**Отчёт по тестовому заданию**

1. **Справочники**

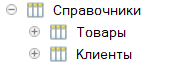


Рисунок – Справочники

* 1. **Товары**

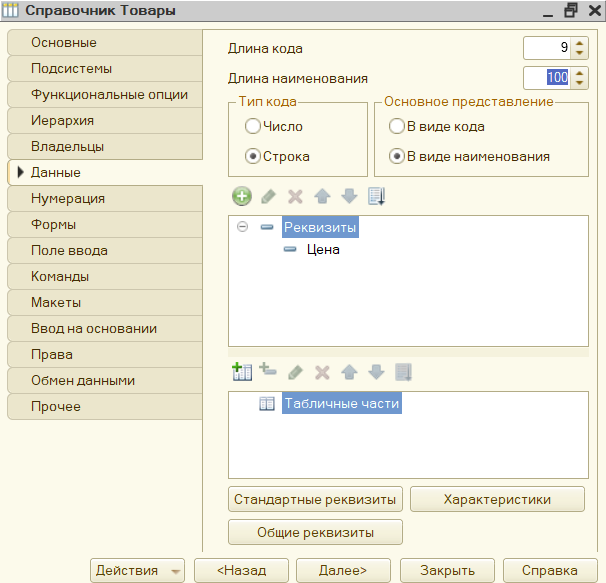


Рисунок – Данные для справочника Товары

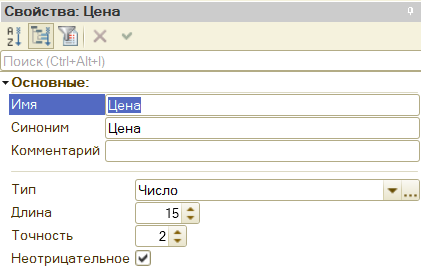


Рисунок – Свойства реквизита “Цена”

Код модуля объекта (Отвечает за проверку уникальности новой записи):

Процедура ПередЗаписью(Отказ)

// Если наименование есть в справочнике, и объект новый, то его нельзя записать

ЕСЛИ Справочники.Товары.НайтиПоНаименованию(ЭтотОбъект.Наименование, Истина) <> Справочники.Товары.ПустаяСсылка() И ЭтотОбъект.ЭтоНовый() ТОГДА

Сообщить("Товар с данным наименованием уже записан.");

Отказ = Истина;

КОНЕЦЕСЛИ;

КонецПроцедуры

Тестирование функционала (При попытке создать новый элемент с существующим наименованием возникнет ошибка):

1. Создание нового элемента с наименованием, которое не встречается в справочнике:

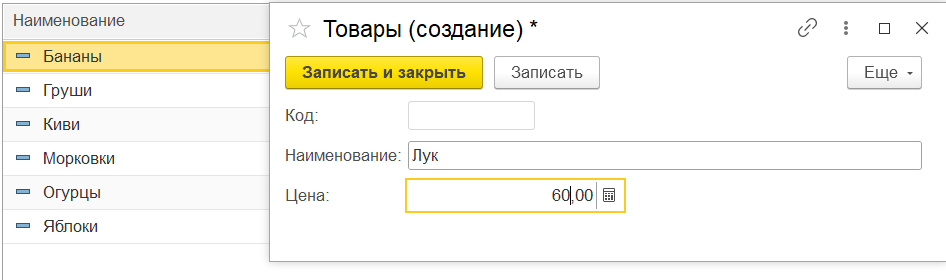


Рисунок – Данные записи и состояние справочника перед записью



Рисунок – Состояние справочника после записи

1. Создание нового элемента с наименованием, которое встречается в справочнике:

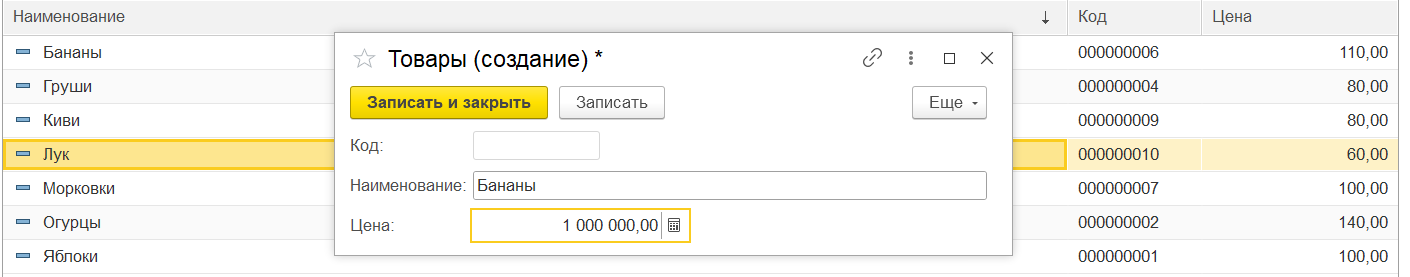


Рисунок – Данные записи и состояние справочника перед записью

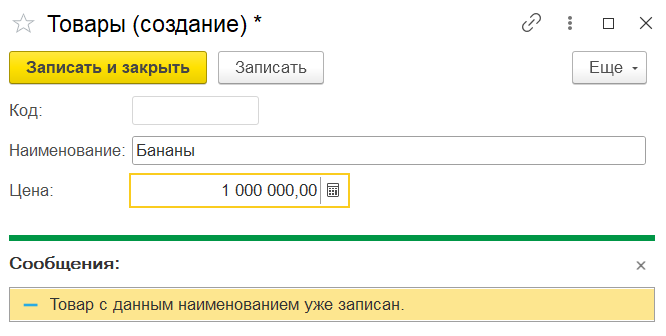


Рисунок – Результат попытки записи

1. Редактирование существующего элемента:

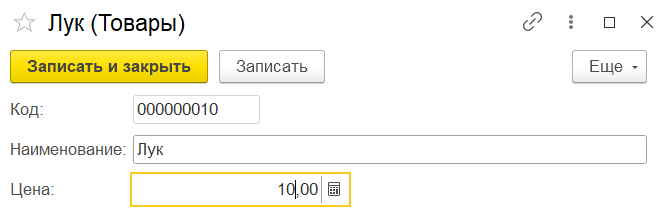


Рисунок – Редактирование элемента



Рисунок – Состояние справочника после редактирования

* 1. **Клиенты**

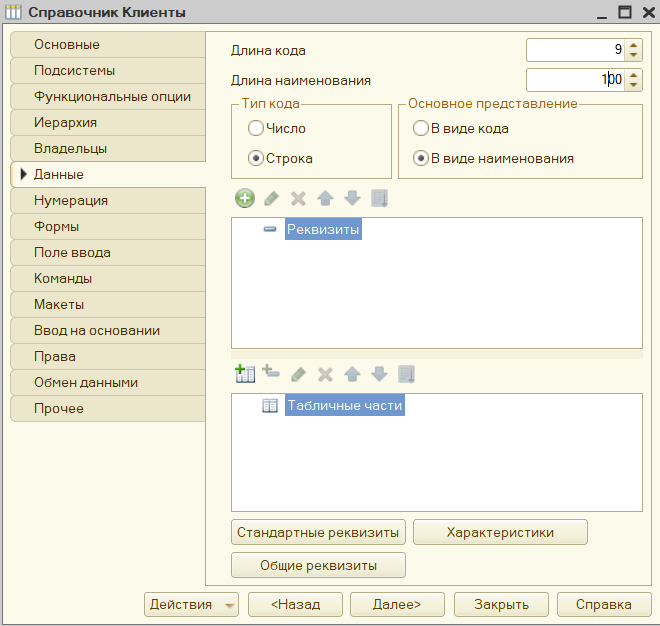


Рисунок – Данные для справочника Клиенты

1. **Документы**

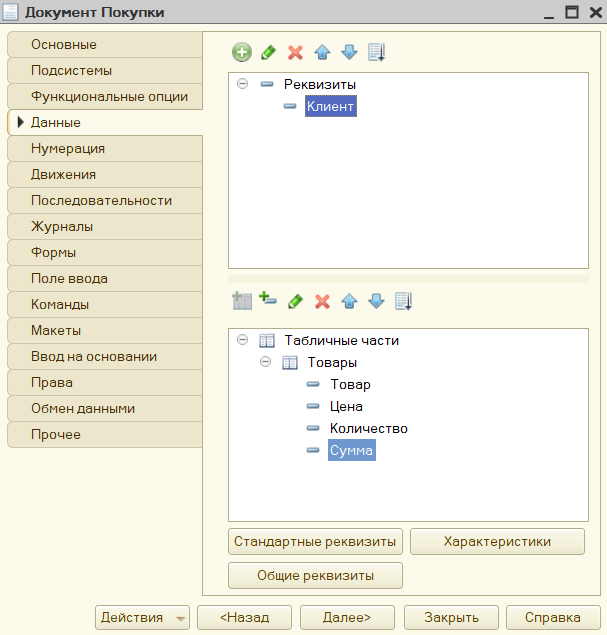


Рисунок – Данные документа Покупки

Типы данных:

Клиент – СправочникСсылка.Клиенты

Товар – СправочникСсылка.Товары

Цена – Число (Длина 15, Точность 2, Неотрицательное)

Количество – Число (Длина 10, точность 0, Неотрицательное)

Сумма – Число (Длина 15, Точность 2, Неотрицательное)

**Код формы элемента:**

//Получение цены на сервере

&НаСервереБезКонтекста

Функция ТоварыТоварПриИзмененииНаСервере(Наим)

Возврат Справочники.Товары.НайтиПоНаименованию(Наим).Цена;

КонецФункции

//Автоматическая подстановка цены и изменение суммы при изменении товара

&НаКлиенте

Процедура ТоварыТоварПриИзменении(Элемент)

ЭтаФорма.Элементы.Товары.текущиеДанные.Цена = ТоварыТоварПриИзмененииНаСервере(ЭтаФорма.Элементы.Товары.текущиеДанные.Товар);

ОбновитьСумму();

КонецПроцедуры

//Метод для изменения суммы

&НаКлиенте

Процедура ОбновитьСумму()

ЭтаФорма.Элементы.Товары.текущиеДанные.Сумма = ЭтаФорма.Элементы.Товары.текущиеДанные.Цена \* ЭтаФорма.Элементы.Товары.текущиеДанные.Количество;

КонецПроцедуры

//Изменение суммы при изменении количества товара

&НаКлиенте

Процедура ТоварыКоличествоПриИзменении(Элемент)

ОбновитьСумму();

КонецПроцедуры

//Изменение константы Печать

&НаСервереБезКонтекста

Процедура РежимПриИзмененииНаСервере(Значение)

Константы.Печать.Установить(Значение);

КонецПроцедуры

//Вызов процедуры для изменения константы Печать при изменении режима

&НаКлиенте

Процедура РежимПриИзменении(Элемент)

РежимПриИзмененииНаСервере(Режим);

КонецПроцедуры

Внешний вид формы:

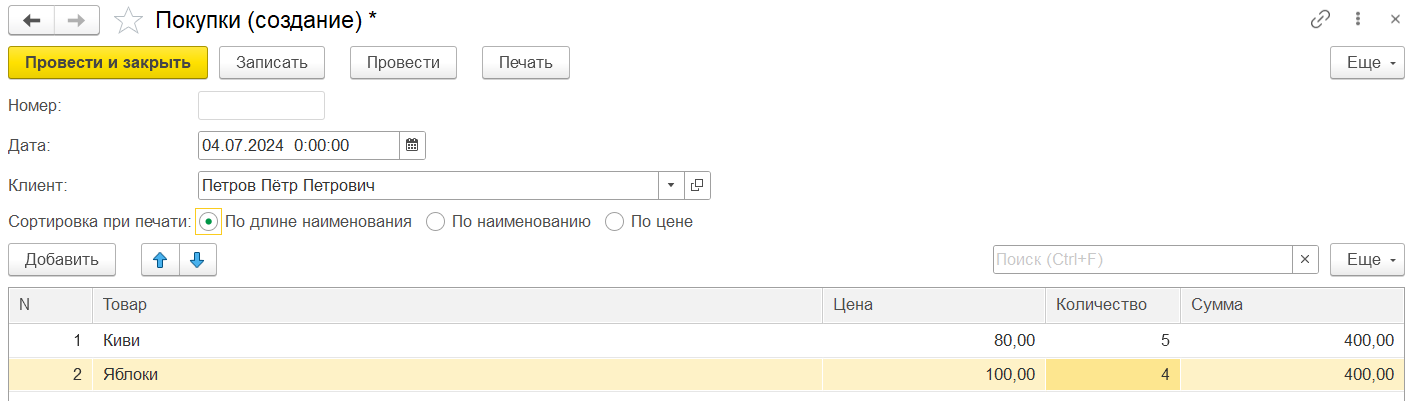


Рисунок – Внешний вид формы

Тестирование автоматической подстановки:

1. Изменение суммы при изменении количества:

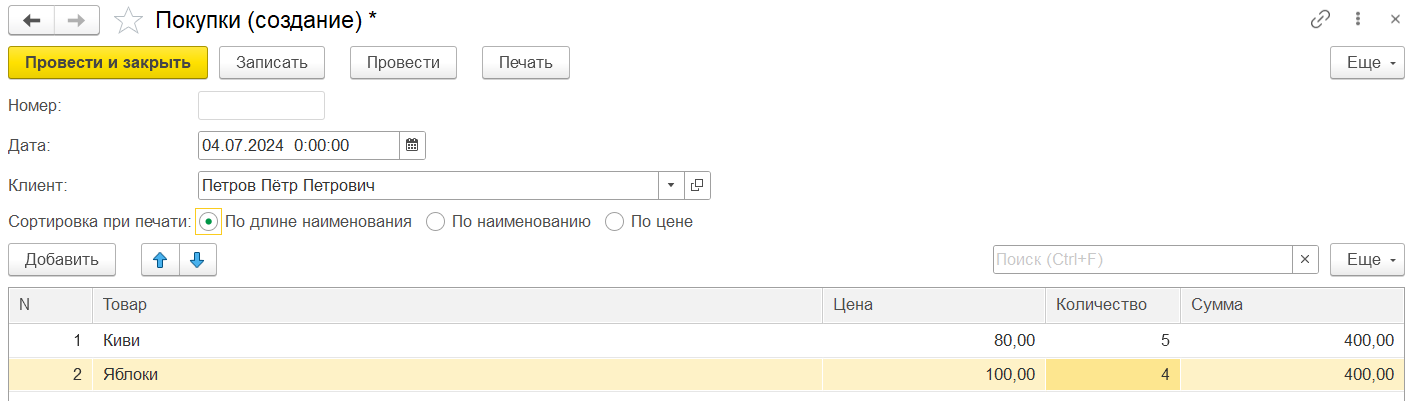


Рисунок – До изменения количества киви

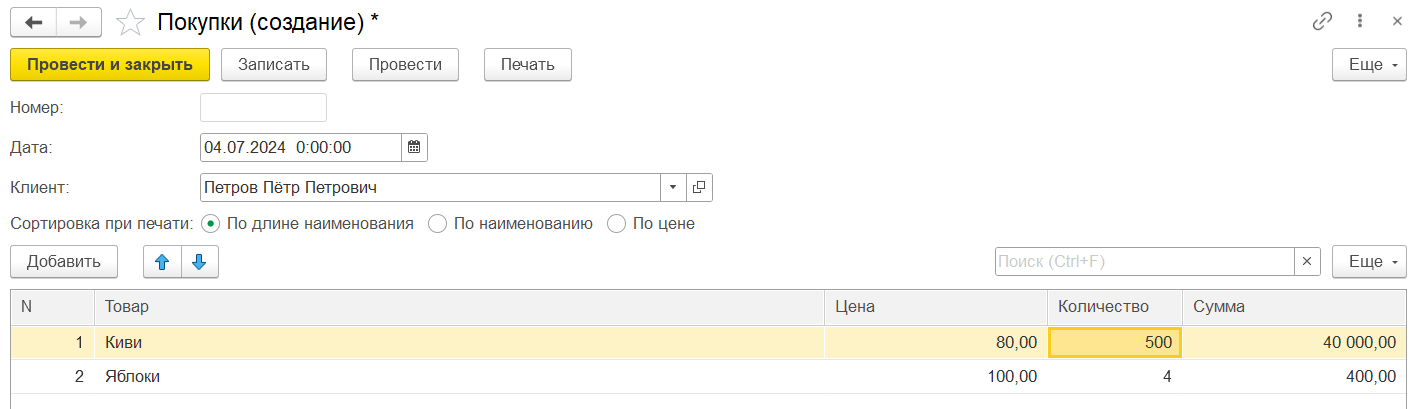


Рисунок – После изменения количества киви

1. Изменение цены и суммы при изменении товара

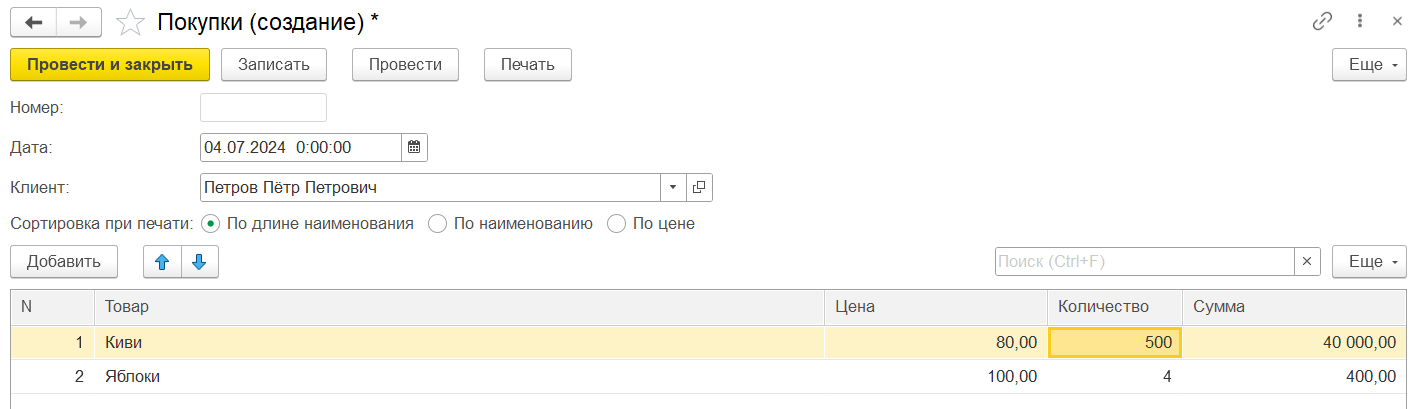


Рисунок – До замены киви на морковки

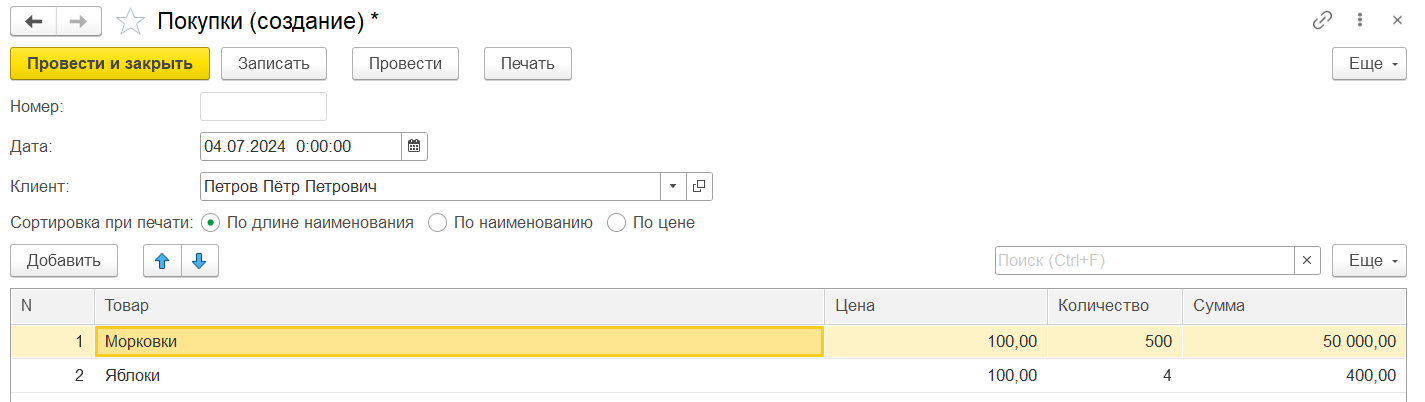


Рисунок – После замены киви на морковки

Цену и сумму пользователь изменить напрямую не может.

Тестирование печатной формы:

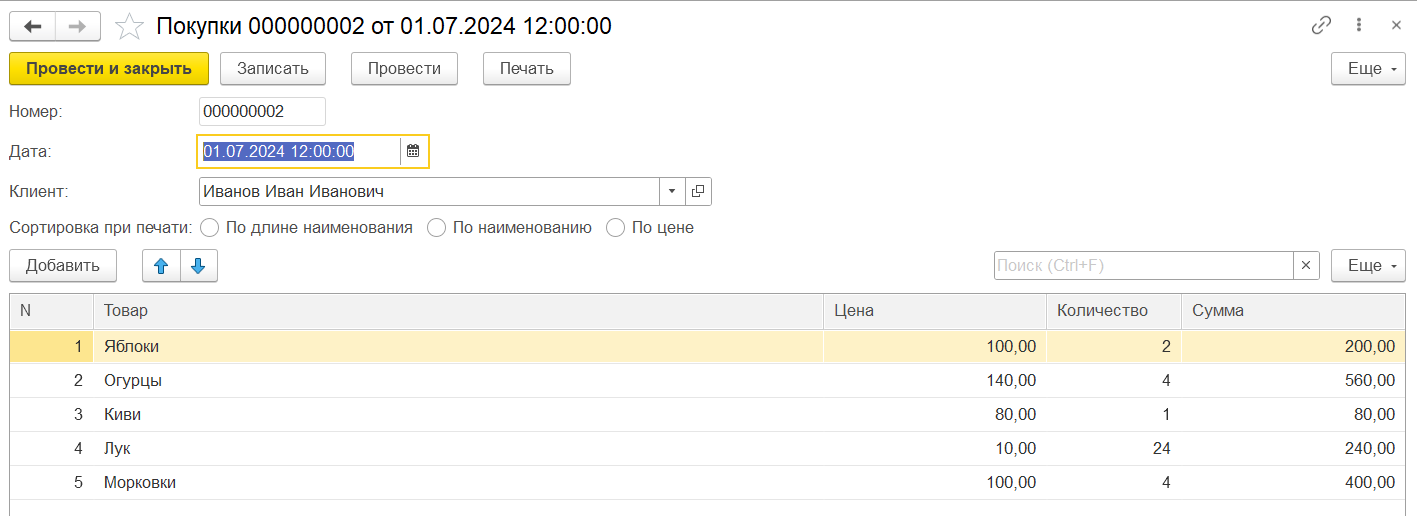


Рисунок – Документ, для которого будет тестироваться печатная форма

1. Открытие с установкой сортировки по длине наименования:

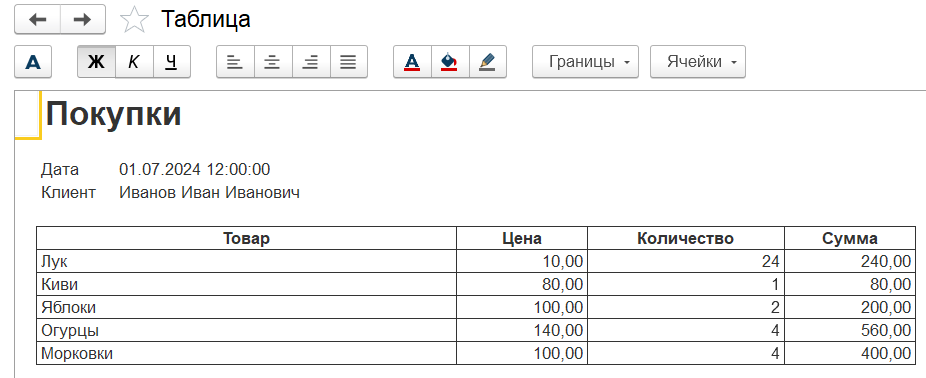


Рисунок – Сортировка по длине наименования

1. Открытие с установкой сортировки по наименованию:

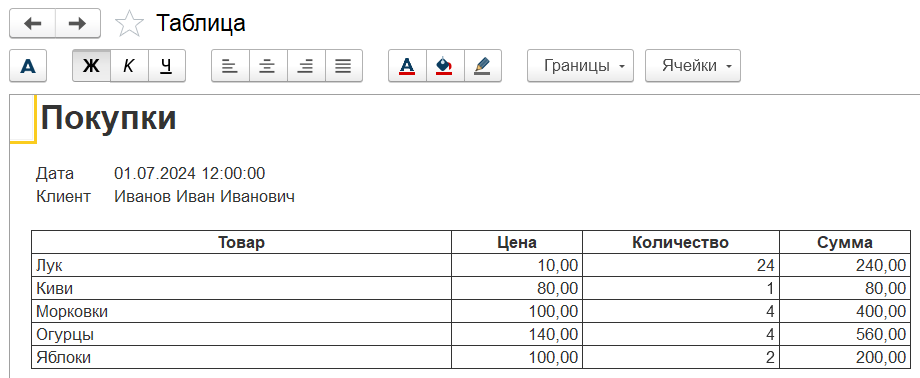


Рисунок – Сортировка по наименованию

1. Открытие с установкой сортировки по цене:

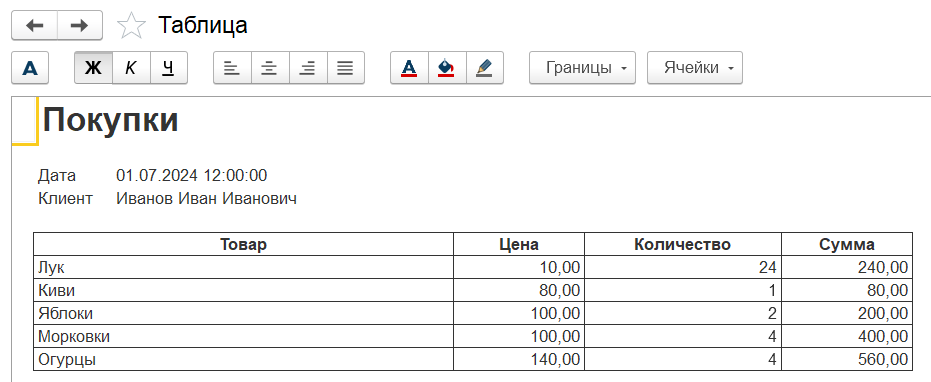


Рисунок – Сортировка по цене

**Движения документа по регистру накопления:**

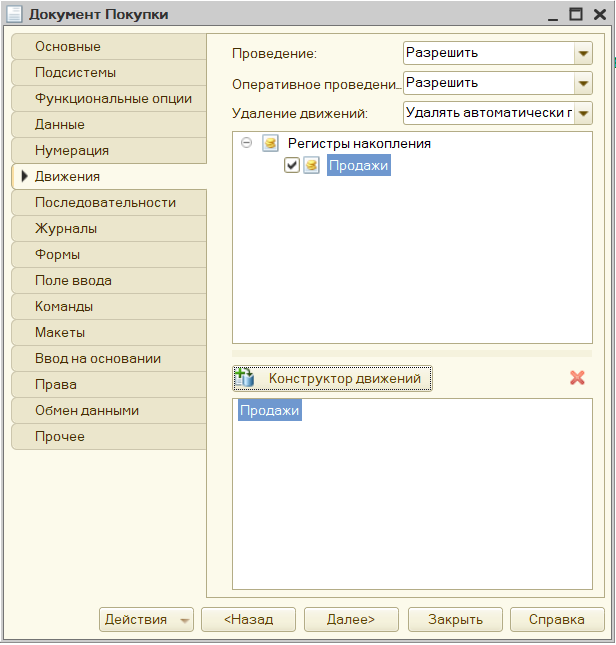


Рисунок – Движения документа

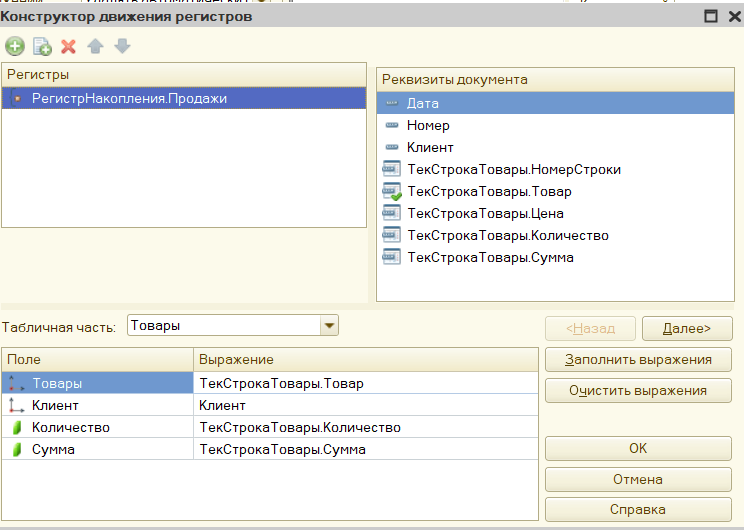


Рисунок – Конструктор движения

1. Регистр накопления

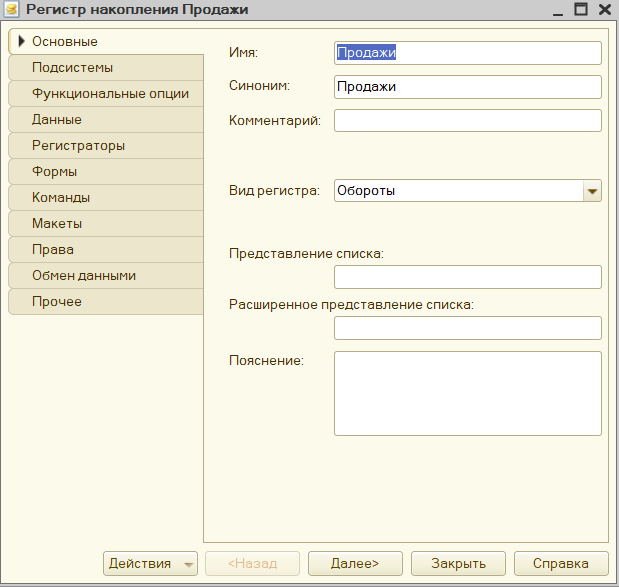


Рисунок – Основные настройки регистра накопления

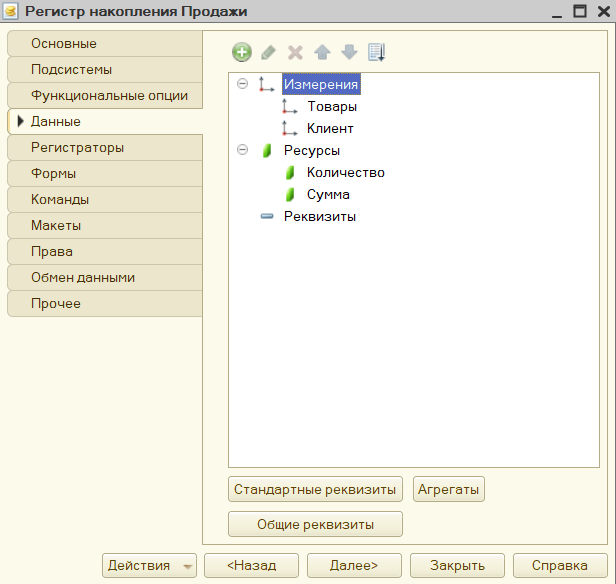


Рисунок – Данные регистра накопления

**Измерения:**



Рисунок – Товары

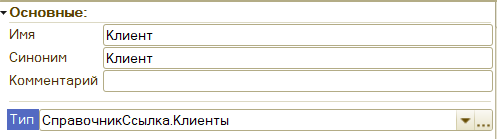


Рисунок – Клиент

Ресурсы:

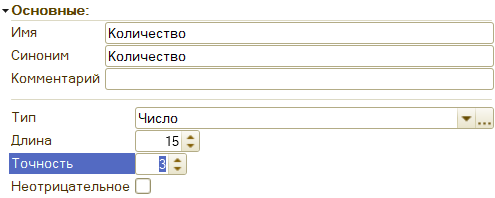


Рисунок – Количество

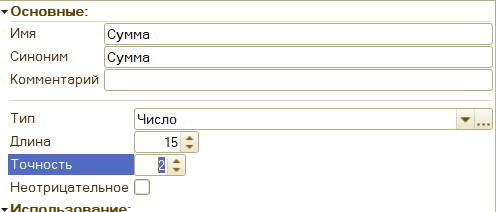


Рисунок – Сумма

1. **Отчёт**

**Запрос для получения данных:**

ВЫБРАТЬ

ПродажиОбороты.Товары КАК Товары,

ПродажиОбороты.Клиент КАК Клиент,

ПродажиОбороты.КоличествоОборот КАК КоличествоОборот,

ПродажиОбороты.СуммаОборот КАК СуммаОборот

ИЗ

РегистрНакопления.Продажи.Обороты КАК ПродажиОбороты

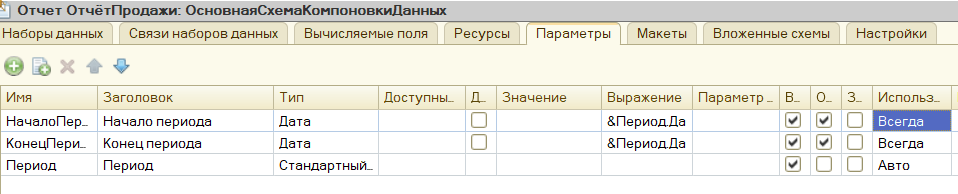


Рисунок – Параметры отчёта

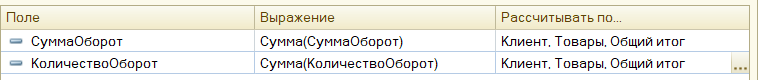


Рисунок – Ресурсы отчёта

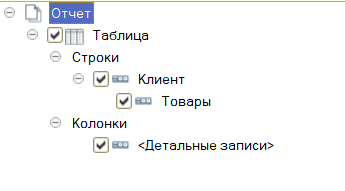


Рисунок – Схема отчёта

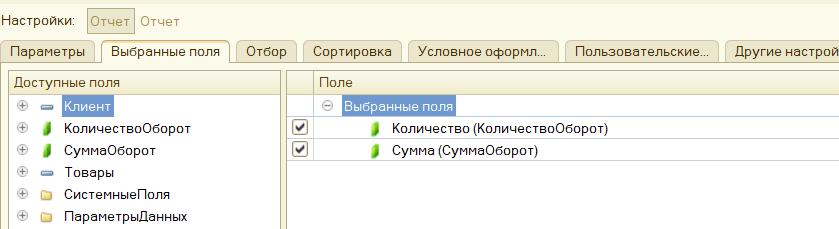


Рисунок – Выбранные поля

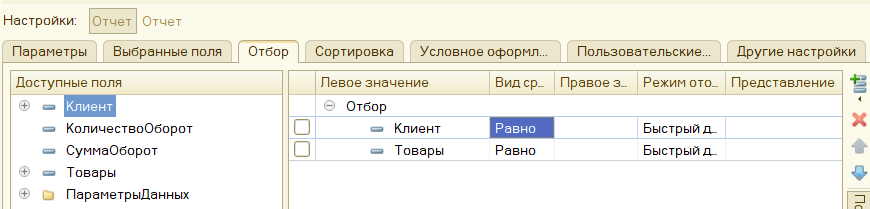


Рисунок – Отбор (Обе строки включены в элементы пользовательских настроек)

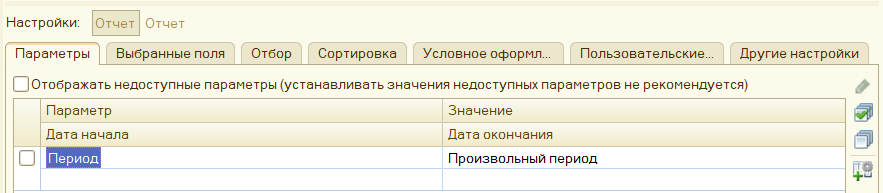


Рисунок – Параметр (Период) (Включён в элементы пользовательских настроек)

Тестирование отчёта:

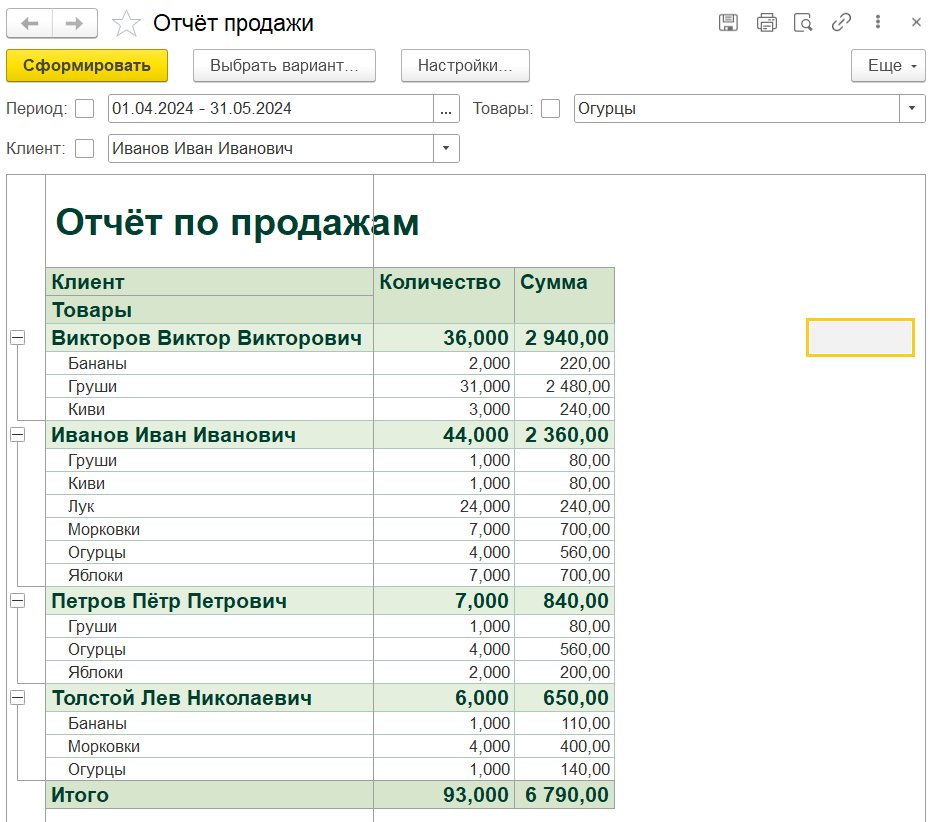


Рисунок – Отчёт без условий отбора

1. Отбор по периоду

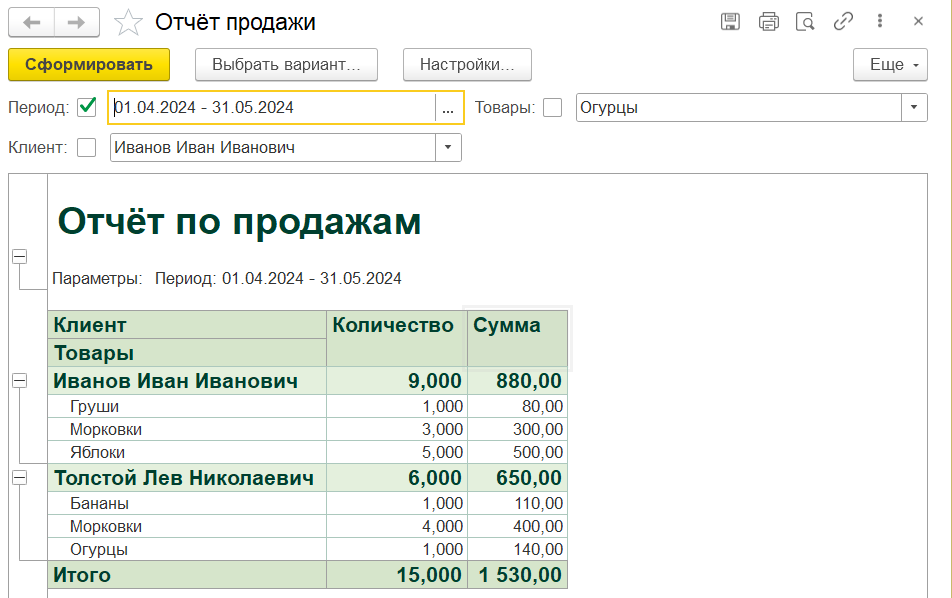


Рисунок – Отчёт по продажам с начала апреля по конец мая

1. Отбор по клиенту

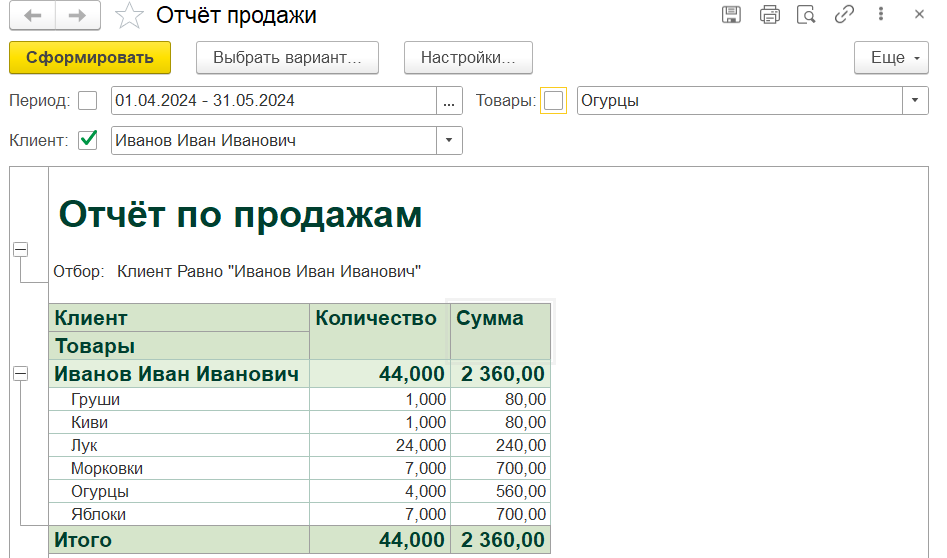


Рисунок – Продажи Иванову Ивану Ивановичу

1. Отбор по товару

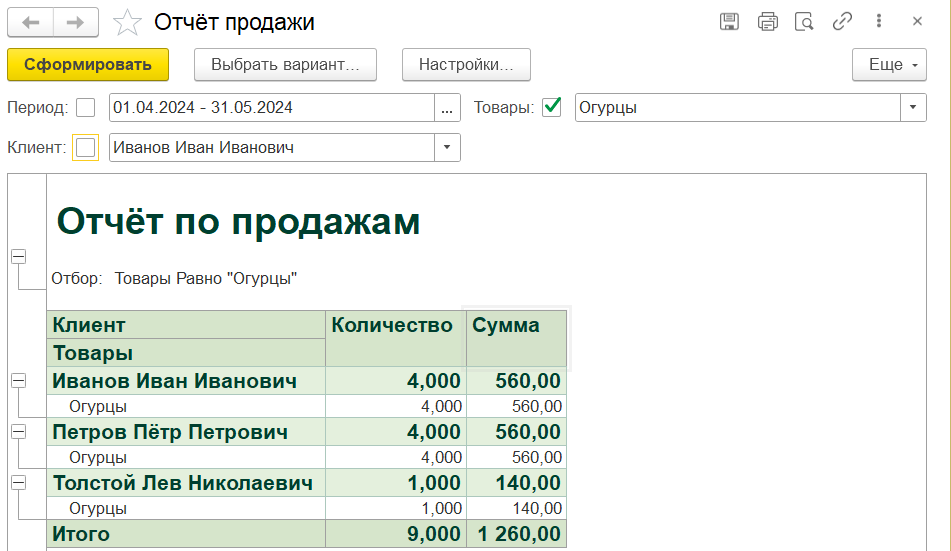


Рисунок – Продажи огурцов по клиентам

1. Отбор по нескольким полям сразу:

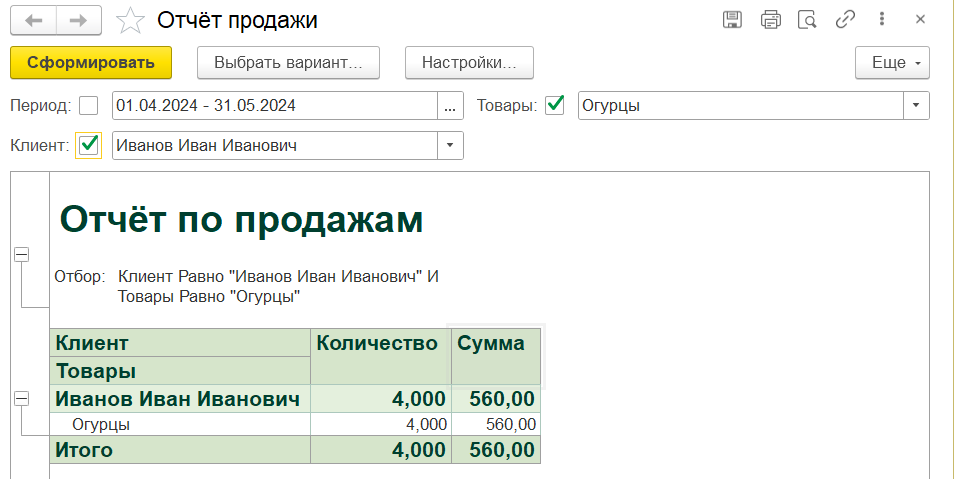


Рисунок – Продажи огурцов Иванову Ивану Ивановичу

Код модуля менеджера документа:

Процедура Печать(ТабДок, Ссылка) Экспорт

//{{\_КОНСТРУКТОР\_ПЕЧАТИ(Печать)

Макет = Документы.Покупки.ПолучитьМакет("Печать");

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Покупки.Дата,

| Покупки.Клиент,

| Покупки.Товары.(

| Товар,

| Цена,

| Количество,

| Сумма

| )

|ИЗ

| Документ.Покупки КАК Покупки

|ГДЕ

| Покупки.Ссылка В (&Ссылка)";

ЕСЛИ Константы.Печать.Получить() = 1 ТОГДА

ЗАПРОС.Текст = ЗАПРОС.Текст + "

|УПОРЯДОЧИТЬ ПО ДлинаСтроки(Покупки.Товары.Товар.Ссылка.Наименование) ВОЗР";

КонецЕсли;

ЕСЛИ Константы.Печать.Получить() = 2 ТОГДА

ЗАПРОС.Текст = ЗАПРОС.Текст + "

|УПОРЯДОЧИТЬ ПО Покупки.Товары.Товар УБЫВ";

КонецЕсли;

ЕСЛИ Константы.Печать.Получить() = 3 ТОГДА

ЗАПРОС.Текст = ЗАПРОС.Текст + "

|УПОРЯДОЧИТЬ ПО Покупки.Товары.Цена ВОЗР";

КонецЕсли;

Запрос.Параметры.Вставить("Ссылка", Ссылка);

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

ОбластьЗаголовок = Макет.ПолучитьОбласть("Заголовок");

Шапка = Макет.ПолучитьОбласть("Шапка");

ОбластьТоварыШапка = Макет.ПолучитьОбласть("ТоварыШапка");

ОбластьТовары = Макет.ПолучитьОбласть("Товары");

ТабДок.Очистить();

ВставлятьРазделительСтраниц = Ложь;

Пока Выборка.Следующий() Цикл

Если ВставлятьРазделительСтраниц Тогда

ТабДок.ВывестиГоризонтальныйРазделительСтраниц();

КонецЕсли;

ТабДок.Вывести(ОбластьЗаголовок);

Шапка.Параметры.Заполнить(Выборка);

ТабДок.Вывести(Шапка, Выборка.Уровень());

ТабДок.Вывести(ОбластьТоварыШапка);

ВыборкаТовары = Выборка.Товары.Выбрать();

Пока ВыборкаТовары.Следующий() Цикл

ОбластьТовары.Параметры.Заполнить(ВыборкаТовары);

ТабДок.Вывести(ОбластьТовары, ВыборкаТовары.Уровень());

КонецЦикла;

ВставлятьРазделительСтраниц = Истина;

КонецЦикла;

//}}

КонецПроцедуры