# Основные блоки

## Старт

С блока Старт начинается работа диаграммы. В диаграмме может быть только один такой блок.

|  |  |
| --- | --- |
| Записать в лог | [Любой тип данных] Числовая, текстовая константа (в кавычках), имя переменной (начинающееся с $) или любое выражение, значение которого будет выведено в лог в процессе работы сценария робота. Значение может в том числе иметь сложный тип данных, например Список, Таблица или Словарь. |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Логировать в оркестратор | Если выбрано, то текст будет также логироваться в оркестратор, если подключение к нему активно |
| Скриншот в оркестратор | Если выбрано, то будет создан скриншот экрана и отправлен в оркестратор, если подключение к нему активно |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то, если это главная диаграмма, будет использоваться значение "Handle", иначе будет использоваться значение блока "Старт" главной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то, если это главная диаграмма, будет использоваться значение "Bottom", иначе будет использоваться значение блока "Старт" главной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то, если это главная диаграмма, будет использоваться значение "Debug", иначе будет использоваться значение блока "Старт" главной диаграммы |

## Конец

Блок обозначает завершение работы сценария или возврат из диаграммы подпроцесса в основной процесс. В диаграмме можно использовать более одного блока Конец. В диаграмме подпроцесса должен быть хотя бы один такой блок, иначе после завершения работы подпроцесса не произойдёт возврат в основной процесс.

|  |  |
| --- | --- |
| Записать в лог | [Любой тип данных] Числовая, текстовая константа (в кавычках), имя переменной (начинающееся с $) или любое выражение, значение которого будет выведено в лог в процессе работы сценария робота. Значение может в том числе иметь сложный тип данных, например Список, Таблица или Словарь. |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Логировать в оркестратор | Если выбрано, то текст будет также логироваться в оркестратор, если подключение к нему активно |
| Скриншот в оркестратор | Если выбрано, то будет создан скриншот экрана и отправлен в оркестратор, если подключение к нему активно |

## Процесс

Блок позволяет создавать сценарии, состоящие из нескольких диаграмм. Выполнение сценария продолжится с блока Старт указанной диаграммы проекта. Выполнение сценария вернётся к текущему блоку и продолжится в текущей диаграмме как только будет достигнут блок Конец во внешней диаграмме.

|  |  |
| --- | --- |
| Имя диаграммы | [Текст] Имя файла диаграммы, в которой продолжится выполнение сценария. Чтобы создать новый подпроцесс, задайте в этом свойстве имя новой диаграммы, а затем нажмите на кнопку "Открыть" в самом блоке. |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Аргументы | [Текст] Аргументы для вызова диаграммы. Для редактирования значения необходимо использовать кнопку на блоке с подсказкой "Аргументы" |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Комментарий

Блок позволяет разместить в диаграмме любой текстовый комментарий. Блок не является частью сценария



## ToDo

Блок можно использовать как заглушку или напоминание для планируемых к реализации функций. В отличие от блока Комментарий данный блок является элементом сценария, но не выполняет никаких действий.



## Лог

Позволяет выводить в лог произвольные сообщения и/или значения переменных в процессе работы сценария робота. Сообщения будут доступны для просмотра в панели Лог программы-дизайнера, а также в текстовых файлах логов программы-робота.

|  |  |
| --- | --- |
| Значение | [Любой тип данных] Числовая, текстовая константа (в кавычках), имя переменной (начинающееся с $) или любое выражение, значение которого будет выведено в лог в процессе работы сценария робота. Значение может в том числе иметь сложный тип данных, например Список, Таблица или Словарь. |
| Сделать скриншот | Создает скриншот и сохраняет его в папке с логами |
| Оркестратор. Логировать | Если выбрано, то текст будет также логироваться в оркестратор, если подключение к нему активно |
| Оркестратор. Скриншот | Если выбрано, то будет создан скриншот экрана и отправлен в оркестратор, если подключение к нему активно |
| Оркестратор. Уровень | Уровень сообщения при отправке лога в оркестратор |
| Отображать в статусе | Если выбрано, то текст будет также выводиться в статусе, в верхней части экрана |

## Условие

Блок проверяет указанное условие на истинность и, в зависимости от результата, выполнение сценария продолжается в сторону выхода "Да" (условие выполнилось) или "Нет" (условие не выполнилось). Например, условие $a == "Привет" проверяет, равно ли значение переменной $a значению "Привет", а условие $Result > 5 проверяет, превышает ли значение переменной $Result значение 5. Условие может содержать математические и логические операции, константы, функции, другие переменные и т.д. Примеры выражений, которые можно использовать в условиях, доступны по нажатию на кнопку ... рядом со свойством Условие.

|  |  |
| --- | --- |
| Условие | [Любое выражение] В зависимости от значения этого выражения выполнение сценария продолжится по выходу "Да" или "Нет". Значение выражения, эквивалентное 0 (нулю) или $False приравнивается к "Нет", все остальные значения - приравниваются к "Да". Например, если в переменную $a было ранее записано значение 5, то условие $a>2 является истинным, а условие $a == 6 является ложным. |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Switch/Case

Оператор switch сравнивает выражение с указанными значениями и выходит по указанному коннектору. Если совпадений не будет обнаружено, то выходит по коннектору "По умолчанию"

|  |  |
| --- | --- |
| Выражение | [Любой тип] Сравниваемое выражение |
| Условие 1 | [Любой тип] Условие 1. При совпадении будет произведен выход по коннектору 1, а также выполнено выражение 1 если оно задано. Если коннектор не задан, то выход будет произведен по умолчанию |
| Выполнить выражение 1 | [Выражение] При выполнении условия 1 будет выполнено данное выражение и произведен выход по соответствующему коннектору. Если коннектор не подключен, то выход будет производится по умолчанию |
| Условие 2 | [Любой тип] Условие 2. При совпадении будет произведен выход по коннектору 2, а также выполнено выражение 2 если оно задано. Если коннектор не задан, то выход будет произведен по умолчанию |
| Выполнить выражение 2 | [Выражение] При выполнении условия 2 будет выполнено данное выражение и произведен выход по соответствующему коннектору. Если коннектор не подключен, то выход будет производится по умолчанию |
| Условие 3 | [Любой тип] Условие 3. При совпадении будет произведен выход по коннектору 3, а также выполнено выражение 3 если оно задано. Если коннектор не задан, то выход будет произведен по умолчанию |
| Выполнить выражение 3 | [Выражение] При выполнении условия 3 будет выполнено данное выражение и произведен выход по соответствующему коннектору. Если коннектор не подключен, то выход будет производится по умолчанию |
| Условие 4 | [Любой тип] Условие 4. При совпадении будет произведен выход по коннектору 4, а также выполнено выражение 4 если оно задано. Если коннектор не задан, то выход будет произведен по умолчанию |
| Выполнить выражение 4 | [Выражение] При выполнении условия 4 будет выполнено данное выражение и произведен выход по соответствующему коннектору. Если коннектор не подключен, то выход будет производится по умолчанию |
| Условие 5 | [Любой тип] Условие 5. При совпадении будет произведен выход по коннектору 5, а также выполнено выражение 5 если оно задано. Если коннектор не задан, то выход будет произведен по умолчанию |
| Выполнить выражение 5 | [Выражение] При выполнении условия 5 будет выполнено данное выражение и произведен выход по соответствующему коннектору. Если коннектор не подключен, то выход будет производится по умолчанию |
| Выполнить выражение по умолчанию | [Выражение] Если ни одно из условий не будет выполнено, то будет выполнено данное выражение |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Присвоить значение переменной

Блок устанавливает новые значения для одной или нескольких переменных. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. Переменная может быть выбрана из числа объявленных в панели Переменные, также может быть вручную введено имя новой переменной или ранее использовавшейся в других блоках переменной, даже если эта переменная не была объявлена в панели Переменные. Примеры выражений, которые можно использовать в свойствах Значение, доступны по нажатию на кнопки ... рядом с соответствующими свойствами.

|  |  |
| --- | --- |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 6 | [Переменная] Имя шестой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 6 | [Любой тип данных] Новое значение шестой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 7 | [Переменная] Имя седьмой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 7 | [Любой тип данных] Новое значение седьмой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 8 | [Переменная] Имя восьмой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 8 | [Любой тип данных] Новое значение восьмой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 9 | [Переменная] Имя девятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 9 | [Любой тип данных] Новое значение девятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 10 | [Переменная] Имя десятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 10 | [Любой тип данных] Новое значение десятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Цикл Для каждого (Таблица Данных)

Циклично перебираются все строки Таблицы Данных, которые выводятся в отдельный коннектор. После перебора всех строк таблицы выполнение сценария продолжается в сторону нижнего выхода блока. Не забудьте соединить выход последнего блока в теле цикла со входом данного блока

|  |  |
| --- | --- |
| Таблица | [Таблица Данных] Таблица данных, в которой перебираются строки |
| Строка | [Ссылка] Возвращается ссылка на строку |
| Индекс строки | [Число] Порядковый номер строки. Нумерация начинается с нуля |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Цикл Для каждого (Список)

Перебирает все элементы в заданном Списке, в каждой итерации цикла присваивает текущий элемент переменной, которая указана в свойстве Элемент, а затем выполнение сценария продолжается в сторону выхода Элемент данного блока (в так называемое "тело цикла"). После перебора всех элементов списка выполнение сценария продолжается в сторону нижнего выхода блока. Не забудьте соединить выход последнего блока в теле цикла со входом данного блока

|  |  |
| --- | --- |
| Список | [Список] Список, элементы которого необходимо перебрать в цикле |
| Элемент | [Объект] Переменная, которой будет присвоен текущий элемент списка при каждом проходе цикла |
| Индекс элемента | [Число] Порядковый номер элемента. Нумерация начинается с нуля |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Цикл Для каждого (Словарь)

Перебирает все элементы (пары Ключ-Значение) в заданном Словаре, в каждой итерации цикла присваивает текущий элемент словаря переменным, которые указаны в свойствах Ключ и Значение, а затем выполнение сценария продолжается в сторону выхода Элемент данного блока (в так называемое "тело цикла"). После перебора всех элементов Словаря выполнение сценария продолжается в сторону нижнего выхода блока. Не забудьте соединить выход последнего блока в теле цикла со входом данного блока

|  |  |
| --- | --- |
| Словарь | [Словарь] Словарь, элементы которого необходимо перебрать |
| Ключ | [Текст] Переменная, в которую будет записан Ключ текущего элемента словаря перед каждой итерацией |
| Значение | [Объект] Переменная, в которую будет записано Значение текущего элемента словаря перед каждой итерацией |
| Индекс элемента | [Число] Порядковый номер элемента. Нумерация начинается с нуля |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Цикл

Цикл позволяет повторить цепочку действий (блоков) несколько раз и при этом посчитать количество итераций цикла, записывая текущее количество в специальную переменную (указанную в свойстве Цикл), которую можно затем использовать в других блоках. При первом входе в блок значение переменной из свойства Цикл приравнивается к Начальному значению, а затем выполнение сценария продолжается в сторону выхода Цикл данного блока (в так называемое "тело цикла"). Как только выполнение сценария вновь дойдет до данного блока, значение переменной Цикл увеличится на значение Шаг, и вновь произойдёт переход в сторону выхода Цикл. Как только значение переменной Цикл станет больше или равно Конечному значению, выполнение сценария продолжится в сторону нижнего выхода блока.
Не забудьте соединить выход последнего блока в теле цикла со входом данного блока или какого-либо из предшествующих блоков.

|  |  |
| --- | --- |
| Начальное значение | [Число] Начальное значение переменной из свойства Цикл, будет присвоено этой переменной в начале цикла |
| Конечное значение | [Число] Конечное значение переменной из свойства Цикл. Итерации цикла будут выполняться пока значение переменной Цикл меньше заданного в этом свойстве |
| Шаг | [Число] Шаг приращения значения переменной из свойства Цикл в каждой итерации цикла. Шаг может иметь отрицательное значение, это полезно когда Начальное значение больше Конечного значения, и требуется считать в обратную сторону. |
| Цикл | [Число] Переменная, в которую будет записано текущее значение цикла. Например: если Начальное значение = 0, Конечное значение = 7, а Шаг = 2, то значениями цикла будут последовательно числа 0,2,4,6. |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Цикл Пока

Цикл позволяет повторить цепочку действий (блоков) несколько раз, пока выполняется условие, заданное в виде выражения. При каждом входе в блок проверяется выражение, указанное в свойстве Условие, и если оно истинно, то выполнение сценария продолжится в сторону выхода Условие данного блока (в так называемое "тело цикла"). Как только при очередном проходе через данный блок Условие перестало выполняться, сценарий продолжится в сторону нижнего выхода блока. Не забудьте соединить выход последнего блока в теле цикла со входом данного блока или какого-либо из предшествующих блоков.

|  |  |
| --- | --- |
| Условие | [Любой тип] Пока указанное выражение имеет истинное значение, цикл продолжит выполняться. Значение выражения, эквивалентное 0 (нулю) или $False считается ложным, а любые другие значения - истинными. Например, если в переменную $a было ранее записано значение 5, то условие $a>2 является истинным, а условие $a == 6 является ложным. Условие может содержать математические и логические операции, константы, функции, другие переменные и т.д. Примеры выражений, которые можно использовать в условиях, доступны по нажатию на кнопку ... рядом со свойством Условие. |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Пауза

Приостанавливает выполнение работы сценария на заданное время

|  |  |
| --- | --- |
| Единица измерения | Длительность паузы может задаваться в секундах или миллисекундах |
| Длительность | [Число/TimeSpan] Значение промежутка времени в секундах или миллисекундах, на который будет приостановлена работа сценария. При указании объекта [TimeSpan] свойство "Единица измерения" игнорируется |
| Писать в лог | При включении перед началом паузы в лог будет отправляться сообщение |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Выполнить выражение PowerShell

Выполнение одного или нескольких выражений на языке, совместимом с PowerShell. Выражения записываются без кавычек

|  |  |
| --- | --- |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 6 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 7 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 8 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 9 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 10 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Выполнить код C#

Выполнение кода на языке C# с передачей параметров из сценария и обратно

|  |  |
| --- | --- |
| Конфигурация | [Текст] Строка конфигурации. Для редактирования значения необходимо использовать кнопку на блоке "Редактировать" |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Выполнить код Python

Выполнение кода на языке Python с передачей параметров из сценария и обратно

http://sherparpa.ru/videos/ru/invokepythoncode.mp4

|  |  |
| --- | --- |
| Конфигурация | [Текст] Строка конфигурации. Для редактирования значения необходимо использовать кнопку на блоке "Редактировать" |
| Стандартный вывод | [Текст] После выполнения скрипта данная переменная будет содержать значение стандартного потока вывода (stdout). Например: при использовании в скрипте оператора print, его значение будет содержаться в этой переменной |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Выполнить код VB

Выполнение кода на языке Visual Basic с передачей параметров из сценария и обратно

|  |  |
| --- | --- |
| Конфигурация | [Текст] Строка конфигурации. Для редактирования значения необходимо использовать кнопку на блоке "Редактировать" |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Try/Catch

Блок реализует конструкцию Try/Catch. Для правильной работы блоки этого контейнера должны заканчиваться блоком "Конец". При возникновении ошибки в блоках будет реализован выход по коннектору "Если ошибка". В противном случае по коннектору "Выход". Для имитации конструкции Finally требуется ручное соединение блоков после обработки ошибки с блоками коннектора "Выход". Если к коннектору "Если ошибка" не подключен блок, то ошибка будет игнорироваться, и выход будет произведен по коннектору "Выход". Если коннектор "Выход" не подключен, то при отсутсвии ошибки, будет произведен возврат из текущей диаграммы

|  |  |
| --- | --- |
| Количество попыток | [Текст] Блок не выйдет по коннектору "Если ошибка" пока не закончится указанное число попыток |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Контейнер

Блок позволяет сворачивать содержимое. Под содержимым понимаются блоки, исходящие от текущего через нижний коннектор. Для правильной работы блоки этого контейнера должны заканчиваться блоком "Конец". По окончании работы внутренних блоков будет произведен переход по коннектору "Выход"

|  |  |
| --- | --- |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |

## Прервать

Блок прерывает работу сценария или генерирует ошибку. Если выбран тип ошибки "System", то так же будет задано значение переменной $SystemException. Если выбран тип ошибки "Business", то так же будет задано значение переменной $BusinessException. Переменные SystemException и BusinessException являются глобальными

|  |  |
| --- | --- |
| Генерировать ошибку | [Текст] Заданное значение будет сгенерировано как ошибка. Если значение не задано, то сценарий будет прерван |
| Тип ошибки | Выбор типа генерируемой ошибки. Возможные значения: Default - обычное исключение; System - системное исключение (устанавливается значение переменной $SystemException); Business - бизнес исключение (устанавливается значение переменной $BusinessException) |
| Записать в лог | [Любой тип данных] Числовая, текстовая константа (в кавычках), имя переменной (начинающееся с $) или любое выражение, значение которого будет выведено в лог в процессе работы сценария робота. Значение может в том числе иметь сложный тип данных, например Список, Таблица или Словарь. |
| Логировать в оркестратор | Если выбрано, то текст будет также логироваться в оркестратор, если подключение к нему активно |
| Скриншот в оркестратор | Если выбрано, то будет создан скриншот экрана и отправлен в оркестратор, если подключение к нему активно |
| Установить статус Failed | При включении статус работы будет принудительно установлен как Failed. Свойство работает только если значение свойства "Генерировать ошибку" не задано |

## Прервать контейнер

Блок позволяет выйти в любом месте из контейнера (Try/Catch, Контейнер). По умолчанию для выхода из контейнера необходимо устанавливать блок "Конец". После этого выход из контейнера будет произведен по коннектору "Выход". Если возникает необходимость выйти из контейнера в любом месте, то необходимо установить данный блок и уже после него продолжить сценарий. Если далее в сценарии встретится блок "Конец", то он завершит или предыдущий контейнер или всю диаграмму. Если блок "Прервать контейнер" стоит вне работы контейнера, то он не выполнит возврат из диаграммы

|  |  |
| --- | --- |
| Записать в лог | [Любой тип данных] Числовая, текстовая константа (в кавычках), имя переменной (начинающееся с $) или любое выражение, значение которого будет выведено в лог в процессе работы сценария робота. Значение может в том числе иметь сложный тип данных, например Список, Таблица или Словарь. |
| Переменная 1 | [Переменная] Имя первой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 1 | [Любой тип данных] Новое значение первой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 2 | [Переменная] Имя второй переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 2 | [Любой тип данных] Новое значение второй переменной. Свойство игнорируется, если не задано имя второй переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 3 | [Переменная] Имя третьей переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 3 | [Любой тип данных] Новое значение третьей переменной. Свойство игнорируется, если не задано имя третьей переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 4 | [Переменная] Имя четвертой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 4 | [Любой тип данных] Новое значение четвертой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Переменная 5 | [Переменная] Имя пятой переменной, которой необходимо присвоить новое значение. Можно выбрать из выпадающего списка переменную, которая была ранее объявлена в панели Переменные, а также ввести имя новой или использовавшейся ранее в другом блоке переменной, даже если она не была объявлена в панели Переменные. |
| Значение 5 | [Любой тип данных] Новое значение пятой переменной. Свойство игнорируется, если не задано имя первой переменной. Новое значение переменной может быть как константой (числом, текстовой строкой), так и выражением с использованием других переменных и/или функций. В новом значении переменной разрешено использовать выражение с участием этой же переменной. Не забудьте, что здесь и в других блоках используемые в выражениях названия переменных должны начинаться с символа $, а строковые константы должны заключаться в кавычки. |
| Выражение 1 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 2 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 3 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 4 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Выражение 5 | [Выражение] Выражение, которое необходимо выполнить. Должно быть записано в формате языка PowerShell |
| Логировать в оркестратор | Если выбрано, то текст будет также логироваться в оркестратор, если подключение к нему активно |
| Скриншот в оркестратор | Если выбрано, то будет создан скриншот экрана и отправлен в оркестратор, если подключение к нему активно |

## Параллельный выбор

Блок позволяет псевдопараллельно выполнить один из непосредственно подключенных к своим выходам блоков, выбрав тот из блоков, в котором не происходит ошибка, и после успешного выполнении такого блока продолжить работу сценария. Подключенный блок считается успешно выполненным, если он не вызвал ошибку (исключение). Коннектор "Если ошибка" у всех непосредственно подключенных блоков должен оставаться пустым. Если ни один из непосредственно подключенных к выходам блоков не выполнился успешно (все блоки вызвали исключение), попытки выполнить эти блоки повторяются пока один из блоков не выполнится успешно, либо пока не наступит таймаут. Популярные сценарии использования: ожидание появления одного из нескольких селекторов на экране (необходимо подключить к выходам блоки Ожидать появление элемента), ожидать появление файлов в папках и т.д. Если подключаемый блок имеет собственный настраиваемый таймаут, рекомендуется установить его в минимально возможное значение

http://sherparpa.ru/videos/ru/parallelselect.mp4

|  |  |
| --- | --- |
| Время ожидания | [Число] Предельное время ожидания выполнения блоков в секундах |
| Выбранный блок | [Число] Возвращает номер выполненного блока. Номер начинается с 1 и соответствует номеру коннектора. Если статус равен нулю, то время ожидания блока закончилось и ни один из подключенных блоков не был успешно выполнен |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |

## Запустить робота

Блок запускает ещё один экземпляр робота в качестве основного или фонового процесса, в том числе с возможностью запуска нескольких фоновых процессов параллельно. При запуске указывается его уникальное имя (чтобы затем отличить один запущенный экземпляр робота от другого) и файл .robot со сценарием, который необходимо выполнить. Блок может как ожидать завершения запущенного в новом экземпляре робота сценария (синхронный запуск), так и сразу продолжать выполнение текущего сценария (асинхронный, фоновый запуск). Текущий сценарий может содержать несколько блоков для асинхронного запуска роботов, что позволяет запустить несколько параллельно выполняющихся экземпляров роботов и сценариев. В случае асинхронного запуска проверить статус запущенного фонового сценария робота можно с помощью блока "Статус робота".

|  |  |
| --- | --- |
| Файл robot | [Текст] Файл робота с расширением .robot. Указывается или относительный путь или полный путь к файлу |
| Режим запуска | Режим запуска робота. Возможные варианты: Background - робот запускается в фоновом режиме, Foreground (PiP) - робот будет запущен в режиме PiP (Картинка в Картинке) |
| Уникальное имя | [Текст] Уникальное имя робота при запуске. Это имя будет учитываться при записи логов робота и при отправке сообщений в оркестратор. Все сообщения в оркестраторе будут начинаться с указанного в этом свойстве значения |
| Ожидать завершения | При включении блок будет ожидать завершения запущенного робота |
| Время ожидания | [Число] Время ожидания завершения запущенного робота. Задается в секундах. Свойство работает если включено свойство "Ожидать завершения". Значение -1 указывает неопределенно долгое ожидание |
| Аргументы | [Текст] Аргументы для запуска робота. Для редактирования значения необходимо использовать кнопку на блоке с подсказкой "Аргументы". Эту функцию можно использовать только если включено свойство "Ожидать завершения". Переменные, используемые в аргументах, в вызываемом роботе должны быть заданы как глобальные |
| ID робота | [Ссылка] Переменная, в которой хранится идентификатор робота |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Проверить завершение робота

Блок позволяет проверить статус завершения выполнения сценария, запущенного с помощью блока "Запустить робота", в том числе позволяет проверить статус сразу нескольких запущенных параллельно роботов

|  |  |
| --- | --- |
| ID робота | [Ссылка/Список/Массив] Переменная, в которой хранится идентификатор робота. При необходимости можно задать сразу несколько идентификаторов в массиве или списке. Например: @($RobotID1, $RobotID2, $RobotID3) |
| Ожидать завершение | Блок будет ожидать завершения выполнения робота или списка роботов и только после этого выйдет по коннектору "Да" |
| Статус работы | [Логический] Если возвращается $true, значит робот работает |
| Статус ошибки | [Логический] Если возвращается $true, значит робот завершился с ошибкой |
| Уровень обработки | Выбор уровня обработки ошибок. Возможные значения: "Default" - по умолчанию; "Ignore" - ошибки игнорируются; "Handle" - ошибки обрабатываются. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Направление выхода | Выбор направления выхода из блока при возникновении ошибки. Возможные значения: "Default" - по умолчанию; "Right" - направо; "Bottom" - вниз. Данное свойство позволяет определить направление выхода при возникновении ошибок для блоков не содержащих коннектора "Если ошибка" (например блок "Условие"). Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Уровень сообщений | Выбор уровня сообщений, которые будут выводить блоки при работе. Возможные значения: "Default" - по умолчанию; "Release" - вывод отключен; "Debug" - вывод основной информации; "Detailed" - вывод подробной информации. Если выбрано значение "Default", то будет использоваться значение блока "Старт" данной диаграммы |
| Текст ошибки | [Текст] Возвращает подробную информацию об ошибке в случае некорректного выполнения работы блока |

## Коннектор

Коннектор