



Использование алгоритмов в программе КонструкторБизнесПроцессов[©]

[Таблица использования алгоритмов](#)

[Переменные](#)

[Дополнительные реквизиты шагов сценария](#)

[Процедуры и функции подсистемы](#)

[Процедуры и функции расширения «Сигнал»](#)

[Описание алгоритмов при запуске процессов и задач](#)

[Вызов алгоритмов по имени](#)

[Динамические команды на форме задачи](#)

[Встроенный редактор программного кода](#)

[Пример программной настройки меню выбора для чат-ботов Telegram](#)

[Дополнительные возможности программной настройки чат-ботов Telegram](#)

Актуально для версии Основной поставки: 2.0.3.21
Актуально для версии Расширения «Сигнал»: 1.3.3.8

Готово

Проверить

_Результат = <...>

?

Варианты выбора:

Отгружено

Не взяли в работу через 5 часов

Не отгружено через 2 дня

Ожидание отгрузки...

```
1 // Контроль отгрузки...
2 Запрос = Новый Запрос;
3 Запрос.Параметры.Вставить("Заказ", _Процесс.Объект);
4
5 Запрос.Текст = "ВЫБРАТЬ РАЗЛИЧНЫЕ
6               | Ордера.Ссылка КАК Ордер,
7               | Ордера.Ссылка.Статус = ЗНАЧЕНИЕ(Перечисление.СтатусыРасходныхОрдеров.От
8               | Из
9               | Документ.РасходныйОрдерНаТовары.ТоварыПоРаспоряжениям КАК Ордера
10              | ГДЕ
11              | Ордера.Распоряжение = &Заказ
12              | И Ордера.Ссылка.Проведен";
13
14 Выборка = Запрос.Выполнить().Выбрать();
15
16 _Результат = Истина;
17
18 ЕстьОрдера = Ложь;
19
20 Пока Выборка.Следующий() Цикл
21     ЕстьОрдера = Истина;
22     _Результат = _Результат И Выборка.Отгружен
23
24
25
26 КонецЦикла;
```

**ВСТРОЕННЫЙ
РЕДАКТОР
ИСПОЛНЯЕМОГО
ПРОГРАММНОГО
КОДА.**



**БизнессПроцессы
Для Профессионалов.**

Таблица использования алгоритмов

В таблице ниже перечислены места использования алгоритмов и программного кода.

Алгоритм – элемент справочника [Алгоритмы](#), содержащий исполняемый программный код и дополнительные настройки.

Программный код – исполняемый программный код, не хранимый в справочнике [Алгоритмы](#).

Объект системы	Использование		Алгоритм	Программный код
Сценарии	Автозапуск по произвольному условию		✓	✓
	Динамический отбор объекта процесса			✓
	Динамические отборы в таблице объектов процесса			✓
	Параметры шаблона наименования процесса			✓
	Алгоритм, выполняемый при создании процесса		✓	
	Параметры описаний шагов сценария		✓	✓
	Отложенный запуск шагов сценария по условию		✓	✓
	 Действие	Параметры шаблона наименования		
		Настройка динамических кнопок на формах задач	✓	✓
		Алгоритм, выполняемый при запуске	✓	
		Условие автоматического и ручного завершения	✓	
		Динамический отбор объекта		✓
		Динамические отборы в таблице объектов		✓
		Настройка адресации: Исполнитель / Роль исполнителя / Наблюдатель	✓	✓
	 Обработка	Тело обработки	✓	✓
	 Условие	Автоматическая проверка условий	✓	✓
	 Выбор варианта	Автоматический выбор вариантов	✓	✓
Причины остановок процессов и задач	Условия автоматических остановок		✓	
Причины отмены процессов	Условия автоматической отмены		✓	
 Система оповещений «Сигнал»	Настройка оповещений	Определение получателей оповещений	✓	
		Параметры шаблона текста оповещений		✓
		Определение контекста оповещений для системы взаимодействия	✓	
	Чат-боты Telegram	Дополнительные алгоритмы авторизации	✓	
		Шаблон текста запроса для получения дополнительных вложений		✓
	Ключевые слова чат-ботов	Алгоритмы обработки входящих сообщений/запросов	✓	

Переменные

1. Для ссылочных значений доступно обращение к реквизитам через знак «.». Пример: `_Процесс.Сценарий`.
2. Переменные с назначением «Общее» доступны при **вызове алгоритмов по имени**.

Имя	Тип	Назначение	Описание
<code>_Процесс</code>	бип Процесс	Общее	Текущий процесс.
<code>_ПроцессОбъект</code>	бип Процесс	Общее	Текущий процесс. Доступен только в алгоритмах, выполняемых при запуске новых процессов.
<code>_ТаблицаДанных</code>	ТаблицаЗначений	Общее	Таблица данных текущего процесса. Заполняется в алгоритмах, выполняемых при запуске новых процессов, если в сценарии настроено использование таблицы данных. Доступ к таблице при автоматическом запуске через переменную <code>_ТаблицаДанных</code> . Доступ к таблице в алгоритме заполнения при ручном запуске процесса через переменную <code>_ПроцессОбъект.ТаблицаДанных</code> . В прочих случаях, доступ к таблице данных возможен через <code>_Процесс.ТаблицаДанных</code> .
<code>_Процесс.Объект</code>	СправочникСсылка ДокументСсылка	Общее	Основной объект процесса. Если процесс запускается по событию, в реквизите хранится ссылка на элемент справочника или документа, по которому произошло событие.
<code>_Процесс.Основание</code>	бип Процесс	Общее	Родительский процесс текущего процесса.
<code>_Задача</code>	бип Задача	Общее	Предшествующая текущему шагу задача. Текущая задача, если выполняется алгоритм при запуске задачи или алгоритм проверки автоматического завершения задачи по условию .
<code>_ВыполненаВручную</code>	Булево	Общее	Признак интерактивного выполнения задачи (нажатием кнопки Выполнена). Используется только в алгоритмах проверки автоматического завершения задач по условию .
<code>_ИмяШага</code>	Строка	Общее	Имя текущего шага, как оно задано в настройках сценария. Например, <code>&Действие1</code> . Используется в алгоритмах настроек адресации.
<code>_ПредставлениеШага</code>	Строка	Общее	Представление текущего шага, как оно задано в настройках сценария. Используется в алгоритмах настроек адресации.
<code>_ПараметрN</code> (N = 1,2,3,4,5)	Произвольный	Общее	Фиксированное, константное значение, указанное в настройках сценария.
<code>_ПараметрNИмя</code> (N = 1,2,3,4,5)	Строка	Общее	Имя фиксированного, константного значения, указанное в настройках сценария.
<code>_Исполнитель</code>	Строка Пользователи	Общее	Исполнитель текущей задачи. Доступен только в алгоритмах, выполняемых при запуске новых задач и при использовании алгоритмов в расширении «Сигнал» : для программного определения получателей оповещений и при выполнении алгоритмов в чат-ботах – в этом случае в переменной содержится значение текущего идентифицированного пользователя Telegram. В прочих случаях для получения исполнителя задачи следует использовать функцию <code>НайтиЗадачу()</code> .
<code>_Процесс.Автор</code> <code>_Задача.Автор</code> <code>_Задача.Исполнитель</code> <code>_Задача.Наблюдатель</code>	Строка Пользователи Роли НаборыИсполнителей	Общее	Автор процесса. Автор, Исполнитель, Наблюдатель задачи.
<code>_ДанныеДляОбработки</code>	Соответствие	Общее	Переменная используется при обработке данных, полученных от чат-ботов расширения «Сигнал» . Содержит историю текущего взаимодействия (контекст «беседы») – результаты выбора из меню, введенный комментарий, прочие сведения, в т.ч. добавляемые программно через алгоритмы ключевых слов.

Имя	Тип	Назначение	Описание
			Очищается при обработке ключевого слова с видом Текстовый ответ (если не включена настройка ключевого слова Сохранять контекст). Очищается при выполнении алгоритмов ключевых слов вида Выполнение алгоритма (если значение переменной _Результат не является массивом или структурой). Очищается при отказе от подтверждения выбора для ключевых слов вида Выбор . Также, см. Руководство пользователя «Сигнал» (раздел Ключевые слова), а также разделы Пример программной настройки меню выбора для чат-ботов Telegram и Дополнительные возможности программной настройки чат-ботов Telegram .
_Результат	Булево Произвольный	Общее	Переменная, в которую помещается результат вычисления какого-либо значения. Используется при проверке условий, при программном определении исполнителей задач и т.п. При вызове алгоритма, как процедуры или функции (см. реквизит ИмяДляВызова на форме алгоритма), переменная доступна при вызове каждого алгоритма. Таким образом, переменную можно использовать для передачи значения при вызове нескольких алгоритмов. При программном определении исполнителей задач в переменную _Результат передаётся значение Исполнителя, Роли исполнителя или Массив, содержащий исполнителей и/или роли. При этом для групповых задач, если Массив содержит Роль исполнителя, то исполнителем задачи устанавливается роль, а не пользователи, обладающие данной ролью. Если в переменную передана Структура вида Новый Структура("ОбъектыПоИсполнителям, Исполнители", Истина, Массив), то каждая групповая задача будет привязана к объекту процесса из табличной части Объекты, если количество исполнителей совпадает с количеством объектов в процессе. Пример: Массив содержит значения «Иванов», «Петров», «Логисты». При обработке группового шага сценария будет создано 3 задачи: задача для Иванова, задача для Петрова и задача для Логистов. Любой из логистов сможет взять задачу для исполнения. При использовании Массива для не групповой задачи, в массиве учитываются только исполнители, роли в массиве игнорируются. В расширении «Сигнал» используется для передачи результатов выполнения алгоритмов для ключевых слов вида Выполнение алгоритма. Также, см. Руководство пользователя «Сигнал» (раздел Ключевые слова), а также раздел Пример программной настройки меню выбора для чат-ботов Telegram.
_Результат	Булево	Проверка условий	Условие проверяется по значению переменной. Условие выполнено, если Результат = Истина .
_РасшифровкаРезультата	Строка		В переменной может быть указана дополнительная информация о результатах проверки условия или выбора вариантов. Значение переменной выводится: – в сообщении пользователю при выполнении задачи с включенной настройкой Автоматически завершать задачу по условию. – на карте процесса, как дополнительная информация для шагов вида Условие и ВыборВарианта.
_Источник	СправочникСсылка ДокументСсылка	Автозапуск по событию	Источник возникновения события, которое привело к запуску текущего процесса.
_ИсточникДо	Структура		Данные (стандартные реквизиты, реквизиты и табличные части) источника возникновения события, которое привело к запуску текущего процесса, до текущего изменения (записи). Используется для запуска процессов при изменении реквизитов источника.

Имя	Тип	Назначение	Описание
_ЭтоНовый	Булево	Автозапуск по условию	Признак первой записи источника события (Признак создания).
_ПользовательСобытия	Строка Пользователи		Пользователь, под которым произошло событие.
_ВыходныеДанные	Массив		Массив, в который могут быть добавлены произвольные данные. Используется для передачи данных в создаваемые на основании сценария процессы. см. _РазделятьПоПроцессам
_РазделятьПоПроцессам	Булево	Расширение «Сигнал» Параметры в текстах оповещений	Признак создания нескольких процессов. Каждому элементу массива _ВыходныеДанные соответствует один создаваемый процесс. Если _РазделятьПоПроцессам = <i>Ложь</i> , все объекты из массива _ВыходныеДанные будут записаны в табличную часть Объекты нового процесса.
Параметры.Объект	Произвольный		Значение, которое будет указано в качестве объекта нового процесса.
Параметры.Сценарий	бип_Сценарии		Текущий сценарий.
_ПолучательОповещения	СправочникСсылка		Используется в алгоритмах оповещений и при расчете программных параметров в текстах оповещений.
_ТекстОповещения	Строка		Используется в алгоритмах оповещений. Содержит текст оповещения.
_ТекстОповещенияHTML	Строка		Используется в алгоритмах оповещений. Содержит текст оповещения в формате HTML.

Дополнительные реквизиты шагов сценария

- Для шагов вида **Условие** и **ВыборВарианта**. Для получения данных о пройденных шагах сценария можно обращаться к ним **по имени** в виде &Действие1, &Условие2 и т.п.
- Имена шагов** сценария можно узнать, дважды щёлкнув по любой из линий связи, относящихся к шагу, или наведя курсор на элемент графической схемы.

Реквизит шага сценария	Тип	Назначение	Описание
.ДатаСоздания	Дата	Получение доп. сведений о шагах сценария	Дата создания шага.
.ДатаВыполнения	Дата		Дата выполнения шага.
.Просрочено	Булево		Признак просроченной задачи.
.Результат	Строка		Результат проверки  Условия («Да»/«Нет»). Результат  Выбора варианта .
.КоличествоПовторов	Число		Количество повторов текущего шага.

Процедуры и функции подсистемы

Имя функции	Параметры	Возвращаемое значение	Описание
ДанныеШага	– Процесс (бип_Процесс) процесс, к которому относится шаг. – ИмяНомер (Строка, Число) имя шага сценария или номер шага на карте процесса. Например, «&Действие1» или 3.	Структура состоит из: – Период (Дата) - Дата шага. – Процесс (бип_Процесс) Процесс. – НомерШага (Число) - Номер шага. – Имя (Строка) - Имя шага. – УсловиеВыбор (Строка) Результат обработки выбора или условия. – Действие (бип_Задача, бип_Процесс) Задача или вложенный процесс.	Возвращает структуру данных шага процесса по его имени или порядковому номеру на карте процесса. Если по имени найдено несколько шагов, то функция возвращает массив структур.

Имя функции	Параметры	Возвращаемое значение	Описание
		– ОтложеннаяДата (<i>Дата</i>) Дата отложенного запуска шага. – РасшифровкаРезультата (<i>Строка</i>) Подробное описание результата проверки условия. Массив	
НайтиЗадачу	– Процесс (<i>бип_Процесс</i>) процесс, к которому относится задача. – Имя (<i>Строка</i>) имя шага сценария в виде «&ДействиеN» или в виде шаблона для поиска нескольких объектов (по шаблону «&ВложенныйПроцесс%» будут найдены все дочерние процессы).	Структура состоит из: – Задача (<i>бип_Задача, бип_Процесс</i>) Найденная задача или вложенный процесс. Если задача или процесс не найдены, то возвращается пустое значение бип_Задача.ПустаяСсылка. – Автор (<i>Пользователи</i>) Автор задачи. – Исполнитель (<i>Пользователи</i>) Текущий исполнитель задачи. Массив	Возвращает структуру, содержащую ссылку на задачу или вложенный процесс по указанному имени, относящиеся к указанному процессу. Дополнительно, в структуре содержится информация об авторе и исполнителе задачи. Если по имени найдено несколько задач или вложенных процессов, то функция возвращает массив структур.
СостояниеЗадачи	– Задача (<i>бип_Задача</i>) задача. – Надату (<i>Дата, Неопределено</i>) дата получения состояния. Если не указана, будет получено текущее состояние.	Структура состоит из: – Период (<i>Дата</i>) Дата установки текущего состояния. – Состояние (<i>бип_СостоянияЗадач</i>) Текущее состояние задачи. – ПричинаОстановкиОтмены (<i>бип_ПричиныОстановок, бип_ПричиныОтмены</i>) Причина остановки/отмены задачи. – Комментарий (<i>Строка</i>) Дополнительный комментарий при остановке/отмене задачи. – НомерШага (<i>Число</i>) Порядковый номер шага задачи в истории процесса.	Возвращает состояние задачи на указанную дату.
ДанныеПроцесса	– Процесс (<i>бип_Процесс</i>) процесс, по которому требуется получить данные.	Структура состоит из: – Ссылка (<i>бип_Процесс</i>) Ссылка на процесс. Если процесс не найден (передана некорректная ссылка), то возвращается бип_Процесс.ПустаяСсылка. – Автор (<i>Пользователи</i>) Автор процесса. – Период (<i>Дата</i>) Дата установки текущего состояния. – Состояние (<i>бип_СостоянияПроцессов</i>) Текущее состояние процесса. – ПричинаОстановкиОтмены (<i>бип_ПричиныОстановок, бип_ПричиныОтмены</i>) Причина остановки/отмены процесса.	Возвращает структуру, содержащую данные указанного процесса.

Имя функции	Параметры	Возвращаемое значение	Описание
		– Комментарий (Строка) Дополнительный комментарий при остановке/отмене процесса.	
НовоеСостояние	– Объект (бип_Процесс, бип_Задача) Ссылка на процесс или задачу. – Состояние (бип_СостоянияПроцессов, бип_СостоянияЗадач) Новое состояние. – Период (Неопределено, Дата) Дата установки нового состояния. – Причина ОстановкиОтмены (бип_ПричиныОстановок, бип_ПричиныОтмены) Причина остановки/отмены процесса/задачи. – Комментарий (Неопределено, Строка) Дополнительный комментарий при остановке/отмене процесса/задачи. – ТекстОшибки (Неопределено, Строка) Переменная, в которую возвращается текст ошибки. Если не указана, ошибка записывается в журнал регистрации.	Булево – Истина новое состояние установлено. – Ложь Ошибка при установке нового состояния.	Устанавливает новое состояние для процесса или задачи.
_СохранитьЗначение	– Имя (Строка) имя, под которым сохраняется значение. – Значение (Произвольный) сохраняемое значение. Если не заполнено, сохраненное ранее по указанному имени значение удаляется. – Процесс (бип_Процесс, Неопределено) Процесс, для которого сохраняется значение. Если не указан, значение сохраняется для текущего процесса.		Сохраняет переданное значение по указанному имени. Значение используется только в рамках указанного процесса. Если не указан, то в рамках текущего процесса.
_ПолучитьЗначение	– Имя (Строка) имя получаемого значения. – Процесс (бип_Процесс,	Произвольный Полученное значение. Неопределено , если значение по имени не найдено.	Возвращает сохраненное значение по указанному имени для указанного процесса. Если процесс не указан, возвращается

Имя функции	Параметры	Возвращаемое значение	Описание
	Неопределено Процесс, для которого получается значение. Если не указан, то для текущего процесса.		значение текущего процесса.
бип_НаСервере.ВыполнитьЗадачу	– Задача (бип Задача, Строка) ссылка на задачу или имя шага сценария. – Процесс (бип Процесс, Неопределено) используется, если в параметр Задача передано имя шага сценария. – ЗаполнитьРеквизитыПоУмолчанию (Булево) используется для автоматического заполнения реквизитов. Обязательные для заполнения (но незаполненные) реквизиты задачи будут заполнены: Условие - "Нет", Выбор варианта - последний вариант, Вариант из списка - последний из списка вариантов. – ПроверятьЗаполнениеРеквизитов (Булево) признак проверки реквизитов задачи. Если Ложь, задача будет завершена без проверок с признаком "Завершена автоматически".	Структура состоит из: – Выполнена (Булево) Результат выполнения задачи. – Расшифровка (Строка) Описание ошибки выполнения задачи. Если задача выполнена, не заполняется.	Используется для программного выполнения задачи, например, при встраивании выполнения задач в объекты основной конфигурации. В функцию может быть передана ссылка на задачу или имя шага сценария в виде «&ДействиеN» и ссылка на процесс, к которому относится задача.
бип_Автозапуск.ЗапуститьНовыйПроцесс	– Параметры (Структура) состоит из: – Сценарий (бип Сценарий) Сценарий, по которому запускается процесс. – Дата (Дата) Дата и время нового процесса. – Событие (Перечисление.бип ТипыСобытий) Событие запуска. – Комментарий (Строка) Дополнительный комментарий к процессу. – Объект (Неопределено, Число, Строка, Дата, Булево, ЛюбаяСсылка) Объект, который записывается в процесс.	Структура состоит из: – Процесс (бип Процесс) Ссылка на новый процесс. – ТекстОшибки (Строка) Описание ошибки запуска нового процесса. Если процесс запущен, не заполняется.	Используется для программного создания и запуска нового процесса по указанному сценарию и объекту/объектам. Может использоваться в алгоритмах системы и/или в прочих случаях, когда требуется программное создание и запуск новых процессов, например в http-сервисах.

Имя функции	Параметры	Возвращаемое значение	Описание
	Если не указан, запущенный процесс не будет связан ни с каким объектом системы. – Объекты (Массив, Неопределено) Массив объектов, записываемых в процесс в табличную часть "Объекты".		
бип_НаСервере. РассчитатьПлановуюДатуВыполнения	– Дата (Дата) Начальная дата. – Параметры (Структура) состоит из: – Срок (Число) Срок в секундах. – СрокДата (Дата) Срок в виде конкретной даты. – СрокКонецДня (Булево) Признак установки срока в конце рабочего дня. – НеУчитыватьВыходные (Булево) Признак учета выходных при расчете срока. – ТолькоРабочиеДни (Булево) Признак расчета срока только по рабочим дням. – Календарь (СправочникСсылка.Календари, Неопределено) Календарь.	Дата Расчетная плановая дата.	Используется для расчета срока выполнения от указанной даты. Если указан параметр Календарь, выходные дни и почасовое расписание берутся из него. Если не указан, определяется по константе БСП «Основной календарь предприятия». Если не удастся получить основной календарь, или календарь не заполнен, то выходными считаются только суббота и воскресенье. Если не заполнено расписание календаря (или календарь не определен), то начало и окончание рабочего дня берутся из модуля повторного использования.

Процедуры и функции расширения «Сигнал»

Имя функции	Параметры	Возвращаемое значение	Описание
нс_Телеграм. ПолучитьДанныеБуфера	– ДанныеДляОбработки (Соответствие) История текущего взаимодействия.	Строка	Возвращает строковое представление переданного соответствия.
нс_Телеграм. СохранитьВложения	– ДанныеДляОбработки (Соответствие) История текущего взаимодействия. – Каталог (Строка) Каталог сохранения. – РаздельноеХранение (Булево) Признак раздельного сохранения вложений по папкам в виде Каталог\КлючПолучателя\Дата(в формате ДФ=ууууММддННммсс). Значение по умолчанию: Истина		Сохраняет вложения из полученного сообщения пользователя. Результаты сохранения записываются в переданный параметр ДанныеДляОбработки, с ключами СохранениеФайла, ПутьКФайлу и error.

Алгоритм, выполняемый при запуске нового процесса

Путь: Форма сценария — Закладка  «Дополнительно» — поле «**При создании процесса выполнять алгоритм**».

Описание: Алгоритм выполняется после начала транзакции записи нового процесса, но до начала его записи. В процедуре **ПередЗаписью()** модуля документа.

Обращение к объекту текущего процесса возможно через переменную **_ПроцессОбъект(бип_ПроцессОбъект)**.

Алгоритм —Основной обработчик процесса

Путь: Форма сценария — Закладка  «Дополнительно» — поле «**Основной обработчик**».

Описание: Алгоритм «*постоянного действия*». Выполняется для каждого процесса каждый раз при выполнении основного регламентного задания.

Алгоритм, выполняемый при завершении процесса

Путь: Форма сценария — Закладка  «Дополнительно» — поле «**При завершении процесса выполнять алгоритм**».

Описание: Алгоритм выполняется в транзакции записи статуса  **Завершен**.

Алгоритм, выполняемый после расчета срока выполнения процесса или задачи

Путь: Форма сценария — Форма настройки срока выполнения процесса или действия — поле «**После расчета срока выполнить алгоритм**».

Описание: Алгоритм выполняется после начала транзакции записи нового процесса или задачи, но до начала их записи. В процедуре **ПередЗаписью()** модуля соответствующего документа.

Обращение к процессу возможно через переменную **_ПроцессОбъект(бип_ПроцессОбъект)**. Обращение к задаче возможно через переменную **_Задача(бип_ЗадачаОбъект)**.

Алгоритм, выполняемый при запуске новой задачи

Путь: Форма сценария — Форма настройки запуска действия — поле «**При запуске выполнять алгоритм**».

Описание: Алгоритм выполняется после записи новой задачи и после записи дополнительных данных в регистры сведений (данные об авторе, исполнителях, наблюдателях, статусе текущей задачи и технических данных по процессу).

Обращение к ссылке на задачу возможно через переменную **_Задача(бип_Задача)**.

Алгоритм проверки завершения/автозавершения задачи

Путь: Форма сценария — Форма настройки действия — поле «**Алгоритм проверки условия**» при включенной настройке «**Завершать задачу по условию**».

Описание: Алгоритм выполняется при записи задачи (при выполнении) и в регламентном задании подсистемы, проверяющем условия автозавершения задач. Задача не сможет быть завершена, если переменная **_Результат = Ложь**. Сообщение пользователю будет содержать одно из значений: значение переменной **_РасшифровкаРезультата**, комментарий к алгоритму или наименование алгоритма.

Обращение к задаче возможно через переменную **_Задача(бип_Задача)** в регламентном задании и через переменную **_Задача(бип_ЗадачаОбъект)** при записи задачи.

Имя алгоритма

Имя алгоритма указывается без пробелов.

Пример: `АлгоритмПроверки`

Переменные

При вызове алгоритма следует добавлять знак `#`.

При вызове алгоритма по имени могут быть переданы параметры. Параметры указываются через запятую.

Пример: `#АлгоритмПроверки(_Процесс, "Действие1", 1, Истина)`

В качестве параметров могут быть переданы **только** значения и общие переменные подсистемы (`_Процесс`, `_Источник` и т.д.).

Значение параметров в коде алгоритма доступны через переменные `Параметр1`, `Параметр2`, `Параметр3` и т.д.

Динамические команды на форме задачи

1. Динамические команды на форме задачи могут использоваться для разных целей. Например, для открытия формы отчёта по связанному с задачей объекту.
2. Программный код и код алгоритмов динамических команд, настроенных в сценарии для задач, выполняется **&НаКлиенте**.
3. Переменные подсистемы (**_Процесс**, **_Задача** и т.д.) не инициализируются и недоступны, т.к. все необходимые значения можно получить из реквизитов формы.
4. **Программный код** следует использоваться при вычислении простых выражений. Для многострочного кода следует использовать **алгоритмы**.
5. Для получения реквизитов объектов в конфигурациях, разработанных с использованием БСП, следует использовать типовые функции общего модуля **ОбщегоНазначенияУТВызовСервера**:
 - Функция **ЗначенияРеквизитовОбъекта(Ссылка, Реквизиты) Экспорт**
 - Функция **ЗначениеРеквизитаОбъекта(Ссылка, ИмяРеквизита) Экспорт**
 - Функция **ЗначенияРеквизитовОбъектов(МассивСсылок, ИменаРеквизитов, ВыбратьРазрешенные = Ложь) Экспорт**
 - Функция **ЗначениеРеквизитаОбъектов(МассивСсылок, ИмяРеквизита, ВыбратьРазрешенные = Ложь) Экспорт**.
6. Для выполнения кода на сервере следует использовать функцию **ВыполнитьНаСервере(ПрограммныйКод)**. В параметр **ПрограммныйКод** следует передать программный код, который требуется выполнить на сервере. Функция возвращает произвольное значение (переменная **Результат**). В функции доступен основной реквизит **Объект** (тип **бип_Задача**) и контекст формы задачи.

Пример с вызовом функции **ВыполнитьНаСервере**:

Описание шага "Выполнить действие"

Инструкция...

Справка по использованию команд ?

Добавить

Заголовок кнопки	Описание	Программный код (&НаКлиенте)	Алгоритм
Два плюс два	Вычисляет выражение 2+2	ПоказатьПредупреждение(, "2+2 = " + ВыполнитьНаСервере("Результат = 2+2"))	

Описание

Параметры

Кнопки

★ Выполнить действие

Выполнена В работе Показывать описание Подзадача Передать... Закрыть ?

№: 000000243 от: 31.10.2022 16:55

Основание: Сценарий №15

Срок: : : Важная

Подробное описание не задано

Комментарий:

Два плюс два

Вычисляет выражение 2+2

Основные Дополнительно

2+2 = 4

OK

Встроенный редактор программного кода (начиная с версии 1.4.6)

Некоторые возможности **встроенного редактора программного кода**:

- Использование индивидуальных **шаблонов программного кода**, доступных для быстрого выбора при редактировании программного кода.
 - При вводе символов, система предлагает список не только ключевых слов встроенного языка, но и готовых шаблонов. Предварительный выбор такого шаблона отображается с просмотром всего текста шаблона, который встраивается в программный код (отображается серым цветом).
 - Для установки курсора в тексте программного кода после выбора шаблона, используется установка курсора (символы «<?»> в тексте шаблона).
 - Шаблоны могут быть настроены индивидуально для каждого пользователя и/или сразу для всех.
- Система отображает знаки табуляции и «лишние» проблемы в конце строк (символ *).
- Автоматическое добавление символа «|» в текстах запроса:
- Автоматическая установка пробелов после знаков «//».
- Автоматическая установка знаков «;» в конце строк.
- Автоматические заглавные буквы для ключевых слов.
- Автоматическое добавление кавычек.
- Цветовое выделение переменных и функций подсистемы.
- Подсказки для функций переменных подсистемы.
- Подсветка встречаемости выделенного текста с отметкой строк, содержащих выделенный текст.
- Групповое комментирование/раскомментирование выделенного текста клавишами **Shift + /**.
- Тёмный режим.

```
1 Запрос = Новый Запрос;
2 шаблон: Запрос = Новый Запрос <...>
3
4 Запрос.Текст = "";
5
6 Выборка = Запрос.Выполнить().Выбрать();
7 Пока Выборка.Следующий() Цикл
8   <?>
9 КонецЦикла;
10
11 ВыходныеДанные = РезультатЗапроса.ВыгрузитьКолонку("Ссылка");
12 // Выборка = Запрос.Выполнить().Выбрать();
```

```
Запрос = Новый Запрос;
Запрос.Текст = "ВЫБРАТЬ
Данные.Ссылка
ИЗ
Справочник." + ИмяСправочника + " КАК Данные";
```

```
Запрос.Текст = "ВЫБРАТЬ
```

```
1 // Алгоритм получения
2 // Может использоваться
3 // В качестве примера
```

```
5 ИмяСправочника = "бип_Инструкции";
6 ДатаЗадачи = _Задача.Дата;
7
8 Запрос = Новый
9
10 Запрос.Текст =
11
12
13
14
15 РезультатЗапроса = Запрос.Выполнить().Выгрузить();
16
17 ВыходныеДанные = РезультатЗапроса.ВыгрузитьКолонку("Ссылка");
18 // Выборка = Запрос.Выполнить().Выбрать();
19 //
```

Тип: бип_Задача
Описание:
Предшествующая текущему шагу задача.
Текущая задача, если выполняется алгоритм при запуске задачи или алгоритм проверки автоматического завершения задачи по условию.
Подробнее: См. справку

```
7
8 Запрос = Новый Запрос;
9
10 Запрос.Текст = "ВЫБРАТЬ
11   Данные.Ссылка
12   ИЗ
13   Справочник." + ИмяСправочника + " КАК Данные";
14
15 РезультатЗапроса = Запрос.Выполнить().Выгрузить();
16
17 ВыходныеДанные = РезультатЗапроса.ВыгрузитьКолонку("Ссылка");
18 // Выборка = Запрос.Выполнить().Выбрать();
19 //
```

Варианты выбора:

Счёт оплачен
Ожидание оплаты
Счёт не оплачен через 3 дня

```
1 Заказ = _Процесс.Объект;
2
3 Если ЗначениеЗаполнено(Заказ.Состояние = Новый Структ
4   Документы.ЗаказКлиента.Р
5   _Результат = Состояние.С
6
7
8
9
10
11 Иначе
12   _Результат = Ложь
13
14
15 КонецЕсли
```

Варианты выбора:

Счёт оплачен
Ожидание оплаты
Счёт не оплачен через 3 дня

```
1 Заказ = _Процесс.Объект;
2
3 Если ЗначениеЗаполнено(Заказ) Тогда
4   Состояние = Новый Структура("СуммаОплаты");
5   Документы.ЗаказКлиента.РассчитатьСостояние(Заказ, Заказ.Договор, Состояние);
6   _Результат = Состояние.СуммаОплаты >= Заказ.СуммаДокумента
7
8 Иначе
9   _Результат = Ложь
10
11 КонецЕсли
```

Пример программной настройки меню выбора для чат-ботов Telegram

Ниже приводится описание функционала расширения  «Сигнал», начиная с версии 1.2.3.7.

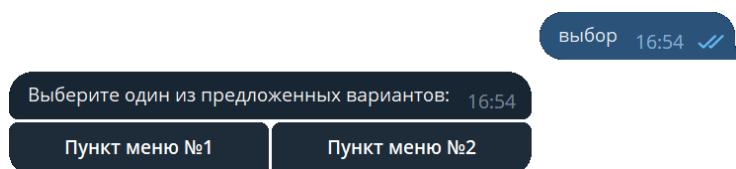
Меню выбора может быть реализовано с помощью настроек справочника **Ключевые слова** 2 способами:

- настройкой ключевых слов с видом **Выбор**,
- настройка с помощью алгоритмов.

Смотрите [Руководство пользователя «Сигнал»](#) (раздел **Ключевые слова**).

Ниже будет рассмотрен пример настройки с помощью второго способа.

Если добавить новое ключевое слово с видом **Выполнение алгоритма** и в этом алгоритме в переменную **_Результат** передать **массив**, то элементы этого массива будут являться кнопками меню выбора.

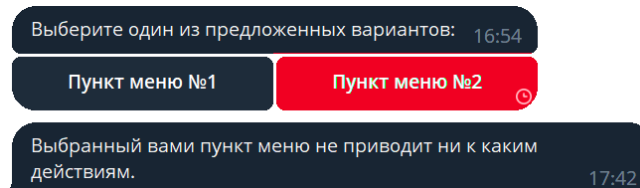
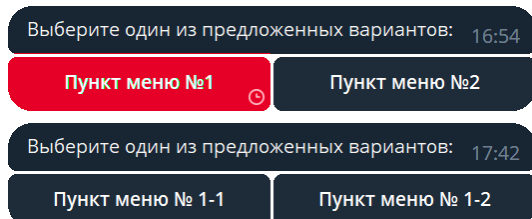


Для обработки нажатия кнопок добавим новое ключевое слово, подчиненное ключевому слову **Выбор**.

Это ключевое слово будет, также, с видом **Выполнения алгоритма**.

Для каждой кнопки определим значение переменной **_Результат**.

Для кнопки «Пункт меню №1» это будет новый **массив кнопок**.
Для кнопки «Пункт меню №2» это будет **сообщение** от чат-бота.



Для обработки подпунктов меню «Пункт меню №1» добавим новое ключевое слово, подчиненное предыдущему и настроим для него **алгоритм обработки выбора**.

В этот раз, вместо массива в переменную **_Результат** передадим **Структуру**, которая будет содержать следующие настройки:

- <Заголовок выбора> (тип **Строка**) – заголовок выбора пунктов меню;
- <Выбор> (тип **Массив**) – массив кнопок выбора;
- <ПодтверждатьВыбор> (тип **Булево**) – признак подтверждения выбранного действия;
- <ЗаголовокКомментария> (тип **Строка**) – если указан, то чат-бот попросит отправить дополнительное сообщение с указанным заголовком.

```

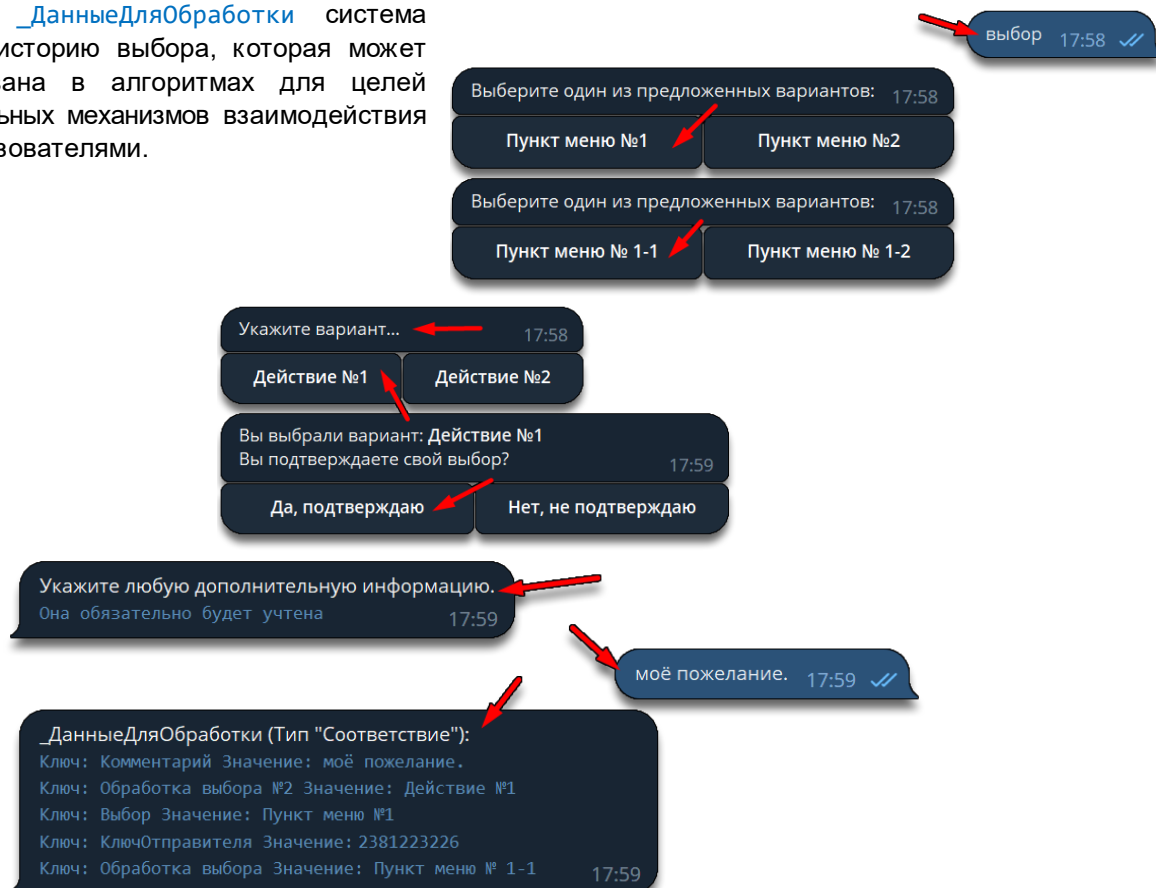
1 ТекущийОбработчикВыбора = _ДанныеДляОбработки.Получить("ТекущийОбработчикВыбора");
2 ВыбранноеЗначение = _ДанныеДляОбработки[ТекущийОбработчикВыбора];
3
4 Если ВыбранноеЗначение = "Пункт меню № 1-1" Тогда
5     Выбор = Новый Массив;
6     Выбор.Добавить("Действие №1");
7     Выбор.Добавить("Действие №2");
8
9     _Результат = Новый Структура;
10    _Результат.Вставить("ЗаголовокВыбора", "Укажите вариант...");
11    _Результат.Вставить("Выбор", Выбор);
12    _Результат.Вставить("ПодтверждатьВыбор", Истина);
13    _Результат.Вставить("ЗаголовокКомментария", "Укажите любую дополнительную информацию. Она обязательно будет учтена");
14
15 ИначеЕсли ВыбранноеЗначение = "Пункт меню № 1-2" Тогда
16     _Результат = "Текущая дата: " + ТекущаяДата();
17
18 КонецЕсли

```

Для обработки нажатия кнопок «Действие №1» и «Действие №2» добавим алгоритм, который выведет в чат-бот «содержание» переменной **_ДанныеДляОбработки** (тип **Соответствие**).

fn	Выбор	Меню выбора
fn	Обработка выбора	Обработка выбора (меню "Выбор")
fn	Обработка выбора №2	Обработка выбора №2 (меню "Выбор")
fn	Вывод истории выбора	Вывод истории выбора

В переменной **_ДанныеДляОбработки** система сохраняет всю историю выбора, которая может быть использована в алгоритмах для целей реализации реальных механизмов взаимодействия чат-ботов с пользователями.



Дополнительные возможности программной настройки чат-ботов Telegram

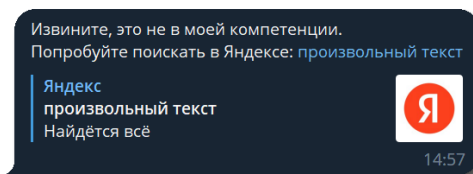
ОБРАБОТКА НЕИЗВЕСТНОГО ЗАПРОСА

По умолчанию, при получении чат-ботом сообщения, для которого не предусмотрено ключевое слово, чат-бот ответит стандартной фразой, указанной в настройках чат-бота.

Основные Авторизация Дополнительно **Тексты**

Ответ на неизвестный запрос:

Извините, это не в моей компетенции.
Попробуйте поискать в Яндекске



произвольный текст 14:57 ✓

Для обработки таких «*неизвестных запросов*» можно использовать **ключевое слово**.

Для этого достаточно выбрать в контекстном меню поля «**Ключевое слово**» пункт **Неизвестный запрос**.

В поле «**Ключевое слово**» будет добавлена строка регулярного выражения.

☆ /(\\S+\\s)/g (Ключевые слова)

Записать и закрыть Записать

Еще - ?

Чат-бот: Основной Код: 000000039

Ключевое слово: /(\\S+\\s)/g

Вид: ☐ Текстовый ответ ☒ Выполнение алгоритма ☐ Выбор ☐ ключевое слово

Алгоритм: Обработка неизвестного запроса

В качестве примера можно настроить обработку неизвестного запроса с помощью алгоритма.

Самый простой алгоритм обработки неизвестного запроса может быть описан одной строкой.

```
1 _Результат = "Извините, я не знаю, что вам на это ответить."
```

произвольный текст 15:12 ✓

Извините, я не знаю, что вам на это ответить. 15:12

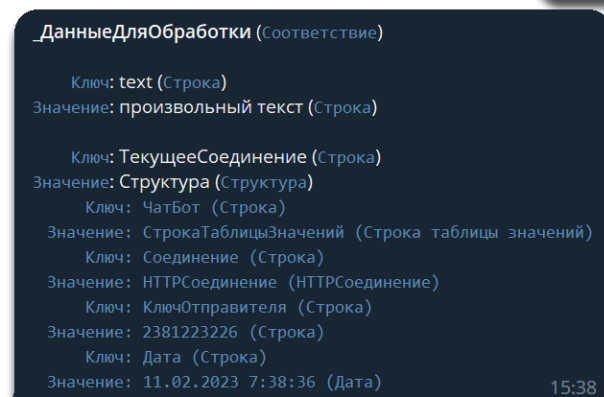
Чтобы получить **данные сообщения**, можно использовать переменную **_ДанныеДляОбработки**.

В качестве примера выведем содержимое переменной **_ДанныеДляОбработки** с помощью встроенной функции **нс_Telegram.ПолучитьДанныеБуфера**.

```
1 _Результат = нс_Telegram.ПолучитьДанныеБуфера(_ДанныеДляОбработки)
```

произвольный текст 15:38 ✓

Переменная **_ДанныеДляОбработки** (тип **Соответствие**) может содержать разные значения, в зависимости от полученных данных.



СОХРАНЕНИЕ ВЛОЖЕНИЙ

В переменной могут присутствовать поля, связанные с **вложениями**, присланными пользователем.

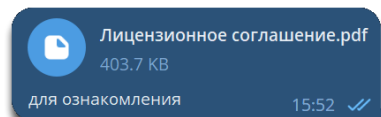
С помощью встроенной функции `нс_Телеграм.СохранитьВложения` можно сохранить полученные вложения в указанный каталог. Подробнее, см. [Процедуры и функции расширения «Сигнал»](#).

Система умеет сохранять вложения следующих типов: **фотография, видео, голосовое сообщение, произвольный файл**.

В качестве примера сохраним полученный от пользователя файл в папку «C:\temp\» и выведем содержимое переменной `_ДанныеДляОбработки`.

```
1 нс_Телеграм.СохранитьВложения(_ДанныеДляОбработки, "C:\Temp\");  
2 _Результат = нс_Телеграм.ПолучитьДанныеБуфера(_ДанныеДляОбработки)
```

Перед выполнением сохранения, система предварительно проверяет, есть ли во входящем сообщении вложения.



`_ДанныеДляОбработки` (Соответствие)

Ключ: СохранениеФайла (Строка)

Значение: Отсутствуют файлы для сохранения (Строка)

При неудачной попытке сохранения в переменной `_ДанныеДляОбработки` будет добавлено значение с ключом `<error>`.

`_ДанныеДляОбработки` (Соответствие)

Ключ: error (Строка)

Значение: Не удалось загрузить файл: "Лицензионное соглашение.pdf". См. журнал регистрации. (Строка)

`_ДанныеДляОбработки` (Соответствие)

Ключ: document (Строка)

Значение: Структура (Структура)

Ключ: ПутьКФайлу (Строка)

Значение: C:\temp\Лицензионное соглашение.pdf (Строка)

Ключ: caption (Строка)

Значение: для ознакомления (Строка)

Ключ: ТекущееСоединение (Строка)

Значение: Структура (Структура)

Ключ: ЧатБот (Строка)

Значение: СтрокаТаблицыЗначений (Строка таблицы значений)

Ключ: Соединение (Строка)

Значение: HTTPСоединение (HTTPСоединение)

Ключ: КлючОтправителя (Строка)

Значение: 2381223226 (Строка)

Ключ: Дата (Строка)

Значение: 11.02.2023 7:52:10 (Дата)

15:52

ПЕРЕАДРЕСАЦИЯ ОБРАБОТКИ ВХОДЯЩЕГО СООБЩЕНИЯ

Если внутри алгоритма требуется **передать** обработку входящего сообщения другому обработчику ключевых слов, то для этого в переменную `_ДанныеДляОбработки` следует добавить ключ `<Переадресация>` и в качестве его значения передать **ссылку на ключевое слово**. При этом, можно передать только ключевые слова 1-го уровня.

В качестве примера передадим обработку неизвестного запроса ключевому слову «Выбор», которое приводилось в качестве примера в разделе [Пример программной настройки меню выбора для чат-ботов Telegram](#).

```
1 _ДанныеДляОбработки.Вставить("Переадресация", Справочники.нс_КлючевыеСлова.НайтиПоНаименованию("Выбор"));  
2 _Результат = Истина
```

произвольный текст 16:18 ✓

Выберите один из предложенных вариантов: 16:18

Пункт меню №1

Пункт меню №2

