

00001	Система компоновки данных.
-------	----------------------------

	Система компоновки данных.
1	
2	
3	
4	
5	ЧтениеXML = Новый ЧтениеXML; ЧтениеXML.УстановитьСтроку(ЭлементыФормы.ТекстСхемыКомпоновкиДанных.ПолучитьТекст()); СКД = СериализаторXDTO.ПрочитатьXML(ЧтениеXML, Тип("СхемаКомпоновкиДанных"));
6	{ВЫБРАТЬ Номенклатура, Склад}
7	{ГДЕ Номенклатура.*, Склад } {ГДЕ Документ.Дата >= &ДатаНачала, Документ.Дата <= &ДатаКонца}
8	{ХАРАКТЕРИСТИКИ ТИП(Справочник.Номенклатура) ВИДЫХАРАКТЕРИСТИК (ВЫБРАТЬ ВидыДопСвойств.Ссылка, ВидыДопСвойств.Наименование, ВидыДопСвойств.ТипЗначения ИЗ ПланВидовХарактеристик.ВидыДопСвойств КАК ВидыДопСвойств) ПОЛЕКЛЮЧА Ссылка ПОЛЕИМЕНИ Наименование ПОЛЕТИПАЗНАЧЕНИЯ ТипЗначения ЗНАЧЕНИЯХАРАКТЕРИСТИК РегистрСведений.ДопСвойства ПОЛЕОБЪЕКТА Номенклатура ПОЛЕВИДА ВидСвойства ПОЛЕЗНАЧЕНИЯ Свойство }
9	ВЫБРАТЬ УчетНоменклатурыОбороты.Номенклатура КАК Номенклатура, УчетНоменклатурыОбороты.Склад КАК Склад, УчетНоменклатурыОбороты.КоличествоПриход КАК КоличествоПриход, УчетНоменклатурыОбороты.КоличествоРасход КАК КоличествоРасход ИЗ РегистрНакопления.УчетНоменклатуры.Обороты({&ДатаНачала, &ДатаКонца}, {Номенклатура.*, Склад.*}) КАК УчетНоменклатурыОбороты
10	Процедура КоманднаяПанельРедактораОтчета (Кнопка) Конструктор = Новый КонструкторСхемыКомпоновкиДанных; Конструктор.УстановитьСхему(ПолучитьСхемуКомпоновкиДанных()); Конструктор.Редактировать(ЭтаФорма); КонецПроцедуры Процедура ОбработкаВыбора(ЗначениеВыбора, Источник) Если ТипЗнч(Источник) = Тип("КонструкторСхемыКомпоновкиДанных") Тогда СхемаКомпоновкиДанных = Источник.ПолучитьСхему(); ЗаписьXML = Новый ЗаписьXML; ЗаписьXML.УстановитьСтроку(); СериализаторXDTO.ЗаписатьXML(ЗаписьXML, СхемаКомпоновкиДанных, "dataCompositionScheme", "http://v8.1c.ru/8/data-composition-system/scheme"); ЭлементыФормы.ТекстСхемыКомпоновкиДанных. УстановитьТекст(ЗаписьXML.Заккрыть()); КонецЕсли;

	КонецПроцедуры
<div>11</div>	<div><div><div>ВЫБРАТЬ ПриходТовара.Дата, ПриходТовара.Номер, ПриходТовара.Поставщик, ПриходТовара.Склад, ПриходТовара.Товары.(НомерСтроки, Товар, Цена, Количество, Сумма) {ВЫБРАТЬ Дата, Номер, Поставщик.*, Склад.*, Товары.(НомерСтроки, Товар.*, Цена, Количество, Сумма)} ИЗ Документ.ПриходТовара КАК ПриходТовара {ГДЕ ПриходТовара.Дата, ПриходТовара.Номер, ПриходТовара.Поставщик.*, ПриходТовара.Склад.*, ПриходТовара.Товары.(НомерСтроки, Товар.*, Цена, Количество, Сумма)} 1 1</div></div></div>
<div>12</div>	"Литерал ""в кавычках"""
<div>13</div>	10.5 200
<div>14</div>	<div>// Шестое января 1975 года ДАТАВРЕМЯ(1975, 1, 06) // Второе декабря 2006 года, 23 часа 56 минут 57 секунд ДАТАВРЕМЯ(2006, 12, 2, 23, 56, 57)</div>
<div>16</div>	ЗНАЧЕНИЕ(ВидСчета.Активный)
<div>17</div>	Номенклатура.Артикул Продажи.СуммаОборот
<div>18</div>	&Контрагент &ДатаНачала
<div>19</div>	-Продажи.Количество
<div>20</div>	+Продажи.Количество
<div>21</div>	ОстИОбрт.НачальныйОстаток — ОстИОбрт.КонечныйОстаток ОстИОбрт.НачальныйОстаток — 100

	400 — 357
2 2	ОстИОбрт.НачальныйОстаток + ОстИОбрт.Оборот ОстИОбрт.НачальныйОстаток + 100 400 + 357
2 3	Номенклатура.Цена * 1.2 2 * 3.14
2 4	Номенклатура.Цена / 1.2 2 / 3.14
2 5	Номенклатура.Цена % 1.2 2 % 3.14
2 6	Номенклатура.Артикул + ": " + Номенклатура.Наименование
2 7	"%АБВ[0-9][абвг]_абв%" СПЕЦСИМВОЛ "\"
2 8	Продажи.Контрагент = Продажи.НоменклатураОсновнойПоставщик
2 9	Продажи.Контрагент <> Продажи.НоменклатураОсновнойПоставщик
3 0	ПродажиТекущие.Сумма < ПродажиПрошлые.Сумма
3 1	ПродажиТекущие.Сумма > ПродажиПрошлые.Сумма
3 2	ПродажиТекущие.Сумма <= ПродажиПрошлые.Сумма
3 3	Номенклатура В (&Товар1, &Товар2)
3 4	Продажи.Контрагент В Контрагенты
3 5	Продажи.Контрагент ЕСТЬ NULL
3 6	НЕ Документ.Грузополучатель = Документ.Грузоотправитель
3 7	Документ.Грузополучатель = Документ.Грузоотправитель И Документ.Грузополучатель = &Контрагент
3 8	Документ.Грузополучатель = Документ.Грузоотправитель ИЛИ Документ.Грузополучатель = &Контрагент
3 9	СУММА(Продажи.СуммаОборот)
4 0	КОЛИЧЕСТВО(Продажи.Контрагент)
4 1	КОЛИЧЕСТВО(Различные Продажи.Контрагент)
4 2	МАКСИМУМ(Остатки.Количество)
4 3	МИНИМУМ(Остатки.Количество)
4 4	СРЕДНЕЕ(Остатки.Количество)
4 5	ФункцииОтчетов.СтандартноеОтклонение(Массив(Сумма))
4 6	ФункцииОтчетов.ТаблицуЗначенийВСтроку(ТаблицаЗначений(Склад КАК Склад, КоличествоОстаток КАК Остаток))
4 7	Свернуть(ТаблицаЗначений(НомерТелефона, Адрес) ,"НомерТелефона");
4 8	ПолучитьЧасть(Свернуть(ТаблицаЗначений(НомерТелефона, Адрес) ,"НомерТелефона"),"НомерТелефона");
4 9	Упорядочить(ТаблицаЗначений(НомерТелефона, Адрес, ДатаЗвонка),"ДатаЗвонка Убыв");
5 0	СоединитьСтроки(ТаблицаЗначений(НомерТелефона, Адрес));
5	// Рассчитать ABC классификацию.

```

1 Функция АВСКлассификация(Данные) Экспорт
Перем ТаблицаЗначений;

Если Данные = Null Тогда
    Возврат Null;
КонецЕсли;

Если ТипЗнч(Данные) <> Тип("ДанныеГрупповойОбработкиКомпоновкиДанных") Тогда
    Сообщить(ТипЗнч(Данные));
    ВызватьИсключение "В функцию АВСКлассификация() можно передавать только объект типа
ДанныеГрупповойОбработкиКомпоновкиДанных";
КонецЕсли;

Если Не Данные.ВременныеДанныеОбработки.Свойство("АВСКлассификацияТаблицаЗначений",
ТаблицаЗначений) Тогда
    // выполним расчет классификации один раз при первом вызове
    // а потом запоем выполненный расчет и дальше будем этот
    // расчет использовать
    ТаблицаЗначений = Данные.Данные.Скопировать();
    ТаблицаЗначений.Колонки.Добавить("Номер", Новый ОписаниеТипов("Число"));
    Номер = 0;
    ОбщаяСумма = 0;
    Для Каждого СтрокаТаблицыЗначений Из ТаблицаЗначений Цикл
        СтрокаТаблицыЗначений.Номер = Номер;
        Номер = Номер + 1;

        Если СтрокаТаблицыЗначений[0] <> NULL Тогда

            ОбщаяСумма = ОбщаяСумма + СтрокаТаблицыЗначений[0];

        КонецЕсли;
    КонецЦикла;

    ТаблицаЗначений.Сортировать(ТаблицаЗначений.Колонки[0].Имя + " Убыв");
    ТаблицаЗначений.Индексы.Добавить("Номер");
    НакопленнаяСумма = 0;
    ИндексКлассаА = Неопределено;
    ИндексКлассаВ = Неопределено;
    Для Каждого СтрокаТаблицыЗначений Из ТаблицаЗначений Цикл

        Если СтрокаТаблицыЗначений[0] <> NULL Тогда

            НакопленнаяСумма = НакопленнаяСумма + СтрокаТаблицыЗначений[0];

        КонецЕсли;

        Если ОбщаяСумма = 0 Тогда

            Процент = 1;

        Иначе

            Процент = НакопленнаяСумма / ОбщаяСумма;

        КонецЕсли;

        Если Процент > 0.75 Тогда
            Если ИндексКлассаА = Неопределено Тогда
                ИндексКлассаА = ТаблицаЗначений.Индекс(СтрокаТаблицыЗначений);
            ИначеЕсли Процент > 0.90 Тогда
                Если ИндексКлассаВ = Неопределено Тогда
                    ИндексКлассаВ = ТаблицаЗначений.Индекс(СтрокаТаблицыЗначений);
                КонецЕсли;
            Прервать;
        КонецЕсли;
    КонецЦикла;

```

	<p>Данные.ВременныеДанныеОбработки.Вставить("АВСКлассификацияТаблицаЗначений", ТаблицаЗначений);</p> <p>Данные.ВременныеДанныеОбработки.Вставить("АВСКлассификацияИндексКлассаА", ИндексКлассаА);</p> <p>Данные.ВременныеДанныеОбработки.Вставить("АВСКлассификацияИндексКлассаВ", ИндексКлассаВ);</p> <p>КонецЕсли;</p> <p>Если Данные.ТекущийЭлемент = Неопределено Тогда</p> <p>// Итог по группировке.</p> <p>Возврат Null;</p> <p>Иначе</p> <p>Строка = ТаблицаЗначений.Найти(Данные.Данные.Индекс(Данные.ТекущийЭлемент), "Номер");</p> <p>Если Строка = Неопределено Тогда</p> <p>Возврат Null;</p> <p>Иначе</p> <p>Индекс = ТаблицаЗначений.Индекс(Строка);</p> <p>Если Индекс <= Данные.ВременныеДанныеОбработки.АВСКлассификацияИндексКлассаА Тогда</p> <p>Возврат 1;</p> <p>ИначеЕсли Индекс <= Данные.ВременныеДанныеОбработки.АВСКлассификацияИндексКлассаВ Тогда</p> <p>Возврат 2;</p> <p>Иначе</p> <p>Возврат 3;</p> <p>КонецЕсли;</p> <p>КонецЕсли;</p> <p>КонецЕсли;</p> <p>КонецФункции</p>
5 2	АВСКлассификация(ГрупповаяОбработка("Сумма(СуммаОборот)"))
5 3	ВЫБОР Когда Сумма > 1000 Тогда Сумма Иначе 0 Конец
5 4	Сумма(Продажи.СуммаОборот) / ВЫЧИСЛИТЬ("Сумма(Продажи.СуммаОборот)", "ОбщийИтог")
5 5	Сумма(Продажи.СуммаОборот)/Вычислить("Сумма(Продажи.СуммаОборот)", "ОбщийИтог")
5 6	ВычислитьВыражение("Сумма(СуммаОборот)", , , "Первая", "Текущая")
5 7	ВычислитьВыражение("Курс", , , "Предыдущая")
5 8	ВычислитьВыражениеСГруппировкойМассив("Сумма(СуммаОборот)", "Контрагент")
5 9	СоединитьСтроки(Массив(Упорядочить(ВычислитьВыражениеСГруппировкойТаблицаЗначений("Представле ние(Сумма(НаборДанных.СуммаОборот)),Сумма(НаборДанных.СуммаОборот)", "НаборДанных.Контрагент"), "2"))))
6 0	Максимум(ВычислитьВыражениеСГруппировкойМассив ("Сумма(СуммаОборот)", "Контрагент"));
6 1	ВычислитьВыражениеСГруппировкойТаблицаЗначений ("Контрагент, Сумма(СуммаОборот)", "Контрагент")
6 2	СоединитьСтроки(ПолучитьЧасть(Упорядочить(ВычислитьВыражениеСГруппировкойТаблицаЗначений ("НаборДанных.Контрагент, НаборДанных.КонтрагентПредставление, Сумма(НаборДанных.СуммаОборот),

	Представление(НаборДанных.СуммаОборот), НаборДанных.ПолеУпорядочивание", "НаборДанных.Контрагент"), "5, 1, 3"), "2, 4"))
6 3	ВычислитьВыражениеСГруппировкойТаблицаЗначений ("Контрагент КАК Контрагент, Сумма(СуммаОборот) Как ОбъемПродаж", "Контрагент")
6 4	УРОВЕНЬ()
6 5	УРОВЕНЬВГРУППИРОВКЕ()
6 6	НОМЕРПОПОРЯДКУ()
6 7	НОМЕРПОПОРЯДКУВГРУППИРОВКЕ()
6 8	ФОРМАТ(РасходныеНакладные.СуммаДок, "ЧДЦ=2")
6 9	НАЧАЛОПЕРИОДА(ДатаВремя(2002, 10, 12, 10, 15, 34), "Месяц")
7 0	01.10.2002 0:00:00
7 1	КОНЕЦПЕРИОДА(ДатаВремя(2002, 10, 12, 10, 15, 34), "Неделя")
7 2	13.10.2002 23:59:59
7 3	ДОБАВИТЬКДАТЕ(ДатаВремя(2002, 10, 12, 10, 15, 34), "Месяц", 1)
7 4	12.11.2002 10:15:34
7 5	РАЗНОСТЬДАТ(ДАТАВРЕМЯ(2002, 10, 12, 10, 15, 34), ДАТАВРЕМЯ(2002, 10, 14, 9, 18, 06), "ДЕНЬ")
7 6	2 ТЕКУЩАЯДАТА()
7 7	ПОДСТРОКА(Контрагенты.Адрес, 1, 4)
7 8	ДЛИНАСТРОКИ(Контрагенты.Адрес)
7 9	ГОД(РасхНакл.Дата)
8 0	КВАРТАЛ(РасхНакл.Дата)
8 1	МЕСЯЦ(РасхНакл.Дата)
8 2	ДЕНЬГОДА(РасхНакл.Дата)
8 3	ДЕНЬ(РасхНакл.Дата)
8 4	НЕДЕЛЯ(РасхНакл.Дата)
8 5	ДЕНЬНЕДЕЛИ(РасхНакл.Дата)
8 6	ЧАС(РасхНакл.Дата)
8 7	МИНУТА(РасхНакл.Дата)
8 8	СЕКУНДА(РасхНакл.Дата)
8 9	ВЫРАЗИТЬ(Данные.Реквизит1, "Число(10,3)")
9 0	ЕСТЬNULL(Сумма(Продажи.СуммаОборот), 0)
9 1	Представление(Контрагент)
9 2	Строка(ДатаПродажи)

9 3	СокращенноеНаименование(Докум.Ссылка, Докум.Дата, Докум.Номер)
9 4	<p>СхемаКомпоновкиДанных = ПолучитьСхемуКомпоновкиДанных(); ИсполняемыеНастройки = ПолучитьИсполняемыеНастройки(); КомпоновщикМакета = Новый КомпоновщикМакетаКомпоновкиДанных; МакетКомпоновки = КомпоновщикМакета.Выполнить(СхемаКомпоновкиДанных, ИсполняемыеНастройки);</p>
9 5	<p>ЭлементыФормы.ТДРезультатТабличныйДокумент.Очистить();</p> <p>МакетКД = ПолучитьМакетКомпоновки();</p> <p>ПроцессорКД = Новый ПроцессорКомпоновкиДанных; ПроцессорКД.Инициализировать(МакетКД);</p> <p>ПроцессорВывода = Новый ПроцессорВыводаРезультатаКомпоновкиДанныхВТабличныйДокумент; ПроцессорВывода.УстановитьДокумент(ЭлементыФормы.ТДРезультатТабличныйДокумент); ПроцессорВывода.НачатьВывод();</p> <p>Пока Истина Цикл</p> <p>ЭлементРезультатаКД = ПроцессорВывода.Следующий(); Если ЭлементРезультатаКД = Неопределено Тогда</p> <p>Прервать;</p> <p>КонецЕсли;</p> <p>ПроцессорВывода.ВывестиЭлемент(ЭлементРезультатаКД);</p> <p>КонецЦикла;</p> <p>ПроцессорВывода.ЗакончитьВывод();</p>
9 6	<p>ЗаписьXML = Новый ЗаписьXML; ЗаписьXML.УстановитьСтроку(); ЗаписьXML.ЗаписатьНачалоЭлемента("result");</p> <p>МакетКомпоновкиДанных = ПолучитьМакетКомпоновки();</p> <p>ПроцессорКомпоновкиДанных = Новый ПроцессорКомпоновкиДанных; ПроцессорКомпоновкиДанных.Инициализировать(МакетКомпоновкиДанных);</p> <p>Пока Истина Цикл</p> <p>ЭлементРезультатаКомпоновкиДанных = ПроцессорКомпоновкиДанных.Следующий(); Если ЭлементРезультатаКомпоновкиДанных = Неопределено Тогда</p> <p>Прервать;</p> <p>КонецЕсли;</p> <p>СериализаторXDTO.ЗаписатьXML(ЗаписьXML, ЭлементРезультатаКомпоновкиДанных, "item", "http://v8.1c.ru/8/data-composition-system/result");</p> <p>КонецЦикла;</p> <p>ЗаписьXML.ЗаписатьКонецЭлемента();</p>
9 7	<p>ЭлементыФормы.ТДРезультатТабличныйДокумент.Очистить();</p> <p>МакетКомпоновкиДанных = ПолучитьМакетКомпоновки();</p> <p>ПроцессорКомпоновкиДанных = Новый ПроцессорКомпоновкиДанных; ПроцессорКомпоновкиДанных.Инициализировать(МакетКомпоновкиДанных);</p>

	<p>ПроцессорВывода = Новый ПроцессорВыводаРезультатаКомпоновкиДанныхВТабличныйДокумент; ПроцессорВывода.УстановитьДокумент(ЭлементыФормы.ТДРезультатТабличныйДокумент); ПроцессорВывода.НачатьВывод();</p> <p>Пока Истина Цикл</p> <p>ЭлементРезультатаКомпоновкиДанных = ПроцессорКомпоновкиДанных.Следующий(); Если ЭлементРезультатаКомпоновкиДанных = Неопределено Тогда</p> <p>Прервать;</p> <p>КонецЕсли;</p> <p>ПроцессорВывода.ВывестиЭлемент(ЭлементРезультатаКомпоновкиДанных);</p> <p>КонецЦикла;</p> <p>ПроцессорВывода.ЗакончитьВывод();</p>
9 8	ПроцессорВывода.Вывести(ПроцессорКомпоновкиДанных);
	<p>КомпоновщикМакета = Новый КомпоновщикМакетаКомпоновкиДанных; МакетКомпоновкиДанных = КомпоновщикМакета.Выполнить(СхемаКомпоновкиДанных,КомпоновщикНастроек.Настройки, , , Тип("ГенераторМакетаКомпоновкиДанныхДляКоллекцииЗначений"));</p> <p>ПроцессорКомпоновкиДанных = Новый ПроцессорКомпоновкиДанных; ПроцессорКомпоновкиДанных.Инициализировать(МакетКомпоновкиДанных);</p> <p>ПроцессорВывода = Новый ПроцессорВыводаРезультатаКомпоновкиДанныхВКоллекциюЗначений; ПроцессорВывода.УстановитьОбъект(ДеревоРезультата); ПроцессорВывода.НачатьВывод();</p> <p>Пока Истина Цикл</p> <p>ЭлементРезультатаКомпоновкиДанных = ПроцессорКомпоновкиДанных.Следующий(); Если ЭлементРезультатаКомпоновкиДанных = Неопределено Тогда</p> <p>Прервать;</p> <p>КонецЕсли;</p> <p>ПроцессорВывода.ВывестиЭлемент(ЭлементРезультатаКомпоновкиДанных);</p> <p>КонецЦикла;</p>
9 9	ПроцессорВывода.ЗакончитьВывод();
1 0 0	<p>ВЫБРАТЬ Номенклатура.Ссылка КАК Ссылка, Номенклатура.Родитель КАК Родитель ИЗ Справочник.Номенклатура КАК Номенклатура ГДЕ Номенклатура.Ссылка В(&Ссылки)</p>
	<p>ВЫБРАТЬ Номенклатура.Ссылка, Номенклатура.Родитель, Номенклатура.Код, Номенклатура.Наименование, Номенклатура.ЭтоГруппа ИЗ Справочник.Номенклатура КАК Номенклатура ГДЕ Номенклатура.Родитель В (&Родители)</p>
	<p>ВЫБРАТЬ Дети.Ссылка</p>

	ИЗ Справочник.Дети как Дети
	ВЫБРАТЬ ДетиСотрудников.Ребенок КАК Ссылка, ДетиСотрудников.РодительРебенка КАК Родитель ИЗ РегистрСведений.ДетиСотрудников КАК ДетиСотрудников ГДЕ ДетиСотрудников.Ребенок В(&Ссылка) ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ Сотрудники.Ссылка, NULL ИЗ Справочник.Сотрудники КАК Сотрудники ГДЕ Сотрудники.Ссылка В(&Ссылка)
	СкомпоноватьРезультат(РежимКомпоновкиРезультата.Непосредственно);
	ВЫБРАТЬ Док.Ссылка.Дата, Док.Ссылка.Номер, Док.Номенклатура, Остатки.Остаток ИЗ Документ.РасходнаяНакладная.Состав КАК Док ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.УчетНоменклатуры.Остатки Остатки ПО Док.Номенклатура = Остатки.Номенклатура