



# Использование алгоритмов в программе Конструктор Бизнес Процессов<sup>©</sup>

[Таблица использования алгоритмов](#)

[Переменные](#)

[Дополнительные реквизиты шагов сценария](#)

[Процедуры и функции подсистемы](#)

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

[Вызов алгоритмов по имени](#)

[Динамические команды на форме задачи](#)

Версия 1.4.5.78

Настройка выбора варианта "Контроль отгрузки"

Готово Проверить

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   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
7   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
8   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
9   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
10  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
11  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
12  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
13  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
14  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
15  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
16  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
17  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
18  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
19  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
20  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
21  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
22  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
23  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
24  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
25  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
26  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
27  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
28  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
29  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
30  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
31  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
32  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
33  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
34  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
35  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
36  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
37  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
38  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
39  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
40  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
41  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
42  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
43  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
44  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
45  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
46  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
47  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
48  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
49  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
50  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
51  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
52  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
53  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
54  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
55  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
56  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
57  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
58  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
59  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
60  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
61  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
62  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
63  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
64  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
65  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
66  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
67  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
68  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
69  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
70  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
71  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
72  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
73  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
74  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
75  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
76  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
77  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
78  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
79  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
80  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
81  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
82  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
83  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
84  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
85  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
86  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
87  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
88  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
89  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
90  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
91  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
92  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
93  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
94  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
95  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
96  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
97  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
98  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
99  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
100 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
```

Выборка = Запрос.Выполнить().Выбрать();

\_Результат = Истина;

ЕстьОрдера = Ложь;

Пока Выборка.Следующий() Цикл

ЕстьОрдера = Истина;

\_Результат = \_Результат И Выборка.Отгружен

КонецЦикла;

\_Результат = ЕстьОрдера И \_Результат

ВСТРОЕННЫЙ РЕДАКТОР ИСПОЛНЯЕМОГО ПРОГРАММНОГО КОДА.






БизнесПроцессы 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>	<a href="#">бип Процесс</a>	Общее	Текущий процесс.
<code>_ПроцессОбъект</code>	<a href="#">бип Процесс</a>	Общее	Текущий процесс. Доступен только в <b>алгоритмах, выполняемых при запуске</b> новых процессов.
<code>_ТаблицаДанных</code>	<a href="#">ТаблицаЗначений</a>	Общее	Таблица данных текущего процесса. Доступна только в <b>алгоритмах, выполняемых при запуске</b> новых процессов и только, если в сценарии настроено использование таблицы данных. В прочих случаях, доступ к таблице данных возможен через <code>_Процесс.ТаблицаДанных</code> .
<code>_Процесс.Основание</code>	<a href="#">бип Процесс</a>	Общее	Родительский процесс текущего процесса.
<code>_Задача</code>	<a href="#">бип Задача</a>	Общее	Предшествующая текущему шагу задача. Текущая задача, если выполняется <b>алгоритм при запуске задачи</b> или <b>алгоритм проверки автоматического завершения задачи по условию</b> .
<code>_ВыполненаВручную</code>	<a href="#">Булево</a>	Общее	Признак интерактивного выполнения задачи (нажатием кнопки <b>Выполнена</b> ). Используется только в <b>алгоритмах проверки автоматического завершения задач по условию</b> .
<code>_ИмяШага</code>	<a href="#">Строка</a>	Общее	Имя текущего шага, как оно задано в настройках сценария. Например, <code>&amp;Действие1</code> . Используется в алгоритмах настроек адресации.
<code>_ПредставлениеШага</code>	<a href="#">Строка</a>	Общее	Представление текущего шага, как оно задано в настройках сценария. Используется в алгоритмах настроек адресации.
<code>_ПараметрN</code> (N = 1, 2, 3, 4, 5)	<a href="#">Произвольный</a>	Общее	Фиксированное, константное значение, указанное в настройках сценария.
<code>_ПараметрИмя</code> (N = 1, 2, 3, 4, 5)	<a href="#">Строка</a>	Общее	Имя фиксированного, константного значения, указанное в настройках сценария.
<code>_Исполнитель</code>	<a href="#">Строка</a> <a href="#">Пользователи</a>	Общее	Исполнитель текущей задачи. Доступен только в <b>алгоритмах, выполняемых при запуске новых задач</b> и при использовании алгоритмов в расширении «Сигнал»: для <b>программного определения получателей оповещений</b> и <b>при выполнении алгоритмов в чат-ботах</b> – 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>_ДанныеДляОбработки</code>	<a href="#">Строка</a>	Общее	Переменная используется при обработке данных, полученных от чат-ботов расширения «Сигнал». Содержит текст входящего сообщения, которое требуется обработать программно.
<code>_Результат</code>	<a href="#">Булево</a> <a href="#">Произвольный</a>	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 <b>ИмяДляВызова</b> 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 <code>_Результат</code> передаётся значение <b>Исполнителя</b> , <b>Роли исполнителя</b> или <b>Массив</b> , содержащий исполнителей и/или роли. При этом для групповых задач, если <b>Массив</b> содержит <b>Роль исполнителя</b> , то

Имя	Тип	Назначение	Описание
			исполнителем задачи устанавливается <i>роль</i> , а не пользователи, обладающие данной ролью. Пример: <b>Массив</b> содержит значения « <i>Иванов</i> », « <i>Петров</i> », « <i>Логисты</i> ». При обработке <b>группового</b> шага сценария будет создано 3 задачи: задача для <i>Иванова</i> , задача для <i>Петрова</i> и задача для <i>Логистов</i> . Любой из логистов сможет взять задачу для исполнения. При использовании <b>Массива</b> для <b>не групповой</b> задачи, в массиве учитываются только исполнители, роли в массиве игнорируются.
_Результат	<a href="#">Булево</a>	Проверка условий	Условие проверяется по значению переменной. Условие выполнено, если <b>_Результат = Истина</b> .
_РасшифровкаРезультата	<a href="#">Строка</a>	Проверка условий	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 <b>Автоматически завершать задачу по условию</b> . – на карте процесса, как дополнительная информация для шагов вида <b>Условие и ВыборВарианта</b> .
_Источник	<a href="#">СправочникСсылка</a> <a href="#">ДокументСсылка</a>	Автозапуск по событию	Источник возникновения события, которое привело к запуску текущего процесса.
_ИсточникДо	<a href="#">Структура</a>	Автозапуск по событию	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.
_ЭтоНовый	<a href="#">Булево</a>	Автозапуск по событию	Признак первой записи источника события (Признак создания).
_ВыходныеДанные	<a href="#">Массив</a>	Автозапуск по условию	Массив, в который могут быть добавлены произвольные данные. Используется для передачи данных в создаваемые на основании сценария процессы. см. <a href="#">_РазделятьПоПроцессам</a>
_РазделятьПоПроцессам	<a href="#">Булево</a>	Автозапуск по условию	Признак создания нескольких процессов. Каждому элементу массива <b>_ВыходныеДанные</b> соответствует один создаваемый процесс. Если <b>_РазделятьПоПроцессам = Ложь</b> , все объекты из массива <b>_ВыходныеДанные</b> будут записаны в табличную часть <b>Объекты</b> нового процесса.
Параметры.Объект	<a href="#">Произвольный</a>	Автозапуск по условию	Значение, которое будет указано в качестве объекта нового процесса.
Параметры.Сценарий	<a href="#">бип_Сценарии</a>	Автозапуск по условию	Текущий сценарий.

## Дополнительные реквизиты шагов сценария


1. Для получения данных о пройденных шагах сценария можно обращаться к ним **по имени** в виде &Действие1, &Условие2 и т.п.
2. **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	<a href="#">Дата</a>	Получение доп. сведений о шагах сценария	Дата создания шага.
.ДатаВыполнения	<a href="#">Дата</a>		Дата выполнения шага.
.Просрочено	<a href="#">Булево</a>		Признак просроченной задачи.
.Результат	<a href="#">Строка</a>		Результат проверки  <b>Условия</b> («Да»/«Нет»). Результат  <b>Выбора варианта</b> .
.КоличествоПовторов	<a href="#">Число</a>		Количество повторов текущего шага.

Имя функции	Параметры	Возвращаемое значение	Описание
ДанныеШага	<ul style="list-style-type: none"> <li>– <b>Процесс</b> (<a href="#">бип_Процесс</a>) процесс, к которому относится шаг.</li> <li>– <b>ИмяНомер</b> (<a href="#">Строка</a>, <a href="#">Число</a>) имя шага сценария или номер шага на карте процесса. Например, «&amp;Действие1» или 3.</li> </ul>	<p><a href="#">Структура</a> состоит из:</p> <ul style="list-style-type: none"> <li>– <b>Период</b> (<a href="#">Дата</a>) Дата шага.</li> <li>– <b>Процесс</b> (<a href="#">бип_Процесс</a>) Процесс.</li> <li>– <b>НомерШага</b> (<a href="#">Число</a>) Номер шага.</li> <li>– <b>Имя</b> (<a href="#">Строка</a>) Имя шага.</li> <li>– <b>УсловиеВыбор</b> (<a href="#">Строка</a>) Результат обработки выбора или условия.</li> <li>– <b>Действие</b> (<a href="#">бип_Задача</a>, <a href="#">бип_Процесс</a>) Задача или вложенный процесс.</li> <li>– <b>ОтложеннаяДата</b> (<a href="#">Дата</a>) Дата отложенного запуска шага.</li> <li>– <b>РасшифровкаРезультата</b> (<a href="#">Строка</a>) Подробное описание результата проверки условия.</li> </ul> <p><a href="#">Массив</a></p>	<p>Возвращает структуру данных шага процесса по его имени или порядковому номеру на карте процесса.</p> <p>Если по имени найдено несколько шагов, то функция возвращает массив структур.</p>
НайтиЗадачу	<ul style="list-style-type: none"> <li>– <b>Процесс</b> (<a href="#">бип_Процесс</a>) процесс, к которому относится задача.</li> <li>– <b>Имя</b> (<a href="#">Строка</a>) имя шага сценария в виде «&amp;ДействиеN».</li> </ul>	<p><a href="#">Структура</a> состоит из:</p> <ul style="list-style-type: none"> <li>– <b>Задача</b> (<a href="#">бип_Задача</a>, <a href="#">бип_Процесс</a>) Найденная задача или вложенный процесс. Если задача или процесс не найдены, то возвращается пустое значение <a href="#">бип_Задача.ПустаяСсылка</a>.</li> <li>– <b>Автор</b> (<a href="#">Пользователи</a>) Автор задачи.</li> <li>– <b>Исполнитель</b> (<a href="#">Пользователи</a>) Текущий исполнитель задачи.</li> </ul> <p><a href="#">Массив</a></p>	<p>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.</p> <p>Если по имени найдено несколько групповых задач, то функция возвращает массив структур.</p>
СостояниеЗадачи	<ul style="list-style-type: none"> <li>– <b>Задача</b> (<a href="#">бип_Задача</a>) задача.</li> <li>– <b>НаДату</b> (<a href="#">Дата</a>, <a href="#">Неопределено</a>) дата получения состояния. Если не указана, будет получено текущее состояние.</li> </ul>	<p><a href="#">Структура</a> состоит из:</p> <ul style="list-style-type: none"> <li>– <b>Период</b> (<a href="#">Дата</a>) Дата установки состояния.</li> <li>– <b>Состояние</b> (<a href="#">бип_СостоянияЗадач</a>) Состояние задачи.</li> <li>– <b>ПричинаОстановкиОтмены</b> (<a href="#">бип_ПричиныОстановок</a>, <a href="#">бип_ПричиныОтмены</a>) Причина остановки/отмены задачи.</li> <li>– <b>Комментарий</b> (<a href="#">Строка</a>) Дополнительный комментарий при остановке/отмене задачи.</li> <li>– <b>НомерШага</b> (<a href="#">Число</a>) Порядковый номер шага задачи в истории процесса.</li> </ul>	<p>Возвращает состояние задачи на указанную дату.</p>

Имя функции	Параметры	Возвращаемое значение	Описание
ДанныеПроцесса	<ul style="list-style-type: none"> <li>– <b>Процесс</b> (<a href="#">бип_Процесс</a>) процесс, по которому требуется получить данные.</li> </ul>	<p><a href="#">Структура</a> состоит из:</p> <ul style="list-style-type: none"> <li>– <b>Ссылка</b> (<a href="#">бип_Процесс</a>) Ссылка на процесс. Если процесс не найден (передана некорректная ссылка), то возвращается <a href="#">бип_Процесс.ПустаяСсылка</a>.</li> <li>– <b>Автор</b> (<a href="#">Пользователи</a>) Автор процесса.</li> </ul>	Возвращает структуру, содержащую данные указанного процесса. В структуре содержится ссылка на процесс и автор процесса.
_СохранитьЗначение	<ul style="list-style-type: none"> <li>– <b>Имя</b> (<a href="#">Строка</a>) имя, под которым сохраняется значения.</li> <li>– <b>Значение</b> (<a href="#">Произвольный</a>) сохраняемое значение. Если <a href="#">Неопределено</a>, сохраненное ранее по указанному имени значение удаляется.</li> </ul>		Сохраняет переданное значение по указанному имени. Значение используется <b>только</b> в рамках текущего процесса.
_ПолучитьЗначение	<ul style="list-style-type: none"> <li>– <b>Имя</b> (<a href="#">Строка</a>) имя получаемого значения.</li> </ul>	<p><a href="#">Произвольный</a> Полученное значение. <b>Неопределено</b>, если значение по имени не найдено.</p>	Возвращает сохраненное значение по указанному имени ( <b>только</b> в рамках текущего процесса).

### 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_Процесс)`.

### 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)`.

## Вызов алгоритмов по имени

### 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. **Программный код** следует использоваться при вычислении простых выражений. Для многострочного кода следует использовать **алгоритмы**.
4. Для получения реквизитов объектов в конфигурациях, разработанных с использованием БСП, следует использовать типовые функции общего модуля **ОбщегоНазначенияУТВывозСервера**:
  - **Функция ЗначенияРеквизитовОбъекта(Ссылка, Реквизиты) Экспорт**
  - **Функция ЗначениеРеквизитаОбъекта(Ссылка, ИмяРеквизита) Экспорт**
  - **Функция ЗначенияРеквизитовОбъектов(МассивСсылок, ИменаРеквизитов, ВыбратьРазрешенные = Ложь) Экспорт**
  - **Функция ЗначениеРеквизитаОбъектов(МассивСсылок, ИмяРеквизита, ВыбратьРазрешенные = Ложь) Экспорт.**
5. 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**:

