**Язык 1С: общие конструкции встроенного языка**

1С - весь программный код пишется на русском языке! Цветовое оформление в примерах полностью соответствует раскраске программного кода в среде 1С: Предприятие 8.

|  |  |
| --- | --- |
| **КОНСТРУКЦИЯ ЯЗЫКА 1С** | **ПРИМЕЧАНИЕ** |
| //Это комментарий | Так оформляются комментарии. Комментарий - это подсказки, пометки разработчика, которые помогают разобраться или вспомнить логику работы программного кода. Каждая новая строка комментария должна начинаться с символов //. |
|  |  |
| Перем ФИО; | Явное объявление переменной. **ФИО** - имя переменной. |
|  |  |
| А = 3; | Переменную можно не объявлять явно. При первом присвоении значения система создает данную переменную. |
|  |  |
| ЭтоЧисло = 23.5+12\*2; | Переменной **ЭтоЧисло**присваивается числовое значение.  С данными числового типа можно выполнять арифметические операции: сложение, вычитание, умножение и деление. В качестве разделителя целой и дробной части используется точка! |
|  |  |
| А = -0.123; | Числовые значения могут быть отрицательными. |
|  |  |
| А = (2+7) \* 3;   // в результате А=27    В = 2 \* ((2+7) - 4); // в результате В = 10 | Приоритет выполнения арифметических операций:  **А =** Сначала вычисляем **2 + 7**, затем полученное значение умножаем на **3**;  **В =** Сначала вычисляем **2 + 7**, затем от полученного значения отнимаем **4**, затем полученное значение умножаем на **2**. |
|  |  |
| ЭтоСтрока = "Пугачева"; | Переменной **ЭтоСтрока** присваиваем строковое значение. Значение строкового типа пишется в кавычках. |
|  |  |
| ФИО = "Пугачёва" + " " + "Алла" + " "+"Борисовна";  // результат: ФИО = "Пугачёва Алла Борисовна" | Сложение строк. Символ " " мы прибавляем, чтобы между фамилией, именем и отчеством были пробелы. Вторая строчка кода это просто комментарий. |
|  |  |
| ДатаОтчета = '2013.01.01'; | Переменная, которая хранит дату. Значение типа Дата записывается в одинарных кавычках. |
|  |  |
| ЧислоСекунд = '2013.01.02' – '2013.01.01'; // ЧислоСекунд = 86400 | Даты можно вычитать одну из другой. В результате получим разницу между датами, измеренную в секундах. В сутках **86 400**секунд (60 сек \* 60 мин \* 24 ч). |
|  | |
| НоваяДата = '2013.01.01'+86400; //НоваяДата = '2013.01.02' | К дате можно прибавлять и вычитать **число**. В результате к дате либо прибавится, либо отнимется число секунд. |
|  |  |
| Процедура РассчитатьЦену (Товар) КонецПроцедуры | Простая процедура. Между словами Процедура и КонецПроцедуры записывается текст процедуры. |
|  |  |
| Функция РассчитатьНалог(Сотрудник, НекаяДата)    Возврат Налог; КонецФункции | Функция должна возвращать результат в место ее вызова. |
|  |  |
| Если Доход > 20000 Тогда     Результат = "Жить можно"; Иначе     Результат = "Так жить нельзя!"; КонецЕсли; | Простое условие. После слова КонецЕсли должна быть точка с запятой, потому что так заканчивается оператор Если. |
|  |  |
| Результат = ?(Доход > 20000, "Жить можно ", "Так  жить нельзя!"); | Сокращенное Если. Краткая запись предыдущего простого условия. |
|  |  |
| Если Доход > 20000 Тогда     Результат = "Жить можно";  ИначеЕсли Доход > 10000 Тогда     Результат = "Плохо";  Иначе     Результат = "Так жить нельзя!"; КонецЕсли; | Множественное условие. Если первое условие не выполняется, то проверяется второе. Если ни одно из условий не выполняется то выполняется блок Иначе. |
|  |  |
| Если (Доход > 20000) И (КодКатегории = 2) Тогда КонецЕсли; | Составное логическое выражение. |
|  |  |
| Пока Номер <= 15 Цикл КонецЦикла; | Простой цикл Пока (с неизвестным числом повторений). После слова КонецЦикла должна быть точка с запятой, потому что так заканчивается оператор Пока. |
|  |  |
| Для Номер = 1 По 15 Цикл КонецЦикла; | Простой цикл Для (цикл с известным числом повторений). |
|  |  |
| Для каждого  СтрокаТаблицы Из Таблицы Цикл  КонецЦикла; | Еще одна разновидность цикла. Оператор цикла Для каждого предназначен для циклического обхода коллекций значений (табличных частей справочников, документов и т.д.). При каждой итерации цикла возвращается новый элемент коллекции. Обход осуществляется до тех пор, пока не будут перебраны все элементы коллекции. |
|  |  |
| Пока <условие> Цикл    Если <условие> Тогда        Продолжить;    КонецЕсли; КонецЦикла; | Оператор Продолжить передает управление в начало цикла. |
|  |  |
| Пока <условие> Цикл    Если <условие> Тогда        Прервать;    КонецЕсли; КонецЦикла; | Оператор Прервать производит досрочный выход из цикла. Управление передается на операторы после цикла. |