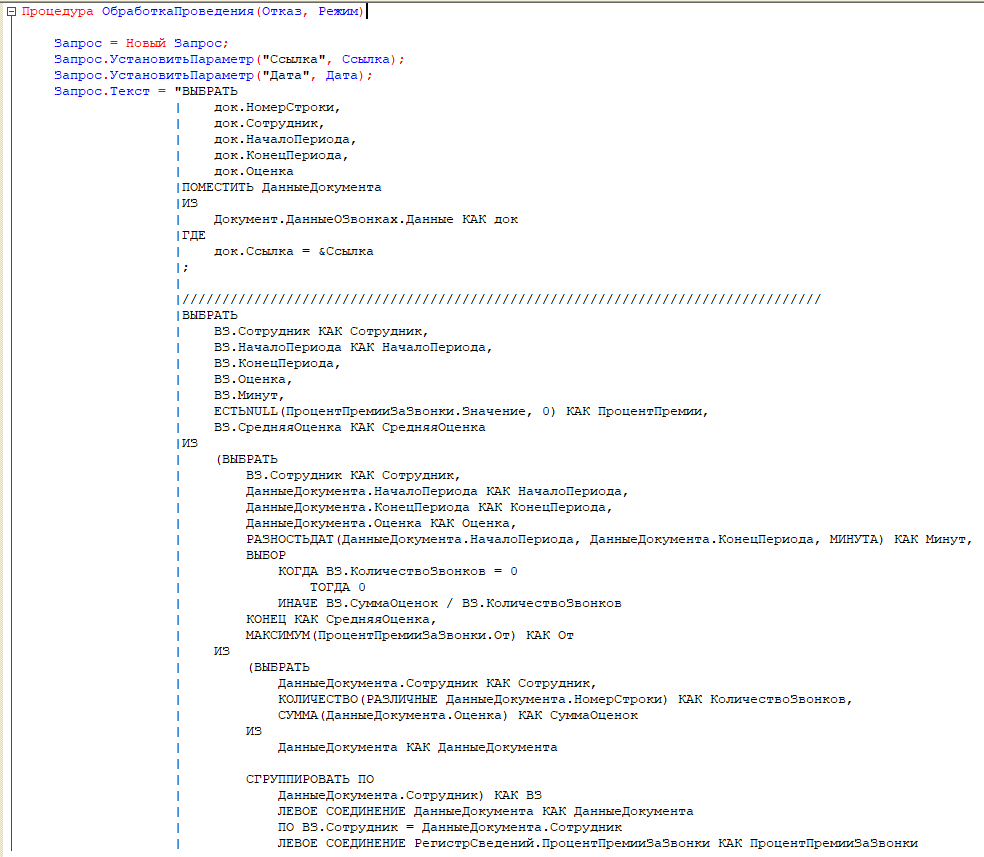








**ПОЛНОСТЬЮ МОДУЛЬ:**





Процедура ОбработкаПроведения(Отказ, Режим)

Запрос = Новый Запрос;

Запрос.УстановитьПараметр("Ссылка", Ссылка);

Запрос.УстановитьПараметр("Дата", Дата);

Запрос.Текст = "ВЫБРАТЬ

| док.НомерСтроки,

| док.Сотрудник,

| док.НачалоПериода,

| док.КонецПериода,

| док.Оценка

|ПОМЕСТИТЬ ДанныеДокумента

|ИЗ

| Документ.ДанныеОЗвонках.Данные КАК док

|ГДЕ

| док.Ссылка = &Ссылка

|;

|

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

|ВЫБРАТЬ

| ВЗ.Сотрудник КАК Сотрудник,

| ВЗ.НачалоПериода КАК НачалоПериода,

| ВЗ.КонецПериода,

| ВЗ.Оценка,

| ВЗ.Минут,

| ЕСТЬNULL(ПроцентПремииЗаЗвонки.Значение, 0) КАК ПроцентПремии,

| ВЗ.СредняяОценка КАК СредняяОценка

|ИЗ

| (ВЫБРАТЬ

| ВЗ.Сотрудник КАК Сотрудник,

| ДанныеДокумента.НачалоПериода КАК НачалоПериода,

| ДанныеДокумента.КонецПериода КАК КонецПериода,

| ДанныеДокумента.Оценка КАК Оценка,

| РАЗНОСТЬДАТ(ДанныеДокумента.НачалоПериода, ДанныеДокумента.КонецПериода, МИНУТА) КАК Минут,

| ВЫБОР

| КОГДА ВЗ.КоличествоЗвонков = 0

| ТОГДА 0

| ИНАЧЕ ВЗ.СуммаОценок / ВЗ.КоличествоЗвонков

| КОНЕЦ КАК СредняяОценка,

| МАКСИМУМ(ПроцентПремииЗаЗвонки.От) КАК От

| ИЗ

| (ВЫБРАТЬ

| ДанныеДокумента.Сотрудник КАК Сотрудник,

| КОЛИЧЕСТВО(РАЗЛИЧНЫЕ ДанныеДокумента.НомерСтроки) КАК КоличествоЗвонков,

| СУММА(ДанныеДокумента.Оценка) КАК СуммаОценок

| ИЗ

| ДанныеДокумента КАК ДанныеДокумента

|

| СГРУППИРОВАТЬ ПО

| ДанныеДокумента.Сотрудник) КАК ВЗ

| ЛЕВОЕ СОЕДИНЕНИЕ ДанныеДокумента КАК ДанныеДокумента

| ПО ВЗ.Сотрудник = ДанныеДокумента.Сотрудник

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрСведений.ПроцентПремииЗаЗвонки КАК ПроцентПремииЗаЗвонки

| ПО (ВЫБОР

| КОГДА ВЗ.КоличествоЗвонков = 0

| ТОГДА 0

| ИНАЧЕ ВЗ.СуммаОценок / ВЗ.КоличествоЗвонков

| КОНЕЦ >= ПроцентПремииЗаЗвонки.От)

|

| СГРУППИРОВАТЬ ПО

| ВЗ.Сотрудник,

| ДанныеДокумента.НачалоПериода,

| ДанныеДокумента.КонецПериода,

| ДанныеДокумента.Оценка,

| РАЗНОСТЬДАТ(ДанныеДокумента.НачалоПериода, ДанныеДокумента.КонецПериода, МИНУТА),

| ВЫБОР

| КОГДА ВЗ.КоличествоЗвонков = 0

| ТОГДА 0

| ИНАЧЕ ВЗ.СуммаОценок / ВЗ.КоличествоЗвонков

| КОНЕЦ) КАК ВЗ

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрСведений.ПроцентПремииЗаЗвонки КАК ПроцентПремииЗаЗвонки

| ПО ВЗ.От = ПроцентПремииЗаЗвонки.От

|ИТОГИ

| МИНИМУМ(НачалоПериода),

| МАКСИМУМ(ПроцентПремии),

| МАКСИМУМ(СредняяОценка)

|ПО

| Сотрудник

|";

ВыборкаСотрудник = Запрос.Выполнить().Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам);

Пока ВыборкаСотрудник.Следующий() Цикл

Если ВыборкаСотрудник.ПроцентПремии > 0 Тогда

Движение = Движения.ЗначенияРазовыхПоказателейРасчетаЗарплатыСотрудников.Добавить();

Движение.Период = НачалоМесяца(Дата);

Движение.ПериодДействия = НачалоМесяца(Дата);

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

Движение.Сотрудник = ВыборкаСотрудник.Сотрудник;

Движение.Организация = Организация;

Движение.Значение = ВыборкаСотрудник.ПроцентПремии;

//это для 3.1.5.99 , 3.0.25.85 измерения ФизическоеЛицо в регистре нет

//Движение.ФизическоеЛицо = ВыборкаСотрудник.Сотрудник.ФизическоеЛицо;

КонецЕсли;

Выборка = ВыборкаСотрудник.Выбрать();

Пока Выборка.Следующий() Цикл

Движение = Движения.ДанныеОЗвонках.Добавить();

ЗаполнитьЗначенияСвойств(Движение, Выборка);

КонецЦикла;

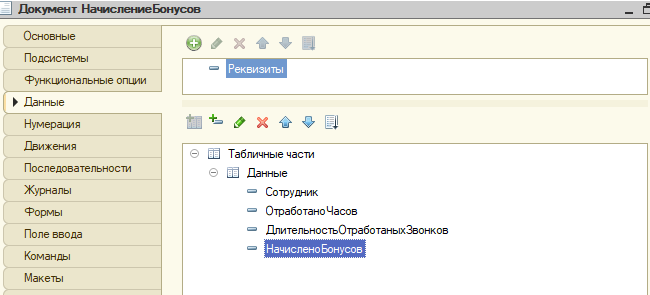
КонецЦикла;

Движения.ДанныеОЗвонках.Записывать = Истина;

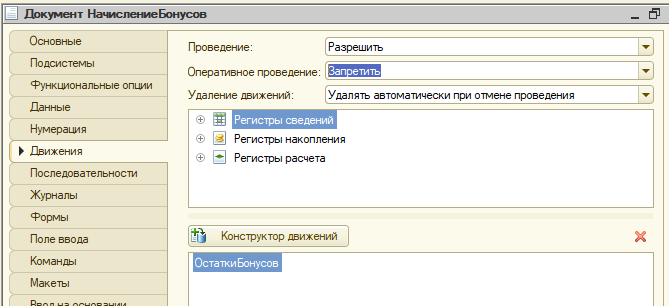
Движения.ЗначенияРазовыхПоказателейРасчетаЗарплатыСотрудников.Записывать = Истина;

КонецПроцедуры

1. **Редактируем документ НачислениеБонусов:**

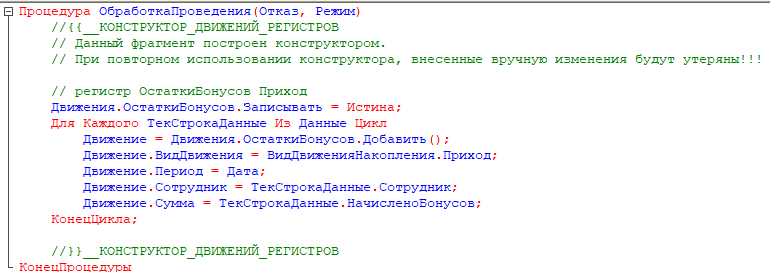


шапка – **отсутствует**, табл. часть – **Данные** (4) - **Сотрудник** (СправочникСсылка.Сотрудники), **ОтработаноЧасов**, **ДлительностьОтработаныхЗвонков** и **НачисленоБонусов** – Число 10.2



**Форму можно не создавать.**

1. **Модуль объекта НачислениеБонусов (конструктором):**



Процедура ОбработкаПроведения(Отказ, Режим)

//{{\_\_КОНСТРУКТОР\_ДВИЖЕНИЙ\_РЕГИСТРОВ

// Данный фрагмент построен конструктором.

// При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!!

// регистр ОстаткиБонусов Приход

Движения.ОстаткиБонусов.Записывать = Истина;

Для Каждого ТекСтрокаДанные Из Данные Цикл

Движение = Движения.ОстаткиБонусов.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

Движение.Период = Дата;

Движение.Сотрудник = ТекСтрокаДанные.Сотрудник;

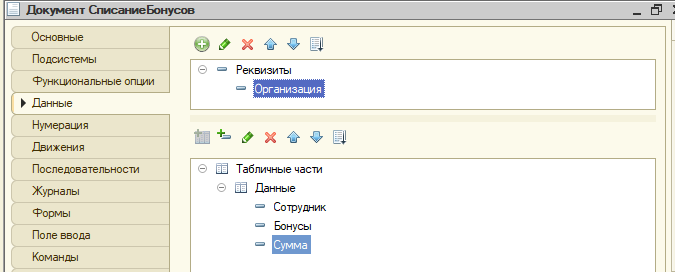
Движение.Сумма = ТекСтрокаДанные.НачисленоБонусов;

КонецЦикла;

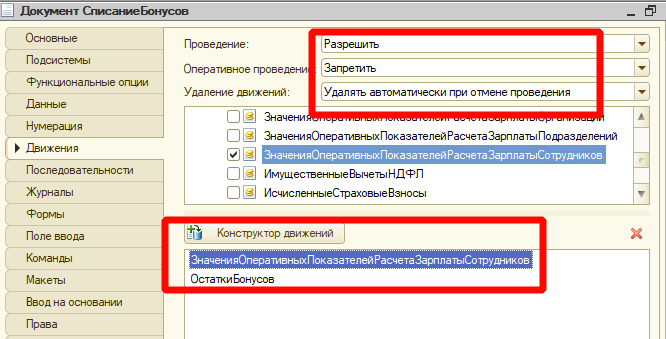
//}}\_\_КОНСТРУКТОР\_ДВИЖЕНИЙ\_РЕГИСТРОВ

КонецПроцедуры

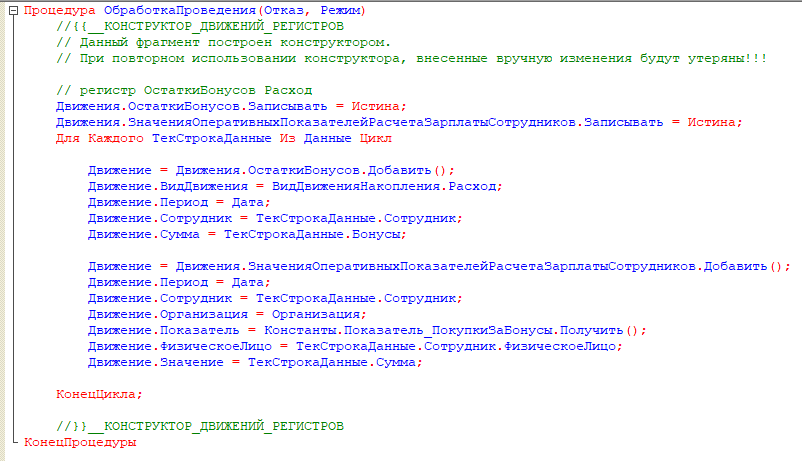
1. **Редактируем документ СписаниеБонусов:**



шапка (1)– Организация (СправочникСсылка.Организации), табл. часть – Данные (3) – Сотрудник (СправочникСсылка.Сотрудники), Бонусы и Сумма (Число 10.2)



1. **Модуль документа СписаниеБонусов (конструктором):**



Процедура ОбработкаПроведения(Отказ, Режим)

//{{\_\_КОНСТРУКТОР\_ДВИЖЕНИЙ\_РЕГИСТРОВ

// Данный фрагмент построен конструктором.

// При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!!

// регистр ОстаткиБонусов Расход

Движения.ОстаткиБонусов.Записывать = Истина;

Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников.Записывать = Истина;

Для Каждого ТекСтрокаДанные Из Данные Цикл

Движение = Движения.ОстаткиБонусов.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Период = Дата;

Движение.Сотрудник = ТекСтрокаДанные.Сотрудник;

Движение.Сумма = ТекСтрокаДанные.Бонусы;

Движение = Движения.ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников.Добавить();

Движение.Период = Дата;

Движение.Сотрудник = ТекСтрокаДанные.Сотрудник;

Движение.Организация = Организация;

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

Движение.ФизическоеЛицо = ТекСтрокаДанные.Сотрудник.ФизическоеЛицо;

Движение.Значение = ТекСтрокаДанные.Сумма;

КонецЦикла;

//}}\_\_КОНСТРУКТОР\_ДВИЖЕНИЙ\_РЕГИСТРОВ

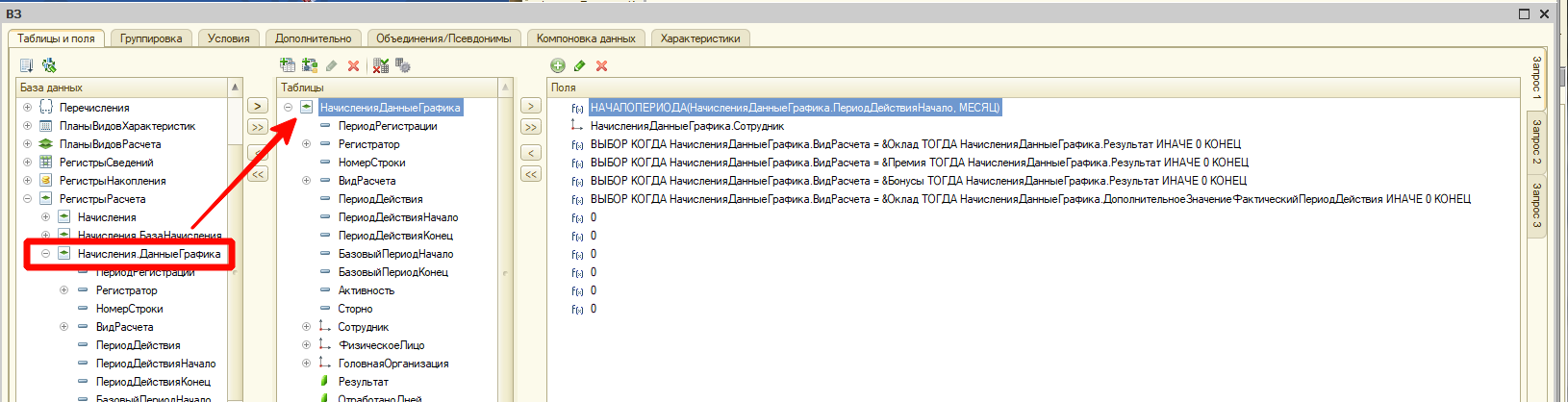
КонецПроцедуры

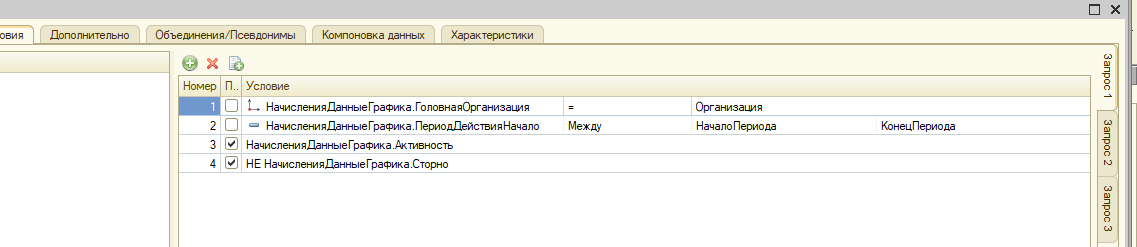
1. **Отчет МотивацияОператоров:**

**ЗАПРОС:**

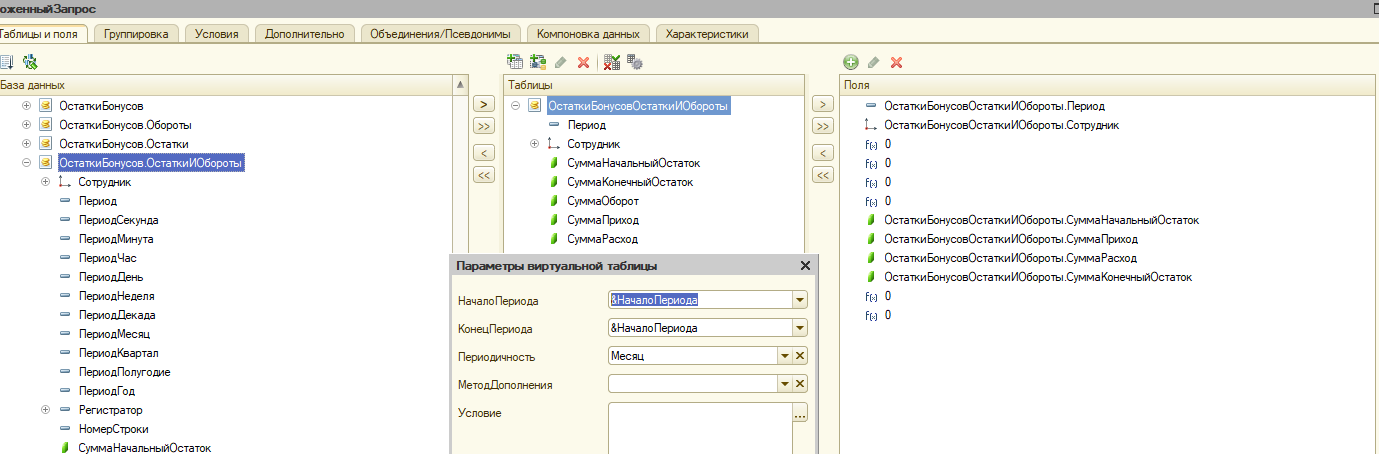
**НАЧАЛО ВЛОЖЕННОГО ЗАПРОСА ВЗ**

**ПЕРВЫЙ ПАКЕТ:**



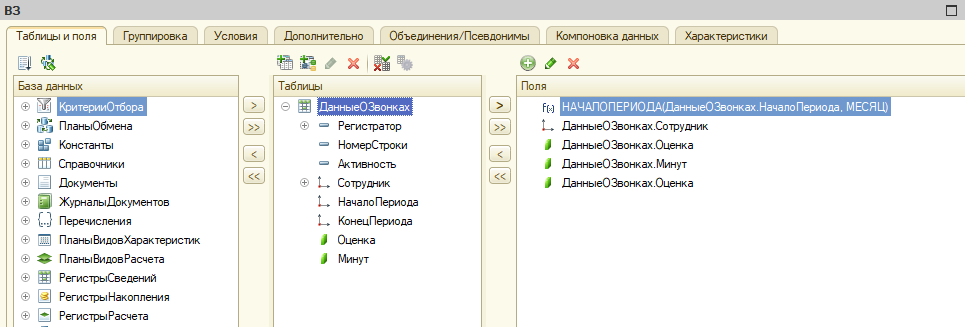


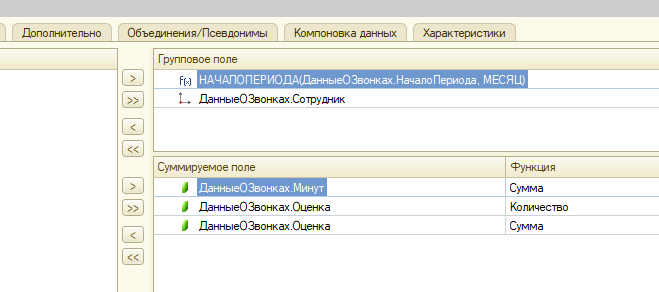
**ВТОРОЙ ПАКЕТ:**

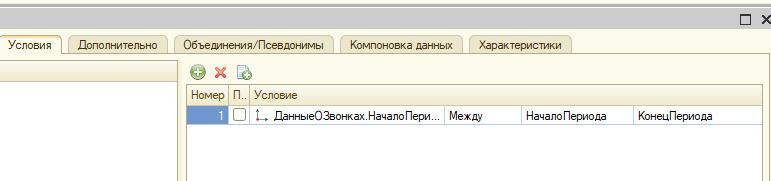


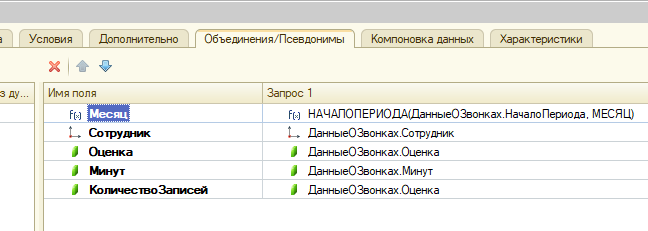
**ТРЕТИЙ ПАКЕТ:**

**ВЛОЖЕННЫЙ ЗАПРОС В ТРЕТИЙ ПАКЕТ:**

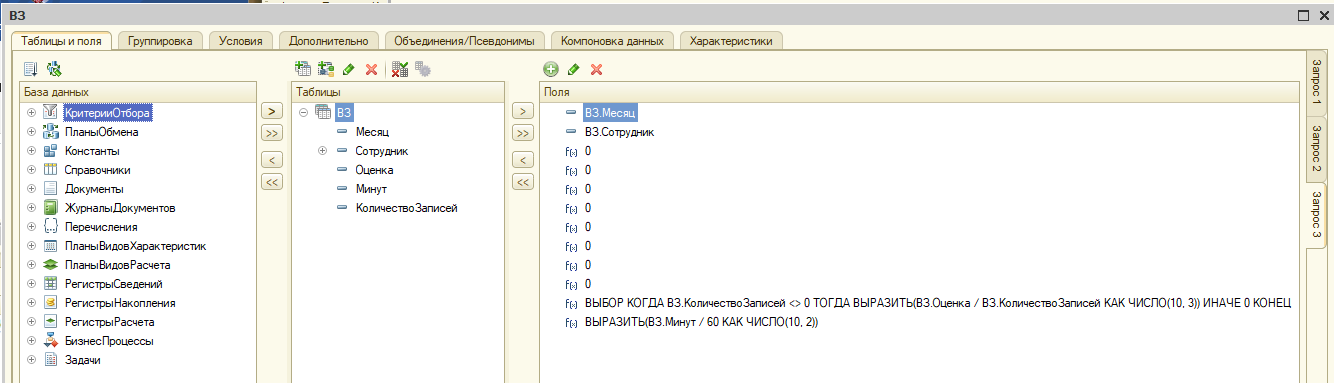




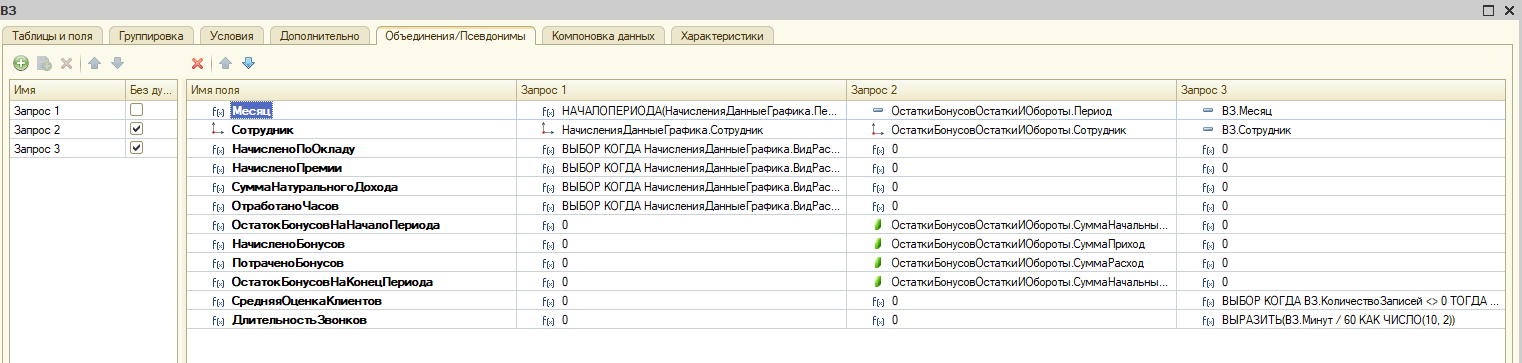




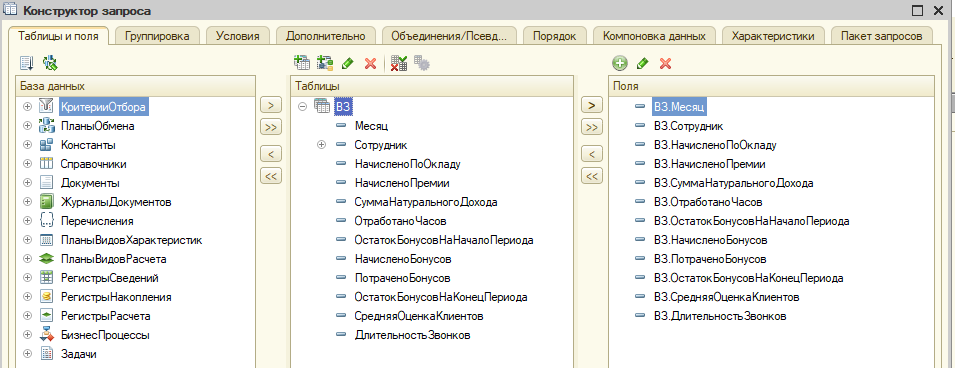
**ОКОНЧАНИЕ ВЛОЖЕННОГО ЗАПРОСА В ТРЕТИЙ ПАКЕТ:**

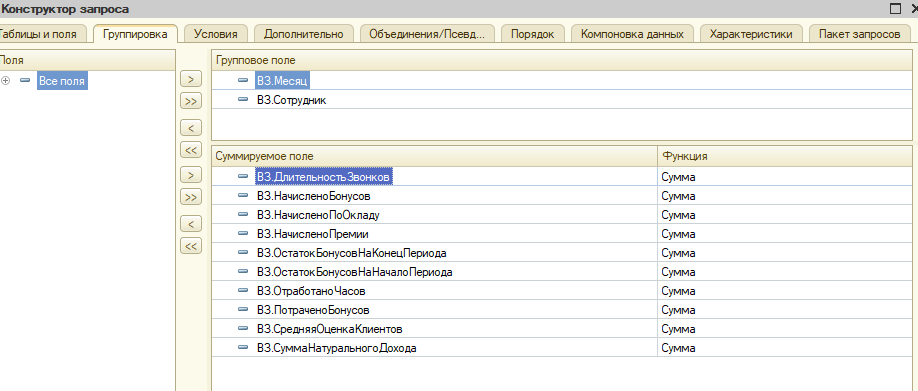


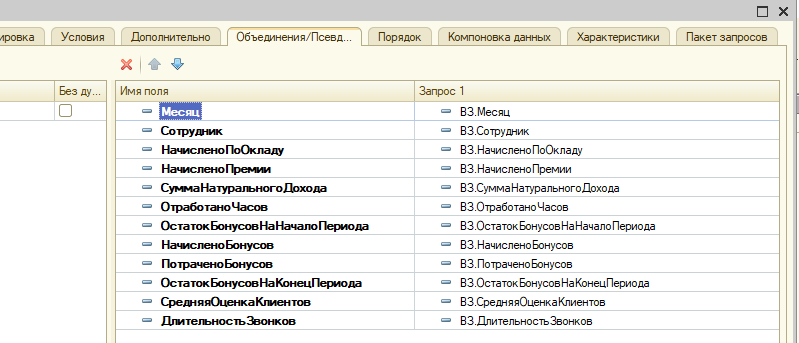
**ОБЪЕДИНЕНИЕ:**



**ОКОНЧАНИЕ ВЛОЖЕННОГО ЗАПРОСА ВЗ**







ВЫБРАТЬ

ВЗ.Месяц КАК Месяц,

ВЗ.Сотрудник КАК Сотрудник,

СУММА(ВЗ.НачисленоПоОкладу) КАК НачисленоПоОкладу,

СУММА(ВЗ.НачисленоПремии) КАК НачисленоПремии,

СУММА(ВЗ.СуммаНатуральногоДохода) КАК СуммаНатуральногоДохода,

СУММА(ВЗ.ОтработаноЧасов) КАК ОтработаноЧасов,

СУММА(ВЗ.ОстатокБонусовНаНачалоПериода) КАК ОстатокБонусовНаНачалоПериода,

СУММА(ВЗ.НачисленоБонусов) КАК НачисленоБонусов,

СУММА(ВЗ.ПотраченоБонусов) КАК ПотраченоБонусов,

СУММА(ВЗ.ОстатокБонусовНаКонецПериода) КАК ОстатокБонусовНаКонецПериода,

СУММА(ВЗ.СредняяОценкаКлиентов) КАК СредняяОценкаКлиентов,

СУММА(ВЗ.ДлительностьЗвонков) КАК ДлительностьЗвонков

ИЗ

(ВЫБРАТЬ

НАЧАЛОПЕРИОДА(НачисленияДанныеГрафика.ПериодДействияНачало, МЕСЯЦ) КАК Месяц,

НачисленияДанныеГрафика.Сотрудник КАК Сотрудник,

ВЫБОР

КОГДА НачисленияДанныеГрафика.ВидРасчета = &Оклад

ТОГДА НачисленияДанныеГрафика.Результат

ИНАЧЕ 0

КОНЕЦ КАК НачисленоПоОкладу,

ВЫБОР

КОГДА НачисленияДанныеГрафика.ВидРасчета = &Премия

ТОГДА НачисленияДанныеГрафика.Результат

ИНАЧЕ 0

КОНЕЦ КАК НачисленоПремии,

ВЫБОР

КОГДА НачисленияДанныеГрафика.ВидРасчета = &Бонусы

ТОГДА НачисленияДанныеГрафика.Результат

ИНАЧЕ 0

КОНЕЦ КАК СуммаНатуральногоДохода,

ВЫБОР

КОГДА НачисленияДанныеГрафика.ВидРасчета = &Оклад

ТОГДА НачисленияДанныеГрафика.ДополнительноеЗначениеФактическийПериодДействия

ИНАЧЕ 0

КОНЕЦ КАК ОтработаноЧасов,

0 КАК ОстатокБонусовНаНачалоПериода,

0 КАК НачисленоБонусов,

0 КАК ПотраченоБонусов,

0 КАК ОстатокБонусовНаКонецПериода,

0 КАК СредняяОценкаКлиентов,

0 КАК ДлительностьЗвонков

ИЗ

РегистрРасчета.Начисления.ДанныеГрафика КАК НачисленияДанныеГрафика

ГДЕ

НачисленияДанныеГрафика.ГоловнаяОрганизация = &Организация

И НачисленияДанныеГрафика.ПериодДействияНачало МЕЖДУ &НачалоПериода И &КонецПериода

И НачисленияДанныеГрафика.Активность

И НЕ НачисленияДанныеГрафика.Сторно

ОБЪЕДИНИТЬ

ВЫБРАТЬ

ОстаткиБонусовОстаткиИОбороты.Период,

ОстаткиБонусовОстаткиИОбороты.Сотрудник,

0,

0,

0,

0,

ОстаткиБонусовОстаткиИОбороты.СуммаНачальныйОстаток,

ОстаткиБонусовОстаткиИОбороты.СуммаПриход,

ОстаткиБонусовОстаткиИОбороты.СуммаРасход,

ОстаткиБонусовОстаткиИОбороты.СуммаНачальныйОстаток,

0,

0

ИЗ

РегистрНакопления.ОстаткиБонусов.ОстаткиИОбороты(&НачалоПериода, &КонецПериода, Месяц, , ) КАК ОстаткиБонусовОстаткиИОбороты

ОБЪЕДИНИТЬ

ВЫБРАТЬ

ВЗ.Месяц,

ВЗ.Сотрудник,

0,

0,

0,

0,

0,

0,

0,

0,

ВЫБОР

КОГДА ВЗ.КоличествоЗаписей <> 0

ТОГДА ВЫРАЗИТЬ(ВЗ.Оценка / ВЗ.КоличествоЗаписей КАК ЧИСЛО(10, 3))

ИНАЧЕ 0

КОНЕЦ,

ВЫРАЗИТЬ(ВЗ.Минут / 60 КАК ЧИСЛО(10, 2))

ИЗ

(ВЫБРАТЬ

НАЧАЛОПЕРИОДА(ДанныеОЗвонках.НачалоПериода, МЕСЯЦ) КАК Месяц,

ДанныеОЗвонках.Сотрудник КАК Сотрудник,

СУММА(ДанныеОЗвонках.Оценка) КАК Оценка,

СУММА(ДанныеОЗвонках.Минут) КАК Минут,

КОЛИЧЕСТВО(ДанныеОЗвонках.Оценка) КАК КоличествоЗаписей

ИЗ

РегистрСведений.ДанныеОЗвонках КАК ДанныеОЗвонках

ГДЕ

ДанныеОЗвонках.НачалоПериода МЕЖДУ &НачалоПериода И &КонецПериода

СГРУППИРОВАТЬ ПО

НАЧАЛОПЕРИОДА(ДанныеОЗвонках.НачалоПериода, МЕСЯЦ),

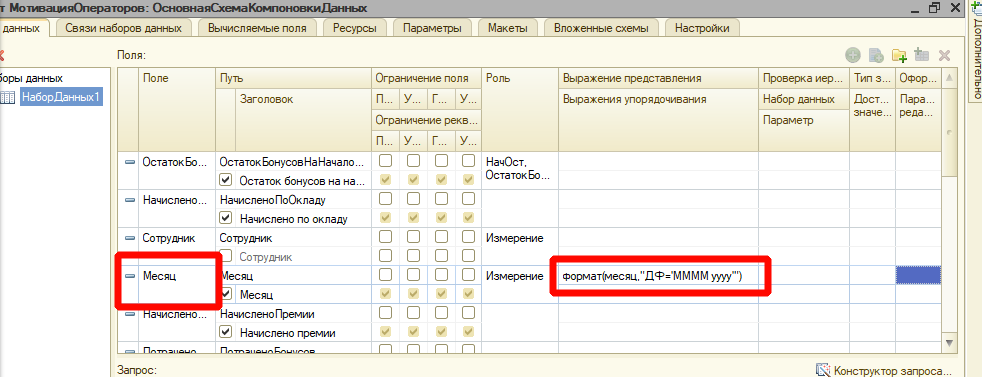
ДанныеОЗвонках.Сотрудник) КАК ВЗ) КАК ВЗ

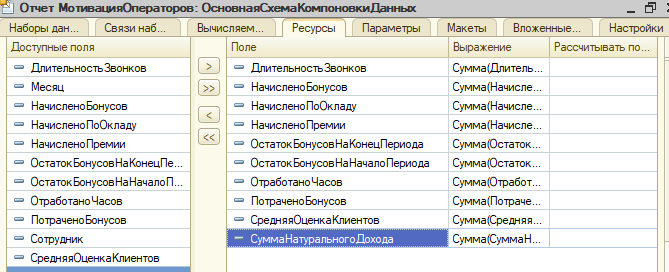
СГРУППИРОВАТЬ ПО

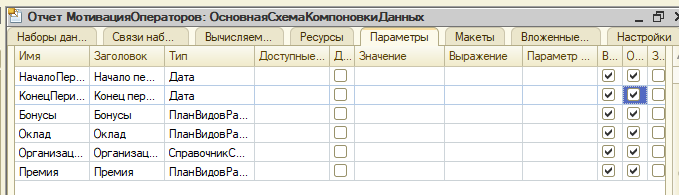
ВЗ.Месяц,

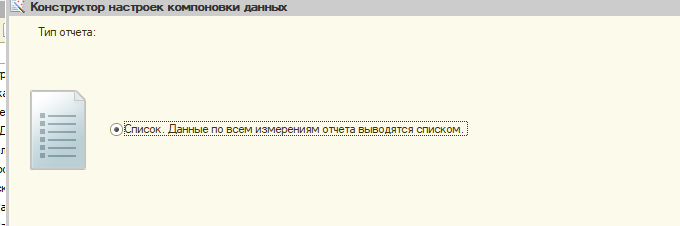
ВЗ.Сотрудник

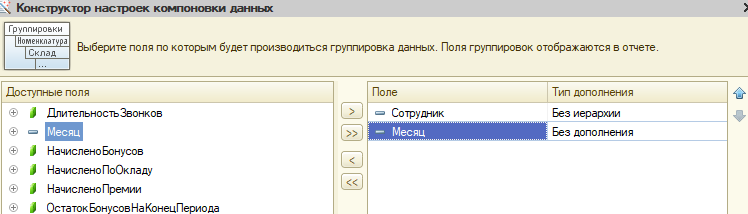
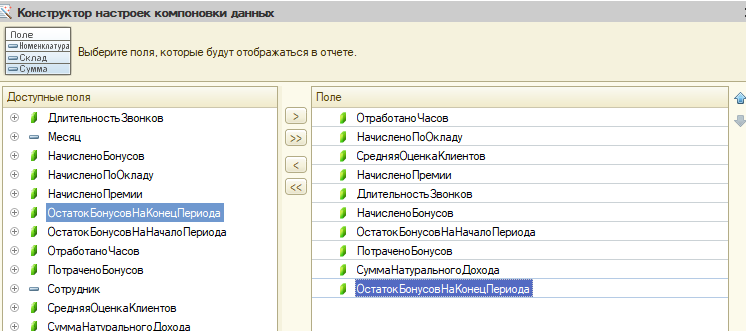
**Далее в СКД:**



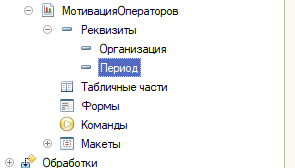


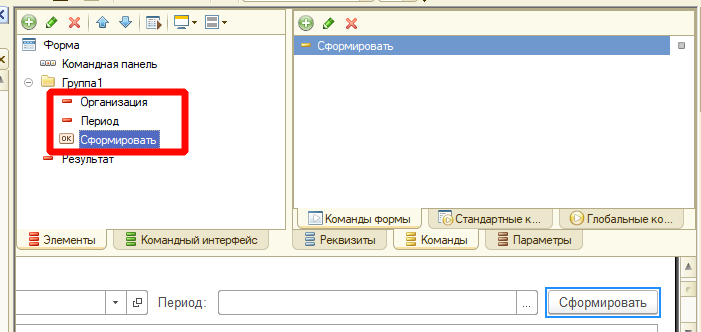






**Добавляем реквизиты Организация и Период. Делаем форму:**

**Сформировать – клиент и сервер.**



**Модуль формы отчета:**



&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(Отчет.Период.ДатаОкончания);

КонецЕсли;

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

ПараметрДанных.Использование = Истина;

ПараметрДанных.Значение = НачалоМесяца(Отчет.Период.ДатаНачала);

КонецЕсли;

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

ПараметрДанных.Использование = Истина;

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

КонецЕсли;

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

ПараметрДанных.Использование = Истина;

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

КонецЕсли;

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

ПараметрДанных.Использование = Истина;

ПараметрДанных.Значение = Отчет.Организация;

КонецЕсли;

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

ПараметрДанных.Использование = Истина;

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

КонецЕсли;

КонецЦикла;

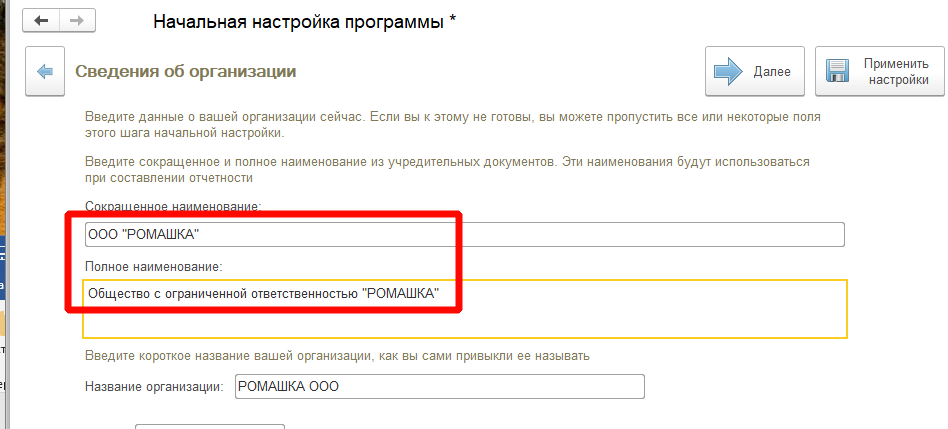
Отчет.КомпоновщикНастроек.ЗагрузитьНастройки(Настройки);

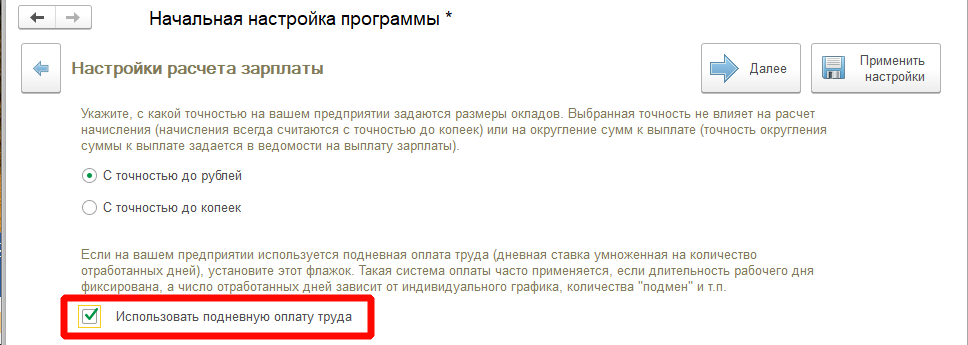
СкомпоноватьРезультат();

КонецПроцедуры

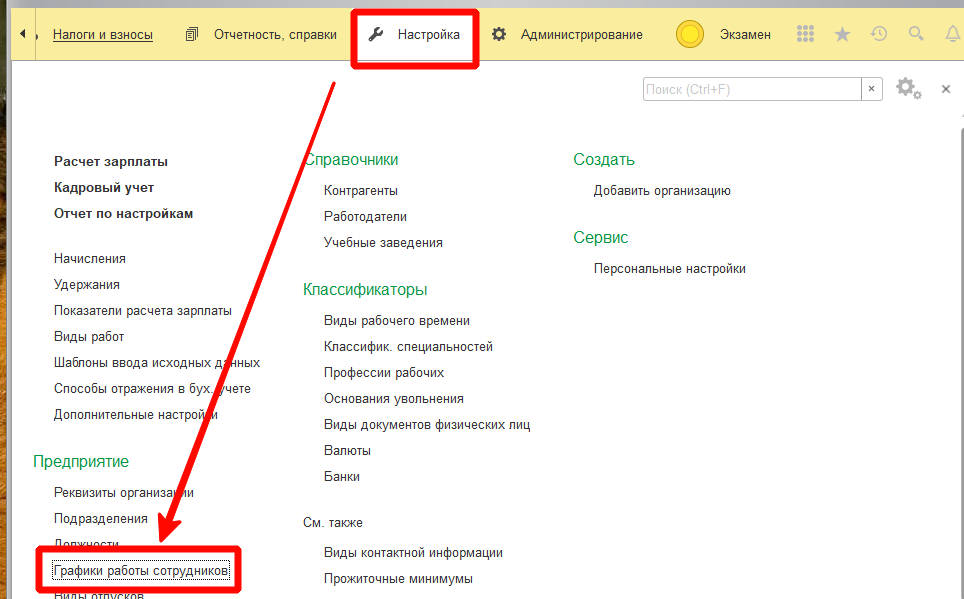
**III Ввод контрольного примера и проверка решения:**

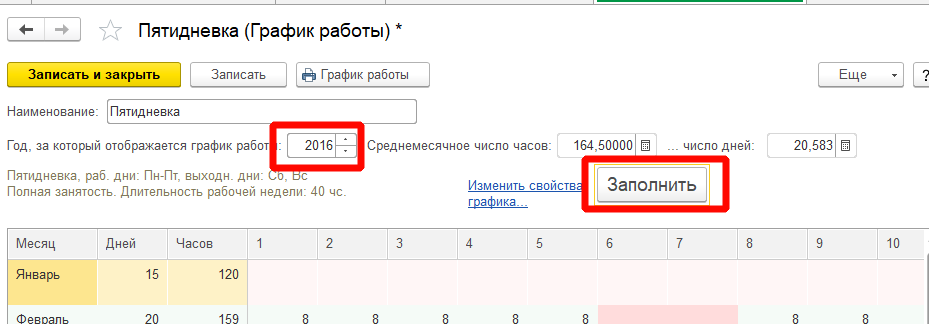
1. **Делаем начальную настройку программы:**



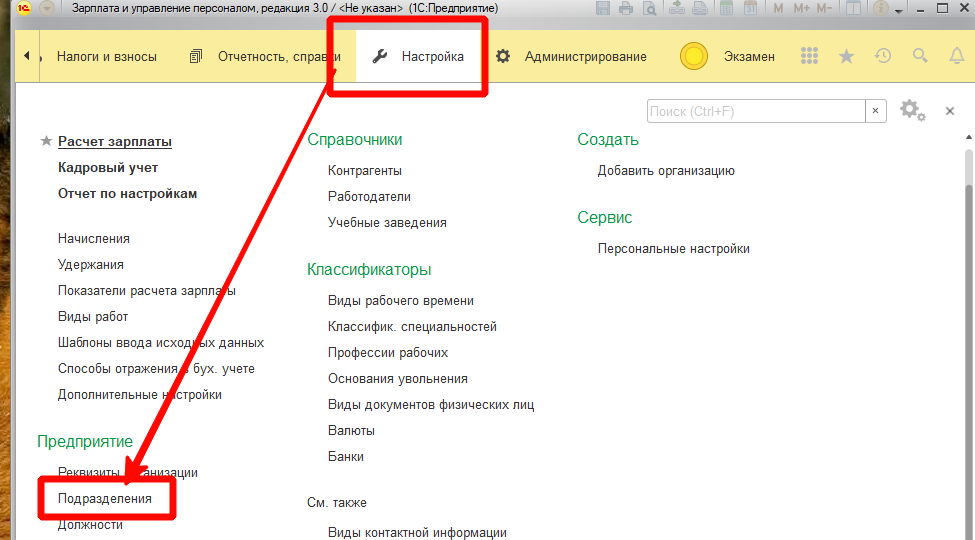


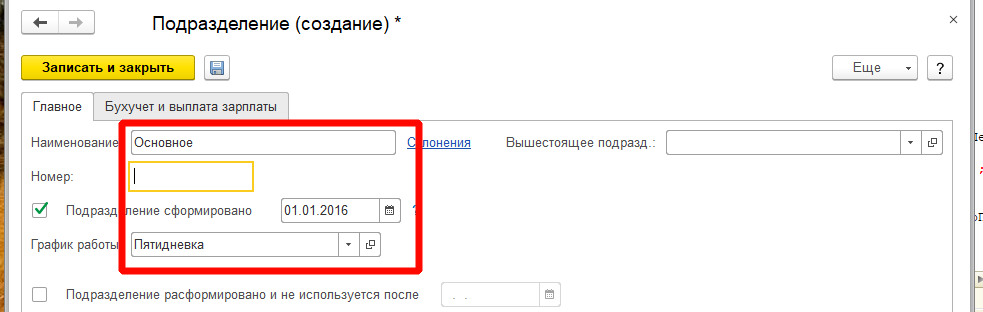
1. **Заполняем график на 2016 год:**



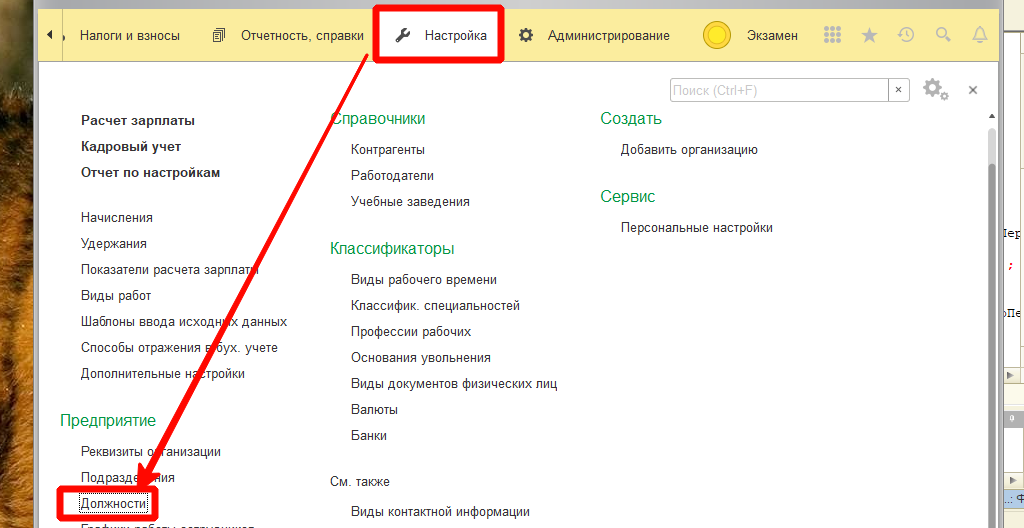


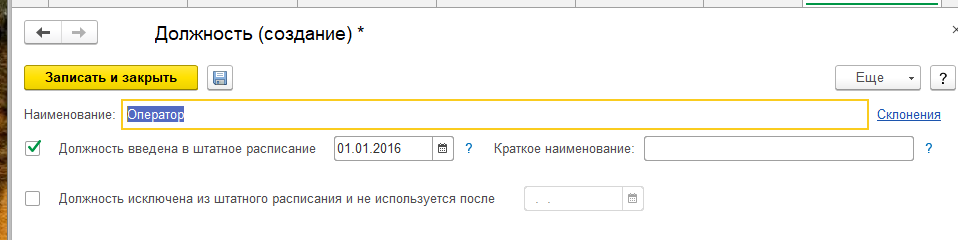
1. **Добавляем подразделение Основное:**



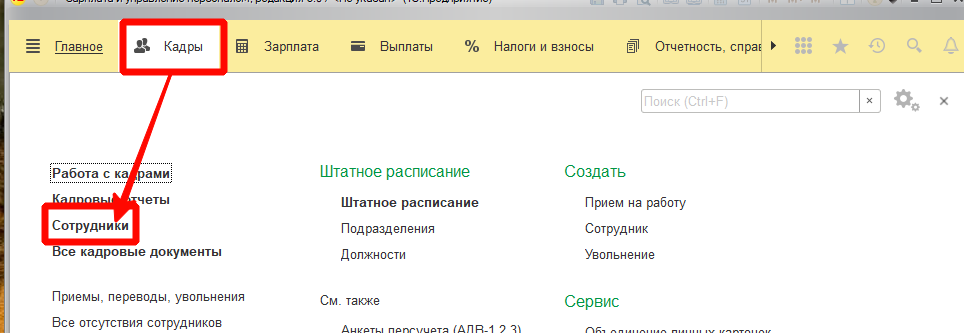


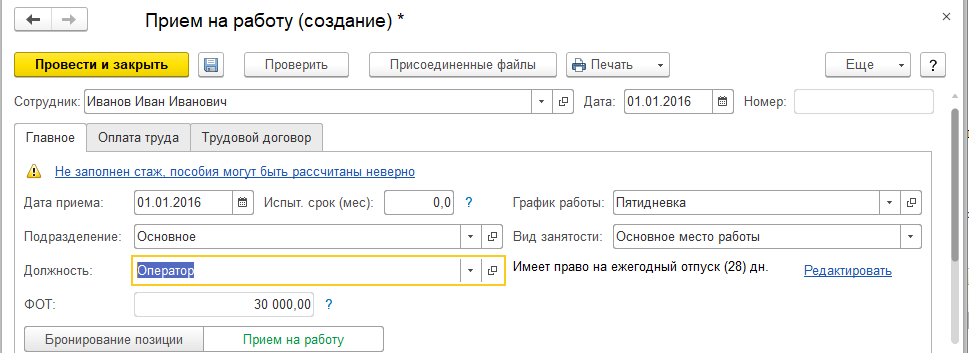
1. **Создаем должность Оператор:**

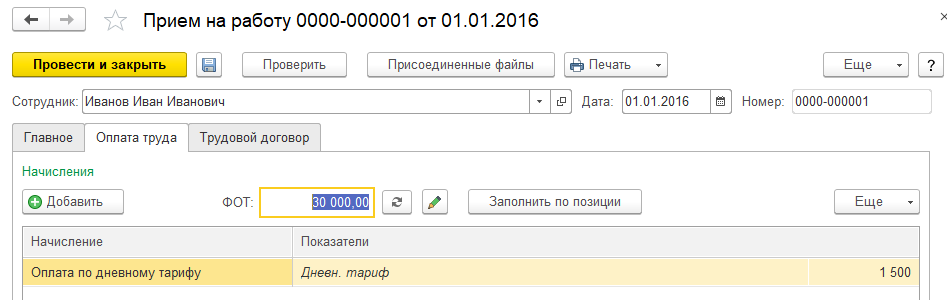


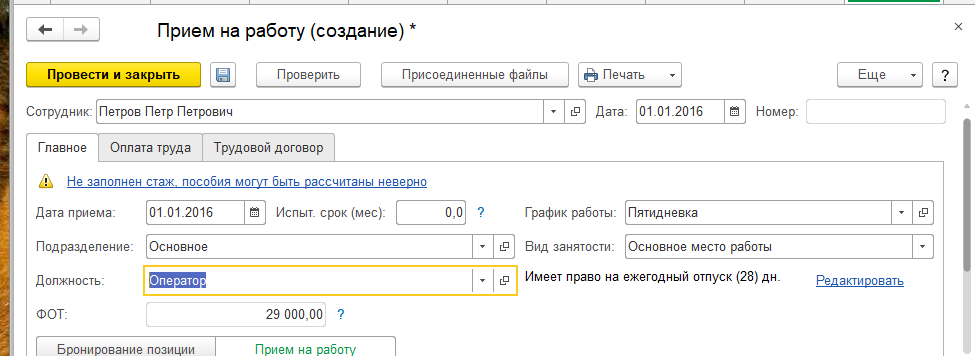


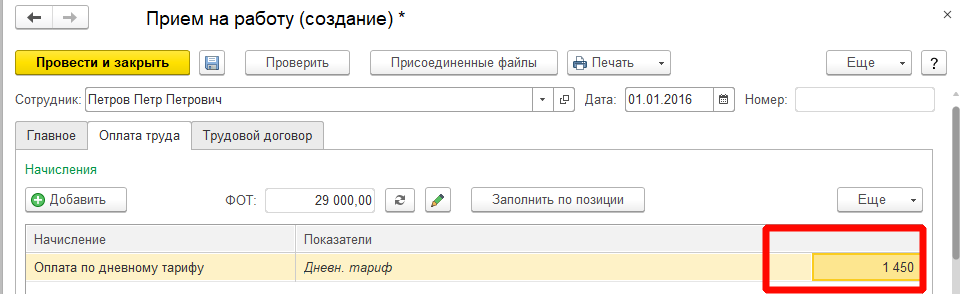
1. **Принимаем на работу Иванова и Петрова:**



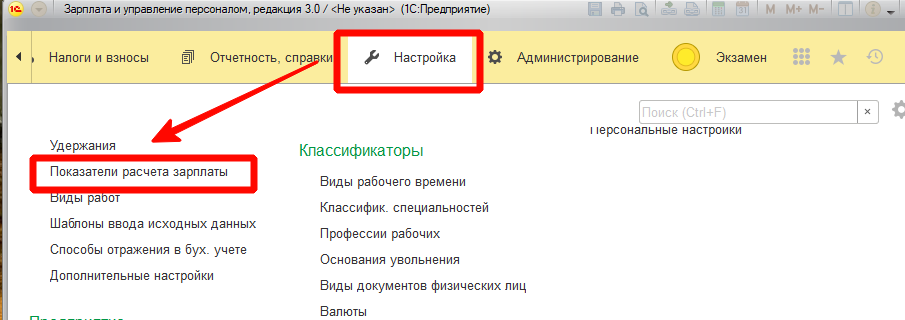


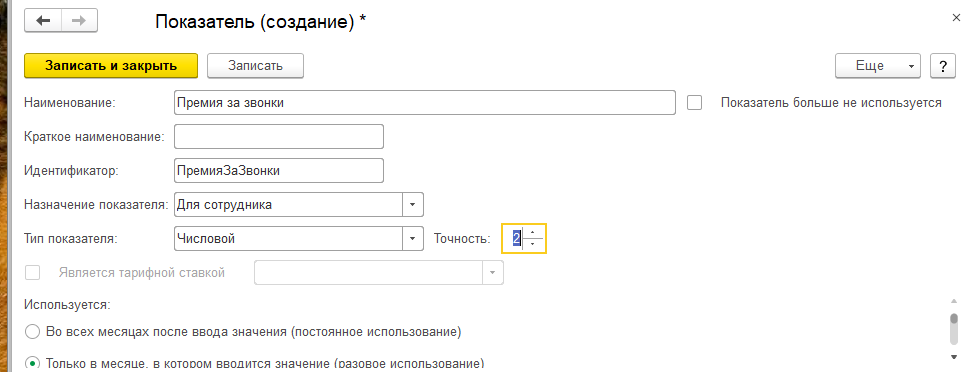


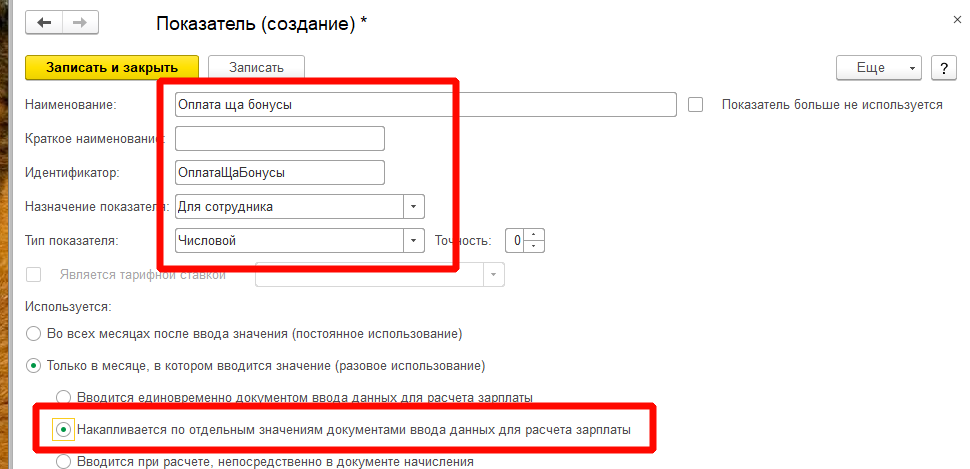




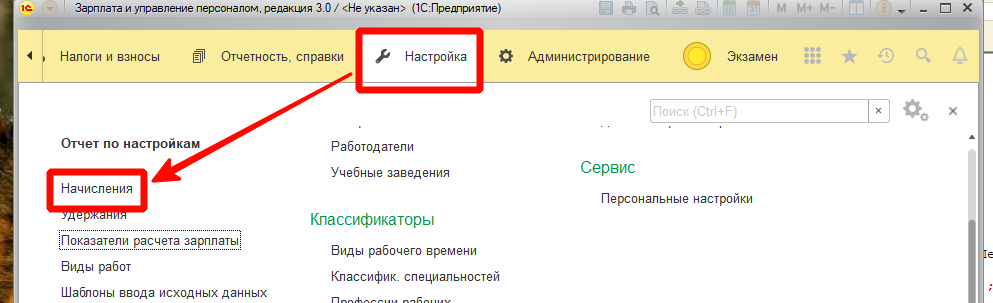
1. **Добавляем два показателя расчета зарплаты:**

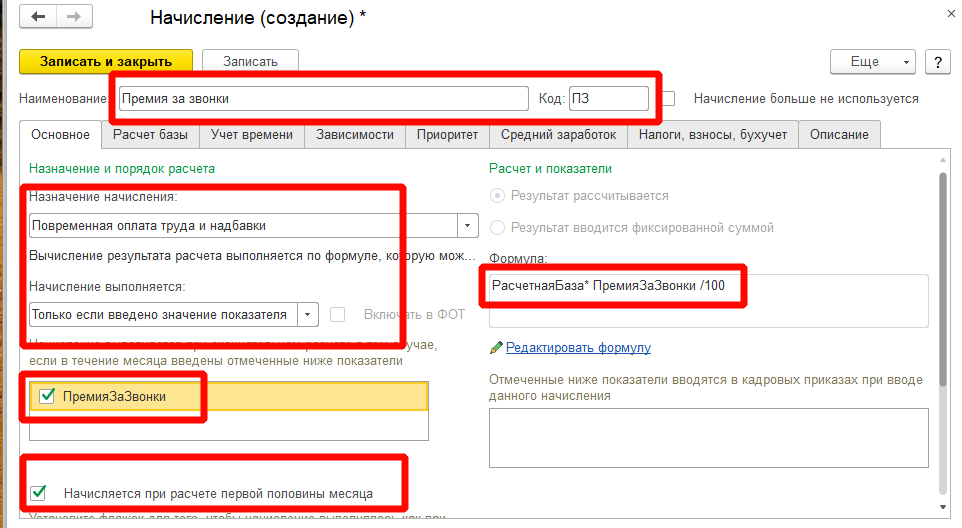


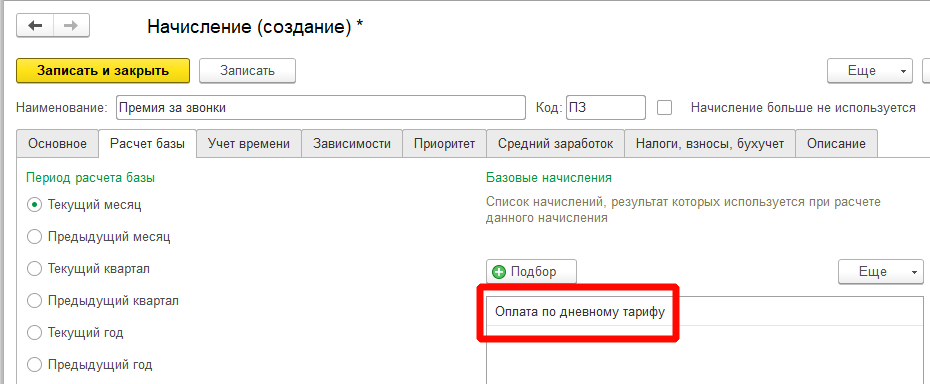


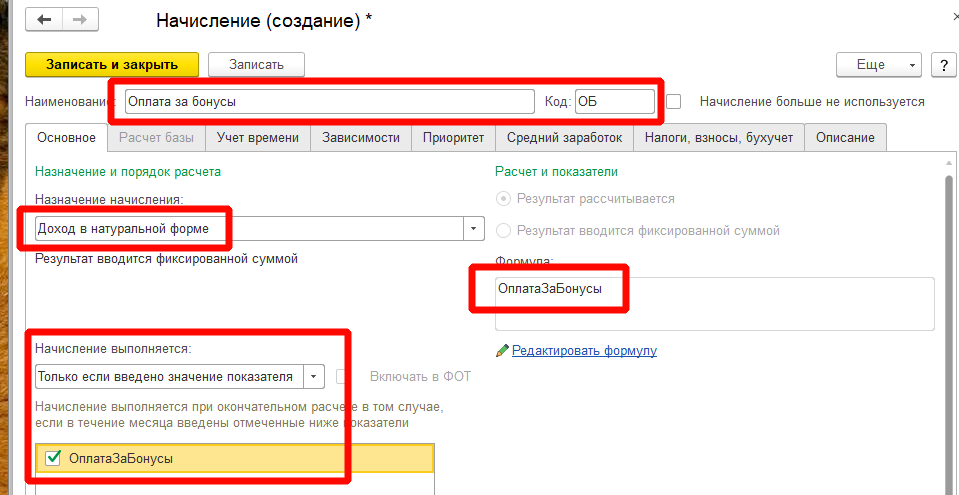


1. **Добавляем два новых начисления:**

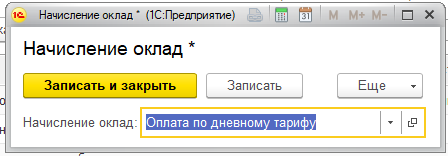


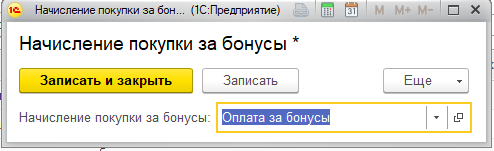


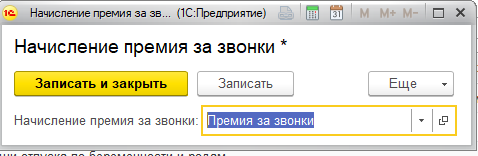


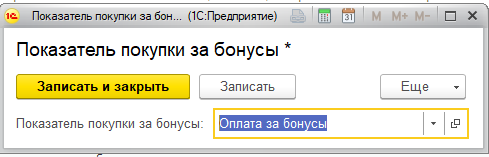


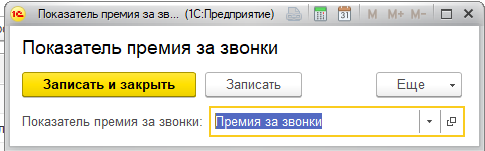
1. **Заполняем константы:**



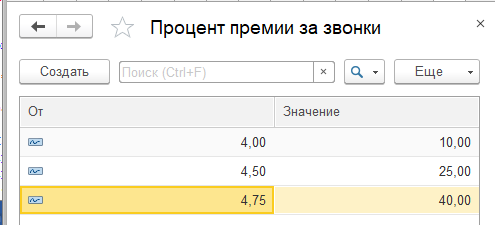




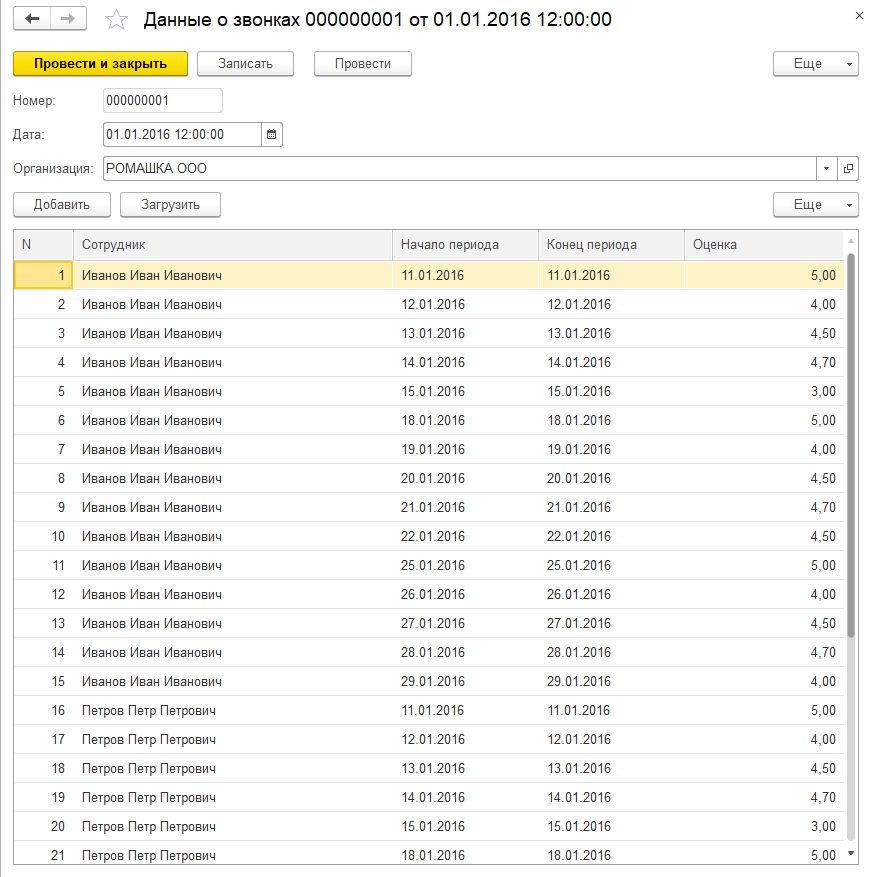


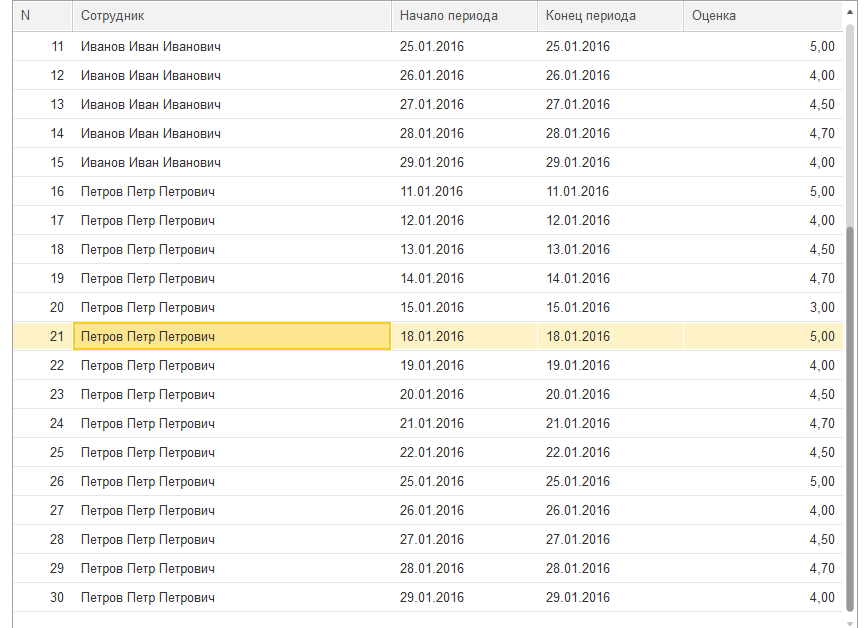


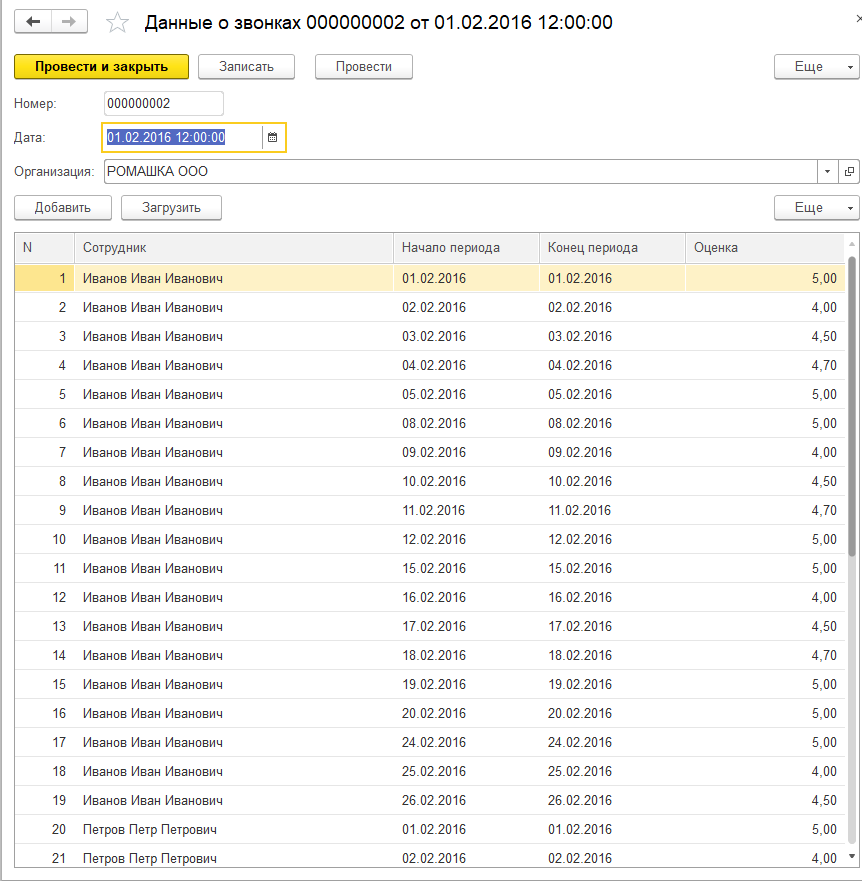
1. **Вводим процент премии за звонки:**

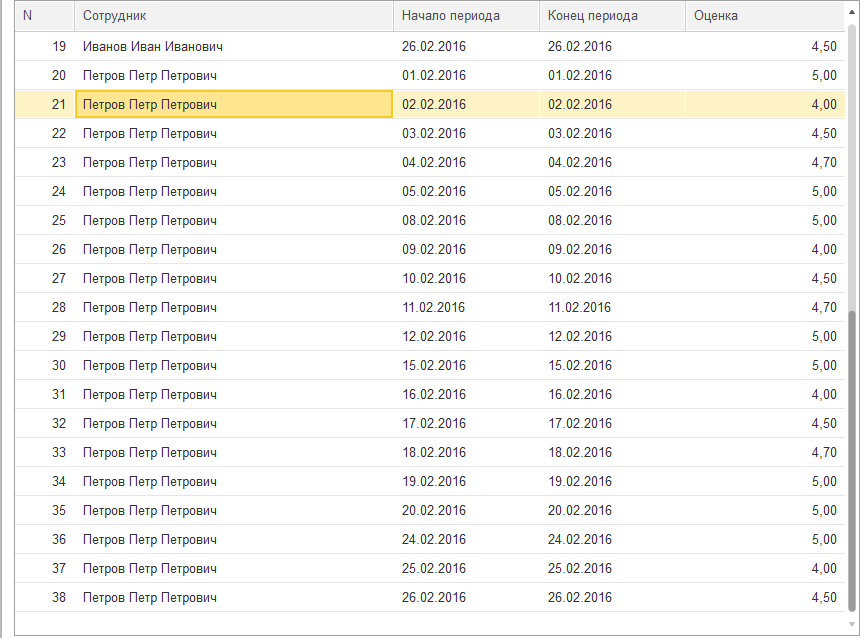


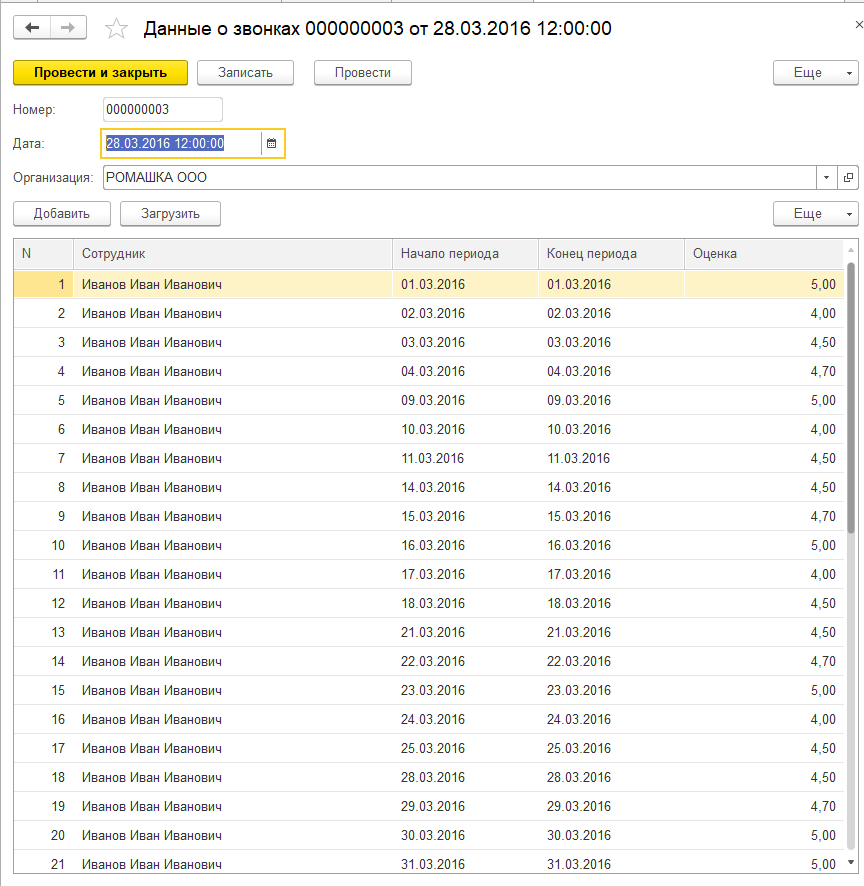
1. **Добавляем три документа Данные о звонках:**

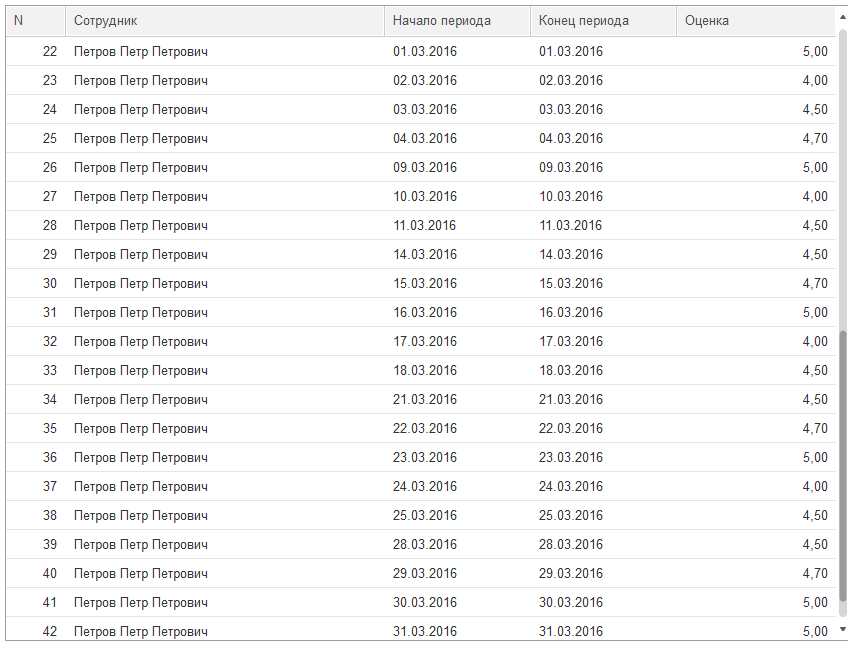


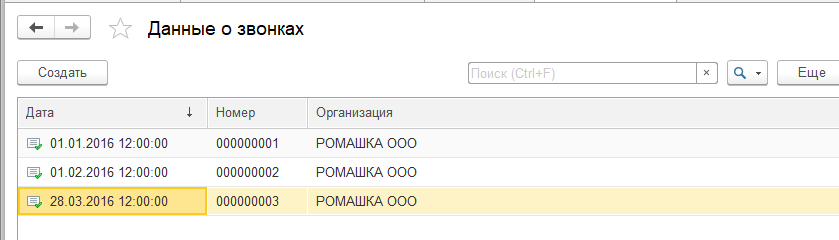




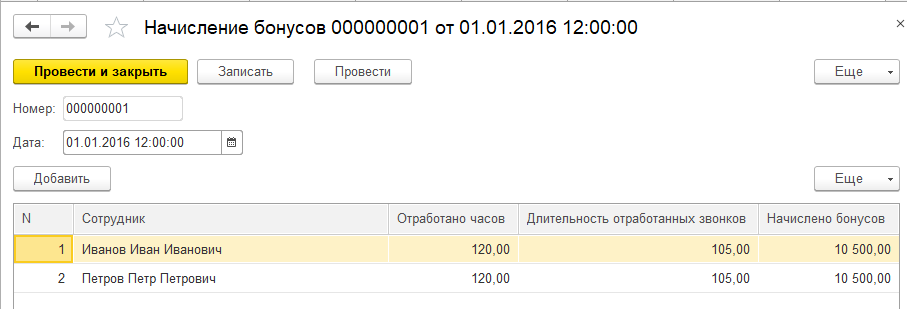


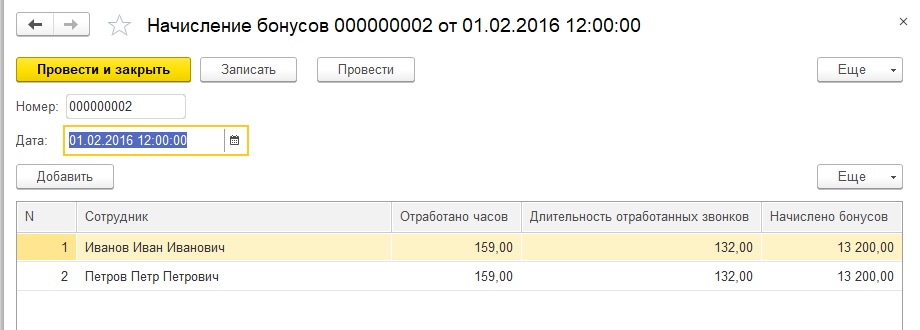


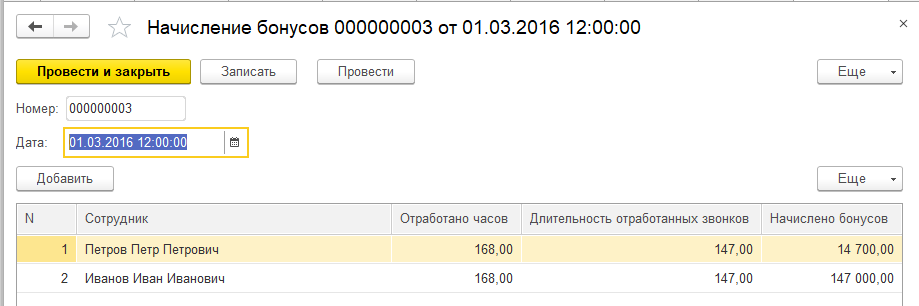




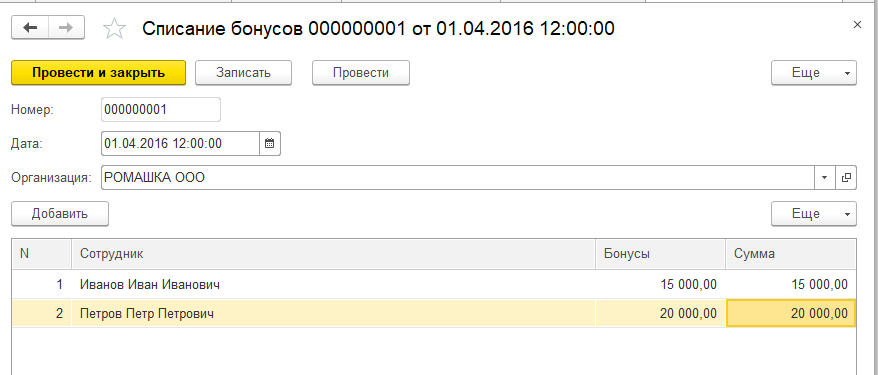
1. **Создаем три документа Начисление бонусов:**



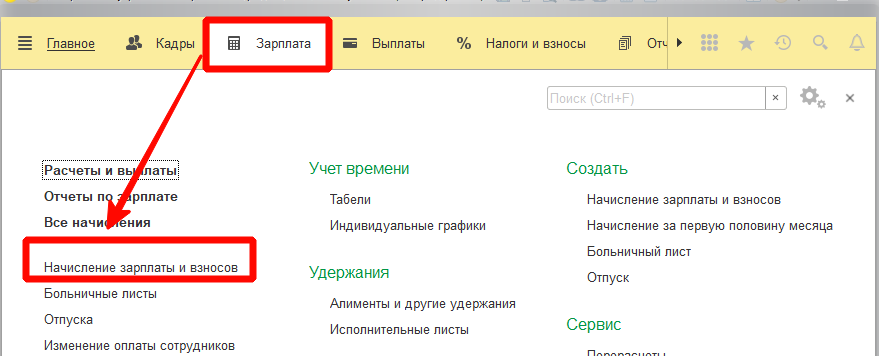


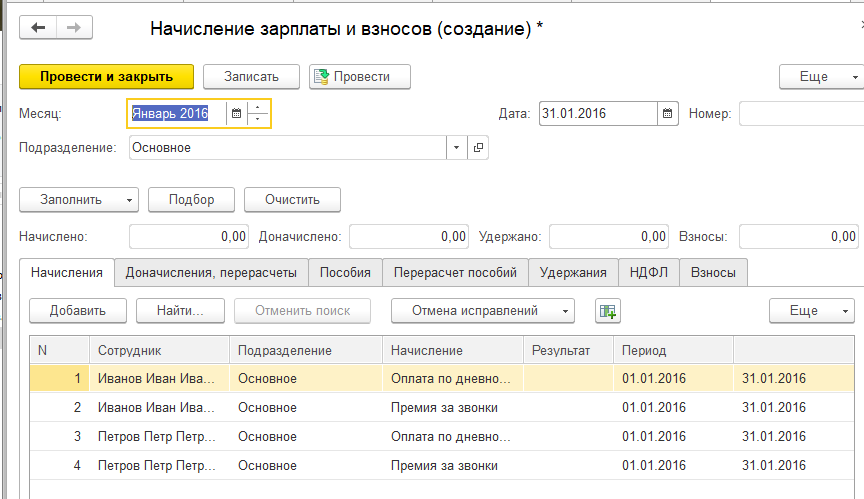


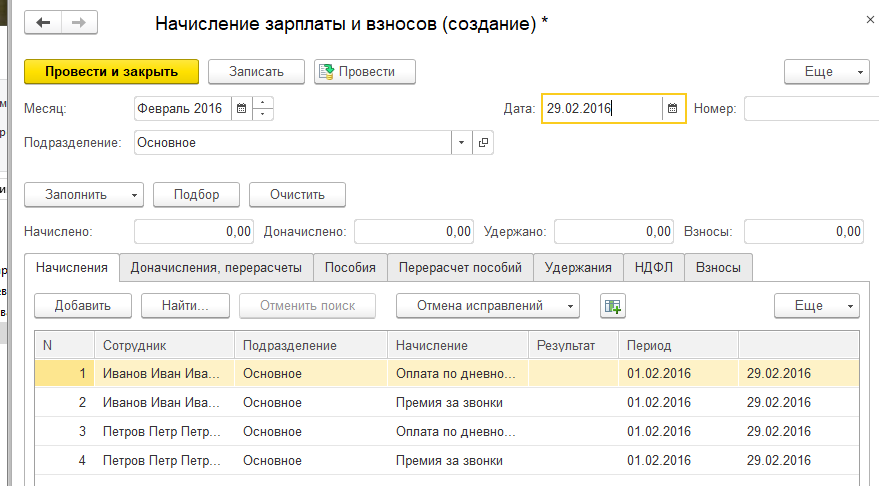
1. **Вводим один документ Списание бонусов:**

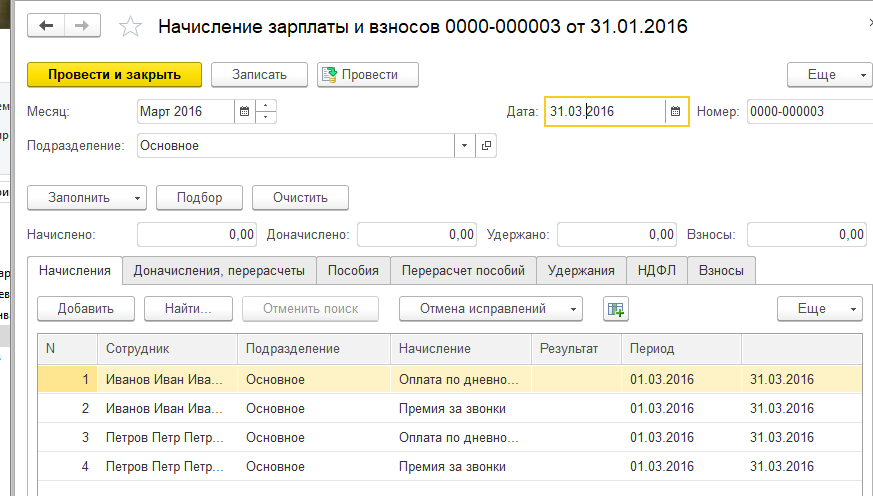


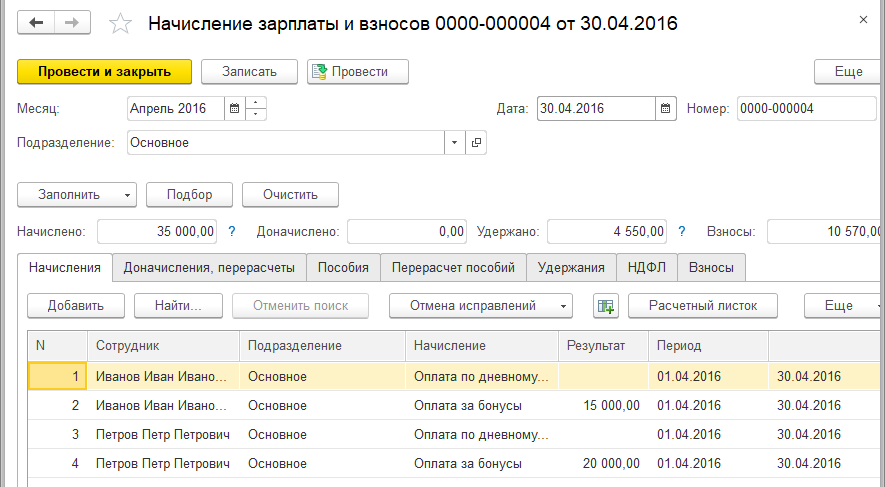
1. **Начисляем ЗП за янв., февр., март, апр. 2016 г:**

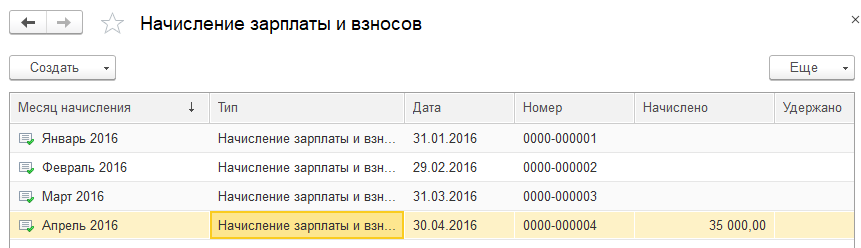




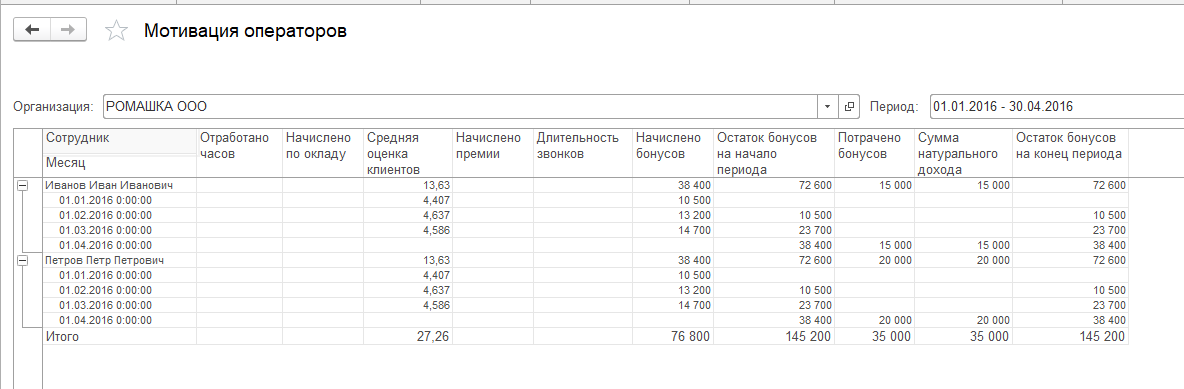








1. **Также смотрим отчет:**



**IV ПОЯСНЕНИЕ К РЕШЕНИЮ**

Для реализации решения созданы 5 констант для хранения показателей расчета зарплаты и начислений.

Созданы документы НачислениБонусов и СписаниеБонусов с информацией о начисленных бонусах и отработанных часах

Для реализации механизма начисления работникам оплаты по окладу, премии и натурального дохода созданы регистры ПроцентПремииЗаЗвонки, ДанныеОЗвонках и ОстаткиБонусов, а для целей обеспечения учета для целей налогообложения, созданные документ СписаниеБонусов делает движение в регистр ЗначенияОперативныхПоказателейРасчетаЗарплатыСотрудников

Создан настраиваемый отчет МотивацияОператоров в соответствии с образцом. Отчет проверен на работоспособность.

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

В базе данных приведен контрольный пример начисления за 4 месяца по 2-м сотрудникам.

Таким образом все требуемые в задании механизмы реализованы.

**V ТЕКСТОВЫЕ ФАЙЛЫ**

|  |  |  |
| --- | --- | --- |
| Данные о звонках Январь 2016 | Данные о звонках Февраль 2016 | Данные о звонках Март 2016 |
| 11.01.2016 10:00;11.01.2016 17:00;0000-00001;5  12.01.2016 10:00;12.01.2016 17:00;0000-00001;4  13.01.2016 10:00;13.01.2016 17:00;0000-00001;4.5  14.01.2016 10:00;14.01.2016 17:00;0000-00001;4.7  15.01.2016 10:00;15.01.2016 17:00;0000-00001;3  18.01.2016 10:00;18.01.2016 17:00;0000-00001;5  19.01.2016 10:00;19.01.2016 17:00;0000-00001;4  20.01.2016 10:00;20.01.2016 17:00;0000-00001;4.5  21.01.2016 10:00;21.01.2016 17:00;0000-00001;4.7  22.01.2016 10:00;22.01.2016 17:00;0000-00001;4.5  25.01.2016 10:00;25.01.2016 17:00;0000-00001;5  26.01.2016 10:00;26.01.2016 17:00;0000-00001;4  27.01.2016 10:00;27.01.2016 17:00;0000-00001;4.5  28.01.2016 10:00;28.01.2016 17:00;0000-00001;4.7  29.01.2016 10:00;29.01.2016 17:00;0000-00001;4  11.01.2016 10:00;11.01.2016 17:00;0000-00002;5  12.01.2016 10:00;12.01.2016 17:00;0000-00002;4  13.01.2016 10:00;13.01.2016 17:00;0000-00002;4.5  14.01.2016 10:00;14.01.2016 17:00;0000-00002;4.7  15.01.2016 10:00;15.01.2016 17:00;0000-00002;3  18.01.2016 10:00;18.01.2016 17:00;0000-00002;5  19.01.2016 10:00;19.01.2016 17:00;0000-00002;4  20.01.2016 10:00;20.01.2016 17:00;0000-00002;4.5  21.01.2016 10:00;21.01.2016 17:00;0000-00002;4.7  22.01.2016 10:00;22.01.2016 17:00;0000-00002;4.5  25.01.2016 10:00;25.01.2016 17:00;0000-00002;5  26.01.2016 10:00;26.01.2016 17:00;0000-00002;4  27.01.2016 10:00;27.01.2016 17:00;0000-00002;4.5  28.01.2016 10:00;28.01.2016 17:00;0000-00002;4.7  29.01.2016 10:00;29.01.2016 17:00;0000-00002;4 | 01.02.2016 10:00;01.02.2016 17:00;0000-00001;5  02.02.2016 10:00;02.02.2016 17:00;0000-00001;4  03.02.2016 10:00;03.02.2016 17:00;0000-00001;4.5  04.02.2016 10:00;04.02.2016 17:00;0000-00001;4.7  05.02.2016 10:00;05.02.2016 17:00;0000-00001;5  08.02.2016 10:00;08.02.2016 17:00;0000-00001;5  09.02.2016 10:00;09.02.2016 17:00;0000-00001;4  10.02.2016 10:00;10.02.2016 17:00;0000-00001;4.5  11.02.2016 10:00;11.02.2016 17:00;0000-00001;4.7  12.02.2016 10:00;12.02.2016 17:00;0000-00001;5  15.02.2016 10:00;15.02.2016 17:00;0000-00001;5  16.02.2016 10:00;16.02.2016 17:00;0000-00001;4  17.02.2016 10:00;17.02.2016 17:00;0000-00001;4.5  18.02.2016 10:00;18.02.2016 17:00;0000-00001;4.7  19.02.2016 10:00;19.02.2016 17:00;0000-00001;5  20.02.2016 10:00;20.02.2016 16:00;0000-00001;5  24.02.2016 10:00;24.02.2016 17:00;0000-00001;5  25.02.2016 10:00;25.02.2016 17:00;0000-00001;4  26.02.2016 10:00;26.02.2016 17:00;0000-00001;4.5  01.02.2016 10:00;01.02.2016 17:00;0000-00002;5  02.02.2016 10:00;02.02.2016 17:00;0000-00002;4  03.02.2016 10:00;03.02.2016 17:00;0000-00002;4.5  04.02.2016 10:00;04.02.2016 17:00;0000-00002;4.7  05.02.2016 10:00;05.02.2016 17:00;0000-00002;5  08.02.2016 10:00;08.02.2016 17:00;0000-00002;5  09.02.2016 10:00;09.02.2016 17:00;0000-00002;4  10.02.2016 10:00;10.02.2016 17:00;0000-00002;4.5  11.02.2016 10:00;11.02.2016 17:00;0000-00002;4.7  12.02.2016 10:00;12.02.2016 17:00;0000-00002;5  15.02.2016 10:00;15.02.2016 17:00;0000-00002;5  16.02.2016 10:00;16.02.2016 17:00;0000-00002;4  17.02.2016 10:00;17.02.2016 17:00;0000-00002;4.5  18.02.2016 10:00;18.02.2016 17:00;0000-00002;4.7  19.02.2016 10:00;19.02.2016 17:00;0000-00002;5  20.02.2016 10:00;20.02.2016 16:00;0000-00002;5  24.02.2016 10:00;24.02.2016 17:00;0000-00002;5  25.02.2016 10:00;25.02.2016 17:00;0000-00002;4  26.02.2016 10:00;26.02.2016 17:00;0000-00002;4.5 | 11.01.2016 10:00;11.01.2016 17:00;0000-00001;5  12.01.2016 10:00;12.01.2016 17:00;0000-00001;4  13.01.2016 10:00;13.01.2016 17:00;0000-00001;4.5  14.01.2016 10:00;14.01.2016 17:00;0000-00001;4.7  15.01.2016 10:00;15.01.2016 17:00;0000-00001;3  18.01.2016 10:00;18.01.2016 17:00;0000-00001;5  19.01.2016 10:00;19.01.2016 17:00;0000-00001;4  20.01.2016 10:00;20.01.2016 17:00;0000-00001;4.5  21.01.2016 10:00;21.01.2016 17:00;0000-00001;4.7  22.01.2016 10:00;22.01.2016 17:00;0000-00001;4.5  25.01.2016 10:00;25.01.2016 17:00;0000-00001;5  26.01.2016 10:00;26.01.2016 17:00;0000-00001;4  27.01.2016 10:00;27.01.2016 17:00;0000-00001;4.5  28.01.2016 10:00;28.01.2016 17:00;0000-00001;4.7  29.01.2016 10:00;29.01.2016 17:00;0000-00001;4  11.01.2016 10:00;11.01.2016 17:00;0000-00002;5  12.01.2016 10:00;12.01.2016 17:00;0000-00002;4  13.01.2016 10:00;13.01.2016 17:00;0000-00002;4.5  14.01.2016 10:00;14.01.2016 17:00;0000-00002;4.7  15.01.2016 10:00;15.01.2016 17:00;0000-00002;3  18.01.2016 10:00;18.01.2016 17:00;0000-00002;5  19.01.2016 10:00;19.01.2016 17:00;0000-00002;4  20.01.2016 10:00;20.01.2016 17:00;0000-00002;4.5  21.01.2016 10:00;21.01.2016 17:00;0000-00002;4.7  22.01.2016 10:00;22.01.2016 17:00;0000-00002;4.5  25.01.2016 10:00;25.01.2016 17:00;0000-00002;5  26.01.2016 10:00;26.01.2016 17:00;0000-00002;4  27.01.2016 10:00;27.01.2016 17:00;0000-00002;4.5  28.01.2016 10:00;28.01.2016 17:00;0000-00002;4.7  29.01.2016 10:00;29.01.2016 17:00;0000-00002;4 |