

Описание

Узел создает новый набор данных с заданным составом и параметрами полей и заполняет его данными в ходе выполнения кода Python. Для заполнения выходного набора в коде возможно использование данных входных портов.

Состав и параметры полей выходного набора могут задаваться как в мастере настройки узла, так и из кода Python.

Примечание: Для работы узлов *Python* может потребоваться предварительная настройка Loginom и установка Python. Подробности в разделе [Параметры компонента: Python](#). Использование Python в Loginom имеет [особенности и ограничения](#).

Вход

-  **Входной источник данных** — таблица данных, необязательный;
-  Добавить еще один порт – создает новый порт Входной источник данных[N], где N – порядковый номер порта;
-  **Входные переменные** — переменные, необязательный.

Выход

-  **Выходной набор данных** — таблица данных.

Мастер настройки

Последовательно выполняются следующие этапы настройки:

- Настройка столбцов выходной таблицы Python.
- Python – ввод кода и предварительный просмотр результатов.

Настройка столбцов выходной таблицы

Столбцы выходного набора можно задать вручную на странице мастера настройки *Настройка столбцов выходной таблицы Python*, либо динамически в ходе исполнения кода Python.

На странице доступны следующие команды:

- **Разрешить формировать выходные столбцы из кода** — при установке флага доступно динамическое создание, изменение и удаление выходных столбцов.
-  **Добавить столбец** — добавляет новый столбец с параметрами по умолчанию.
-  **Клонировать столбец** — добавляет новый столбец с параметрами, как у выбранного столбца.
-  **Редактировать столбец (F2)** — вызывает модальное окно редактирования параметров выбранного столбца.
-  **Переместить вверх (Ctrl+Up)** — поднять выбранный столбец на одну позицию вверх по списку.
-  **Переместить вниз (Ctrl+Down)** — опустить выбранный столбец на одну позицию вниз по списку.
-  **Удалить столбец (Delete)** — удаляет выбранный столбец.
-  **Удалить все столбцы (Shift+Delete)** — удаляет все имеющиеся столбцы.

Python

Страница *Python* содержит редактор исполняемого узлом кода. Запускать написанный скрипт возможно внутри процесса LogiNot или в отдельном процессе (в *Процессе интерпретатора*) при установке флага *Запускать в отдельном процессе*.

По кнопке Предпросмотр... в отдельном окне выводится до 100 первых строк результирующего набора данных и панель вывода сообщений.

Примечание: При нажатии на кнопку *Предпросмотра* активируются все входные порты. При успешном завершении активации открывается окно *Предпросмотра*.

При запуске скрипта на Python внутри процесса LogiNot одновременно может выполняться только один узел *Python*, соответственно последующий в очереди запуска узел *Python* ожидает выполнения предыдущего. Максимальное время ожидания задается параметром *Тайм-аут ожидания запуска (мс)*. По умолчанию тайм-аут неограничен. Если время тайм-аута превышено, выполнение узла завершается с соответствующей ошибкой.

Поддерживается импорт модулей Python (см. Поиск модулей Python внутри процесса LogiNot).

Доступ из кода Python к данным портов и другим встроенным объектам

Для доступа к данным портов и другим встроенным объектам в контексте выполнения кода предусмотрены следующие объекты:

- Входные наборы данных (`InputTables` , `InputTable`);
- Входные переменные (`InputVariables`);
- Выходной набор данных (`OutputTable`);
- Необходимые перечисления (`DataType` , `DataKind` , `UsageType`).

Вышеуказанные объекты импортируются из встроенного модуля `builtin_data` . По умолчанию в текст исполняемого узлом кода добавляется строка импорта этих объектов.

Обработка ошибок

При вызове окна *Предпросмотра* или выполнении узла выводится сообщение об обнаруженных синтаксических ошибках и ошибках хода выполнения с указанием позиции кода, содержащего ошибку.

Глобальные функции

Доступны следующие глобальные функции:

- `filestoragepath` — принимает путь в файловом хранилище и преобразует его в абсолютный путь в файловой системе.

▼ Пример:

```
from builtin_data import OutputTable
import json

with open(filestoragepath("data_file.json")) as f:
    data = json.load(f)
    OutputTable.Append()
    OutputTable.Set(0, data)
```

- `getLocale` — возвращает имя локали текущего узла в формате ICU. Функция не имеет аргументов.

Дополнительных импортов для применения данных функций выполнять не нужно.

Статьи в разделе:

- [Входные наборы](#)
- [Входные переменные](#)
- [Выходной набор](#)
- [Перечисления](#)
- [Панель вывода](#)
- [Описание API](#)
- [Горячие клавиши](#)
- [Ограничения](#)