Кофигурация

**Информационная база**

1. 1с – добавить – создать новую информационную базу – создание информационной базы без конфигурации – добавление конфиг базы: название тип расположения на данном компе или компе в локальной сети – каталог инф базы по умолчанию – дальше:

Вариант аутентификации: выбирать автоматически

Скорость соединения: обычная

Дополнительные параметры запуска: пусто

Основной режим запуска: выбирать автоматически

Версия 1с предприятия: 8.3

Разрядность: пусто

1. Появляется новая база, можно работать в двух режимах: предприятие (где работают пользователи) и конфигуратора (для разработчиков и аналитиков)
2. Работа в конфигураторе:

Открыть окно конфигурации, откроется дерево объектов конфигурации

**Работа в конфигураторе:**

Открыть окно конфигурации (в меню третий рад, первый значок) или через конфигурация – открыть конфигурацию (верхний ряд, конфигурация). Открося дерево конфигурации в котором есть объекты конфигурации

В меню сервис параметры модули можно настроить шрифт текст подсветку: сделать атвозамена – включить с подсказкой

, а ещё выделение – текущий идентификатор и поставить фон, отличный от белого

Програмный модуль – контейнер в котором хранится программный код

**Модуль приложения** модуль сеансамодуль внешнего соединения

Оператор – фрагмент кода до ;

Вывод на экран:

Сообщить(«фраза»); Сообщить(Переменная);

Или

Сообщение = Новый СообщениеПользователю;

Сообщение.Текст = «пам пам пам»;

Сообщение.Сообщить();

Код выполнится когда будет загружаться 1с предприятие

Сохранияем конфиг **f7**

**(ctrl + f1) - справка**

обновляем конфиг или сервис – 1с предприятие **(ctrl + f5)**

еще есть 1с edt

**Переменная**

переменная – контейнер для хранения данных, хранится в оперативной памяти компьютера

1 способ. Явный. В оперативной памяти резервируется определен область и ей дают имя, но там ничего н ехранится

**Перем ИмяПеременной;** ( должна начинаться с \_ либо с буквы, не должна содержать пробел)

2. способ(неявный) **ИмяПеременной = сто;**

Оператор присваивания работает справа налево

Комментарий: //

Многострочный коммент – тоже самое ИЛИ выделить нужные строки **ctrl + slash**

**УСЛОВНЫЕ ОПЕРАТОРЫ**

|  |  |  |
| --- | --- | --- |
| Если <Условие> Тогда  Тут код;  ИначеЕсли  Тут код;  Иначе  Тут код;  КонецЕсли; | **Короткая форма:**  ?(<Условие>, <ЕслиУсловиеИстина>, <ЕслиУсловиеЛожь>)  А = ?(б > c, б - с, с - б) | |
| **Операции сравнения** | **Логические операции** |
| **= -** равно  **> , >=**  - больше (или равно)  **< , <= -** меньше (или равно)  **<> -** не равно | **И**  **ИЛИ**  **НЕ** |

ЧислоА = 7;

ЧислоБ = 5;

ЧислоА = (присваивание (всегда самое левое равно)) ЧислоБ = (сравнение)5; (истина)

Сообщить (ЧислоА); //выведет ДА

**ЦИКЛИЧЕСКИЙ ОПЕРАТОР**

1. **Цикл по условию**

Пока <Условие> Цикл

Тут код;

КонецЦикла;

1. **Цикл по счетчику**

Для Счетчик=<НачальноеЗначение> по <КонечноеЗначение> Цикл

Тут код;

КонецЦикла;

1. **Цикл по коллекции**

Для КАЖДОГО ЭлементКоллекции ИЗ =<Коллекция> Цикл

Тут код;

КонецЦикла;

//Создание массива:

МассивЧисел = Новый Массив;

Для ТекЧисло=1 По 100 Цикл

МассивЧисиел.Добавить(ТекЧисло);

КонецЦикла;

//Заполнение массива

Для каждого ТекущееЧисло Из МассивЧисел Цикл

СуммаЧисел = СуммаЧисел + ТекущееЧисло;

КонецЦикла;

**Пропуск итерации**

//Сумма четных чисел

Для каждого ТекущееЧисло Из МассивЧисел Цикл

Если ТекущееЧисло%2 = 0 Тогда

Продолжить;

КонецЕсли;

СуммаЧетЧисел = СуммаЧетЧисел + ТекущееЧисло;

КонецЦикла;

**Прерывание цикла**

//прервать цикл как только сумма будет больше 1000

Для Число=1 По 100 Цикл

СуммаЧисел = СуммаЧисел + Число;

Если СуммаЧисел >= 1000 Тогда

Прервать;

КонецЕсли;

КонецЦикла;

**ПРИМИТИВНЫЕ ТИПЫ ДАННЫХ**

Типизация неявная. Переменная может быть переопределена по типу

ПеременнаяЧисло = 5;

ПеременнаяЧисло = «пять»;

ПеременнаяЧисло = Новый Массив;

**Примитивные типы:**

* Число // +, - , \*, /, %
* «Строка» // +, когда надо добавить кавычки в строку: «оао “” пум пум “”»
* ‘Дата’ // ‘**ГГГГММДД***ЧЧММСС*’, определены + (учитываем в секундах) и -
* Булево // ИСТИНА (1), ЛОЖЬ (0). пользователь видит ДА / НЕТ. Лог.оп. **НЕ**
* Неопределенно// описать пустое значение неизвестного типа на уровне языка
* Тип // в переменной храним описание типа, исп-ся для усл.опер-ра

ПеременнаяНеизвестногоТипа = «тест»;

Если ТипЗнач(ПеременнаяНеизвестногоТипа) = Тип(«Строка»)Тогда …

* Null // описать пустое значение неизвестного типа на уровне БД

**Процедуры и функции**

**Процедура (типа воид)** больше для структурирования кода

Процедура ИмяПроцедуры(сюда можно передавать переменную)

Пум пум пум;

КонецПроцедуры

Процедура не выполнится, пока ее не вызовут:

ИмяПроцедуры();

**Функция (возвращает значение)**

Функция СуммаЧисел(Параметр1, Параметр2)

СуммаЧисел = Параметр1 + Параметр2

Возврат СуммаЧисел;

КонецФцнкции

СуммаЧисел(19,20)

**Обязательные и необязательные параметры**

Функция СуммаЧисел(Параметр1, Параметр2, Параметр3 = 0)

То есть теперь можно делать так СуммаЧисел(19,20,21) или СуммаЧисел(19,20)

**Структура программного модуля:**

1. Раздел объявления переменных
2. Раздел процедур и функций
3. Раздел основной программы

**Справка f1**

**Синтакс-помощник** (тут есть документация для функций для всех типов данных и куча полезный штук)

**Коллекции**

А = Новый <Тип коллекции>

Позволяют работать с несколькими значениями

Как обращаться к элементам:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Ключ типа **число** | значение |  | Ключ типа **«строка»** | значение |  | Ключ типа ‘**дата’** | значение |
| 0 | 5 |  | «первый» | 5 |  | 01.05.2020 | 5 |
| 1 | 8 |  | «второй» | 8 |  | 02.05.2020 | 8 |
| 2 | 9 |  | «третий» | 9 |  | 03.05.2020 | 9 |
| … | … |  | … | … |  | … | … |
| N - 1 | 114 |  | «последний» | 114 |  | 10.05.2020 | 114 |
|  |  |  |  |  |  |  |  |
| A[0] | |  | A[“Первое число”] | |  | A[‘20200501’] | |

Универсальные(хранят данные в момент выполнения алгоритма) и специализированные (прикладные объекты в 1с типа справочник или документ, хранят данные постоянно) коллекции

**Универсальные коллекции:**

// тут мы создали внешняя обработка – универсальная коллекция

// И добавляем Форму обработки

// На форме обработки создаем группу страницы и туда добавляем 1ую страницу «группа массив», заголовок «массив»

// Добавляем команду «работа с массивом»

// Создаем обработчик действия команд

1. **Массив**

Доступ через числовой индекс: Массив[0]

**Процедура** РаботаСМассивом(Команда)

1. Как ***создать*** массив

МассивСотрудников = Новый Массив; // динамический массив

МассивСотрудников = Новый Массив(2); //фиксированныймассив на 2 элемента

МассивСотрудников = Новый Массив(2,2); // многомерный массив

1. Как ***добавить*** элемент в массив

//Добавить – добавляет в массив новый элемент

//Вставить – вставляет в массив значение по указанному индексу

МассивСотрудников.Добавить(«Иванов»); // 0

МассивСотрудников.Добавить(«Петров»); // 1

МассивСотрудников.Добавить(«Николаев»); // 2

1. Как получить ***количество элементов*** в массиве (работает для всех коллекций)

КоличествоЭлементов = МассивСотрудников.Количество();

1. Как ***найти*** **значение** в массиве

ИндексПетрова = МассивСотрудников.Найти(«Петров»); // вернет 0

ИндексВасильев = МассивСотрудников.Найти(«Васильев»); //вернет неопределенно

1. Как ***получить*** ***значение*** массива (2 способа)

//1 способ

Петров = МассивСотрудников[1];

//2 способ

Петров = МассивСотрудников.Получить(1);

1. ***Обход*** всех элементов массива

//1 способ – через счетчик по индексу

Для Счетчик = 0 По МассивСотрудников.ВГраница() Цикл

// МассивСотрудников.ВГраница() находит индекс последнего элемента

Сообщить(МассивСотрудников(Счетчик));

КонецЦикла;

//2 способ

Для каждого ЭлементМассива из МассивСотрудников Цикл

//ЭлементМассива – САМ ЭЛЕМЕНТ

Сообщить(МассивСотрудников);

КонецЦикла;

1. Как ***удалить*** элемент массива

Николаев = МассивСотрудников.Найти(«Николаев»);

МассивСотрудников.Удалить(Николаев);

1. Как ***очистить*** массив

МассивСотрудников.Очистить();

**КонецПроцедуры**

1. **Структура**

Доступ к значениям через строковый идентификатор

**Процедура** РаботаССтруктурой(Команда)

1. Как **создать** структуру

//1 способ - без ключей и значений

СтруктураДанных = Новый Структура;

//2 способ - перечислить ключи при создании

СтруктураДанных = Новый(«фио,датарождения,адреспрописки,номерпаспорта»);

//3 способ - перечислить ключи и значения при создании

СтруктураДанных = Новый(«фио,датарождения,адреспрописки,номерпаспорта», «Иванов», «19900108», «г москв ул какая-то дом такойто», «МС1214141»); //пропущенные данные типа 19900108 будут как неопределено

1. Как **вставить** элемент в структуру

СтруктураДанных.Вставить(«ТабельныйНомер», «ТТ-14»);

1. Как получить ***количество элементов*** в структуре (работает для всех коллекций)

КоличествоЭлементов = СтруктураДанных.Количество();

1. Как **найти** значение в структуре

ДатаРождения = Неопределено;

СтруктураДанных.Свойство(«ДатаРождения», ДатаРождения); // возвращает булево

1. Как **получить** значение структуры (3 способа)

//1 способ

ФИО = СтруктураДанных(«ФИО»);

//2 способ

ФИО = СтруктураДанных.ФИО;

//3 способ

ФИО = Неопределено;

СтруктураДанных.Свойство(«ФИО», ФИО);

1. **Обход** всех элементов структуры

Для каждого ЭлементСтруктуры из СтруктураДанных Цикл

Сообщить(ЭлементСтруктуры.Ключ + « = » + ЭлементСтруктуры.Значение);

КонецЦикла;

1. Как **удалить** элемент структуры

СтруктураДанных.Удалить(«НомерПаспорта»);

1. Как **очистить** структуру

СтруктураДанных.Очистить();

**КонецПроцедуры**

Перевести данные из массива в структуру:

МассивЧисел = Новый Массив;

Для Число=1 По 5000 Цикл

МассивЧисел.Добавить(Число);

КонецЦикла;

СтрЧисел = Новый Структура;

Для каждого ЗначениеМассива из МассивЧисел Цикл

ИмяКлюча = «Число\_» + Формат(МассивЧисел.Найти(ЗначениеМассива), «ЧГ=0»);

Попытка

СтруктураЧисел.Вставить(ИмяКлюча, ЗначениеМассива);

Исключение

Сообщение = Новый СообщениеПользователю;

Сообщение.Текст = ЗначениеМассива;

Сообщение.Сообщить();

//с 1001 значения начнется ошибка, потому что ключ получится некорректным: Число\_1 000 из-за разделителя, для этого необходимо использовать форматирование

КонецЦикла;

1. **Соответствие**

Доступ к значениям, используя ключ произвольного типа (ссылка, число, строка, дата и т.д.)

1. Как **создать** соответствие

СоответствиеРазмеров = Новый Соответствие;

1. Как **вставить** элемент в соответствии

//СоответствиеРазмеров.Вставить(<Ключ>, <Значение>);

СоответствиеРазмеров.Вставить(42, «XS»);

СоответствиеРазмеров.Вставить(44, «S»);

СоответствиеРазмеров.Вставить(46, «M»);

СоответствиеРазмеров.Вставить(48, «L»);

СоответствиеРазмеров.Вставить(50, «XL»);

1. Как получить ***количество элементов*** в структуре (работает для всех коллекций)

КоличествоЭлементов = СоответствиеРазмеров.**Количество**();

1. Как **найти** значение в соответствии

Размер36 = СоответствиеРазмеров.Получить(36); // ключ

Если Размер36 <> Неопределено Тогда

Сообщить(Размер36);

КонецЕсли;

1. Как **получить** значение в соответствии

//способ 1

Размер42 = СоответствиеРазмеров**[42];**

//способ 2

Размер36 = СоответствиеРазмеров.**Получить**(36); // ключ

Если Размер36 <> Неопределено Тогда

Сообщить(Размер36);

КонецЕсли;

//хотим поменять соответствие для размера 50:

СоответствиеРазмеров[50] = «XXXL»;

1. **Обход** всех элементов соответствия

Для каждого ЭлементСоответсивя из СоответствиеРазмеров Цикл

Сообщить(ЭлементСоответствия.Ключ + « = » + ЭлементСоответствия.Значение);

// может быть ошибка, так как ключом может быть любой типа

КонецЦикла;

1. Как **удалить** элемент соответствия

СоответствиеРазмеров.Удалить(42);

1. Как **очистить** соответствие

СоответствиеРазмеров.Очистить();

Как проверить тип коллекции:

Если ТипЗнч(СоответствиеРазмеров) = Тип(«Массив») Тогда ….

ИначеЕсли ТипЗнч(СоответствиеРазмеров) = Тип(«Структура») Тогда …

ИначеЕсли ТипЗнч(СоответствиеРазмеров) = Тип(«Соответствие») Тогда …

КонецЕсли;

1. **Список значений**

Имеет: индекс, значение, представление (как значение видит пользователь), пометку, картинку. Используется для взаимодействия с пользователем.

1. Как **создать** список значений

СписокДней = Новый СписокЗначений;

1. Как **вставить** элемент в список значений

СписокДней.Добавить(«Пн», «Понедельник»);

СписокДней.Добавить(«Вт», «Вторник»);

СписокДней.Добавить(«Ср», «Среда»);

СписокДней.Добавить(«Чт», «Четверг»);

СписокДней.Добавить(«Пт», «Пятница»);

СписокДней.Добавить(«Сб», «Суббота»);

СписокДней.Добавить(«Вс», «Воскресенье»);

//в конец, обязательно вставить значение («значение», «представление», «истина/ложь»);

1. Как получить ***количество элементов*** в список значений

КоличествоЭлементов = СписокДней.Количество();

1. Как **найти** значение в списке значений //результат поиска – элемент списка значений (строка)

//найти по значению

Среда = СписокДней.НайтиПоЗначение(«Ср»);

//найти по идентификатору

Четверг = СписокДней.НайтиПоИндентификатору(3);

1. Как **получить** значение в списке значений (по индексу)

//способ 1

Пятница = СписокДней.Получить(4);

//способ 2

Пятница = СписокДней[4];

1. **Обход** всех элементов списка значений

Для каждого ЭлементСписка из СписокДней Цикл

Сообщить(ЭлементСписка. Значение + « = (» + ЭлементСтруктуры.Представление+ «) »);

КонецЦикла;

1. показать список значений пользователю для **выбора одного значения**

ЧтоДелатьПослеВыбора = Новый **ОписаниеОповещения**(«ПослеВыбораЗначения», ЭтотОбъект);

СписокДней.ПоказатьВыборЭлемента(ЧтоДелатьПослеВыбора, «Выберете дни удаленной работы»);

Процедура ПослеВыбораЗначения(ВыбранныйЭлемент, ДополнительныеПараметры) Экспорт

Если ВыбранныйЭлемент = Неопределено Тогда

Возврат;

КонецЕсли;

Сообщить(«ВЫ будете работать в » + ВыбранныйЭлемент.Представление);

КонецПроцедуры;

1. показать список значений пользователю для отметки **нескольких значений**

ЧтоДелатьПослеОтметки = Новый **ОписаниеОповещения**(«ПослеОтметкиЗначения», ЭтотОбъект);

СписокДней.ПоказатьОтметкуЭлементов(ЧтоДелатьПослеОтметки, «Выберете дни удаленной работы»);

Процедура ПослеОтметкиЗначения (Список, ДополнительныеПараметры) Экспорт

Если Список = Неопределено Тогда

Возврат;

КонецЕсли;

ДниУдаленнойРаботы = «»;

Для каждого ДеньНедели из Список Цикл

Если ДеньНедели.Пометка Тогда

ДниУдаленнойРаботы = ДниУдаленнойРаботы +?( ДниУдаленнойРаботы <>””, «,» , «») + ДеньНедели.Представление;

КонецЕсли;

КонецЦикла;

Сообщить(«ВЫ будете работать в: » + ВыбранныйЭлемент.Представление);

КонецПроцедуры;

1. Как **удалить** элемент из списка значений

СписокДней.Удалить(0);

1. Как **очистить** список значений

СписокДней.Очистить();

1. **Таблица значений**

Можно считать, как массив, содержащий структуры

Очень тяжелая, доступна только для работы на сервере, толстый клиент

Создаем процедуру на сервере

1. Как **создать** таблицу значений

ТаблицаТоваров = Новый Таблица значений

//создание колонок – еще одна коллекция

//ТаблицаТоваров.Колонки.Добавить(ИмяКолонки, ТипЗначенийВКолонке)

ТаблицаТоваров.Колонки.Добавить(«Артикул»);

ТаблицаТоваров.Колонки.Добавить(«Группа»);

ТаблицаТоваров.Колонки.Добавить(«Наименование»);

ТаблицаТоваров.Колонки.Добавить(«Количество»);

ТаблицаТоваров.Колонки.Добавить(«ЦенаЗакупки»);

ТаблицаТоваров.Колонки.Добавить(«ЦенаПродажи»);

1. Как **вставить** элемент **(строку)** в таблицу значений

//Вставить по индексу, добавить в конец

НоваяСтрока = ТаблицаТоваров.Добавить();

НоваяСтрока.Артикул = «1234»;

НоваяСтрока.Наименование = «Мышь gtag»;

НоваяСтрока.Количество = 15;

НоваяСтрока.ЦенаЗакупки = 20;

НоваяСтрока.ЦенаПродажи = 40;

НоваяСтрока.Группа = «компьютерные комплектующие»;

НоваяСтрока = ТаблицаТоваров.Добавить();

НоваяСтрока.Артикул = «122»;

НоваяСтрока.Наименование = «микрофон»;

НоваяСтрока.Количество = 150;

НоваяСтрока.ЦенаЗакупки = 200;

НоваяСтрока.ЦенаПродажи = 400;

НоваяСтрока.Группа = «микрофоны»;

1. Как получить ***количество элементов*** в таблице (работает для всех коллекций)

КоличествоСтрок = ТаблицаТоваров.Количество();

1. Как **найти** значение в таблице

//Найти – поиск 1ой строки

//Значение – что ищем, колонки – где ищем

Мышь = ТаблицаТоваров.Найти(«Мышь gtag», «Наименование»);

Если Мышь = Неопределено Тогда

Сообщить(«Мышь не найдена»);

КонецЕсли;

//НайтиСтроки – поиск нескольких строк

ВсеКомплектующие = ТаблицаТоваров.НайтиСтроки(«Группа», «компьютерные комплектующие»);

ВсеКомплектующие = ТаблицаТоваров.НайтиСтроки(ПараметрыОтбора);

Если ВсеКомплектующие.Количество()=0 Тогда

Сообщение = Новый СообщениеПользователю;

Сообщение.Текст = «Не найдено комплектующих»;

Сообщение.Сообщить();

КонецЕсли;

Для каждого СтрокаКомплектующих из ВсеКомплектующие Цикл

// элментами массива являются строки таблицы значений

Сообщить(СтрокаКомплектующих.Артикул);

КонецЦикла;

1. **Копия** таблицы значений

//копировать целиком с данными

ТаблицаКомплектующих = ТаблицаТоваров.Скопировать(ВсеКомплектующие);

ТаблицаКомплектующих = ТаблицаТоваров.Скопировать(ВсеКомплектующие, «Наименование»); //скопируется только нужный столбец

//только названия колонок без данных

Полнаякопиябезданных = ТаблицаТоваров.СкопироватьКолонки();

1. **Сортировка** таблицы

//по возрастанию цены

ТаблицаТоваров.Сортировать(«ЦенаПродажи»);

//по убыванию цены

ТаблицаТоваров.Сортировать(«ЦенаПродажи **Убыв**»);

1. Получить итог по колонке числового типа

КоличествоИтог = ТаблицаТоваров.Итог(«Количество»);

1. **Сворачивание** таблицы

КопияТаблицы = ТаблицаТоваров.Скопировать();

КопияТаблицы.Свернуть(«Группа», «Количество»); //ТАБЛИЦА ПОМЕНЯЕТСЯ

1. **Обход** всех элементов таблицы

Для каждого СтрокаТаблицы из ТаблицаТоваров Цикл

Сообщение = Новый СообщениеПользователю;

Сообщение.Текст = СтрокаТаблицы. Группа + «, » + СтрокаТаблицы. Количество;

Сообщение.Сообщить();

КонецЦикла;

1. Как **удалить** элемент таблицы

//зная индекс строки

ТаблицаТоваров.Удалить(0);

//зная строку

НайденнаяСтрока = ТаблицаТоваров.Найти(«1234», «Артикул»);

Если НайденнаяСтрока <>Неопределено Тогда

ТаблицаТоваров.Удалить(НайденнаяСтрока);

КонецЕсли;

1. Как **очистить** таблицу

ТаблицаТоваров.Очистить();

1. Добавим в таблицу миллион строк

Для Счетчик=1 по 1000000 Цикл

Строкатаблицы = ТаблицаТоваров.Добавить();

Строкатаблицы.Артикул = счетчик;

Строкатаблицы.Наименование = «микрофон» + счетчик;

Строкатаблицы.Количество = 150;

Строкатаблицы.ЦенаЗакупки = 200;

Строкатаблицы.ЦенаПродажи = 400;

Строкатаблицы.Группа = «микрофоны»;

КонецЦикла;

//Хотим найти товар с артикулом 14

НайденныйТовар = ТаблицаТоваров.Найти(«14», «Артиул»);

//в отладке нажимаем «размер производительности», F10

1. **Дерево значений**
2. Как **создать** соответствие
3. Как **вставить** элемент в соответствии
4. Как получить ***количество элементов*** в структуре (работает для всех коллекций)

КоличествоЭлементов = СтруктураДанных.Количество();

1. Как **найти** значение в соответствии
2. Как **получить** значение в соответствии
3. **Обход** всех элементов соответствия

Для каждого ЭлементСтруктуры из СтруктураДанных Цикл

Сообщить(ЭлементСтруктуры.Ключ + « = » + ЭлементСтруктуры.Значение);

КонецЦикла;

1. Как **удалить** элемент соответствия

СтруктураДанных.Удалить(«НомерПаспорта»);

1. Как **очистить** соответствие

СтруктураДанных.Очистить();