

ОТЛАДКА

Отладка, это такой специальный режим, который запускается в конфигураторе, и который позволяет нам посмотреть в режиме реального времени, что сейчас выполняет программа, какие она использует значения для расчёта.

Мы с вами в прошлом модуле остановились на том, что узнали о существовании событий элементов форм. И о том, что внутри этих событий программисты пишут код. Собственно говоря, описания всех событий всех элементов всех форм это и есть весь код нашей конфигурации.

Но как понять, что вообще здесь написано, как здесь ориентироваться, если мы не программисты, мы не знаем язык программирования.

Во-первых, хочу заметить, что вам повезло больше чем пользователям других программ - потому что как вы уже могли заметить - здесь всё написано по-русски. И мало того, нужно помнить, что это всё писалось обычными людьми, программистами, которые и придумывали названия этих функций, процедур и переменных. Соответственно часто уже по названию какой-то функции отлично понятно для чего она предназначена

Например, функция, которую мы с вами смотрели - **КонтрагентПриИзменении**. Уже по названию нам понятно что это то что происходит при изменении контрагента. Программист её не назвал **ФункцияНомер435**, хотя он имел возможность это сделать, он мог назвать её абсолютно как угодно.

Дело в том что, существуют некие стандарты программирования, и одним из них является требование о том, что код написанный программистом должен быть понятен другим программистам. О том, что функции и переменные должны называться соответственно тому, для чего они и предназначены.

ЧТО МОЖЕТ РАСПОЛАГАТЬСЯ В МОДУЛЕ.

Здесь много очень много всего написано, но по сути весь этот код состоит из 3 составляющих:

1. Переменных
2. Процедур
3. Функций

Которые по-разному называются и выполняют разные вещи.

Чем отличаются друг от друга Переменные, Процедуры и Функции.

Рассмотрим на примере внешней обработки, которая складывает 2 числа что такое переменная процедура и функция.

&НаКлиенте

Процедура Число1ПриИзменении(Элемент)

Объект.Результат = СложитьЧисла();

КонецПроцедуры

&НаКлиенте

Процедура Число2ПриИзменении(Элемент)

Объект.Результат = СложитьЧисла();

КонецПроцедуры

&НаКлиенте

Функция СложитьЧисла()

Сумма = Объект.Число1+Объект.Число2;

Возврат Сумма;

КонецФункции

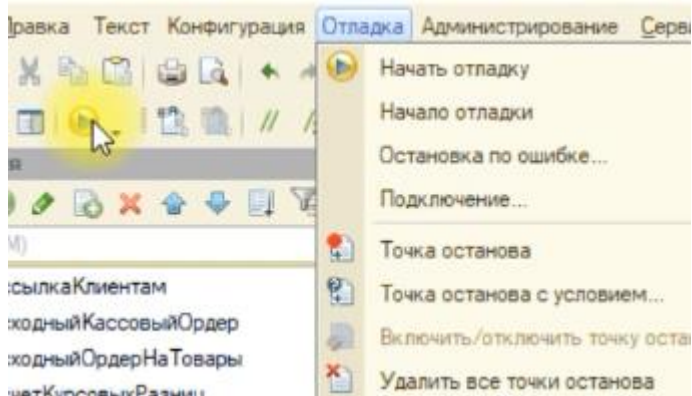
Что у нас в итоге получилось. **2 процедуры**, которые что то выполняют, **1 функция**, которая что-то выполняет и возвращает результат туда, откуда её вызвали и **1 переменная**, которая временно что-то в себе хранит.

Вот и весь код каждого модуля в конфигурации, состоит из процедур функций и переменных. Которые по-разному называются и выполняют разные задачи.

ЗАПУСК ОТЛАДКИ

Как посмотреть в отладке, в режиме реального времени как работает наша обработка, какие она использует значения, что куда передаётся и возвращается.

Как запустить отладку. Когда мы находимся в конфигураторе нажимаем Отладка/Начать отладку. Либо кнопку со стрелочкой:

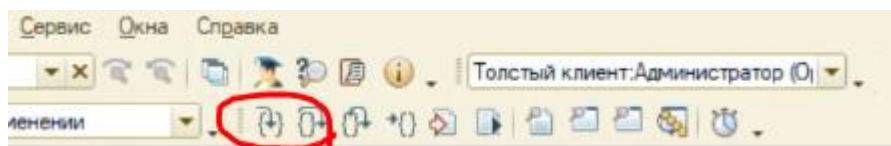


И у нас запускается база в пользовательском режиме. И теперь мы можем поставить точку, на выполнении которой программа должна остановиться.

Установим точку останова (так она называется) дважды щелкаем около нужной строки в процедуре При изменении числа 1 и посмотрим что будет когда мы выполним это событие в пользовательском режиме.

Изменим число 1. Идём меняем число 1 вписываем в него 5, и у нас выполнение программы останавливается на той точке, которую мы поставили в конфигураторе. И мы можем посмотреть в моменте, что содержится в переменных и реквизитах.

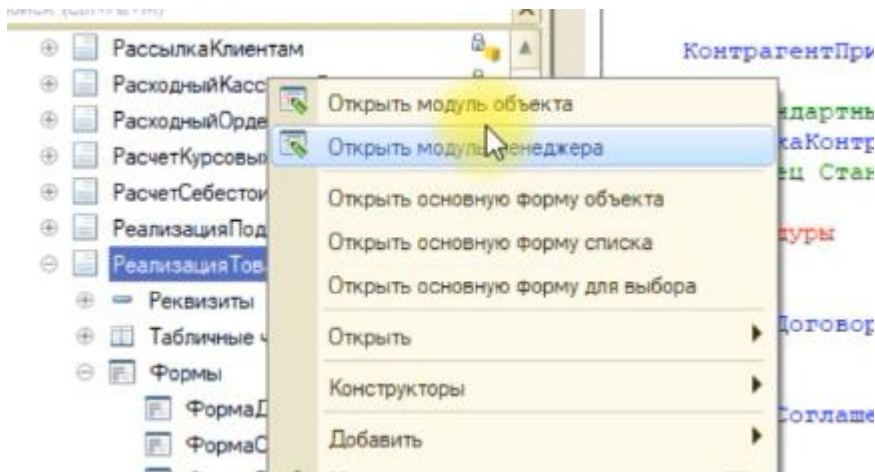
Для того чтобы выполнять программу построчно у вас есть 2 кнопки на панели отладки – **Шагнуть через** - позволяет просто шагнуть на след строку, а кнопка **Шагнуть в** - позволяет провалиться внутрь функции или процедуры, если в данной строке есть функция или процедура.



Если мы просто нажмем **Шагнуть через** которая просто выполняет эту строку, то эта функция выполнится и мы не увидим подробностей, она просто вернёт нам результат. А если мы нажмем кнопку **Шагнуть в** то мы провалимся внутрь функции и можем подробно рассмотреть что она там делает.

Отладкой конечно можно пользоваться не только во внешних обработках, но и в любых формах в вашей конфигурации.

Еще у каждого объекта помимо процедур и функций в модуле формы есть ещё процедуры в **модуле объекта и модуле менеджера.**



И они могут вызываться из модуля формы, когда вы будете шагать по строкам в отладке формы и провалитесь в какую-то процедуру вас может перекинуть ещё и сюда, но потом по окончании этой процедуры вы вернетесь обратно в модуль формы.

И ещё есть **общие модули** в них тоже написаны различные процедуры и тоже вас может перекинуть в какой то общий модуль, он будет автоматически открываться если вы провалитесь в процедуру которая написана в нём.

ПРИМЕР

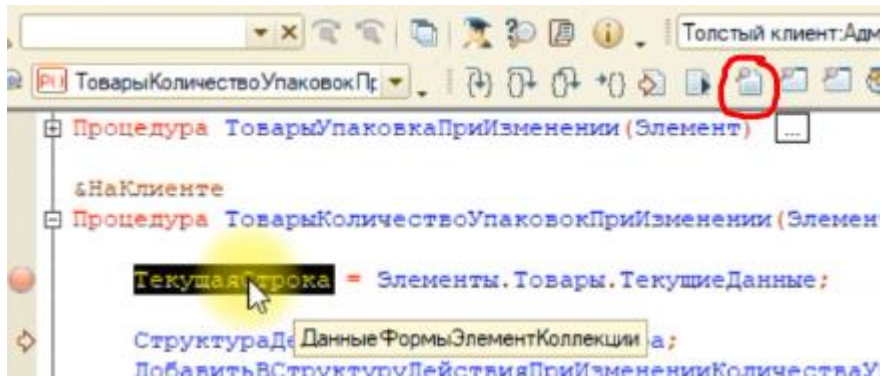
Давайте вернёмся к форме нашей реализации и попробуем посмотреть в отладке какое-то событие реквизита формы.

Например, **при изменении количества в табличной части.**

1. Ставим также точку останова и запускаем отладку.
2. Идём выполняем это событие, меняем количество в табличной части.
3. Останавливаемся, шагаем.

Допустим нам не понятно, как получается такая сумма в строке табличной части, попытаемся это расследовать.

Чтобы посмотреть, что находится внутри переменной есть кнопка - **Вычислить выражение.**



Пока мы не достигнем своей цели проваливаемся в интересующие нас процедуры кнопкой **Шагнуть в**. И шагаем по строкам кнопкой **Шагнуть через**.

```
Процедура ПересчитатьСуммуВСтрокеТЧ(ТекущаяСтрока, СтруктураДействий, КэшированныеЗначения) Экспорт
    ИмяКоличества = Неопределено;

    Если СтруктураДействий.Свойство("ПересчитатьСумму", ИмяКоличества) Тогда
        Если НЕ ЗначениеЗаполнено(ИмяКоличества) Тогда
            ИмяКоличества = "КоличествоУпаковок"
        КонецЕсли;
        Если ТекущаяСтрока <> Неопределено Тогда
            ТекущаяСтрока.Сумма = Округ(ТекущаяСтрока.Цена * ТекущаяСтрока[ИмяКоличества], 2, РежимОкругления.Округ15как20);
        КонецЕсли;
        КонецЕсли;
        1 000
КонецПроцедуры
```

Вот мы нашли - как рассчитывается Сумма.

Нажимаем - **Отладка / Удалить точки останова. Продолжить Отладку**.