|  |
| --- |
| титул |
| **115172, Москва, ул. Б.Каменщики, д. 7;тел., факс: (495) 134 1234; e-mail: spo-54@edu.mos.ru** |

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ**

к лабораторной работе №5:

**Методы вывода сообщений.**

**Функции работы и типами СТРОКА и ЧИСЛО**

по учебной дисциплине

**МДК 01.01 Разработка программных модулей**

Специальность

**09.02.07 «Информационные системы и программирование»**

**Оглавление**

[Введение 3](#_Toc127978441)

[1. Программирование — процедуры и функции (методы вывода сообщений) 6](#_Toc127978442)

[2. Программирование — функции (методы работы с типом «Строка») 8](#_Toc127978443)

[3. Программирование — функции (методы работы с типом «Число») 12](#_Toc127978444)

[Индивидуальное задание 14](#_Toc127978445)

[Контрольные вопросы 15](#_Toc127978446)

# **Введение**

**Цель работы**: знакомство с конфигуратором платформы 1С: Предприятие для наиболее эффективной работы с информационной базой.

**Время выполнения**: 2 часа

**Задание**:

1. Законспектировать теоретические сведения, при наличии.

2. Выполнить лабораторную работу.

3. Выполнить индивидуальное задания, при наличии.

4. Защитить работу, ответив на контрольные вопросы устно.

В первую очередь введем несколько основных определений.

Для вывода информации пользователю в 1С:Предприятии 8.2 существует специальный механизм сообщений. В этом механизме используется объект встроенного языка **СообщениеПользователю**. Предполагается, что сообщения выводятся для того, чтобы сообщить пользователю об ошибках. А для информирования о выполняемом действии рекомендуется использовать метод встроенного языка **ПоказатьОповещениеПользователя().**

**Строка —** это один из примитивных типов данных существующих 1С и работать с ним приходится практически постоянно. Помимо 1С, этот тип данных используется во всех известных языках программирования, обычно он называется «string». Следовательно, необходимо иметь представление о том, какие функции для работы со строками существуют, как и когда они используются, а также, что получается в результате. Этим и займемся.

**Функции для работы со строками:**

1. Строка
2. СтрДлина
3. СокрЛП, СокрЛ, СокрП
4. Лев, Прав, Сред
5. ВРег, НРег, ТРег
6. Найти
7. СтрЧислоВхождений
8. СтрЗаменить
9. ПустаяСтрока
10. СтрЧислоСтрок
11. СтрПолучитьСтроку
12. Символ, КодСимвола
13. Символы
14. ВК (CR) — возврат каретки;
15. ВТаб (VTab) — вертикальная табуляция;
16. НПП (NBSp) — неразрывный пробел;
17. ПС (LF) — перевод строки;
18. ПФ (FF) — перевод формы;
19. Таб (Tab) — символ горизонтальной табуляции.
20. ЗначениеВСтрокуВнутр, ЗначениеИзСтрокиВнутр
21. ВвестиСтроку
22. СтрШаблон
23. СтрСравнить
24. СтрНайти
25. СтрНачинаетсяС, СтрЗаканчиваетсяНа
26. СтрРазделить, СтрСоединить

**Типовые задачи при работе со строками:**

1. Сложение строк
2. Преобразования строка-число, число-строка
3. Кавычки в строке
4. Символ переноса строки
5. Сравнение строк
6. Разделение строки по разделителю

**Число —** это один из примитивных типов данных существующих в 1С. Числа в 1С:Предприятии встречаются абсолютно везде и всюду. Следовательно важность знания функций для работы с числами трудно переоценить.

С числами в 1С можно выполнять все стандартные арифметические операции:

* “+” — сложение;
* “-” — вычитание;
* “\*” — умножение;
* “/” — деление;
* “%” — остаток от деления.

**Функции для работы с числами:**

1. Окр
2. Цел
3. Число
4. Формат
5. Число прописью
6. Мин
7. Макс
8. Вычислить
9. Модуль
10. Sqrt
11. Pow
12. Exp
13. Логарифм
14. Тригонометрия
15. Sin(x) — синус;
16. Cos(x) — косинус;
17. Tan(x) — тангенс;
18. ASin(x) — арксинус;
19. ACos(x) — арккосинус;
20. ATan(x) — арктангенс.
21. Случайные числа
22. Ввести число

**Типовые задачи при работе с числами:**

1. Дробная часть числа
2. Строку в число

# **Программирование — процедуры и функции (методы вывода сообщений)**

**Метод объекта —** это процедура или функция, которая или выполняет какое-то действие над экземпляром этого объекта, или возвращает информацию о состоянии экземпляра этого объекта.

**Процедура —** заранее заготовленный алгоритм, который можно вызвать, обратившись по имени (например, Сообщить()).

**Функция —** аналог процедуры, которая дополнительно возвращает значение (например, Окр()).

***Важно!***

***Результат функции можно присвоить переменной, а процедуры — нет.***

Известный нам метод, «Сообщить()», является процедурой вывода. Сообщить(ТекстВывода, Статус) выводит созданное сообщение в окно сообщений.

Помимо него существуют ещё процедуры вывода, одна из них — это «Предупреждение()», которое выводит сообщение в модальном окне.

Модальное окно — это окно, которое блокирует работу пользователя с приложением до тех пор, пока пользователь его не закроет.

С возникновением Web-клиента появились новые методы, так как модальные окна в данном типе клиента не работали. В новых методах указывается имя процедуры, которая будет выполняться, когда данное окно будет закрыто.

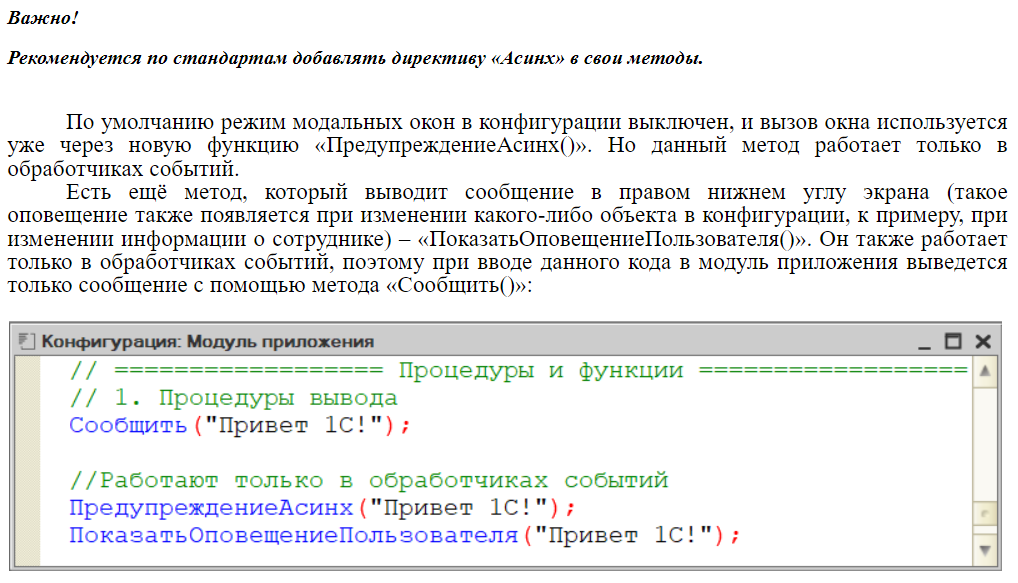
С версией 8.3.18 появилась директива «Асинх», которая означает асинхронные методы. Асинхронные методы управляются разработчиком: именно разработчик решает, будет выполняться программный код после прерывания или нет. Данные методы работают повсеместно, и в названии таких методов добавляется директива «Асинх».

***Важно!***

***Рекомендуется по стандартам добавлять директиву «Асинх» в свои методы.***

По умолчанию режим модальных окон в конфигурации выключен, и вызов окна используется уже через новую функцию «ПредупреждениеАсинх()». Но данный метод работает только в обработчиках событий.

Есть ещё метод, который выводит сообщение в правом нижнем углу экрана (такое оповещение также появляется при изменении какого-либо объекта в конфигурации, к примеру, при изменении информации о сотруднике) — «ПоказатьОповещениеПользователя()». Он также работает только в обработчиках событий, поэтому при вводе данного кода в модуль приложения выведется только сообщение с помощью метода «Сообщить()»:

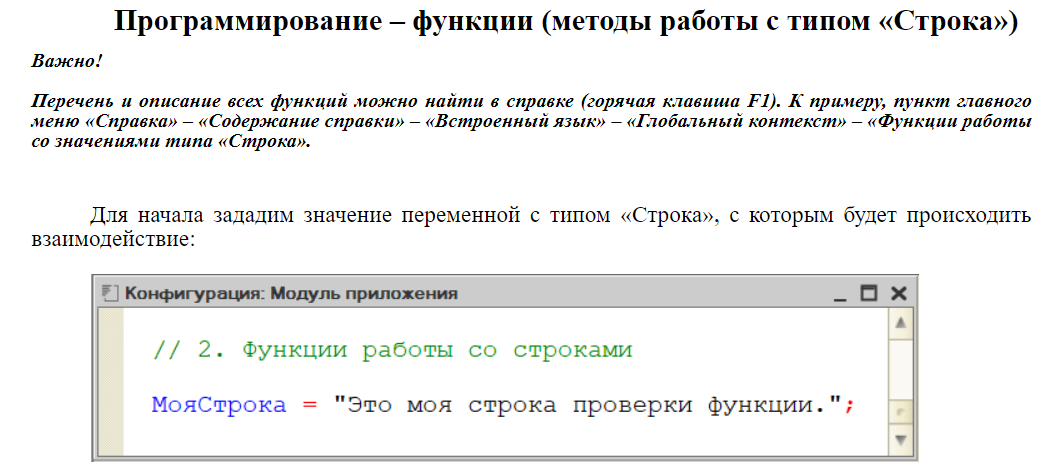


# **Программирование — функции (методы работы с типом «Строка»)**

***Важно!***

***Перечень и описание всех функций можно найти в справке (горячая клавиша Е1). К примеру, пункт главного меню «Справка» — «Содержание справки» — «Встроенный язык» — «Глобальный контекст» — «Функции работы со значениями типа «Строка».***

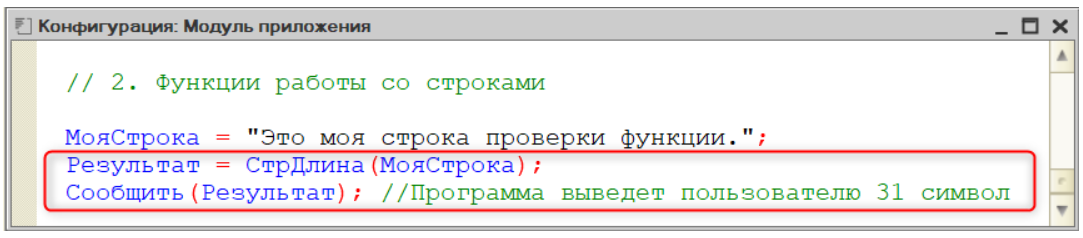
Для начала зададим значение переменной с типом «Строка», с которым будет происходить взаимодействие:



Далее будем записывать в новую переменную «Результат» результаты функций и выводить их. пользователю. Делается это с помощью процедуры вывода «Сообщить».

Первая функция, с которой необходимо познакомиться — это «СтрДлина()». Данная функция считает количество символов в строке и возвращает значение с типом число. Под символами подразумеваются не только буквы, но и знаки препинания, пробелы, разделители строки и т. д.

Для того чтобы вызвать данную функцию, достаточно написать:



***Важно!***

***Просто значение («=») можно только функции, иначе программа выдаст ошибку «Обращение к процедуре как в функции».***

Не менее важная и полезная функция работы со строкой — «СтрНайти()».

Её синтаксис выглядит следующим образом:

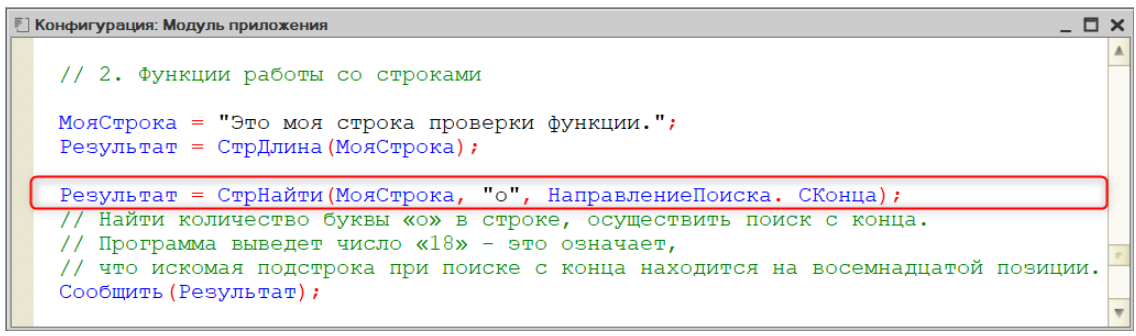


Параметры функции:

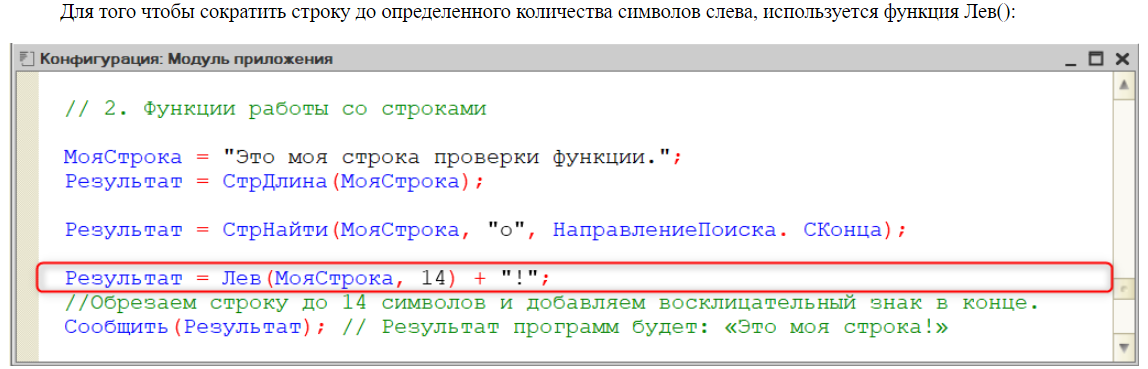
* **Строка (обязательный) —** это исходная строка, в которой будет осуществляться поиск.
* **Подстрока поиска (обязательный) —** это значение с типом «Строка», которое необходимо найти. В данном параметре очень важен регистр букв.
* **Направление поиска (необязательный) —** указывает направление поиска. По умолчанию, если не указывать значение — «НаправлениеПоиска.СНачала». Также существует направление — «НаправлениеПоиска.СКониа».
* **Начальная позиция (необязательный) —** указывает позицию в строке (тип «Число»), с которой начинается поиск.
* **Номер вхождения (необязательный) —** указывает номер вхождения искомой подстроки в исходной строке.

Если строка не содержит указанной подстроки, то возвращается 0. В ином случае функция вернёт номер позиции искомой подстроки.

Пример использования данной функции выглядит следующим образом:



Для того чтобы сократить строку до определенного количества символов слева, используется функция Лев():



В обратную сторону работает функция «Прав()», взаимодействие с ней аналогичное.

Если необходимо вырезать строку посередине, используется функция «Сред». В отличие от предыдущей, она имеет три параметра:



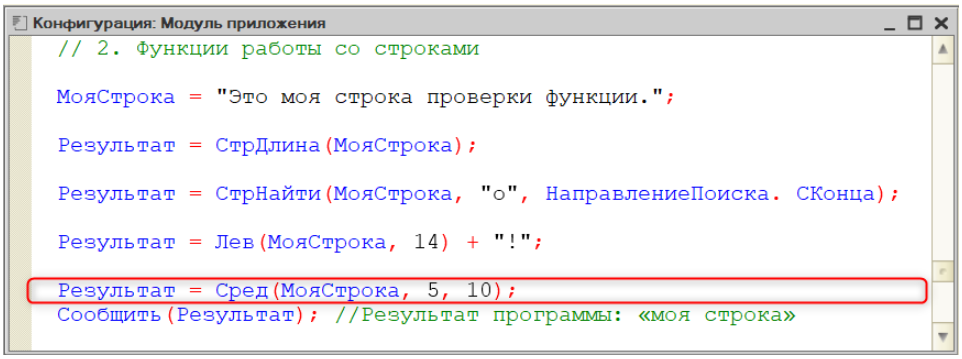
Строка — исходная строка.

Начальный номер — это номер символа, с которого начинается выборка. Нумерация символов в строке начинается с 1.

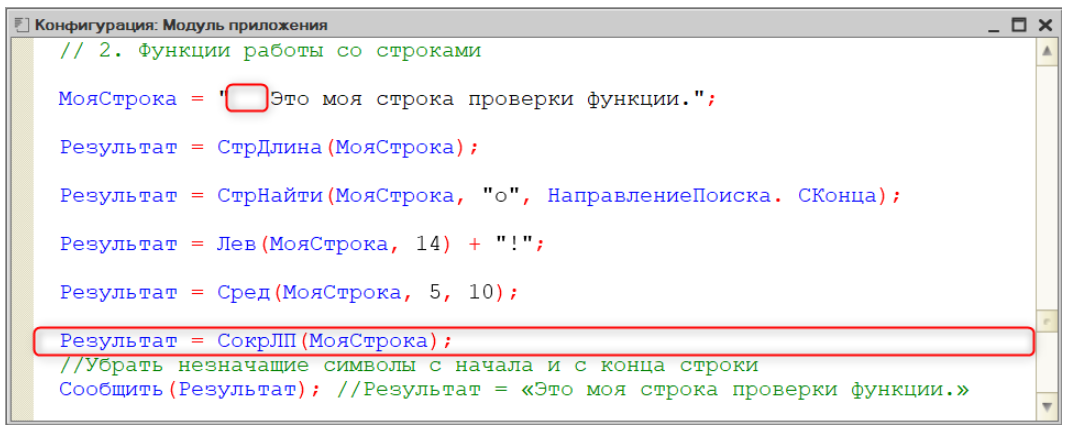
Если указано значение меньшее или равное нулю. то параметр принимает значение 1.

Число символов — количество выбираемых символов. Если параметр не указан, то выбираются символы до конца строки.

Обрежем строку с помощью данной функции:

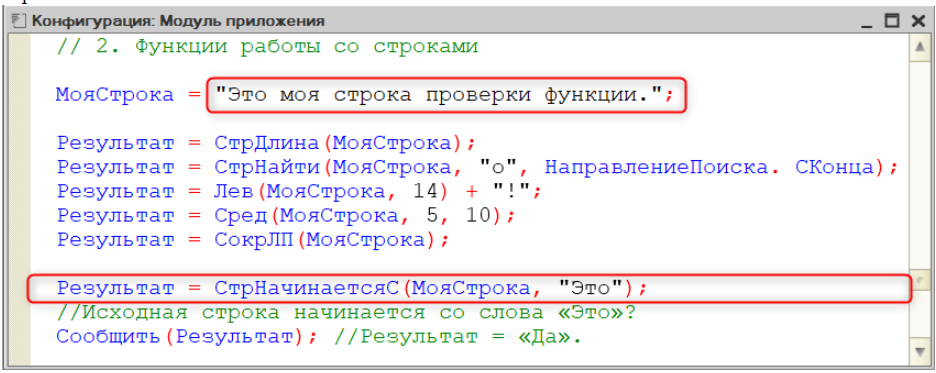


Для того чтобы убрать все незначащие символы (пробел, неразрывный пробел, табуляция и т.д.) слева и справа в строке, используем функцию «СокрЛП()»:

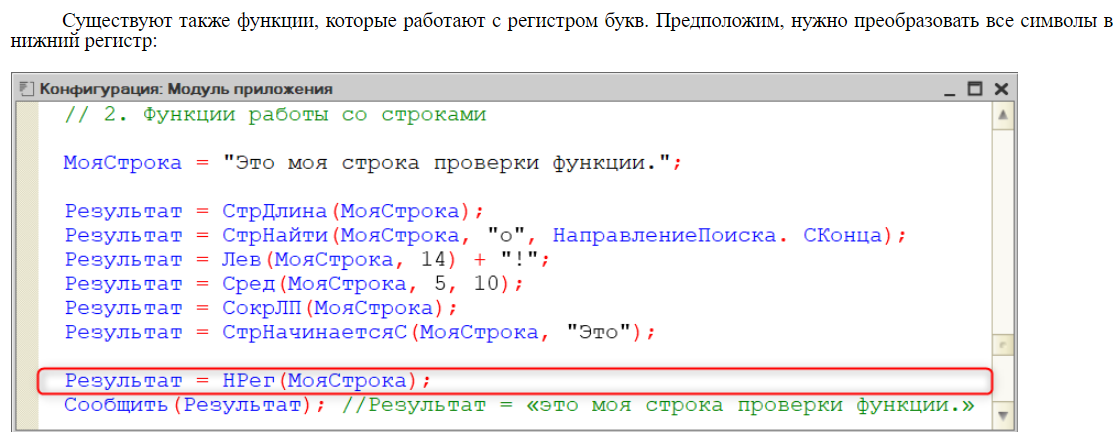


Для того чтобы обрезать незначащие символы слева или справа, используются функции «СокрЛ()» и «СокрП()» соответственно.

Если необходимо проверить («Да» или «Нет»), начинается ли строка с подстроки «Это», то используется функция «СтрНачинаетсяС()». Она имеет два параметра — это исходная строка и искомая подстрока. Используется она следующим образом:

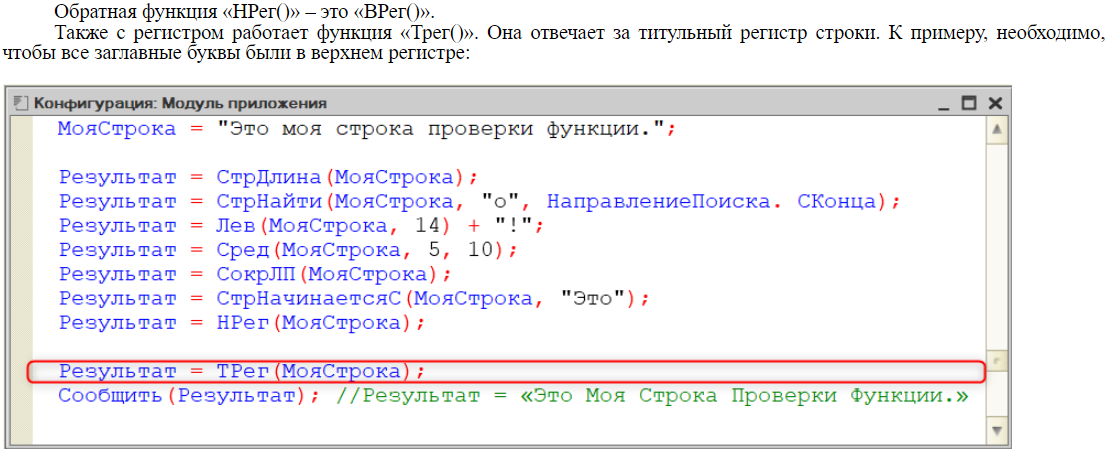


Существуют также функции, которые работают с регистром букв. Предположим, нужно преобразовать все символы в нижний регистр:



Обратная функция «НРег()» - это «ВРег()».

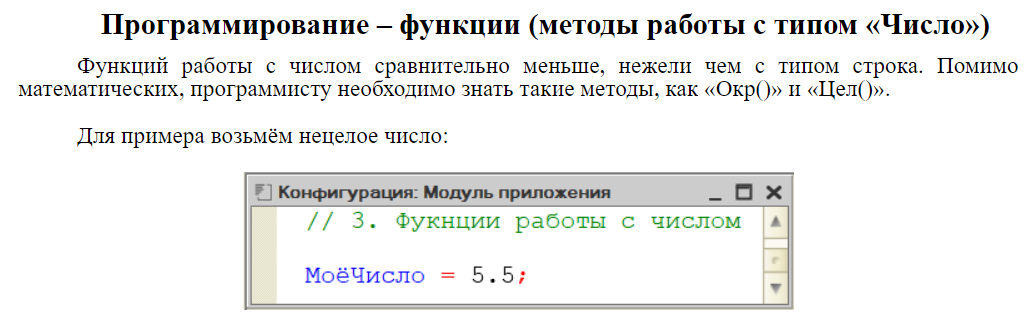
Также с регистром работает функция «ТРег()». Она отвечает за титульный регистр строки. К примеру, необходимо, чтобы все заглавные буквы были в верхнем регистре:



# **Программирование — функции (методы работы с типом «Число»)**

Функций работы с числом сравнительно меньше, нежели чем с типом строка. Помимо математических, программисту необходимо знать такие методы, как «Ок()» и «Цел()».

Для примера возьмём нецелое число:



Синтаксис первой функции выглядит следующим образом:



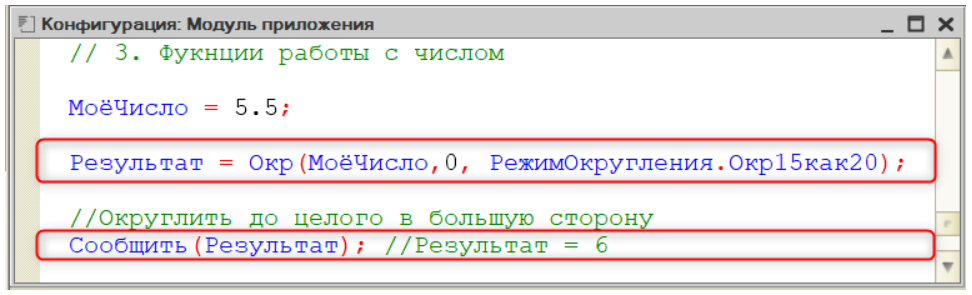
Число (обязательный) — исходное число.

Разрядность (необязательный) — определяет число знаков дробной части, до которых производится округление. Если параметр отрицательный, то число округляется до соответствующего разряда в целой части, начиная с младших разрядов.

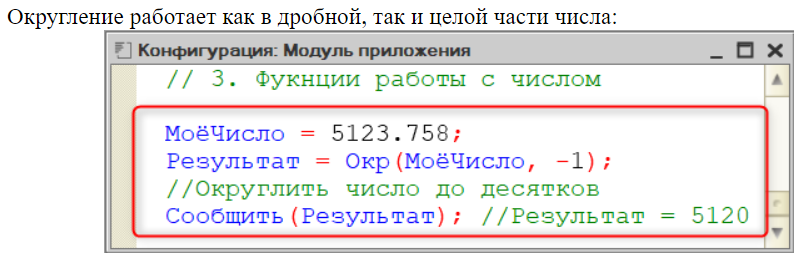
Параметр <Разрядность> обязательный, если указан и третий параметр <РежимОкругления>.

Значение по умолчанию: 0.

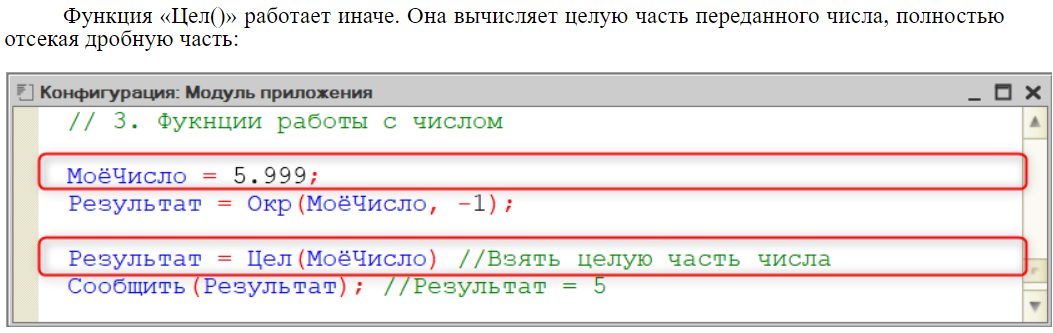
Режим округления — определяет режим округления (в большую или в меньшую сторону). Помимо системного перечисления допускается использование числового параметра: «РежимОкругления.Окр15как10» — если при округлении 1.5 = 1; «РежимОкругления.Окр15как20» — если при округлении 1.5 =2.



Округление работает как в дробной, так и целой части числа:



Функция «Цел()» работает иначе. Она вычисляет целую часть переданного числа, полностью отсекая дробную часть:



# **Индивидуальное задание**

**№ 9. Алгоритмы - «Сколько Вам недель?»**

Необходимо определить1 переменную:

1. Дата рождения

***Требуется посчитать возраст человека в неделях***

**№ 10. Алгоритмы - «Ваша фамилия и инициалы, пожалуйста!»**

Необходимо определить 1 переменную:

1. ФИО – «Платформов Василий Михайлович»

Требуется сформировать следующее сообщение:

«Платформов Василий Михайлович = Платформов В.М.»

***Примечание:*** при определении имени и отчества от полного ФИО используйте функции для работы со строкой.

# **Контрольные вопросы**

1. Что такое «модальное окно»?
2. Чем отличаются асинхронные методы от обычных?
3. Чем отличается процедура от функции?