

Инструкция
пользователя
Lexema-RPA Studio



Инструкция пользователя Lexema-RPA Studio

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	7
Назначение программы	7
Что такое RPA?	7
Что могут роботы?	7
Сколько нужно времени для создания роботов?	8
Области использования роботов	8
С чего начать	9
ЧАСТЬ I. ЗНАКОМСТВО СО СТУДИЕЙ	12
Глава 1. Интерфейс главного окна	12
Главное окно	12
Кнопки управления	13
Меню	16
Обозреватель решений	19
Отладочный вывод	21
Глава 2. Основы работы	23
Переменные	23
Функции	26
Рабочая область	26
Горячие клавиши	29
ЧАСТЬ II. МОДУЛЬ «EXCEL»	30
Глава 3. Основные моменты модуля	30
Интерфейс модуля	30
Раздел «Общие настройки»	31
Разделы «Добавить команду» и «Список команд»	32
Глава 4. Набор полей в составе модуля	36
Перечень возможных полей с правилами их заполнения	36
Глава 5. Описание действий, доступных в модуле	40
Группа «Работа с данными»	40
Группа «Работа со строками»	42
Группа «Работа с колонками»	43
Группа «Работа с листами и файлами»	44
Группа «Поиск»	46

Группа «Другие функции».....	48
Группа «Форматирование»	58
Глава 6. Модуль «Закрывать Excel-файл»	72
ЧАСТЬ III. МОДУЛЬ «КЛИКЕР» И МОДУЛИ ПОИСКА ИЗОБРАЖЕНИЙ	73
Глава 7. Модуль «Кликер».....	73
Создание макроса.....	73
Создание команды	75
Статические поля.....	75
Поля, появляющиеся только при определенных действиях.....	75
Описание действий, доступных в модуле	77
Глава 8. Модуль «Поиск изображения»	82
Интерфейс модуля	82
Глава 9. Модуль «Поиск изображений»	83
Интерфейс модуля	83
Выходные данные	84
ЧАСТЬ IV. МОДУЛЬ «РАБОТА С ФАЙЛАМИ».....	86
Глава 10. Основные моменты модуля.....	86
Интерфейс модуля	86
Функциональность модуля	86
Глава 11. Описание действий, доступных в модуле.....	88
Перечень и описание действий.....	88
ЧАСТЬ V. МОДУЛИ «ЦИКЛ», «УСЛОВИЕ», «WHILE», «BREAK» И «RETURN».....	93
Глава 12. Модуль «Условие»	93
Интерфейс модуля	93
Конструктор условий	93
Глава 13. Модуль «Цикл»	98
Интерфейс модуля	98
Глава 14. Модуль «While».....	100
Интерфейс модуля	100
Глава 15. Модуль «Break».....	102
Глава 16. Модуль «Return»	103
ЧАСТЬ VI. МОДУЛИ «ПОЧТОВЫЕ АГЕНТЫ».....	104
Глава 17. Модуль «Отправить письмо».....	104

Интерфейс модуля	104
Глава 18. Модуль «Считать письма»	109
Интерфейс модуля	109
ЧАСТЬ VII. МОДУЛЬ «РАБОТА С БРАУЗЕРОМ»	113
Глава 19. Основные моменты модуля.....	113
Интерфейс модуля	113
Глава 20. Описание действий	115
Перечень и описание действий.....	115
ЧАСТЬ VIII. МОДУЛЬ «ИНТЕРФЕЙС»	122
Глава 21. Основные моменты	122
Основное окно.....	122
Глава 22. Общие настройки	123
Глава 23. Добавление элемента интерфейса	127
Поле «Тип»	127
Поле «Источник данных»	129
Поля «Колонка» и «Строка»	131
Поля «Объединить колонки» и «Объединить строки»	131
Поле «Переменная»	133
ЧАСТЬ IX. МОДУЛИ ДЛЯ РАБОТЫ С ТЕКСТОМ.....	134
Глава 24. Модуль «Обработка текста»	134
Интерфейс модуля	134
Методы анализа текста.....	135
Глава 25. Модуль «Распознавание текста»	136
Интерфейс модуля	136
Глава 26. Модуль «Распознавание текста АBBYU»	138
Интерфейс модуля	138
Основной текст	138
Табличная часть	139
ЧАСТЬ X. ДРУГИЕ МОДУЛИ	140
Глава 27. Модуль «Вычисления»	140
Интерфейс модуля	140
Поддерживаемые операции	140
Пример работы с модулем	141

Глава 28. Модуль «Запустить процесс».....	143
Интерфейс модуля	143
Глава 29. Модуль «Завершить процесс».....	145
Интерфейс модуля	145
Глава 30. Модуль «Код C#»	146
Интерфейс модуля	146
Глава 31. Модуль «SQL».....	147
Интерфейс модуля	147
Глава 32. Модуль «HTTP_GET».....	148
Интерфейс модуля	148
Глава 33. Модуль «HTTP_POST».....	149
Интерфейс модуля	149
Глава 34. Модуль «Парсинг JSON»	150
Интерфейс модуля	150
Глава 35. Модуль «Пауза»	152
Глава 36. Модуль «Прокси».....	152

ВВЕДЕНИЕ

Руководство предназначено для людей, которые хотят научиться писать роботов в программе Lexema-RPA Studio. Вы познакомитесь со всеми инструментами студии и основными модулями, применяемыми для создания роботов.

Lexema-RPA Studio, далее именуемая просто студия, предоставляет пользователю возможность настраивать последовательность взаимодействий с разнообразными программами. Взаимодействие настраивается с помощью программных модулей, которые поставляются в составе студии. Каждый модуль представляет собой отдельный функционал, необходимый для выполнения конкретного действия, и механизмы для его настройки.

Назначение программы

Программа Lexema-RPA Studio предназначена для разработки программных роботов.

Что такое RPA?

Robotic process automation (или RPA) — это форма технологии автоматизации бизнес-процессов, основанная на метафорическом программном обеспечении роботов (ботов) или работников искусственного интеллекта. Роботы RPA используют пользовательский интерфейс для сбора данных и управления приложениями.

Более простыми словами, *RPA* — это программные роботы, которые эмулируют¹ действия человека, производимые им на компьютере.

Что могут роботы?

Роботы могут выполнять рутинные операции на компьютере вместо человека:

- вводить данные с клавиатуры, манипулировать мышью, считывать информацию с экрана, получать и обрабатывать почту, взаимодействовать с офисными программами;
- интегрировать данные из одной системы в другую без использования дополнительных средств.

¹ Эмуляция (англ. emulation) в вычислительной технике — комплекс программных, аппаратных средств или их сочетание, предназначенное для копирования (или эмулирования) функций одной вычислительной системы (гостя) на другой, отличной от первой, вычислительной системе (хосте) таким образом, чтобы эмулированное поведение как можно ближе соответствовало поведению оригинальной системы (гостя).

Сколько нужно времени для создания роботов?

Роботы создаются в простом графическом интерфейсе, и время, затраченное на работа, напрямую зависит от его сложности – простых роботов можно написать за пару часов, на более трудоемких может уйти несколько дней.

Области использования роботов

Роботы-дублёры (интеграторы)

- Перенос информации из одной системы в другую;
- Перезаполнение данных.

HR-роботы²

- Поиск вакансий по заданным критериям.

Роботы-операторы

- Регистрация и ввод данных по скан-копиям документов.

Роботы-консолидаторы

- Консолидация данных в отчет;
- Массовая отправка или печать отчетов.

Роботы-закупщики

- Мониторинг торговых площадок;
- Выгрузка информации о тендерах³;
- Отбор поставщиков на основании заданных условий.

Роботы-продавцы

- Проверка благонадежности контрагентов⁴;
- Формирование коммерческих предложений.

² HR (англ.яз., Human resources) — область управления персоналом.

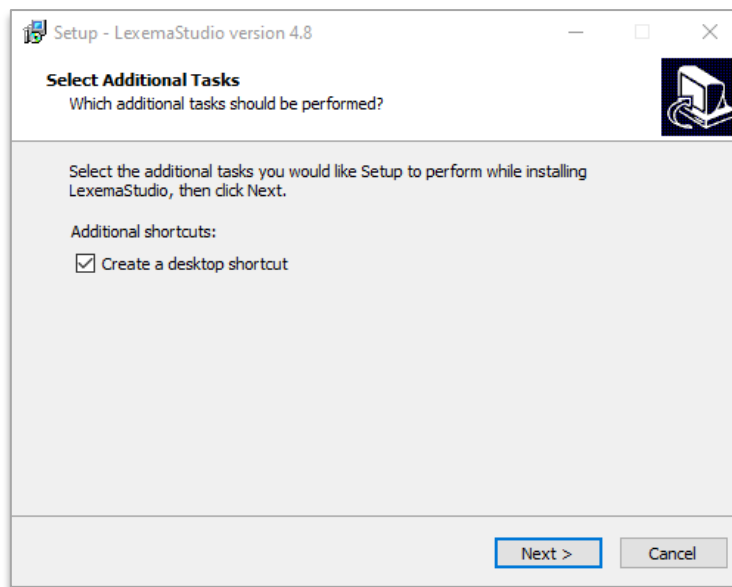
³ Тендер (англ. tender) — заявка на подряд, поданная на конкурсной основе.

⁴ Контрагент – лицо или учреждение, берущее на себя известные обязательства по договору.

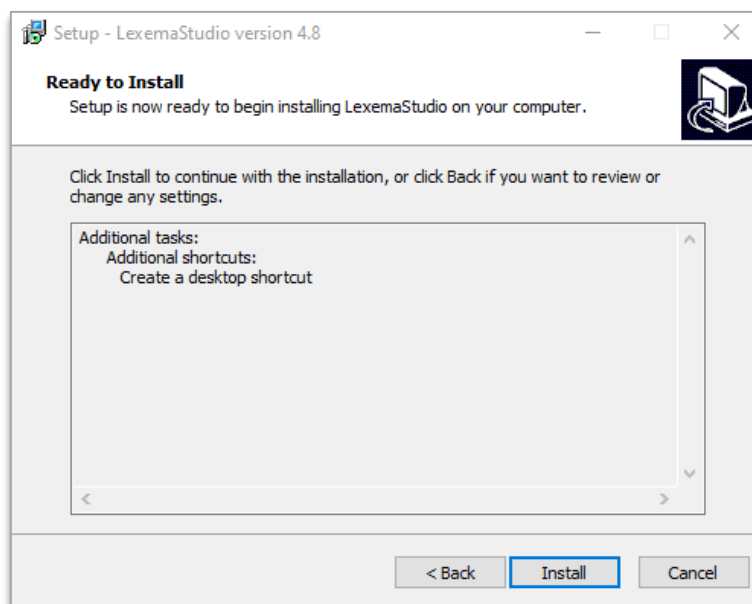
С чего начать

Для начала работы скачайте студию Lexema-RPA Studio *на сайте* <http://lexema.ru/lexemarpastudio/registratsiya/>. Перед началом скачивания вам будет предложена регистрация. Регистрация на сайте обязательна, так как доступ в студию осуществляется после авторизации пользователя. После окончания регистрации начнется скачивание установщика.

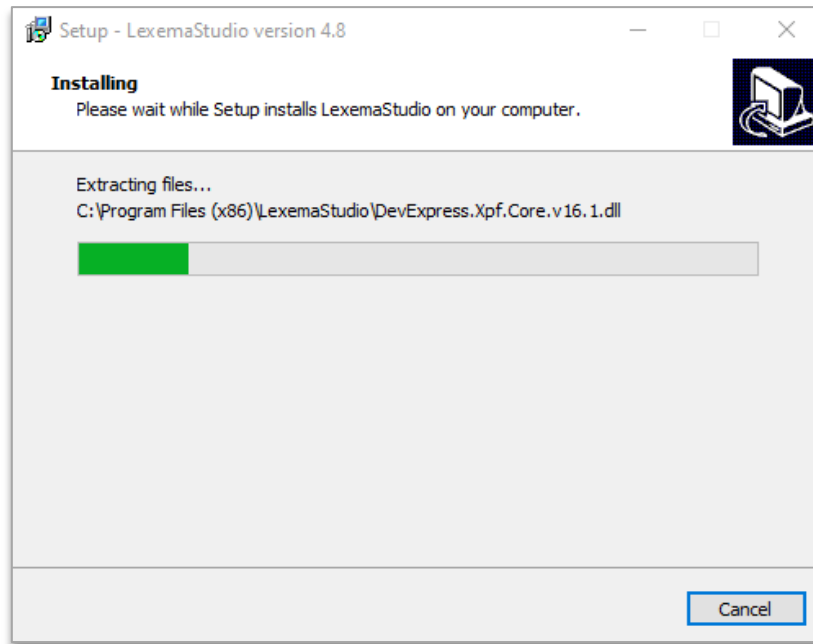
После этого запустите установщик и в открывшемся диалоговом окне нажмите на кнопку «Next».



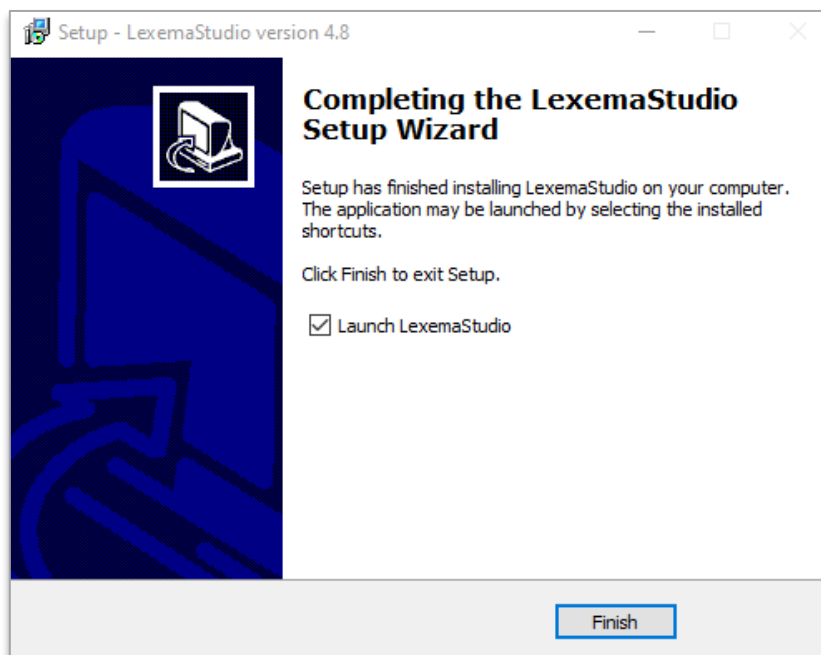
Затем нажмите на кнопку «Install».



После этого начнется установка студии.



Наконец, нажмите на кнопку «Finish».

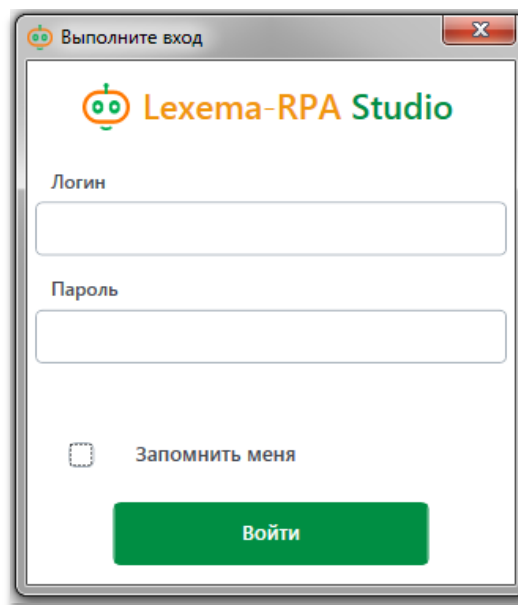


По завершению установки появится ярлык программы на рабочем столе.



СОВЕТ. Если у вас не получилось запустить или установить студию, установите (обновите) *.NET Framework 4.6.1*. Сделать это можно по ссылке <https://www.microsoft.com/ru-ru/download/details.aspx?id=53344> (Центр загрузки Windows).

После запуска программы появится окно авторизации, в которое следует ввести логин и пароль, указанные при регистрации. *Окно авторизации появляется в последних версиях программы.*



Флажок «Запомнить меня» запомнит ваши логин и пароль для последующих входов с текущего устройства.

Кнопка «Войти». По нажатию на кнопку произойдет вход в программу и откроется окно студии.

Если текущая версия программы не последняя, то перед авторизацией вам будет предложено скачать и установить обновление.

ЧАСТЬ I. ЗНАКОМСТВО СО СТУДИЕЙ

Глава 1. Интерфейс главного окна

Пользовательский интерфейс студии Lexema-RPA Studio характеризуется областью для работы с проектами (рабочей областью), расположенной в центре главного окна программы. Прямо над рабочей областью располагается полоса с кнопками управления проектом. Слева от рабочей области находится меню, в котором собраны модули для создания роботов, справа – обзорер решений, прямо под рабочей областью – отладочный вывод.

Главное окно

Главное окно студии имеет несколько разделов и основные кнопки управления проектом (рис. 1.1.1).

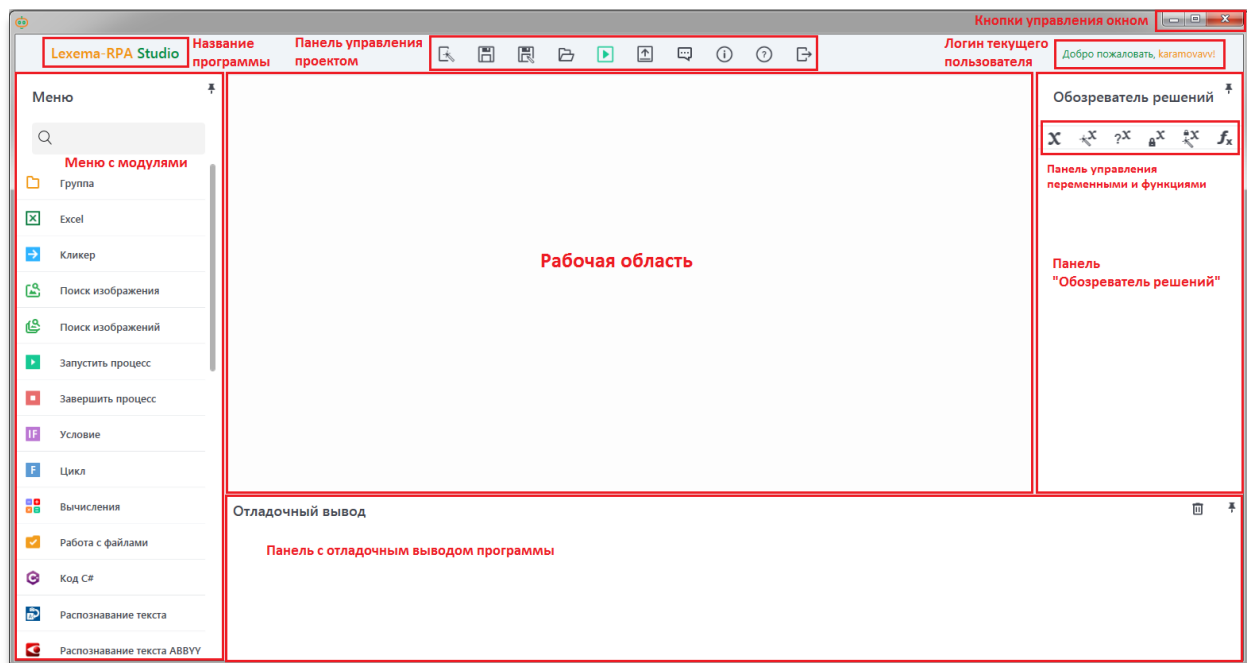



Рис. 1.1.1. Главное окно студии

Все панели (меню, обзорер решений и отладочный вывод) могут быть закреплены или открываться при наведении на них (рис. 1.1.2).

Чтобы открепить или закрепить панель необходимо нажать на иконку с изображением канцелярской кнопки  в правом верхнем углу панели.

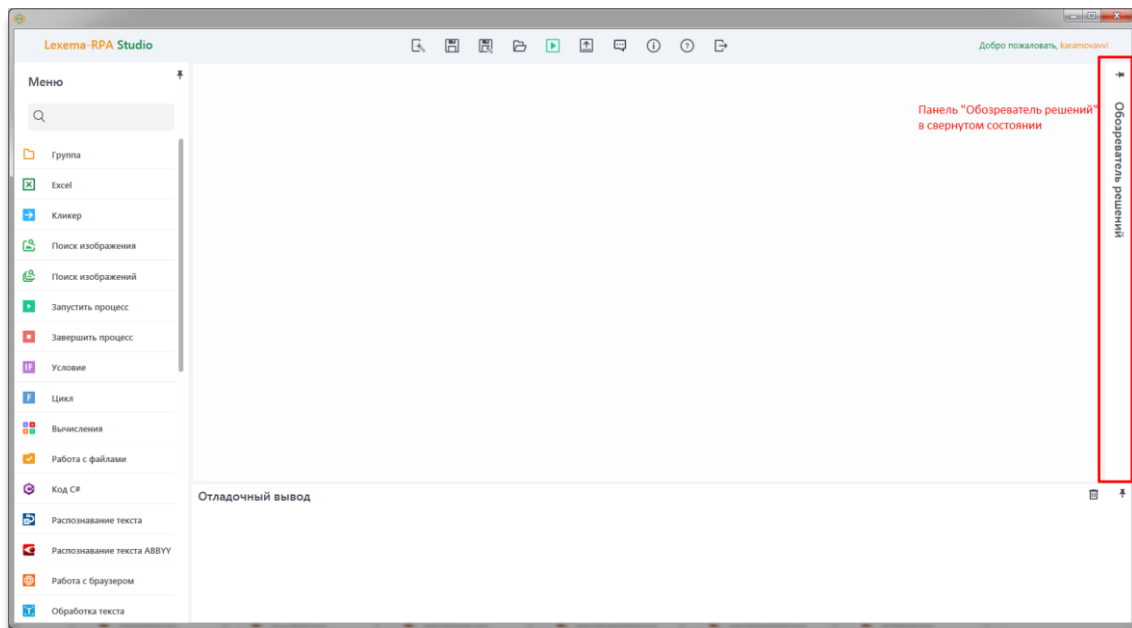







Рисунок 1.1.2. Скрытая панель «Обозреватель решений»

Функционал разделов главного окна студии будет описан ниже.

Кнопки управления

Кнопки управления проектом находятся на верхней панели главного окна студии и предназначены для осуществления основных манипуляций с проектом: создания , сохранения , открытия , выполнения описанных в проекте действий  и выхода из программы .

Также поддерживаются различные комбинации горячих клавиш. Например, для создания проекта можно использовать комбинацию Ctrl+N.

Более подробно поддерживаемые комбинации горячих клавиш описаны в разделе «Горячие клавиши».

Создать

Нажатие на кнопку «Создать» вызывает окно создания нового проекта (рис. 1.1.3).

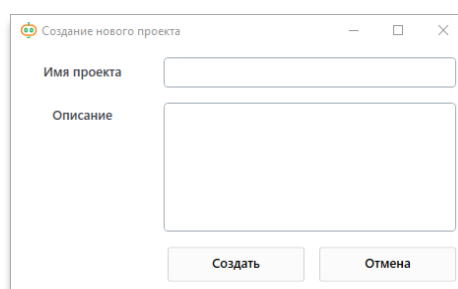


Рис. 1.1.3. Создание нового проекта

В открывшемся окне необходимо ввести имя нового проекта, дополнительно можно указать его описание. Нажатие на кнопку «Создать» в указанном окне завершит процесс создания проекта. Новый проект будет открыт в главном окне студии.

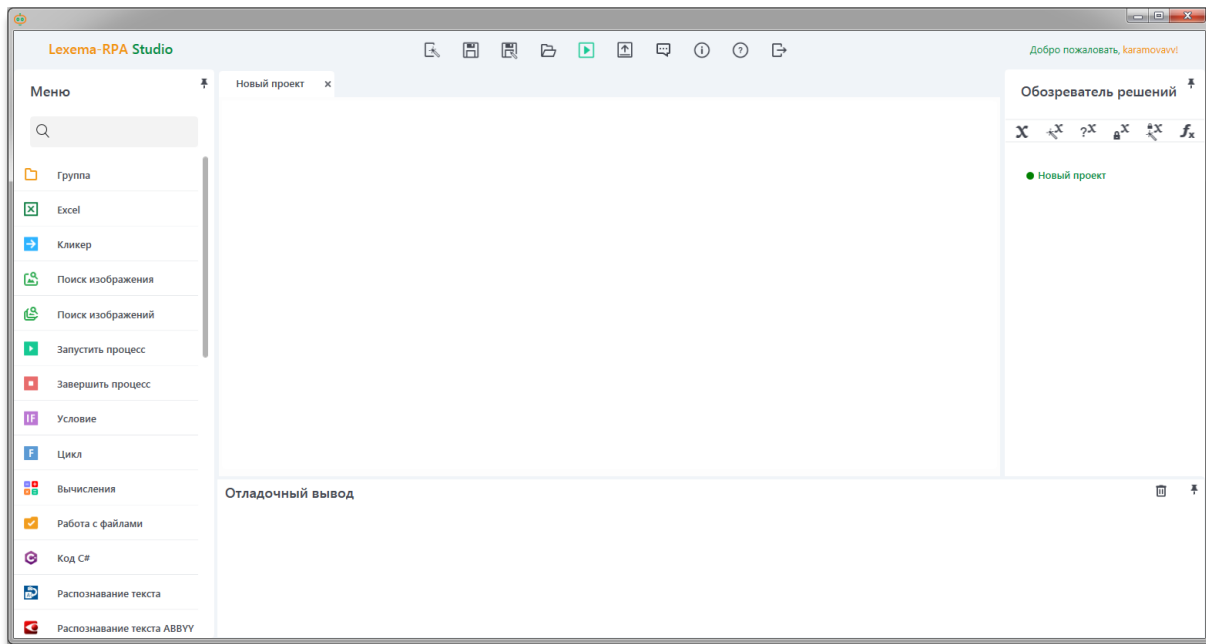


Рис. 1.1.4. Окно нового проекта

Сохранить

Кнопка «Сохранить» вызывает стандартный диалог сохранения файла.

Для нового проекта в диалоговом окне необходимо выбрать папку назначения и название файла, в котором будет сохранен проект (по умолчанию, предлагается название проекта, которое вы задали при его создании). Файлы проектов Lexema-RPA Studio имеют расширение «.гра». После выбора папки для сохранения, необходимо нажать на кнопку «Сохранить» (рис. 1.1.5). По завершении операции, файл проекта будет сохранен по указанному расположению. Существующий проект будет сохранен в его текущем расположении.

Сохранить как

Кнопка «Сохранить как» вызывает стандартный диалог сохранения файла, аналогичный окну сохранения нового проекта.

В диалоговом окне предоставляется возможность для изменения наименования или расположения файла проекта.

Открыть

Кнопка «Открыть» вызывает стандартный диалог выбора файла.

Выбранный файл проекта будет загружен в главное окно студии.

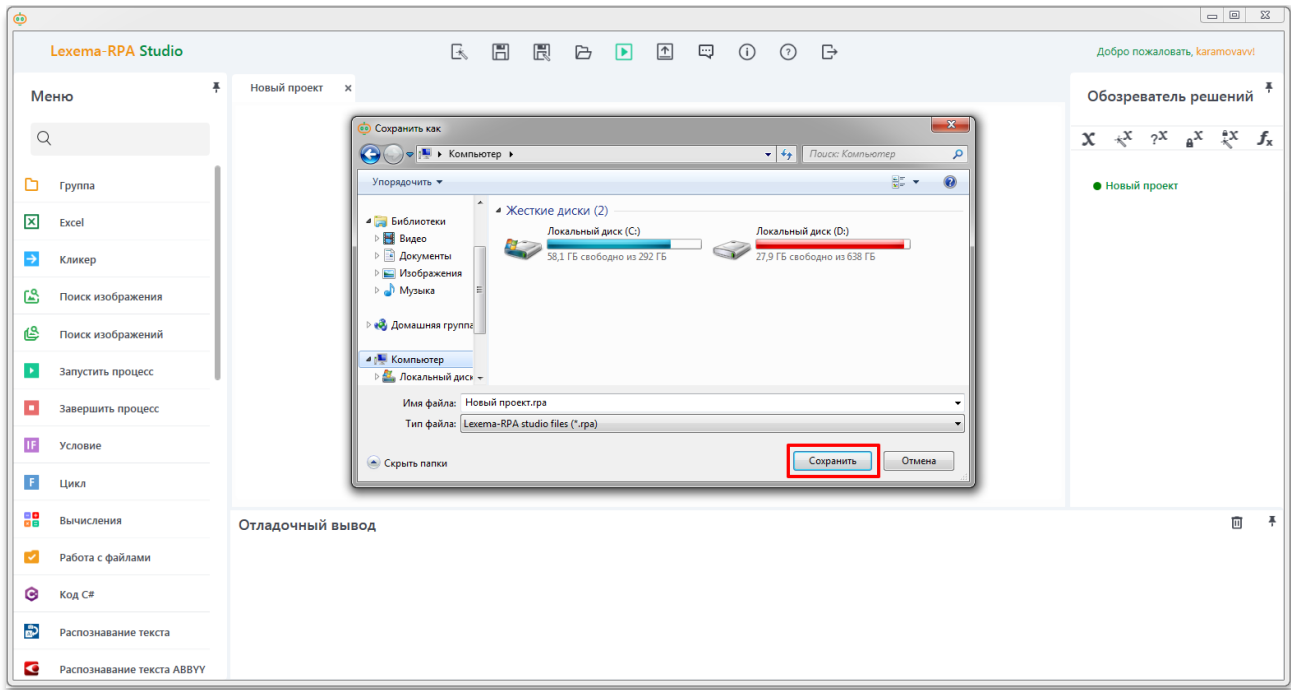


Рис. 1.1.5. Диалоговое окно сохранения проекта


Проиграть

Кнопка «Проиграть» запускает выполнение всех описанных в проекте действий.


Попытка запуска будет осуществляться только в случае, если в проекте имеется хотя бы одно действие. Действия выполняются в порядке, описанном в структуре проекта.

В случае возникновения ошибок или исключительных ситуаций, они будут выведены в область «Отладочный вывод» (далее будет дано подробное описание области отладочного вывода).

Опубликовать


Кнопка «Опубликовать»  вызывает диалог публикации текущего проекта в оркестратор роботов⁵.

Форум

Кнопка «Форум»  позволяет перейти на веб-сайт форума технической поддержки приложения - <http://talk.lexema.ru/>.

⁵ Оркестратор роботов — сервис запуска роботов по составленному расписанию и событиям.

О программе

Кнопка «О программе»  выводит диалоговое окно, содержащее информацию о текущей версии студии.

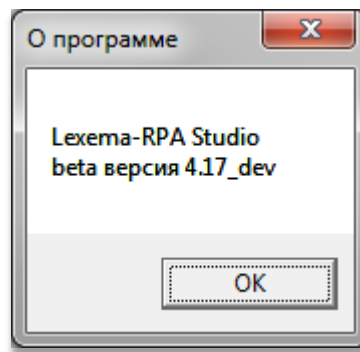


Рис. 1.1.6. Информация о текущей версии студии

Выйти

Кнопка «Выйти»  закрывает студию.

Меню

В разделе «Меню» выводится список доступных для добавления в проект модулей. Каждый модуль отвечает за выполнение конкретного действия или за взаимодействие с определенной программой.

Список доступных модулей

В программе Lexema-RPA Studio имеются следующие модули: Excel, Кликер, Поиск изображения (2 варианта), Запустить/Завершить процесс, Условие, Цикл, Вычисления, Работа с файлами, Код С#, Распознавание текста (обычное и ABBYY), SQL, Отправить/Считать письмо, HTTP GET/POST запросы, Break, JSON, Интерфейс, Закреть Excel-файл, While.

В списке доступных модулей выводятся все имеющиеся модули (рис. 1.1.7).

Поиск модуля

Поиск модуля можно выполнить с помощью текстового поля, расположенного над списком доступных модулей. Список будет фильтроваться по мере ввода поисковой фразы (рис. 1.1.8).

Добавление модуля в проект

Для добавления выбранного модуля в проект необходимо перетащить его в рабочую область с помощью указателя мыши. Новый модуль можно расположить в нужной позиции в общей последовательности действий. Для этого необходимо привести курсор на соседний с нужной позицией модуль. При этом соседний модуль будет подсвечен красным выделением

и положение нового модуля будет отображаться в виде яркого выделения сверху или снизу: сверху – если новый модуль будет добавлен перед соседним, снизу – если после него (рис. 1.1.9 – 1.1.11).

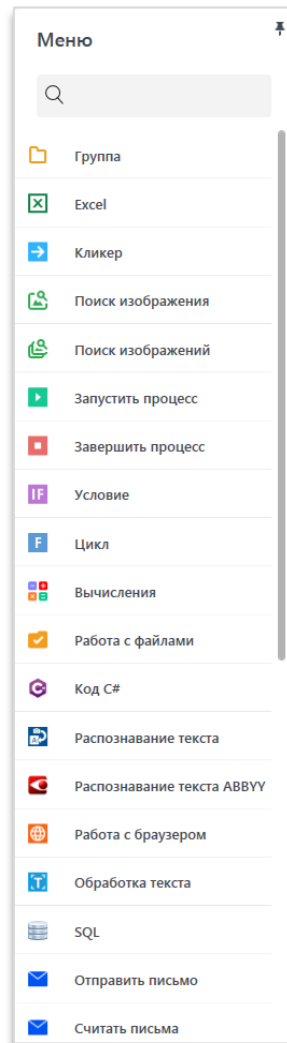


Рис. 1.1.7. Меню доступных модулей

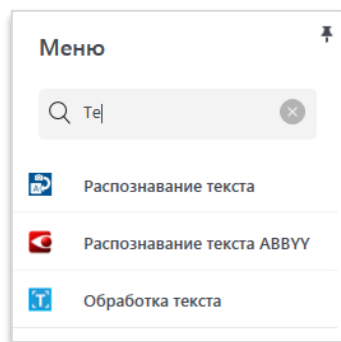


Рис. 1.1.8. Поиск в списке модулей

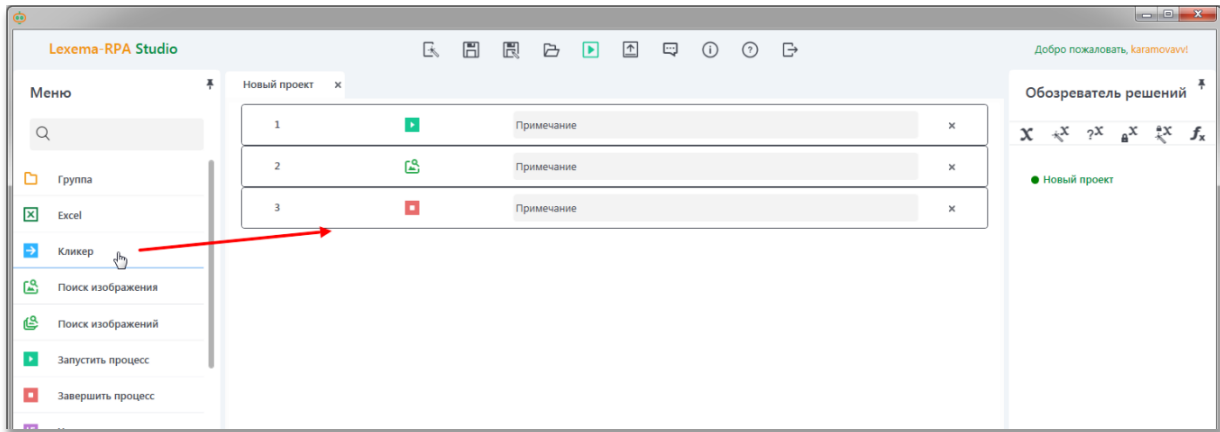


Рис. 1.1.9. Выбор модуля и перетягивание его в нужную область

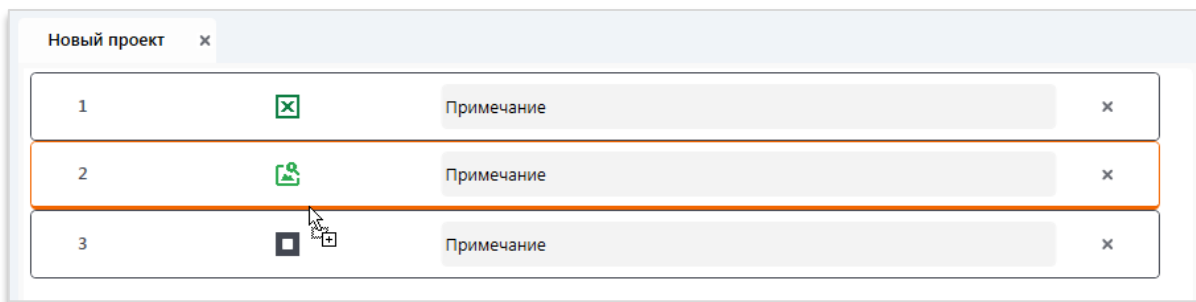


Рис. 1.1.10. Подсветка 2-го модуля при перемещении нового модуля после него

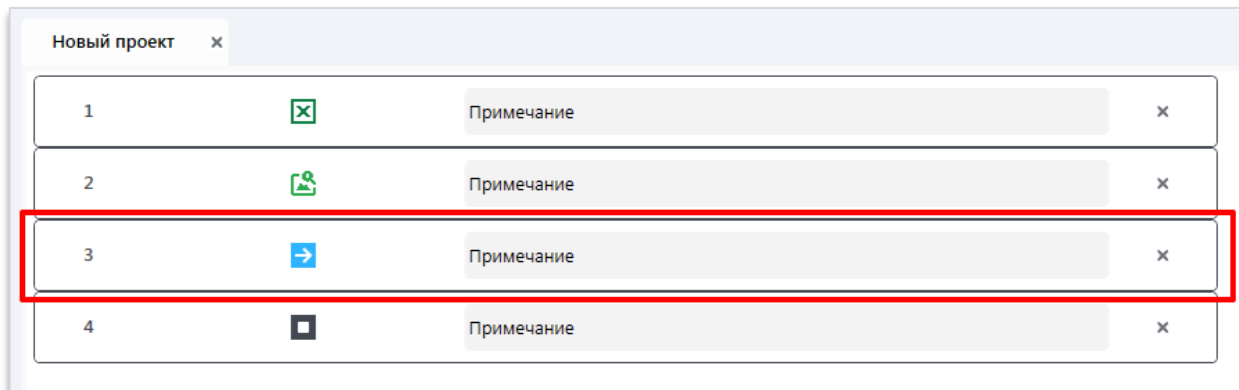


Рис. 1.1.11. Итоговое расположение модуля

Обозреватель решений

Раздел «Обозреватель решений» находится в правой части главного окна. В верхней части раздела расположена панель управления переменными и функциями, прямо под панелью - область текущих проектов. Активный проект (окно которого сейчас открыто) выделен зеленым цветом и помечен маркером зеленая точка.

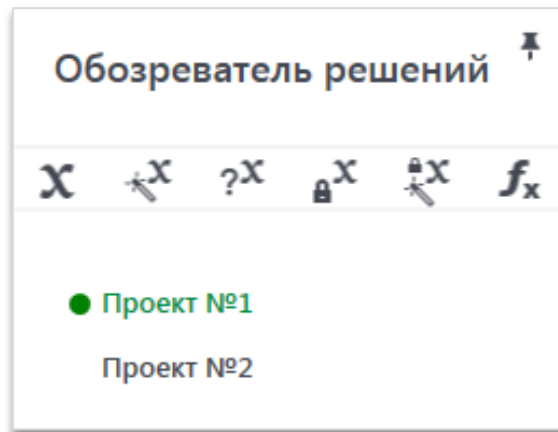


Рис. 1.1.12. Обзор решений

Панель управления переменными и функциями содержит следующие кнопки (перечисление слева направо): «Переменные», «Создать новую переменную», «Узнать значение переменной», «Локальное хранилище», «Создать локальную переменную», «Функции». При нажатии на одну из этих кнопок будут открыты или созданы те переменные или функции, которые относятся к активному проекту. Подробное описание кнопок будет приведено далее.

Обзор переменных

В рабочей области студии можно вывести информацию о доступных в активном проекте переменных.

Окно обзора переменных отображается во вкладке «Переменные» при нажатии на кнопку «Переменные» **X** в обозревателе решений.

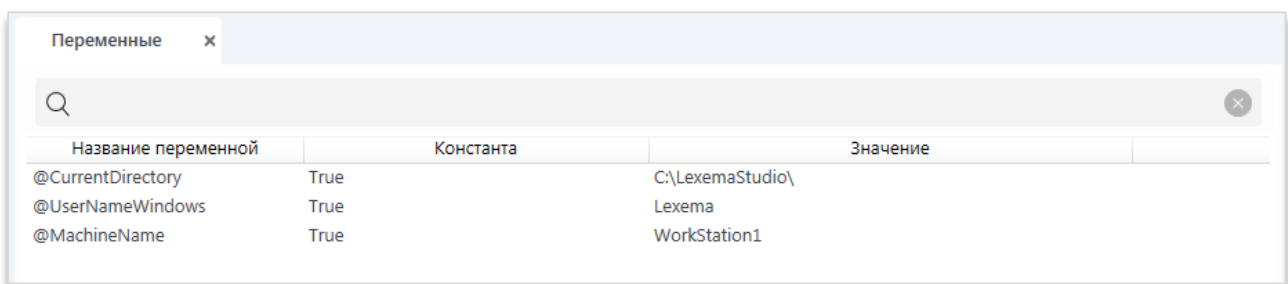


Рис. 1.1.13. Обзор переменных проекта

Окно содержит строку для поиска переменной по её названию и таблицу со списком всех доступных проекту переменных. Таблица имеет три столбца:

- Название переменной
- Константа — флаг, показывающий можно ли изменять значение данной переменной
- Значение переменной

По умолчанию, *переменные автоматически создаются при указании их в том или ином модуле*. При создании проекта автоматически создаются системные переменные:

- @CurrentDirectory – директория, в которой установлена студия;
- @UserNameWindows – название пользователя ОС;
- @MachineName – имя компьютера.

При двойном щелчке по строке с переменной открывается диалоговое окно редактирования свойств переменной. В диалоговом окне предоставляется возможность изменять свойства переменной или удалять переменную из проекта. Подробное описание работы с переменными будет рассмотрено в главе 2.

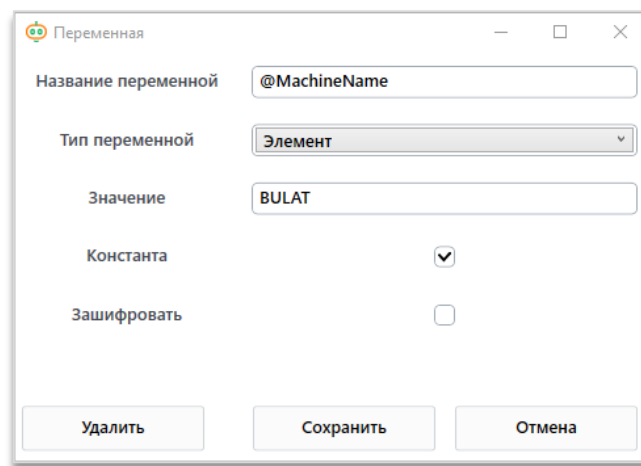


Рис. 1.1.14. Окно настройки переменной

Обзор функций

Доступные в проекте функции выводятся в рабочей области студии во вкладке «Функции» по нажатию на кнопку f_x в обозревателе решений. В окне имеется строка для поиска функции по её названию или описанию.


Список доступных функций представлен в виде таблицы со следующими столбцами:

- Название функции — название функции и пример аргументов, которые она принимает;
- Пример использования — пример результата выполнения функции первого столбца;
- Описание — общее описание функции.

Название функции	Пример использования	Описание
#GetDate()	23.04.2020	Возвращает текущую дату
#GetGuid()	48ff4fcc-1022-4af5-9dd7-f00b3cef632f	Возвращает GUID
#ConvertToSqlString('str')	'str'	Конвертирует переменную в строку для SQL-скриптов
#OperationPlus(10, 20)	30	Операция сложения двух чисел/переменных
#OperationMinus(10, 20)	-10	Операция вычитания двух чисел/переменных
#OperationMultiplication(10, 20)	200	Операция умножения двух чисел/переменных
#OperationDivision(10, 20)	0,5	Операция деления двух чисел/переменных
#GetNumber(A1)	1	Возвращает только цифры
#GetLetter(A1)	A	Возвращает только буквы
#WithoutSpace(string)	string	Убирает пробелы в строке
#Replace("straaiing", "aaa", "")	string	Замена в первом параметре все вхождения второй стрс
#GetColumnName(1)	A	Возвращает название колонки
#DateAdd(day, 1, 12.03.2020)	13.03.2020	Изменяет значение даты в заданном аргументе на указа
#LastDayMonth(3)	31	Возвращает последний день указанного месяца
#GetDay(12.03.2020)	12	Возвращает день указанной даты
#GetMonth(12.03.2020)	3	Возвращает номер месяца указанной даты
#GetYear(12.03.2020)	2020	Возвращает год указанной даты
#Trim(string)	string	Возвращает строку с удаленными в начале и конце прол
#IndexOf(string, str, 0)	0	Возвращает индекс вхождения подстроки в строку
#SubString(string, 2)	ring	Возвращает подстроку из строки, начиная с указанного
#SubStringWithLength(SubString	bSt	Возвращает подстроку из строки, начиная с указанного
#RandomDigit(1, 10)	3	Возвращает случайное целое число в диапазоне указани
#GetTime()	0	Возвращает количество секунд, прошедших с начала ра
#GetTextOnBuffer()		Возвращает текст из буфера обмена
#SetTextOnBuffer()		Записывает текст в буфер обмена
#ClearBuffer()		Очищает буфер обмена
#DateDiff(20.03.2020, 23.03.2020	3	Возвращает количество дней между двумя датами
#ConvertToBitSQL(true)	1	Конвертирует тип bool в bit
#GetValueStringVariable(@test)	123	Получает значение переменной, записанной в другую п
#GetDateSpecPSQL()	23/04/2020	Возвращает текущую дату в формате psql
#GetLenStr(example)	7	Возвращает длину строки
#GetSymbol(example, 2)	a	Возвращает определенный символ строки
#GetLocalValue(@test)		Возвращает значение переменной из локального храни

Рис. 1.1.15. Обзор доступных функций

Обзор локальных переменных

Lexema-RPA Studio позволяет хранить конфиденциальную информацию, например, пароли, логины или почтовые адреса, в переменных, которые будут зашифрованы и сохранены на вашем компьютере. Обзор ваших локальных переменных доступен по кнопке «Локальное хранилище»  в панели управления переменными и функциями. Окно аналогично окну обзора общих переменных студии.

Отладочный вывод

В раздел «Отладочный вывод» главного окна студии выводятся все сообщения, возникающие в ходе выполнения настроенных действий проекта после нажатия на кнопку «Проиграть». Сообщения выводятся для всех действий по порядку их выполнения в проекте.

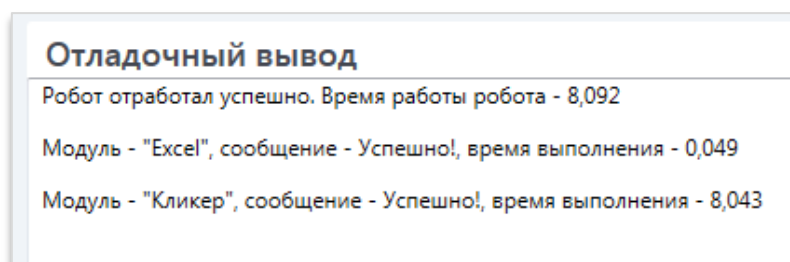
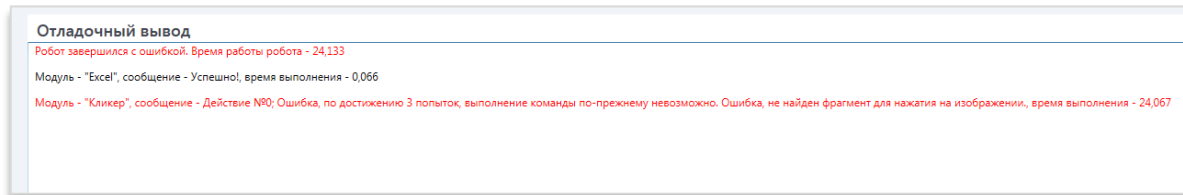


Рис. 1.1.16. Пример успешного вывода

В случае возникновения исключения в работе модуля, робот приостановит выполнение, и в отладочном выводе, в последней строке, будет содержаться информация об ошибке. Текст первой и последней строки отладочного вывода будет написан красным цветом.



```
Отладочный вывод
Робот завершился с ошибкой. Время работы робота - 24,133
Модуль - "Excel", сообщение - Успешно!, время выполнения - 0,066
Модуль - "Кликер", сообщение - Действие №0; Ошибка, по достижению 3 попыток, выполнение команды по-прежнему件 невозможно. Ошибка, не найден фрагмент для нажатия на изображении., время выполнения - 24,067
```

Рис. 1.1.17. Пример вывода при исключительной ситуации

Глава 2. Основы работы

Теперь, когда вы познакомились с интерфейсом студии, пришло время выяснить, как работать с этой программой. В этой главе вы узнаете о переменных и функциях, а также освоите маленькие хитрости (горячие клавиши), которыми сможете воспользоваться для быстрой работы.

Переменные

Переменные предназначены для хранения данных, которые должны быть доступны для всех действий проекта. Большинство модулей поддерживает чтение и запись данных в переменные.

К наименованию переменных предъявляется следующее требование: **название переменной должно всегда начинаться с символа @**.

Обращаться к переменным можно в полях окон настройки модулей приложения. При обращении необходимо указывать полное название переменной согласно описанному выше требованию.

В проекте можно создавать переменные трёх типов:

- Элемент – переменная, имеющая одно значение. Это может быть строка или число;
- Список – переменная, состоящая из нескольких элементов;
- Таблица – переменная в виде таблицы, в ячейках которой содержатся элементы.

Для создания переменной нажмите на кнопку «Создать переменную» в обзревателе решений, предварительно выбрав нужный проект (сделав его активным). Для создания локально хранимой и зашифрованной переменной необходимо нажать кнопку «Создать локальную переменную». После выполнения действия откроется окно создания переменной, это окно аналогично окну редактирования свойств переменной. Локальные переменные могут создаваться только с типом «Элемент», поэтому окно их настройки немного отличается от настройки обычных переменных.

В окне создания переменной укажите название переменной согласно требованиям приложения (**в начале наименования обязательно должен быть символ @**) и выберите тип переменной из списка. При изменении типа переменной изменяются доступные настройки.

Для переменных с типом «Элемент» доступны следующие настройки:

- поле для ввода значения;
- признак «Константа» (для констант обязательно указывать значение);
- шифрование переменной.

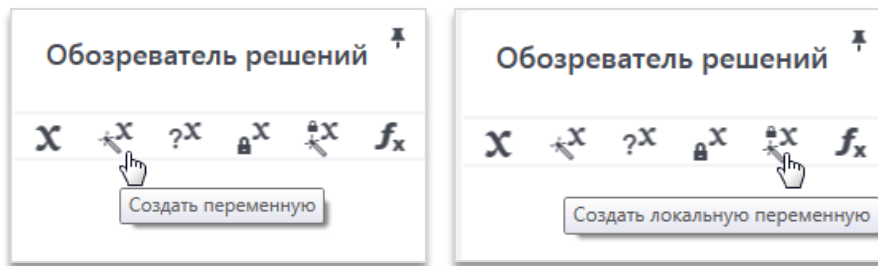


Рис. 1.2.1. Кнопки создания переменной

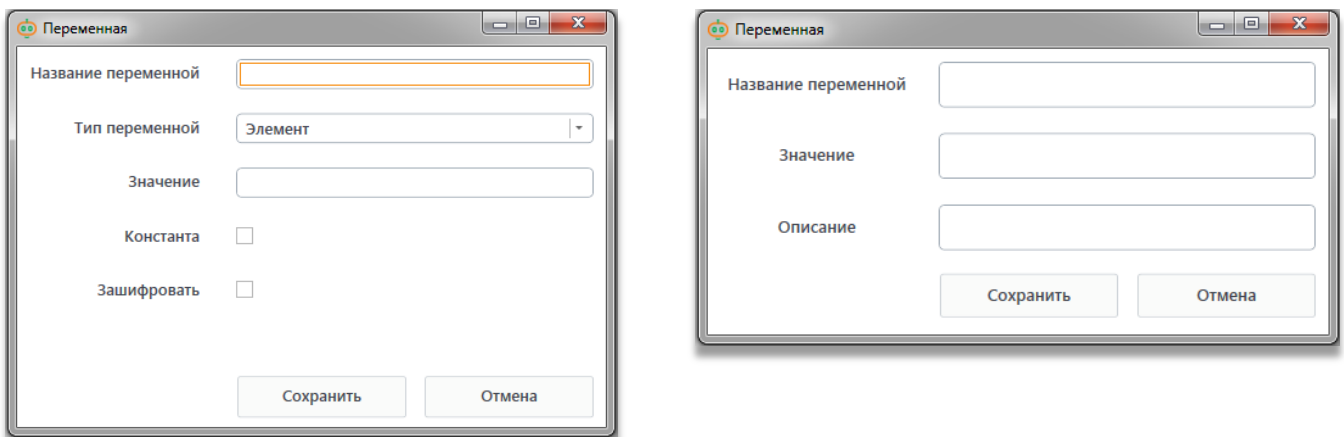


Рис. 1.2.2. Окна создания переменной (справа – локальной)

Переменные с типом «Список» хранят набор (массив) переменных. Обращение к каждому элементу списка осуществляется по его индексу, индекс начинается с 0. Индекс указывается в квадратных скобках, следующих сразу за названием переменной. Например, для обращения ко второму элементу - @array[1].

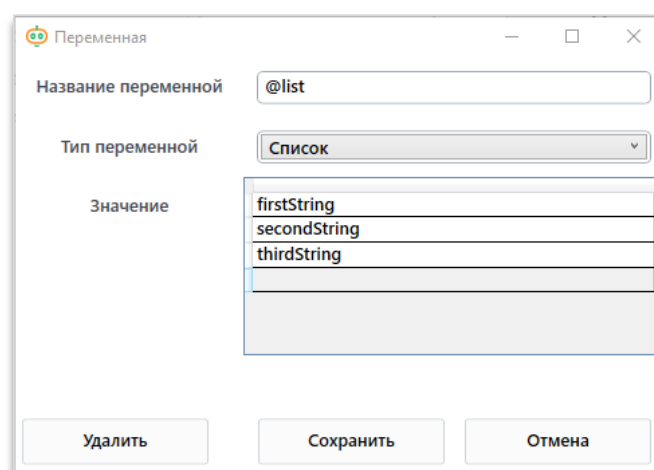


Рис. 1.2.3. Создание переменной-списка

Для создания переменных-списков становится доступной таблица, в которой можно заполнить требуемые значения. Значения вводятся в ячейки. Добавление следующего элемента осуществляется по нажатию клавиши «Enter».

Переменные с типом «Таблица» представляют собой таблицу (двумерный массив), т.е. это список, элементами которого являются списки. Другими словами, это набор однотипных данных, имеющий общее имя, доступ, к элементам которого осуществляется по двум индексам. Наглядно двумерный массив удобно представлять в виде таблицы, в которой n строк и m столбцов, а под ячейкой таблицы, стоящей в i -й строке и j -м столбце понимают некоторый элемент массива $a[i-1, j-1]$ (так как индекс начинается с нуля). Это означает, что для обращения к элементу в 3 строке и 5 столбце, отвечающему за ячейку таблицы, необходимо указывать два индекса: `@table[2, 4]`.

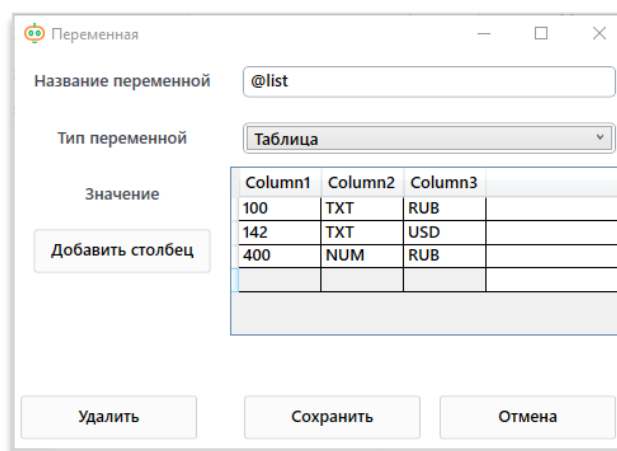


Рис. 1.2.4. Создание переменной-таблицы

Для заполнения значений переменных с типом «Таблица» становится доступной таблица, набор столбцов которой можно расширять с помощью кнопки «Добавить столбец». Добавление строк производится по нажатию клавиши «Enter». Вводить значения можно для каждой ячейки. **Незаполненные столбцы не будут использоваться при создании переменной.**

При нажатии в обозревателе решений на кнопку «Узнать значение переменной» открывается окно, с помощью которого можно проверить значение переменной активного проекта (рис. 1.2.5 – 1.2.6). **Для проверки значения необходимо ввести полное наименование переменной (включая символ @).**

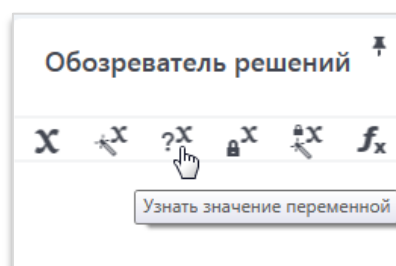


Рис. 1.2.5. Кнопка «Узнать значение переменной»

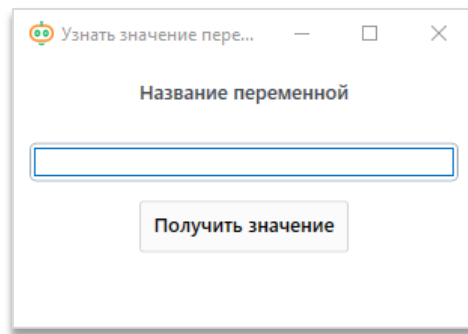


Рис. 1.2.6. Окно проверки значения переменной

Функции

Предустановленный набор вспомогательных функций позволяет выполнять различные вычисления в действиях проекта.

Для вызова встроенной функции необходимо перед её названием ввести знак #, например, #OperationPlus(1, 1). Аргументы функции, над которыми должна производиться операция, перечисляются в круглых скобках через запятую (если функция работает с несколькими аргументами). Между наименованием вызываемой функции и скобками с её аргументами не должно быть пробела.

Рабочая область

Это основной раздел главного окна студии. Здесь осуществляется настройка структуры проекта и выполняемых в нём действий.

Структура проекта

Действия добавляются с помощью меню доступных модулей. После добавления модуля в структуру проекта он отображается в виде элемента, обозначающего выполняемое действие.

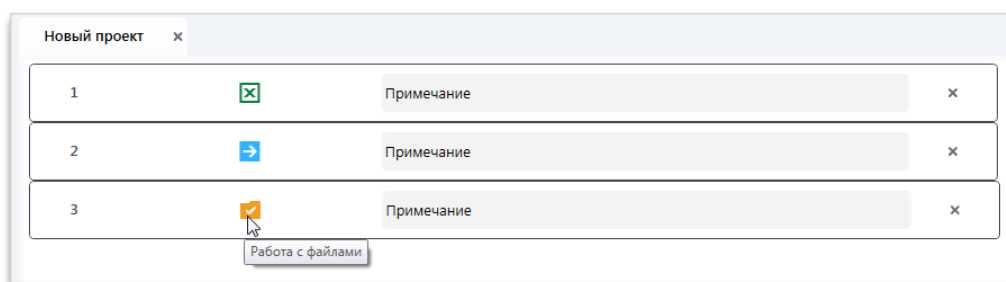


Рис. 1.2.7. Структура проекта

Элемент действия имеет несколько составляющих:

- порядковый номер действия;
- название действия;
- настраиваемое пользователем примечание;
- кнопку удаления действия.

В структуре проекта по нажатию правой кнопкой мыши на добавленное действие доступно контекстное меню со следующими пунктами:

- отключить/включить;
- снять/поставить точку остановки;
- копировать;
- вставить.

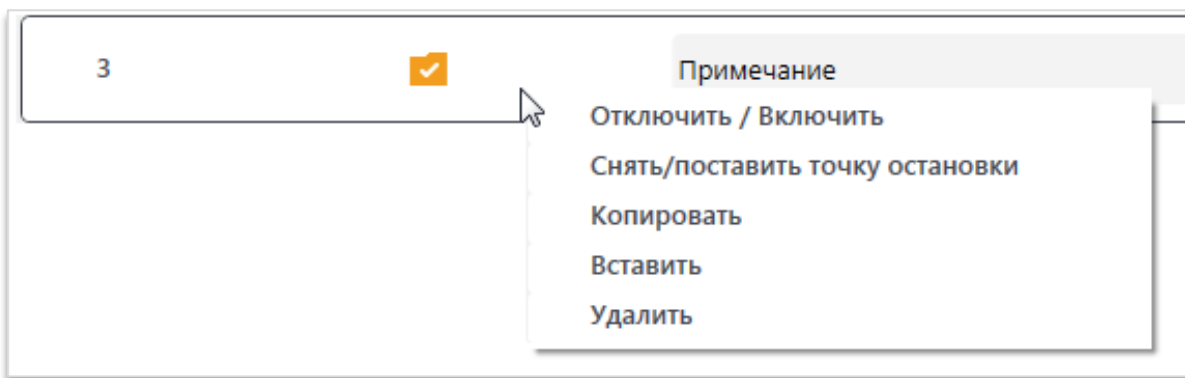


Рис. 1.2.8. Контекстное меню структуры проекта

Пункт **«Отключить/Включить»** позволяет переключать активность выбранного действия. Выключенное действие будет пропущено при запуске проекта.

Пункт **«Снять/поставить точку остановки»** создаёт или отключает точку остановки выполнения проекта. Точки остановки упрощают процесс наладки последовательности действий.

Пункт **«Копировать»** копирует в буфер обмена выбранное действие для последующей вставки.

Пункт **«Вставить»** вставляет скопированное действие после выделенного действия. Данный пункт доступен также при вызове контекстного меню в рабочей области главного окна студии.

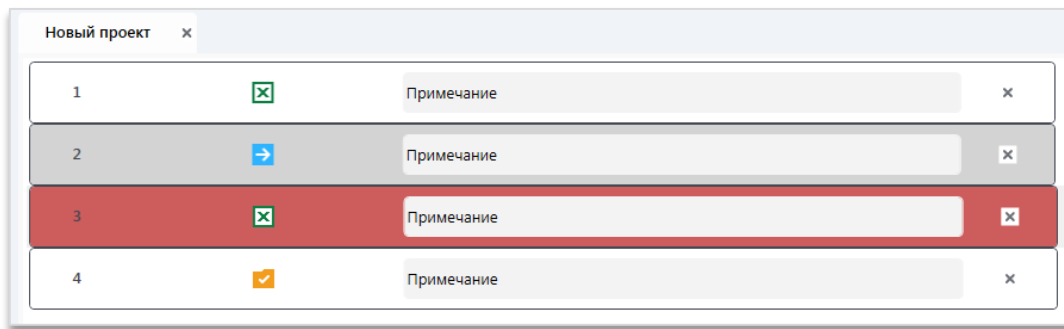


Рис. 1.2.9. Проект с отключенным действием и точкой остановки

Точка остановки

Точка остановки – это инструмент, который помогает отлаживаться при написании робота. Такой инструмент позволяет остановить работу робота после выполнения выбранного модуля, при этом появляется окно с указанием всех переменных, доступных на данный момент выполнения, для ознакомления с их содержанием.

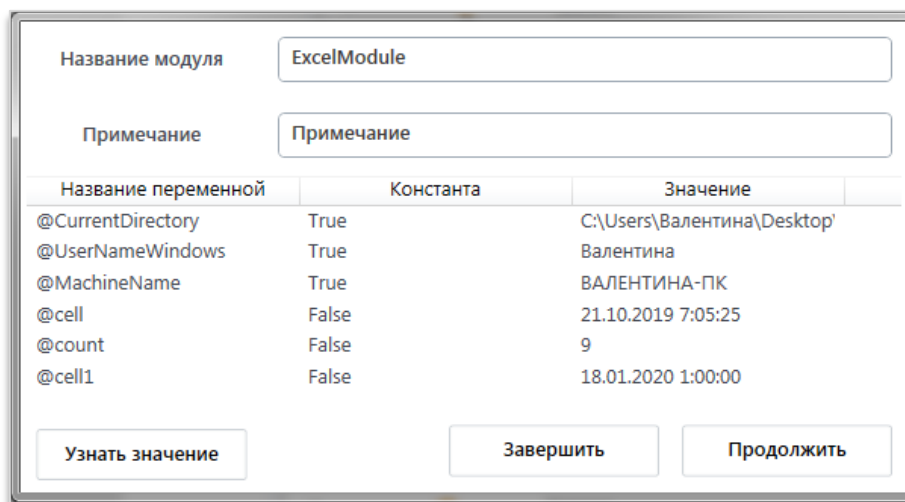


Рис. 1.2.10. Окно со списком переменных, доступных на момент выполнения робота

Редактировать переменные в данном окне нельзя. Для определения значения сложных переменных (переменные типов «Таблица» и «Список» или объекты) можно воспользоваться кнопкой «Узнать значение».

После ознакомления с переменными можно продолжить работу робота, а можно завершить, в любом из случаев это окно будет закрыто.

Настройка добавленных действий

Окно настройки свойств действия, добавленного в структуру проекта, открывается с помощью двойного щелчка левой кнопкой мыши по указанному действию. Набор свойств, доступных в этом окне, зависит от выбранного действия. Подробнее о доступных свойствах каждого действия будет описано далее в руководстве.

Горячие клавиши

Приложение поддерживает следующий набор комбинаций горячих клавиш:

Комбинация клавиш	Действие
<i>Ctrl+N</i>	Создать новый проект. Открывает диалог создания нового проекта.
<i>Ctrl+S</i>	Сохранить проект. Сохраняет проект в текущем расположении.
<i>Ctrl+Shift+S</i>	Сохранить проект как. Аналогично кнопке «Сохранить как» открывает диалог для указания нового расположения или имени файла проекта.
<i>Ctrl+O</i>	Открыть проект. Открывает сохраненный ранее проект с помощью системного диалога выбора файлов.
<i>Esc</i>	Закрыть программу.
<i>F5</i>	Запустить выполнение робота в текущем проекте.
<i>F1</i>	Перейти в справку по адресу http://wikie.lexema.ru/ .

ЧАСТЬ II. МОДУЛЬ «EXCEL»

Глава 3. Основные моменты модуля

Модуль «Excel» является самым обширным модулем в программе. Данный модуль предназначен для обработки Excel-файлов в фоновом режиме, незаметном для пользователя компьютера. Модуль последовательно выполнит заданный ему список команд.

Интерфейс модуля

Окно модуля состоит из нескольких разделов: «Добавить команду», «Общие настройки», «Список команд» и окно с предварительным просмотром файла.

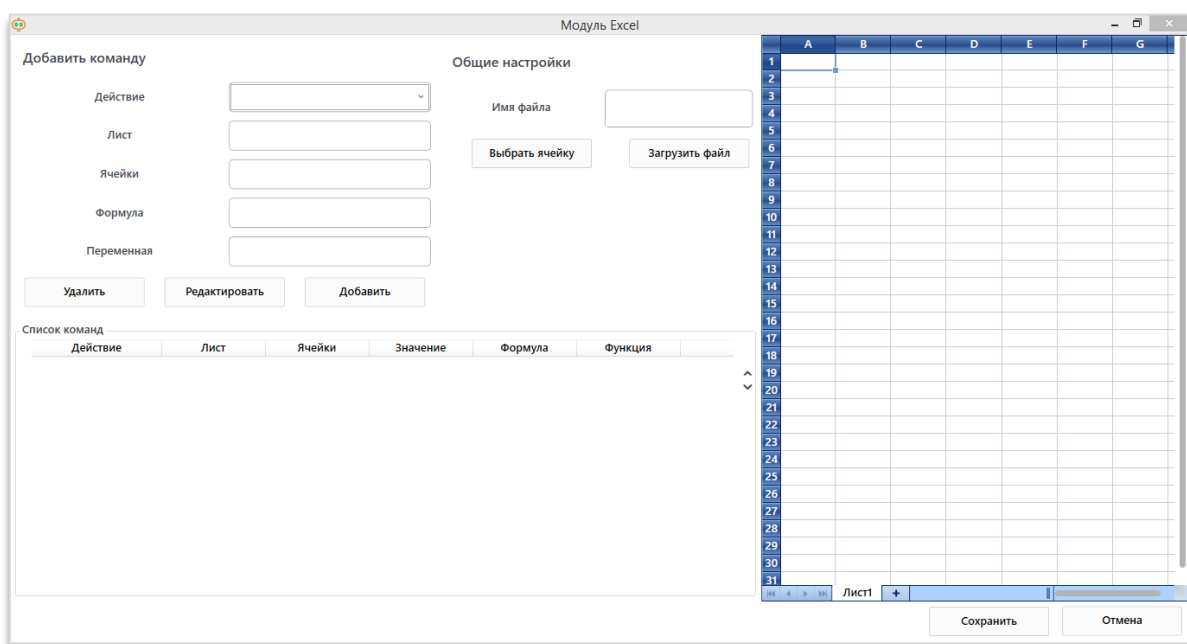


Рис. 2.3.1. Окно модуля Excel

Окно предварительного просмотра позволяет просматривать все листы загруженного файла с помощью вкладок с названиями листов и полосы прокрутки. Также есть возможность вписывать в ячейки свои значения, применять различное форматирование, но эти изменения сохранены не будут, то есть данный функционал доступен только для «примерки».

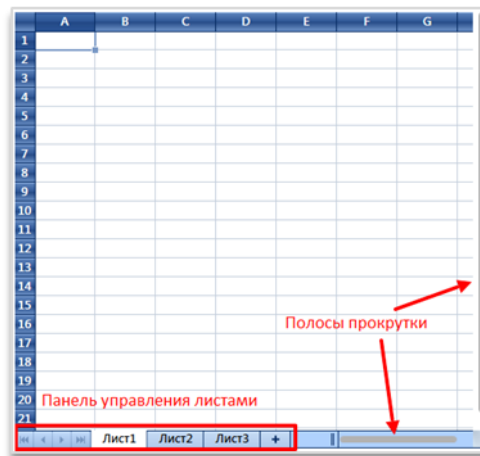


Рисунок 2.3.2. Элементы управления окна предварительного просмотра

Раздел «Общие настройки»

Загрузить файл

Раздел «Общие настройки» состоит из поля «Имя файла» и кнопок «Загрузить файл» и «Выбрать ячейку».

Работу с модулем Excel советуем начинать с выбора рабочего файла. После того, как по кнопке «Загрузить файл» будет выбран файл, необходимо подождать несколько секунд и указанный файл будет отображен в окне предварительного просмотра (рис. 2.3.3).

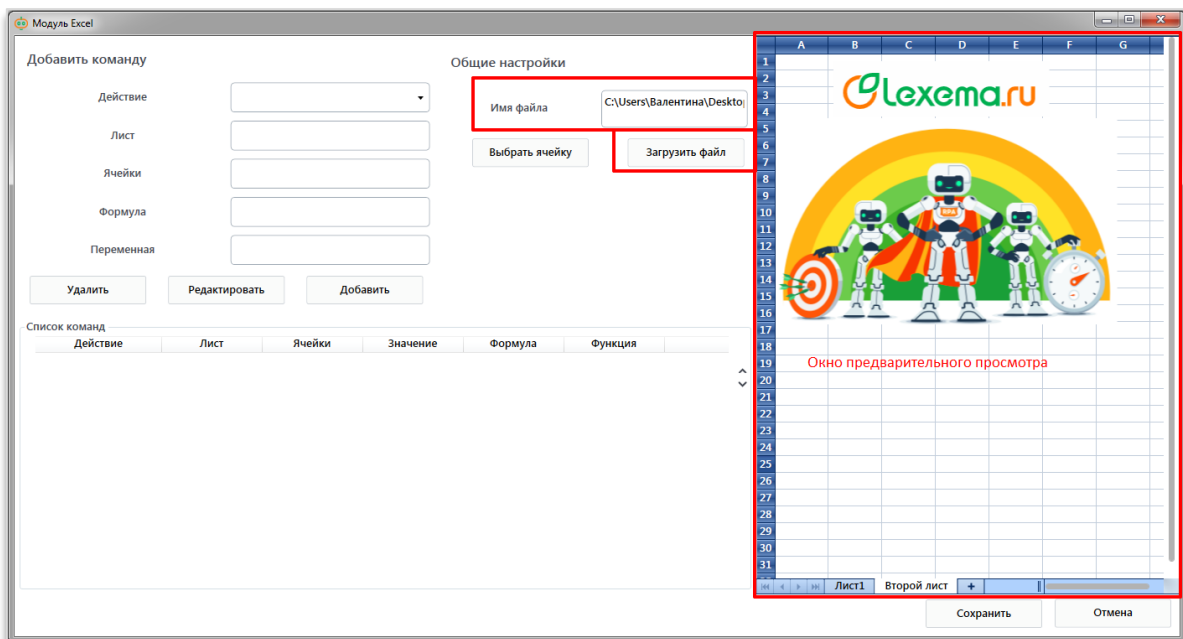


Рис. 2.3.3. Общие настройки

Разделы «Добавить команду» и «Список команд»

Раздел «Добавить команду» является основным «полотном» данного модуля. В нем создаются действия, имитирующие работу пользователя, например, такие как считать некоторый диапазон ячеек, затем перенести его на другой лист, удалить дубликаты и построить круговую диаграмму. В данном случае было описано 4 действия. Эти 4 действия необходимо последовательно выбирать и добавлять в раздел «Список команд» - таблицу, отображающую все действия, которые будут выполняться данным модулем. Набор полей в данном разделе не всегда одинаков – он меняется в зависимости от выбранного действия в первом поле.

Кнопки «Добавить», «Редактировать» и «Удалить» предназначены для работы с разделами «Добавить команду» и «Список команд».

Добавление команды

Для добавления команды предусмотрена кнопка «Добавить». Она заносит созданное действие в таблицу «Список команд».

Действие	Лист	Ячейки	Значение	Формула	Функция
Границы	Лист2	A2:A6	255,0,0,0	Внешние	Штрихпунктир с /
Считать данные	Лист2	A1:A4	@text		

Рис. 2.3.4. Добавление команды

При выборе строки в списке команд информация по выбранному действию отобразится в разделе «Добавить команду».

Помимо описанного функционала, кнопка «Добавить» служит для копирования уже имеющихся действий в списке команд. Для этого необходимо выбрать строку в списке команд и нажать на кнопку «Добавить». Таким образом, если нужно выполнить действие несколько раз, достаточно его копировать, а не создавать вновь.

Действие	Лист	Ячейки	Значение	Формула	Функция
Границы	Лист2	A2:A6	255,0,0,0	Внешние	Штрихпунктир с т
Считать данные	Лист2	A1:A4	@text		

Рис. 2.3.5. Отображение команды

Редактирование команды

Для редактирования команды предусмотрена кнопка «Редактировать». Чтобы изменить команду, выделите ее в списке команд, внесите необходимые изменения в разделе «Добавить команду» и нажмите на кнопку «Редактировать» для сохранения изменений.

Изменение порядка действий

Для изменения порядка действий предусмотрены кнопки с иконками стрелок «Вверх» и «Вниз». Кнопки расположены справа от списка команд.

Изменение порядка действий бывает необходимо, когда требуется добавить действие, которое не предусмотрели сразу и не внесли в список команд. В таком случае вы можете добавить действие стандартным способом, а затем переместить его.

Действие	Лист	Ячейки	Значение	Формула	Функция
Границы	Лист2	A2:A6	255,0,0,0	Внешние	Штрихпунктир с т
Считать данные	Лист2	A1:A4	@text		
Добавить строку	Лист2	12			

Рис. 2.3.6. Изменение порядка действий

Удаление команды

Кнопка «Удалить» удаляет выбранную строку из списка команд.

Отключение команды

Если вы не хотите удалять команду, так как хотели бы, например, проверить, как работает робот без неё, а затем вернуть её в работу, то можно отключить команду, нажав на соответствующую ей строку правой кнопкой мыши и отключить или, наоборот, включить её.

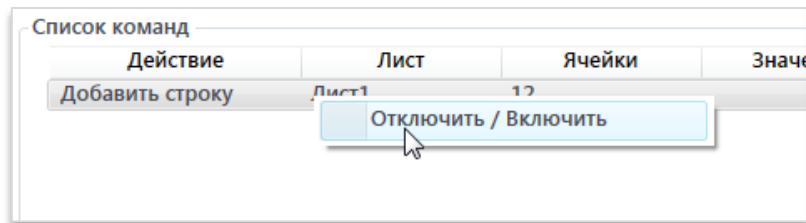


Рис. 2.3.7. Контекстное меню команды

Выбор ячейки

Кнопка «Выбрать ячейку» работает для каждого действия по-своему, но смысл у неё один – она заполняет поле «Ячейки» или «Диапазон» (в зависимости от того, какое из них используется в выбранном действии) названием выбранной в окне предварительного просмотра ячейки / диапазона и вписывает в поле «Лист» название листа, на котором находятся выбранные ячейки. **В некоторых действиях по кнопке заполняется только одна ячейка, хотя был выбран диапазон – это означает, что в данном действии использование диапазона невозможно.**

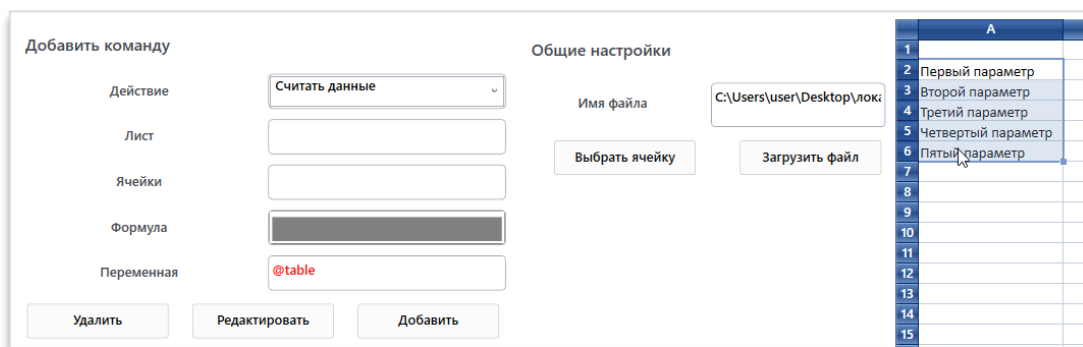


Рис. 2.3.8. Выбор диапазона в окне предварительного просмотра

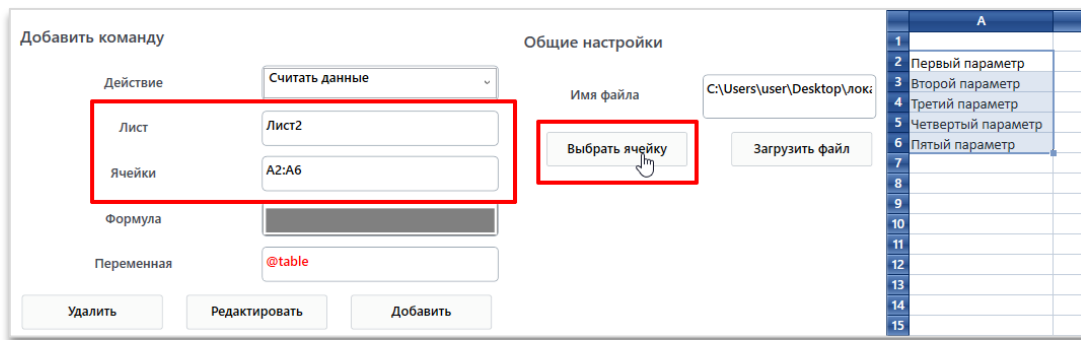


Рис. 2.3.9. Выбрать ячейку

В некоторых действиях кнопка «Выбрать ячейку» заполняет номера колонок, строк, какие-либо значения. При работе с файлами рекомендуется её использование для минимизации возникновения синтаксических ошибок.

Сохранение команды

Для сохранения команды в модуле Excel нажмите на кнопку «Сохранить» в правом нижнем углу модуля.

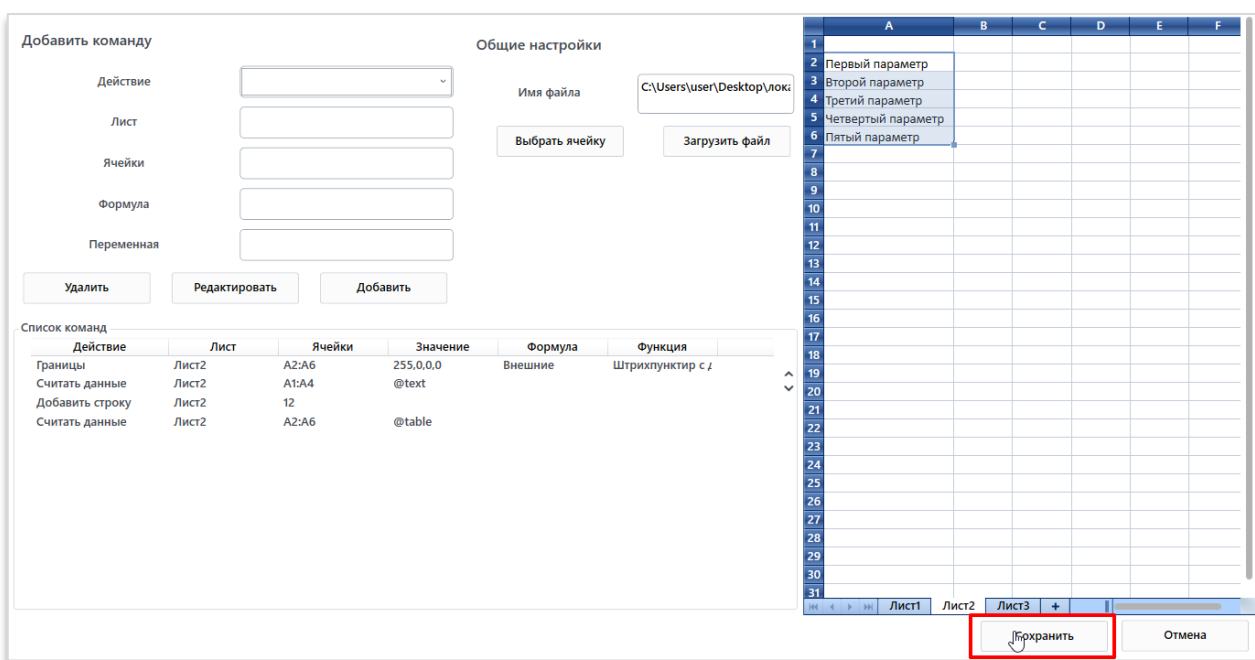


Рис. 2.3.10. Сохранение в модуле Excel

Глава 4. Набор полей в составе модуля

Для каждого действия в разделе «Добавить команду» существует свой набор необходимых для него полей, который будет отображен при выборе действия. По умолчанию, отображены поля «Лист», «Ячейки», «Формула» и «Переменная». Почти все команды имеют именно такой набор полей, но некоторые требуют несколько иной информации для выполнения действия.

Перечень возможных полей с правилами их заполнения

1. Лист – поле, в которое необходимо указать имя или номер листа (начиная с 0), на котором необходимо выполнить указанное действие;
2. Ячейки – поле, в котором ожидается либо адрес одной ячейки (A1), либо диапазон ячеек (A1:C10). Если адрес ячейки вычисляется динамически и неизвестен, то можно использовать номера колонок и строк ячеек, то есть вместо «C1» можно написать «3,1», где 3 – номер колонки, 1 – номер строки. Так же можно указать диапазон – вместо «A3:D4» - «1,3:4,4». Второй формат написания ячеек может быть использован только по отношению к полям «Ячейка(-и)», но невозможен в полях «Формула» (см. след. пункт);
3. Формула – поле для ввода формулы в стандартном формате Excel. Например «=ЕСЛИ(A2 > B2:1:0)». Для более удобного ввода длинных формул поле можно расширить, нажав на кнопку со стрелочкой в поле ввода:



Рис. 2.4.1. Поле «Формула»

4. Переменная – поле для ввода названия переменной, в которую будет помещен результат выполнения действия. Название вводится с символа «@»;
5. Значение – поле для ввода текста или названия переменной, в которой лежит текст, используемый в выбранном действии;
6. Столбец – номер (с 1) или название столбца;
7. Строка – номер строки (с 1). В некоторых действиях («Найти данные», «Найти лист») – текстовая строка для поиска в файле;
8. Путь к файлу – путь до папки, в которую будет помещен файл. Можно выбрать с помощью диалога выбора папки по нажатию на кнопку в поле:



Рис. 2.4.2. Поле «Путь к файлу»

9. Имя файла – название файла, можно указать как с расширением, так и без, в таком случае будет создан файл с расширением.xlsx;
10. Путь копируемого файла – путь до файла вместе с его расширением, который будет использоваться в данном действии помимо основного файла для копирования данных из него. Можно выбрать с помощью диалога выбора файлов по нажатию на кнопку в поле:



Рис. 2.4.3. Поле «Путь копируемого файла»

11. Лист копируемого файла – название или номер листа (с 0) в копируемом файле;
12. Колонки (ч/з ;) – аналогично полю «Столбец», только здесь возможно их перечисление через символ «;»;
13. Значения (ч/з ;) – строковые значения (или переменные, хранящие их), перечисленные через «;»;
14. Строка, колонка – номер строки и номер колонки, перечисленные через запятую;
15. Ячейка из диапазона значений – любая ячейка из таблицы, к которой применяется действие (например, если таблица A1:D10, то любая ячейка из этого диапазона – B5);
16. Функция – выпадающий список с набором функций, который меняется в зависимости от выбранного действия; обязателен выбор одного из представленных значений;
17. По убыванию – флажок, использующийся в действии «Добавить сортировку», который ставится, если необходимо сортировать по убыванию, и не ставится в противном случае;
18. Диапазон таблицы – аналогично полю «Ячейки», но с обязательным условием, что это должен быть именно диапазон (то есть должен присутствовать символ «:»);
19. Лист для вставки – название или номер листа (с 0);
20. Уникальные столбцы – используется в действии «Получить уникальные значения столбца», необходим ввод названий или номеров столбцов (с 1-цы), которые должны быть уникальными;
21. Для колонок – аналогично полю «Столбец»;
22. При каждом изменении в колонке – аналогично полю «Столбец»;
23. Лист, диапазон - поле, в которое должен быть введен лист файла (аналогично полю «Лист»), затем поставлена запятая и указан диапазон или ячейка (аналогично полю «Ячейки»);
24. Лист, диапазон источника – аналогично полю «Лист, диапазон»;

25. Лист, диапазон для вставки – аналогично полю «Лист, диапазон»;
26. Поля в столбцах – используется в действии «Сводная таблица», необходимо ввести название колонки из рабочей таблицы (не А, В, С и т.д., а именно название столбца вашей таблицы, например «Дата создания», без использования кавычек);
27. Поля в значениях – аналогично «Поля в столбцах»;
28. Поля в строках – аналогично «Поля в столбцах»;
29. Копируемый лист – аналогично полю «Лист»;
30. Копируемый диапазон/ячейка – аналогично полю «Ячейки»;
31. Диапазон/ячейка для вставки – аналогично полю «Ячейки»;
32. Файл для вставки (необяз.) – путь до файла с его расширением, является необязательным полем;
33. Расположение (лев. верхн., прав. нижн.) – две ячейки, разделенные запятой, которые обозначают положение - первая – левый верхний угол, вторая – правый нижний;
34. Диапазон названий – аналогично полю «Ячейки»;
35. Диапазон аргументов - аналогично полю «Ячейки»;
36. Диапазон значений - аналогично полю «Ячейки»;
37. Ячейка для вставки - аналогично полю «Ячейки», но только в единственном числе;
38. Тип данных – выпадающий список, обязателен выбор одного значения из представленных;
39. Операция – аналогично «Типу данных»;
40. Критерии (ч/з ;) – аргументы для выбранной операции, могут быть как числами, так и строками, перечисляются через «;»;
41. Цвет заливки – цвет в формате ARGB, с перечислением каждого параметра через запятую, возможен выбор цвета через диалог выбора цвета по кнопке в поле:



Рис. 2.4.4. Поле «Цвет заливки»

42. Цвет текста – аналогично «Цвет заливки»;
43. Цвет – аналогично полю «Цвет заливки»;
44. Значение для сравнения – текстовое или числовое значение;
45. Ячейка/диапазон – аналогично полю «Ячейки»;

46. Столбцы – перечисление номеров (с 1-цы) или названий необходимых столбцов через запятую;
47. Ширина – число, как целое, так и дробное (через «.» или «,»);
48. Строки – перечисление номеров (с 1-цы) строк через запятую;
49. Высота – аналогично полю «Ширина»;
50. Общее положение – выпадающий список, обязателен выбор одного из предложенных значений;
51. Выравнивание содержимого – аналогично полю «Общее положение»;
52. Шрифт – выпадающий список со всеми шрифтами, имеющимися в Microsoft Office, необходимо выбрать один из них;
53. Стиль текста – аналогично полю «Общее положение»;
54. Размер – аналогично полю «Ширина»;
55. Расположение – аналогично полю «Общее положение»;
56. Стиль – аналогично полю «Общее положение».

Во всех полях можно использовать переменные, но в них должно содержаться то, что требуют поля, то есть, в поле «Столбец» можно написать @column, но тогда в @column должен содержаться номер столбца, начинающийся с единицы, либо его название. Поля, которые закрашены серым цветом, не используются в выбранном действии.

Глава 5. Описание действий, доступных в модуле

Действия разделены на группы для удобной навигации по выпадающему списку. Если вы знаете точное название действия, то вы можете начать вводить его в поле «Действие», и программа постарается подсказать вам то действие, которое вы ищите.

Группа «Работа с данными»

Действие «Считать данные»

Данное действие позволяет считать значение из одной конкретной ячейки, либо из диапазона ячеек. Если будет указана одна ячейка, то создастся переменная с типом «Элемент», если же диапазон, то будет создана переменная с типом «Таблица».

Действие	Считать данные
Лист	Лист1
Ячейки	A1:B3
Формула	
Переменная	@table

Рис. 2.5.1. Пример настройки действия «Считать данные»

Действие «Записать данные»

Действие позволяет записать указанное значение в одну или в диапазон ячеек.

Действие	Записать данные
Лист	Лист1
Ячейки	A1
Формула	
Значение	Номер строки

Рис. 2.5.2. Пример настройки действия «Записать данные»

Действие «Формула»

Действие позволяет записать формулу в указанную ячейку (или в диапазон) и вернуть её результат в указанную переменную. Поле «Переменная» можно оставить пустым, если результат выполнения данной формулы не важен.

Действие	Формула
Лист	Лист1
Ячейки	C1
Формула	=СУММ(A1,B1) ↗
Переменная	@sum

Рис. 2.5.3. Пример настройки действия «Формула»

СОВЕТ. Предположим, что вам необходимо написать формулу, применимую к текущей строке, и «растянуть» её на несколько других строк, как в Excel. Начните с создания действия «Формула» для первой строки. Затем «растяните» её с помощью действия «Скопировать и вставить ячейку/диапазон» применительно к настроенной формуле. То есть для выше предложенной формулы «растягивание» будет выглядеть так:

Действие	Скопировать и вставить ячей
Копируемый лист	Лист1
Копируемый диапазон/ячейка	C1
Лист для вставки	Лист1
Диапазон/ячейка для вставки	C2:C10

Рис. 2.5.4. Пример настройки действия «Скопировать и вставить ячейку/диапазон»

Действие «Получить формулу»

Действие возвращает формулу из ячейки в указанную переменную.

Действие	Получить формулу
Лист	Лист1
Ячейки	C1
Формула	
Переменная	@formula

Рис. 2.5.5. Пример настройки действия «Получить формулу»

Действие «Заменить формулу на её значение»

Данное действие заменяет формулу в ячейке или диапазоне ячеек на её результат.

Действие	Заменить формулу на её значе
Лист	Лист1
Ячейки	C1:C10
Формула	
Значение	

Рис. 2.5.5. Пример настройки действия «Заменить формулу на ее значение»

Группа «Работа со строками»**Действие «Добавить строку»**

Действие вставляет новую строку в указанное место.

Действие	Добавить строку
Лист	Лист1
Строка	4
Формула	
Значение	

Рис. 2.5.6. Пример настройки действия «Добавление строки»

Действие «Удалить строку»

Действие удаляет указанную строку.

Действие	Удалить строку
Лист	Лист1
Строка	4
Формула	
Значение	

Рис. 2.5.7. Пример настройки действия «Удаление строки»

Действие «Получить количество строк»

Действие возвращает номер последней используемой строки в листе в указанную переменную.

Действие	Получить количество строк ▾
Лист	Лист1
Ячейки	
Формула	
Переменная	@countRow

Рис. 2.5.8. Пример настройки действия «Получение количества строк»

Действие «Получить количество отфильтрованных строк»

Действие возвращает количество видимых строк при примененном фильтре в указанную переменную.

Действие	Получить количество отфильтр ▾
Лист	Лист1
Ячейки	
Формула	
Переменная	@countFilteredRow

Рис. 2.5.9. Пример настройки действия «Получение количества отфильтрованных строк»

Группа «Работа с колонками»

Действие «Добавить столбец»

Действие добавляет новый столбец в указанное место.

Действие	Добавить столбец ▾
Лист	Лист1
Столбец	С
Формула	
Значение	

Рис. 2.5.10. Пример настройки действия «Добавление столбца»

Действие «Удалить столбец»

Действие удаляет указанный столбец.

Действие	Удалить столбец
Лист	Лист1
Столбец	С
Формула	
Значение	

Рис. 2.5.11. Пример настройки действия «Удаление столбца»

Действие «Получить количество столбцов»

Действие возвращает номер самого последнего используемого столбца.

Действие	Получить количество столбцов
Лист	Лист1
Ячейки	
Формула	
Переменная	@countColumn

Рис. 2.5.12. Пример настройки действия «Получение количества столбцов»

Группа «Работа с листами и файлами»**Действие «Создать новый лист»**

Действие создает новый лист с указанным именем.

Действие	Создать новый лист
Лист	Лист2
Ячейки	
Формула	
Значение	

Рис. 2.5.13. Пример настройки действия «Создание нового листа»

Действие «Удалить лист»

Действие удаляет лист с указанным именем.

Действие	Удалить лист
Лист	Лист2
Ячейки	
Формула	
Значение	

Рис. 2.5.15. Пример настройки действия «Удаление листа»

Действие «Очистить лист»

Действие очищает ячейки листа от данных. **Не очищает форматирование!**

Действие	Очистить лист
Лист	Лист1
Ячейки	
Формула	
Значение	

Рис. 2.5.16. Пример настройки действия «Очищение листа»

Действие «Создать новый файл»

Действие создает в указанной папке новый файл с названием и указанным расширением (если расширение не указано, создается файл .xlsx). Если поле «Лист» заполнено, то первый лист файла будет назван так, как указано в поле, иначе - по умолчанию (Лист 1, Sheet 1, в зависимости от языка системы).

Действие	Создать новый файл
Лист	Лист1
Путь к файлу	C:\Users\user\Desktop\робот
Формула	
Имя файла	Новый тестовый файл

Рис. 2.5.17. Пример настройки действия «Создание нового файла»

Действие «Заполнить лист на основе Excel-файла»

Данное действие производит заполнение листа текущего файла по аналогии с листом какого-либо другого файла. В поле «Лист» указывается лист текущего файла.

Действие	Заполнить лист на основе Excel ▾
Лист	Лист1
Путь копируемого файла	C:\Users\user\Desktop\локализ: 📁
Лист копируемого файла	Лист 1
Значение	

Рис. 2.5.18. Пример настройки действия «Заполнение листа»

Действие «Экспортировать в PDF»

Действие позволяет сохранить Excel-файл в формате PDF стандартными методами Windows, то есть деление листа Excel-файла на листы в документе PDF будет производиться по стандартным настройкам, указанным в файле Excel (по умолчанию размер листа A4, с обычными полями).

Действие	Экспортировать в PDF ▾
Лист	
Ячейки	
Формула	
Значение	

Рис. 2.5.19. Пример настройки действия «Экспортирование в PDF»

Название файла и его расположение будут совпадать с экспортируемым файлом.

Группа «Поиск»**Действие «Найти данные»**

Ищет заданную строку в указанном листе и возвращает адрес первой найденной ячейки.

Действие	Найти данные
Лист	Лист1
Строка	Дата
Формула	
Переменная	@findData

Рис. 2.5.20. Пример настройки действия «Найти данные»

Действие «Поиск строки по значениям колонок»

Данное действие ищет и возвращает номер первой найденной строки, в заданных колонках которой записаны указанные значения.

Действие	Поиск строки по значениям ко. ▾
Лист	Лист1
Колонки (ч/з ;)	A;B
Значения (ч/з ;)	23.03.2020;Уфа
Переменная	@findStr

Рис. 2.5.21. Пример настройки действия «Поиск строки»

В данном случае будет искаться строка, в колонке А которой 23.03.2020, а в колонке В – Уфа.

Действие «Найти лист»

Ищет лист, в котором есть ячейка с указанным в поле «Строка» значением и возвращает его название в указанную переменную.

Действие	Найти лист
Лист	
Строка	Данные, указанные на опр. листе
Формула	
Переменная	@findList

Рис. 2.5.22. Пример настройки действия «Найти лист»

Группа «Другие функции»

Действие «Сместить ячейку»

Смещает заданную ячейку на указанное количество строк и колонок, затем возвращает результат в переменную (в данном примере результатом будет ячейка D13).

Действие	Сместить ячейку
Лист	Лист1
Ячейка	A13
Строка, колонка	0,3
Переменная	@cell

Рис. 2.5.23. Пример настройки действия «Сместить ячейку»

Действие «Наложить фильтр»

Данное действие добавляет фильтр в таблицу на указанном листе, ячейка из которой указана в поле «Ячейка из диапазона значений». В поле столбец записывается столбец, к которому необходимо применить фильтр. Фильтруемый столбец должен быть один. Если необходимо применить несколько фильтров к одной таблице, то необходимо добавить для каждого фильтра свое действие в рамках одного Excel-модуля. Функция, по которой должно происходить сравнение, выбирается из выпадающего списка в поле «Функция», значение, с которым будет происходить сравнение при фильтрации, указано в поле «Значение».

Действие	Наложить фильтр	Имя файла	C:\Users\user\Desktop\
Лист	Лист 2	Выбрать ячейку	Загрузить файл
Столбец	A	Функция	больше
Значение	15		
Ячейка из диапазона значений	A2		

Рис. 2.5.24. Пример настройки действия «Наложить фильтр»

Действие «Удалить фильтр»

Действие удаляет все фильтры с листа.

Действие	Удалить фильтр
Лист	Лист 2
Ячейки	
Формула	
Значение	

Рис. 2.5.25. Пример настройки действия «Удаление фильтра»

Действие «Добавить сортировку»

Действие добавляет сортировку к указанному диапазону по выбранному столбцу. В зависимости от флажка определяется, сортируется столбец по убыванию или по возрастанию. Столбец для сортировки должен быть один.

Действие	Добавить сортировку
Лист	Лист 2
Ячейки	A1:D15
Столбец	B
По убыванию	<input type="checkbox"/>

Рис. 2.5.26. Пример настройки действия «Добавление сортировки»

Действие «Очистить диапазон»

Действие удаляет все значения и форматирования ячеек в указанном диапазоне.

Действие	Очистить диапазон
Лист	Лист 2
Ячейки	A1:D15
Формула	
Значение	

Рис. 2.5.27. Пример настройки действия «Очищение диапазона»

Действие «Получить уникальные значения столбца»

Аналогично действию «Удалить дубликаты» в Excel, данное действие возвращает только уникальные значения указанных столбцов из диапазона. Итоговая таблица может быть вставлена как в тот же лист, так и в другой в лист, но в тот же диапазон. В зависимости от наличия дубликатов, количество строк может быть уменьшено.

Действие	Получить уникальные знач
Лист	Лист 2
Диапазон таблицы	A1:D15
Лист для вставки	Лист 3
Уникальные столбцы	A,B

Рис. 2.5.28. Пример настройки действия «Получение уникальных значений»

Действие «Промежуточные итоги»

Действие добавляет операцию Excel «Промежуточные итоги» на указанный лист к указанному диапазону.

Действие	Промежуточные итоги	Имя файла	C:\Users\user\Desktop\лс
Лист	Лист1	Выбрать ячейку	Загрузить файл
Ячейки	A1:D15		
Для колонок	D		
При каждом изменении в колонке	A	Функция	СЧЁТЗ

Рис. 2.5.29. Пример настройки действия «Промежуточные итоги»

Поле «Для колонок» соответствует полю в Excel «Добавить итоги по» (выделено зеленым на рис. 2.3.30), поле «При каждом изменении в колонке» соответствует аналогичному полю в Excel, но должно быть заполнено названием колонки Excel (A, B, ... или 1, 2, ...) (выделено красным на рис. 2.3.30), поле «Функция» заполняется операцией, которая должна быть применена для выбранных колонок (выделено синим на рис. 2.3.30).

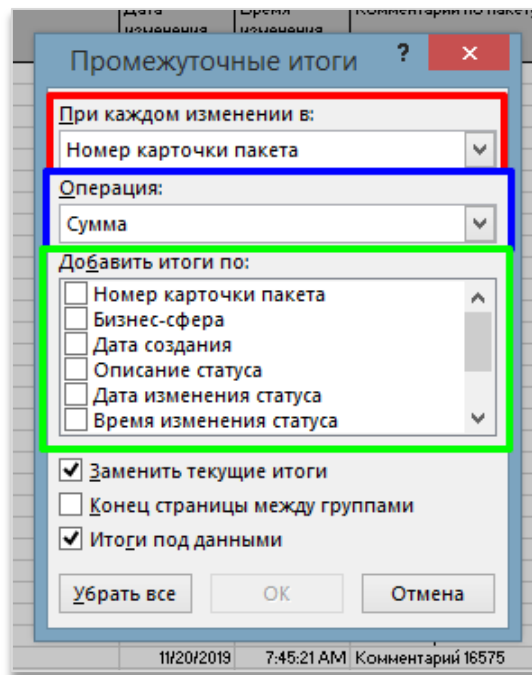


Рис. 2.5.30. Настройка действия «Промежуточные итоги» в Excel

Действие «Сводная таблица»

Добавляет сводную таблицу в указанные лист и диапазон (поле «Лист, диапазон для вставки») на основе данных таблицы из поля «Лист, диапазон источника».

Действие	Сводная таблица	Имя файла	C:\Users\user\Desktop\локс
Лист, диапазон источника	Лист1, A1:E200	Выбрать ячейку	Загрузить файл
Лист, диапазон для вставки	Лист2, A1		
Поля в столбцах	Бизнес-сфера		
Поля в значениях	Нарушение срока	Поля в строках	Дата создания;Номер карт

Рис. 2.5.31. Пример настройки действия «Сводная таблица»

В поля «Поля в столбцах», «Поля в значениях» и «Поля в строках» вписываются **названия колонок таблицы-источника** (а не Excel-таблицы), как в программе Excel:

«Поля в столбцах» соответствуют названиям столбцов таблицы, значения которых должны быть расположены в столбцах сводной таблицы (выделено зеленым на рис. 2.5.32);

«Поля в значениях» соответствуют названиям столбцов таблицы, значения которых должны быть расположены в значениях сводной таблицы (выделено красным на рис. 2.5.32);

«Поля в строках» соответствуют названиям столбцов таблицы, значения которых должны быть расположены в строках сводной таблицы (выделено синим на рис. 2.5.32);

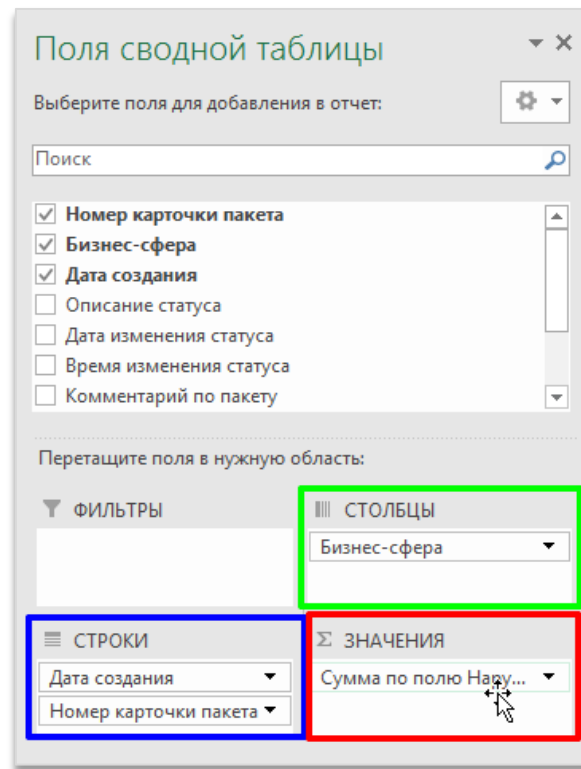


Рис. 2.5.32. Настройка действия «Сводная таблица» в Excel

Действие «Скопировать и вставить ячейку/диапазон»

Данное действие позволяет скопировать данные и форматирование и вставить их в тот же или в другой файл. С помощью него можно «растягивать» формулы (см. «Действие «Формула»).

Действие	Скопировать и вставить ячейку/д	Имя файла	C:\Users\user\Desktop\локэ
Копируемый лист	Лист1	Выбрать ячейку	Загрузить файл
Копируемый диапазон/ячейка	A1:D15		
Лист для вставки	Лист2		
Диапазон/ячейка для вставки	D5	Файл для вставки (необяз.)	C:\Users\user\Desktop\лр

Рис. 2.5.33. Пример настройки действия «Скопировать и вставить ячейку/диапазон»

Действие «Отобразить в виде процентов»

Действие переводит выбранный диапазон в проценты.

Действие	Отобразить в виде процентов ▾
Лист	Лист1
Ячейки	A1:D15
Формула	
Значение	

Рис. 2.5.34. Пример настройки действия «Отображение в виде процентов»

Действие «Круговая диаграмма»

Действие создает на указанном листе круговую диаграмму. Значения диаграммы берутся из диапазона, указанного в поле «Ячейки», диапазон с названиями значений указывается в поле «Диапазон названий». В поле «Расположение» указываются границы расположения круговой диаграммы, а именно левый верхний угол и правый нижний через запятую.

Действие	Круговая диаграмма ▾
Лист	Лист2
Ячейки	B2:B3
Расположение (лев. верхн., прав. нижн.)	A5,E10
Диапазон названий	A2:A3

Рис. 2.5.35. Пример настройки действия «Создание круговой диаграммы»

ПРИМЕР.

Построение круговой диаграммы для двух параметров, указанных в столбце А, по соответствующим им параметрам, указанным в столбце В. Заполнение действия для данного примера указано на рисунке 2.5.35.

Пример исходных данных представлен на рисунке 2.5.36.

	A	B	C
1			
2	Первый параметр	1002	
3	Второй параметр	580	
4			
5			
6			
7			
8			

Рис. 2.5.36. Пример исходных данных

Результат построения круговой диаграммы представлен на рис. 2.5.37.

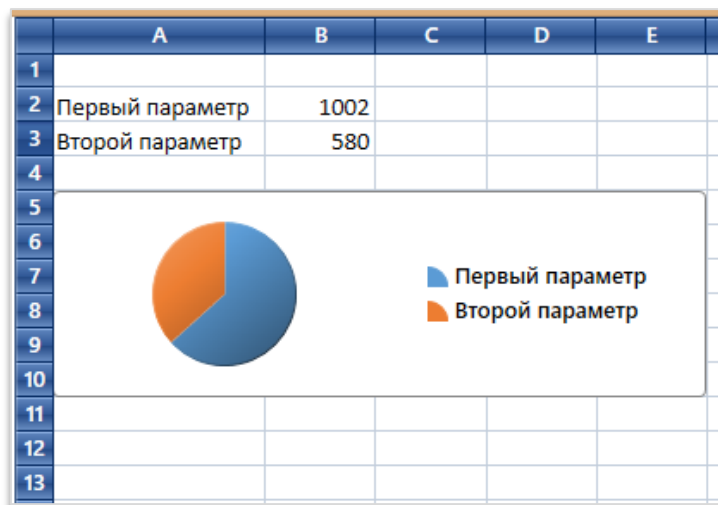


Рис. 2.5.37. Результат выполнения действия «Круговая диаграмма»

Действие «График»

Действие создает линейный график с одним или несколькими рядами. Ряд необходимо указать в поле «Диапазон значений» в виде одномерного диапазона (по одной колонке или по одной строке). Если рядов больше, чем один, то последующие ряды нужно указать в таком же виде через «;», как указано на рис. 2.5.38. Диапазон аргументов должен быть один и указывается аналогично ряду. «Ячейки с названиями рядов» заполняются названиями ячеек через «;», их количество должно быть равно количеству рядов, первая ячейка отображает название первого ряда и т.д. В ячейку для вставки записывается ячейка, в которой будет располагаться левый верхний угол графика.

Действие	График	Имя файла	C:\Users\user\Desktop\лок:
Лист	Лист1	Выбрать ячейку	Загрузить файл
Диапазон значений	B2:B19;C2:C19		
Диапазон аргументов	A2:A19		
Ячейки с названиями рядов	B1:C1	Ячейка для вставки	A21

Рис. 2.5.38. Пример настройки действия «График»

ПРИМЕР.

В качестве примера возьмем курс доллара и евро за некоторый промежуток времени. Так как валют две, то график должен состоять из двух рядов. В качестве аргументов выступит столбец с указанием дат – А, в качестве значений будет два столбца, первый – В, курс евро, и второй – С, курс доллара. Заполнение действия представлено на рис. 2.5.38.

Пример исходных данных представлен на рис. 2.5.39.

	А	В	С
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8896
6	18.03.2020	82,3056	74,1262
7	17.03.2020	82,7471	73,1882
8	14.03.2020	81,861	74,0274
9	13.03.2020	83,6584	71,472
10	12.03.2020	81,0207	72,0208
11	11.03.2020	81,8588	67,5175
12	07.03.2020	75,8424	66,1854
13	06.03.2020	73,6842	66,0784
14	05.03.2020	73,7369	66,4437
15	04.03.2020	73,9385	66,3274
16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213
20			

Рис. 2.5.39. Пример настройки действия «График»

Результат работы программы представлен на рис. 2.5.40.

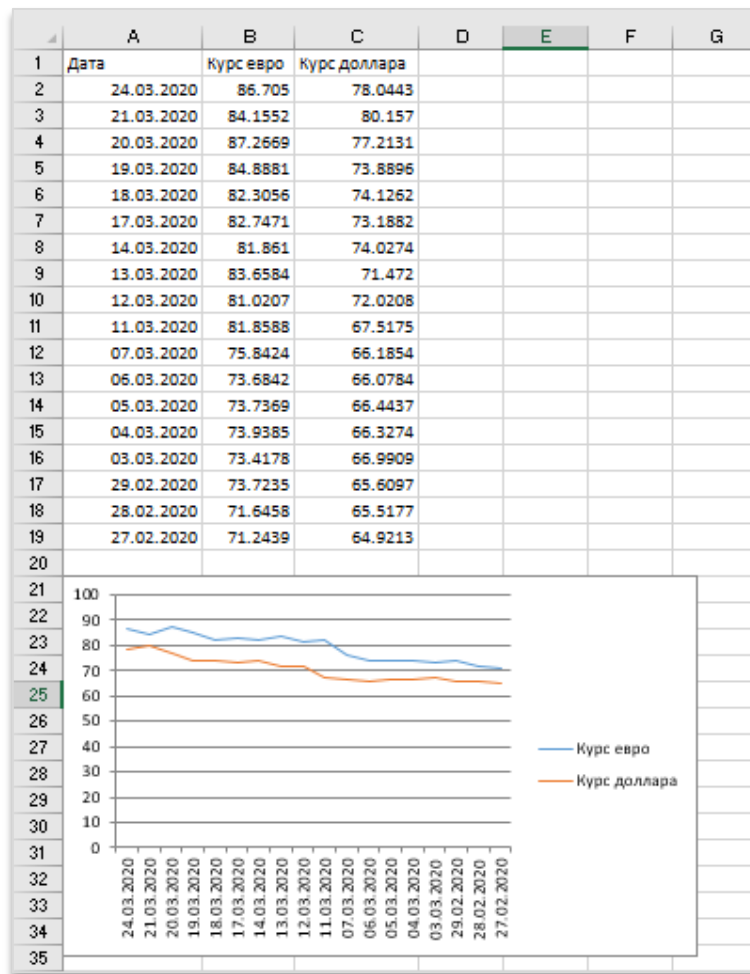


Рис. 2.5.40. Результат выполнения действия «График»

Действие «Проверка значений»

Действие создает ограничение типов данных, которые можно ввести в выбранные ячейки, путем выбора предложенных правил. В поле «Лист» вводится лист, в ячейки которого добавляются ограничения, в поле «Ячейки» вводится диапазон ячеек, на которых будет проводиться проверка значений при вводе. В поле «Тип данных» выбирается один из предложенных типов данных – целое или действительное число, список, дата, длина текста, время. Поле «Операция» состоит из списка доступных операций для составления правила проверки – равно, не равно, больше, меньше, больше или равно, меньше или равно, между и вне. При использовании типа данных «Список» заполнение поля «Операция» не требуется, поэтому оно не доступно для редактирования.

Последнее поле меняется в зависимости от типа данных – для типа «Список» это будет «Источник данных», для всех остальных – «Критерии (ч/з ;)». Поле «Источник данных» заполняется диапазоном значений (внутри одного столбца или одной строки), в которой будет указан в списке (если список статичный, то указывайте абсолютную ссылку). Поле «Критерии (ч/з ;)» заполняется значениями, относительно которых будет применяться правило проверки.

Поле «Сообщение об ошибке» заполняется текстом ошибки, которая будет отображаться, если в ячейку введено значение, не удовлетворяющее заданному для неё правилу. Является необязательным полем. Если оставить его пустым, то будет отображаться ошибка «Введенное значение неверно. Набор значений, которые могут быть введены в ячейку, ограничен».

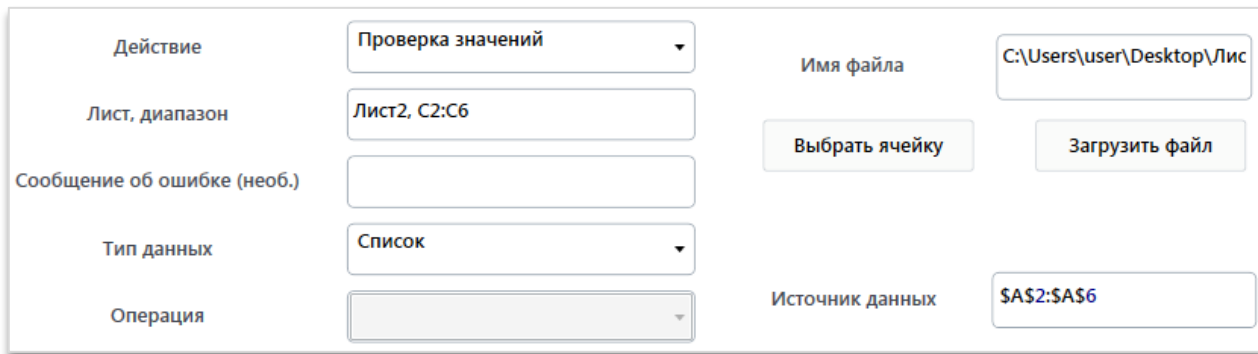


Рис. 2.5.41. Пример настройки действия «Проверка значений»

ПРИМЕР.

Для создания списка, состоящего из некоторых заданных в какой-либо колонке значений, действие заполняется так, как указано на рисунке 2.5.41. Пусть в ячейках A2:A6 будут следующие значения:

	A	B	C
1			
2	Первый параметр		
3	Второй параметр		
4	Третий параметр		
5	Четвертый параметр		
6	Пятый параметр		
7			

Рис. 2.5.42. Пример исходных данных

Тогда после работы робота при нажатии на любую ячейку из диапазона C2:C6 будет появляться стрелочка для открытия выпадающего списка, в котором будут содержаться выбранные значения:

	A	B	C	D
1				
2	Первый параметр			
3	Второй параметр			
4	Третий параметр			
5	Четвертый параметр			
6	Пятый параметр			
7				
8				
9				
10				
11				
12				

Рис. 2.5.43. Результат выполнения действия «Проверка значений»

Второй пример - создание правила, позволяющего вводить в диапазон ячеек A1:A5 только действительные числа, находящиеся в промежутке между 5 и 10. Пусть при введении неверного значения будет выдаваться ошибка: «Введено неверное значение. Должно быть введено действительное число между 5 и 10»:

Действие	Проверка значений	Имя файла	C:\Users\user\Desktop\Лис
Лист, диапазон	Лист1,A1:A5	Выбрать ячейку	Загрузить файл
Сообщение об ошибке (необ.)	Введено неверное значение. Долж		
Тип данных	Действительное число	Критерии (ч/з :)	5;10
Операция	между		

Рис. 2.5.44. Пример создания правила для проверки значений

После работы робота при попытке ввести число, находящееся вне диапазона от 5 до 10, появляется ошибка:

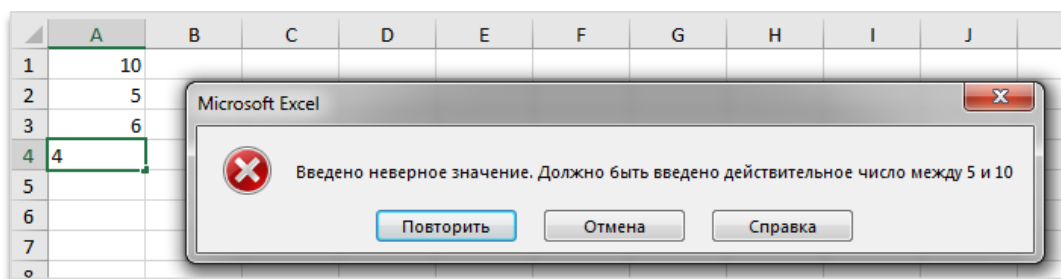


Рис. 2.5.45. Результат выполнения действия «Проверка значений»

Группа «Форматирование»

Действие «Условное форматирование»

Действие форматирует указанные ячейки на выбранном листе в зависимости от того, удовлетворяют ли они указанному условию. В поле «Лист, диапазон» через запятую вводятся лист и диапазон или одна ячейка из этого листа, к которым будет применяться условное форматирование. В полях «Цвет заливки» и «Цвет текста» указываются цвета, которыми будет залита ячейка и окрашен текст в ней, если она будет удовлетворять условию. Условие составляется из выбора функции и указания значения для сравнения. Доступные функции – равно, не равно, больше, меньше, больно или равно, меньше или равно, формула. Если выбрана функция «формула», то в поле «Значение для сравнения» следует вписать формулу в формате Excel, начиная со знака «=».

Рис. 2.5.46. Пример настройки действия «Условное форматирование»

ПРИМЕР.

Окрасим все ячейки в столбцах В и С в зеленый цвет, если значение курса меньше или равно 82. Действие заполняется как на рисунке 2.5.46.

Исходные данные представлены на рис. 2.5.47:

	А	В	С
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8896
6	18.03.2020	82,3056	74,1262
7	17.03.2020	82,7471	73,1882
8	14.03.2020	81,861	74,0274
9	13.03.2020	83,6584	71,472
10	12.03.2020	81,0207	72,0208
11	11.03.2020	81,8588	67,5175
12	07.03.2020	75,8424	66,1854
13	06.03.2020	73,6842	66,0784
14	05.03.2020	73,7369	66,4437
15	04.03.2020	73,9385	66,3274
16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213
20			

Рис. 2.5.47. Пример исходных данных

Результат работы программы представлен на рис. 2.5.48.

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8896
6	18.03.2020	82,3056	74,1262
7	17.03.2020	82,7471	73,1882
8	14.03.2020	81,861	74,0274
9	13.03.2020	83,6584	71,472
10	12.03.2020	81,0207	72,0208
11	11.03.2020	81,8588	67,5175
12	07.03.2020	75,8424	66,1854
13	06.03.2020	73,6842	66,0784
14	05.03.2020	73,7369	66,4437
15	04.03.2020	73,9385	66,3274
16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213

Рис. 2.5.48. Результат выполнения действия «Условное форматирование»

Действие «Цвет ячейки/диапазона»

Действие меняет цвет заливки текста и самого текста в указанном диапазоне / ячейке.

Действие	Цвет ячейки/диапазона
Лист	Лист1
Ячейка/диапазон	A17:C19
Цвет заливки	255 128 255 255
Цвет текста	255 128 0 128

Рис. 2.5.49. Пример настройки действия «Цвет ячейки/диапазона»

ПРИМЕР.

На рисунке 2.5.49 представлено, как залить диапазон ячеек A17:C19 в голубой цвет, а текст в этих ячейках окрасить в фиолетовый.

Исходные данные представлены на рис. 2.5.50:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8896
6	18.03.2020	82,3056	74,1262
7	17.03.2020	82,7471	73,1882
8	14.03.2020	81,861	74,0274
9	13.03.2020	83,6584	71,472
10	12.03.2020	81,0207	72,0208
11	11.03.2020	81,8588	67,5175
12	07.03.2020	75,8424	66,1854
13	06.03.2020	73,6842	66,0784
14	05.03.2020	73,7369	66,4437
15	04.03.2020	73,9385	66,3274
16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213
20			

Рис. 2.5.50. Пример исходных данных

Результат работы робота представлен на рис. 2.5.51:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8896
6	18.03.2020	82,3056	74,1262
7	17.03.2020	82,7471	73,1882
8	14.03.2020	81,861	74,0274
9	13.03.2020	83,6584	71,472
10	12.03.2020	81,0207	72,0208
11	11.03.2020	81,8588	67,5175
12	07.03.2020	75,8424	66,1854
13	06.03.2020	73,6842	66,0784
14	05.03.2020	73,7369	66,4437
15	04.03.2020	73,9385	66,3274
16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213
20			

Рис. 2.5.51. Результат выполнения действия «Цвет ячейки/диапазона»

Действие «Ширина столбца(-ов)»

Действие изменяет ширину указанных столбцов. Столбцы можно перечислять через запятую, писать их номера вместо названий (с 1-цы). Ширина указывается в той же единице измерения, которая используется в Excel.

Действие	Ширина столбца(-ов)
Лист	Лист1
Столбцы	B,C
Формула	
Ширина	16

Рис. 2.5.52. Пример настройки действия «Ширина столбца(-ов)»

ПРИМЕР.

На рисунке 2.5.52 представлено, как заполнить действие для установки ширины столбцов B и C в 16 пунктов. В поле «Столбцы» можно было бы написать «2,3», то есть второй и третий столбцы.

Исходные данные представлены на рис. 2.5.53:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131

Рис. 2.5.53. Пример исходных данных

Результат работы представлен на рис. 2.5.54:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131

Рис. 2.5.54. Результат выполнения действия «Ширина столбца(-ов)»

Действие «Высота строки (-ок)»

Действие изменяет высоту указанных строк. Используется та же единица измерения, что и в Excel.

Действие	Высота строки(-ок)
Лист	Лист1
Строки	1
Формула	
Высота	30

Рис. 2.5.55. Пример настройки действия «Высота строки(-ок)»

ПРИМЕР.

Изменим высоту первой строки, сделав её равной 30 пунктам. Для этого заполняем действие так, как указано на рисунке 2.5.55.

Исходные данные представлены на рис. 2.5.56:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8886

Рис. 2.5.56. Пример исходных данных

Результат работы робота представлен на рис. 2.5.57:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131

Рис. 2.5.57. Результат выполнения действия «Высота строки(-ок)»

Действие «Объединить ячейки»

Действие объединяет указанные ячейки на выбранном листе. В модуле предусмотрено три вида объединений:

- простое объединение – то есть объединяются и колонки, и столбцы диапазона;
- объединение по строкам;
- объединение по столбцам.

Рис. 2.5.58. Пример настройки действия «Объединить ячейки»

ПРИМЕР.

На рисунке 2.5.58 представлено заполнение действия для объединения колонок в диапазоне ячеек A2:C4.

Исходные данные представлены на рис. 2.5.59:

	A	B	C
1	Дата	Курс евро	Курс доллара
2	24.03.2020	86,705	78,0443
3	21.03.2020	84,1552	80,157
4	20.03.2020	87,2669	77,2131
5	19.03.2020	84,8881	73,8896
6	18.03.2020	82,3056	74,1262
7	17.03.2020	82,7471	73,1882
8	14.03.2020	81,861	74,0274
9	13.03.2020	83,6584	71,472

Рис. 2.5.59. Пример исходных данных

Результат работы представлен на рис. 2.5.60:

	A	B	C
1	Дата	Курс евро	Курс доллара
2			
3			
4	24.03.2020	86,705	78,0443
5	19.03.2020	84,8881	73,8896

Рис. 2.5.60. Результат выполнения действия «Объединить ячейки»

Действие «Очистить форматирование»

Действие очищает форматирование в указанном диапазоне выбранного листа.

Действие	Очистить форматирование ▾
Лист	Лист1
Ячейки	A17:C19
Формула	
Значение	

Рис. 2.5.61. Пример настройки действия «Очистить форматирование»

ПРИМЕР.

Очистим форматирование, которое содержится в диапазоне A17:C19. Для этого заполним действие так, как указано на рисунке 2.5.61 и запустим робота.

Исходные данные представлены на рисунке 2.5.62:

16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213
20			

Рис. 2.5.62. Пример исходных данных

Результат работы представлен на рисунке 2.5.63:

16	03.03.2020	73,4178	66,9909
17	29.02.2020	73,7235	65,6097
18	28.02.2020	71,6458	65,5177
19	27.02.2020	71,2439	64,9213
20			

Рис. 2.5.63. Результат выполнения действия «Очистить форматирование»

Действие «Формат ячейки»

Действие меняет формат указанной ячейки на выбранный. По умолчанию, все значения, заносимые в Excel студией, имеют строковый формат, поэтому, для корректного пересчета формул или сортировки, необходимо привести данные к нужному формату.

Действие	Формат ячейки
Лист	Лист2
Ячейки	A2:A6
Тип данных	Дата
Значение	

Рис. 2.5.64. Пример настройки действия «Формат ячейки»

Действие «Формат числа»

Действие отображает число по определенным правилам числового формата Excel.

Действие	Формат числа
Лист	Лист3
Ячейки	A1
Формула	
Значение	\$#,##0.00

Рис. 2.5.65. Пример настройки действия «Формат числа»

Примеры форматов можно посмотреть в программе Excel, кликнув правой кнопкой левой кнопкой мыши по ячейке – «Формат ячейки» – вкладка «Число» - (все форматы):

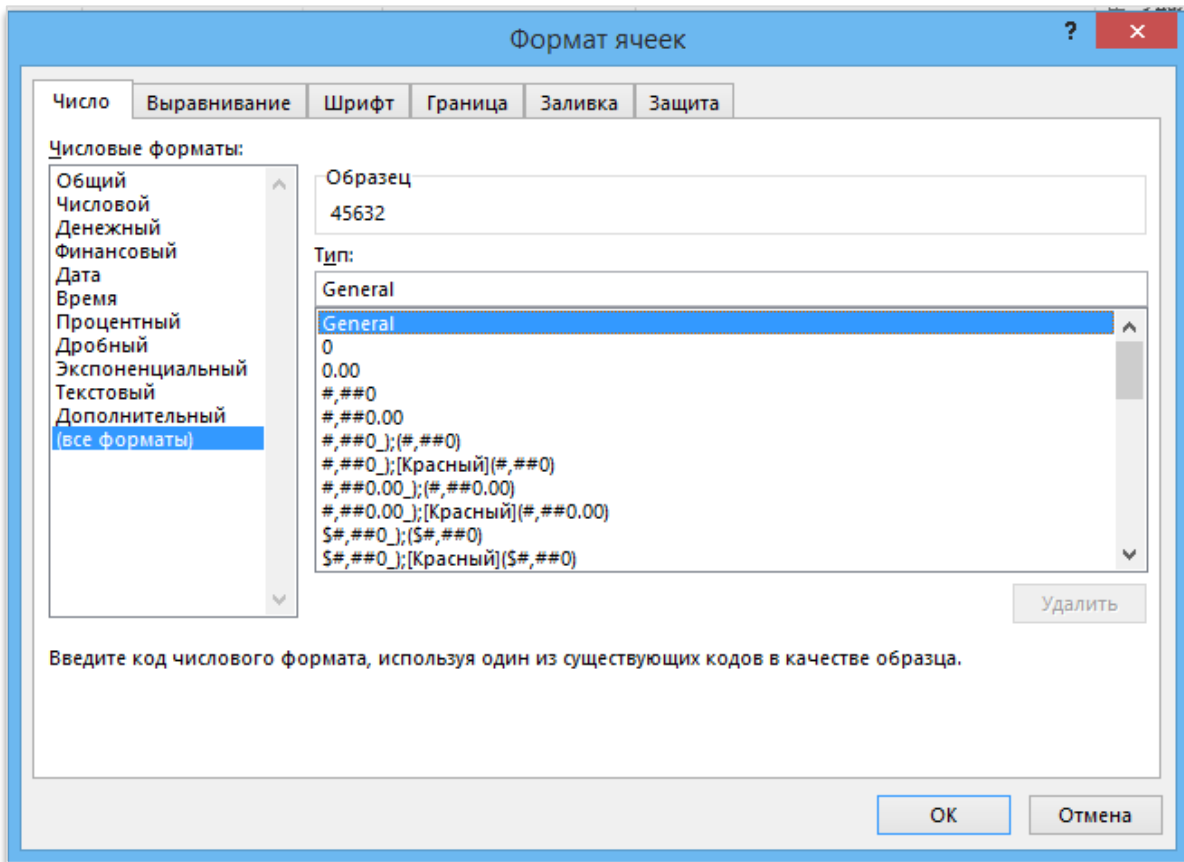


Рис. 2.5.66. Просмотр примеров форматов числа в Excel

Наиболее часто используемые форматы числа предлагаются в студии при наведении на поле для ввода формата.

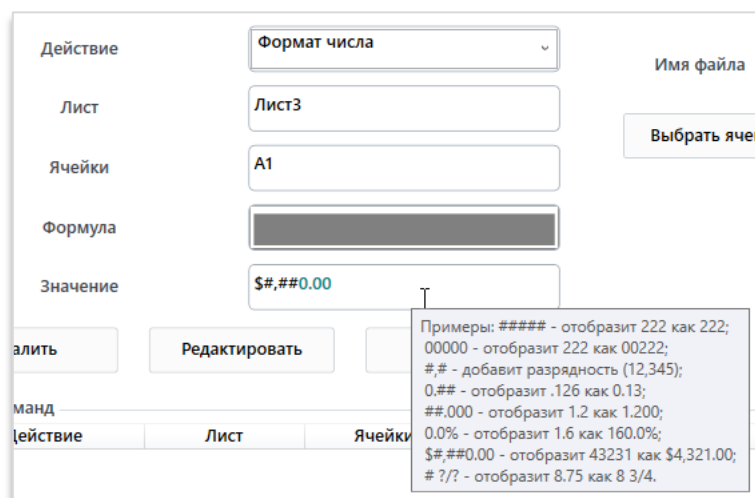


Рис. 2.5.67. Просмотр примеров форматов числа в студии

ПРИМЕР.

Отформатируем число, содержащееся в ячейке A1 в денежный долларový формат. Для этого заполним действие так, как указано на рисунке 2.5.56.

Исходные данные представлены на рисунке 2.5.68:

	A	B
1	123456,00	
2	45632,00	
3	452,37	
4	12,45	
5		

Рис. 2.5.68. Пример исходных данных

Результат работы представлен на рисунке ниже:

	A
1	\$123 456,00
2	45632,00
3	452,37
4	12,45
5	

Рис. 2.5.69. Результат выполнения действия «Формат числа»

Действие «Выравнивание»

Действие применяет заданное выравнивание в указанных ячейках. В поле «Общее положение» предлагается выбрать одно из двух – по вертикали или по горизонтали применять выравнивание, выбранное в поле «Выравнивание содержимого», которое, в свою очередь, предлагает выбрать выравнивание содержимого по левому/правому краю, либо по верхнему/нижнему краю, либо по центру.

Действие	Выравнивание
Лист	Лист1
Ячейки	A1
Общее положение	По вертикали
Выравнивание содержимого	По центру

Рис. 2.5.70. Пример настройки действия «Выравнивание»

ПРИМЕР.

В заполненном на рисунке 2.5.70 действии настраивается выравнивание ячейки A1 по центру относительно вертикали.

Исходное положение текста в ячейке представлено на рисунке 2.5.71:

	A	B	C
1	Дата	Курс евро	Курс доллара
2			
3			

Рис. 2.5.71. Пример исходных данных

Результат работы представлен на рисунке 2.5.72:

	A	B	C
1	Дата	Курс евро	Курс доллара
2			

Рис. 2.5.72. Результат выполнения действия «Выравнивание»

Действие «Шрифт»

Действие меняет шрифт, стиль или размер выбранных ячеек на указанном листе. Модуль предоставляет выбор всех стандартных шрифтов, доступных в пакете Microsoft Office.

Действие	Шрифт	Имя файла	C:\Users\user\Desktop\лок
Лист	Лист1	Выбрать ячейку	Загрузить файл
Ячейки	B1		
Шрифт	Batang		
Стиль текста	Полужирный курсив	Размер	15

Рис. 2.5.73. Пример настройки действия «Изменение шрифта»

ПРИМЕР.

Изменим шрифт текста, находящегося в ячейке В1 так, как указано в действии на рисунке 2.5.73 – сам шрифт изменим на Batang, зададим стиль текста – полужирный курсив, а размер текста 16.

Исходное форматирование текста представлен на рисунке 2.5.74:

	А	В	С
1	Дата	Курс евро	Курс доллара
2			

Рис. 2.5.74. Пример исходных данных

Результат работы представлен на рисунке 2.5.75:

	А	В	С
1	Дата	<i>Курс евро</i>	Курс доллара
2			

Рис. 2.5.75. Результат выполнения действия «Шрифт»

Действие «Границы»

Действие добавляет или изменяет границы выбранных ячеек на указанном листе. Предполагает настройку расположения границ – внешние, внутренние, только правая и др., цвет границ и их стиль – толстая, тонкая, штриховая линия и др.

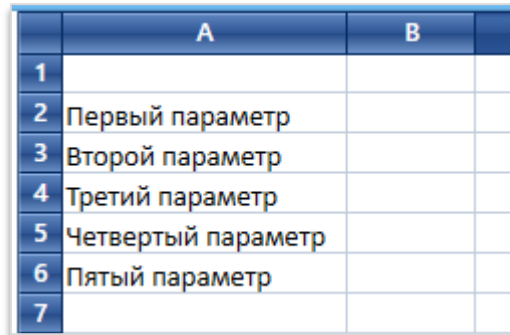
Действие	Границы	Имя файла	C:\Users\user\Desktop\лок
Лист	Лист2	Выбрать ячейку	Загрузить файл
Ячейки	A2:A6		
Расположение	Внешние	Стиль	Штрихпунктир с двумя т
Цвет	255 0 0 0		

Рис. 2.5.76. Пример настройки действия «Границы»

ПРИМЕР.

Добавим штрихпунктирную внешнюю границу для блока текста, находящегося в диапазоне A2:A6. Для этого необходимо заполнить действие так, как указано на рисунке 2.5.76.

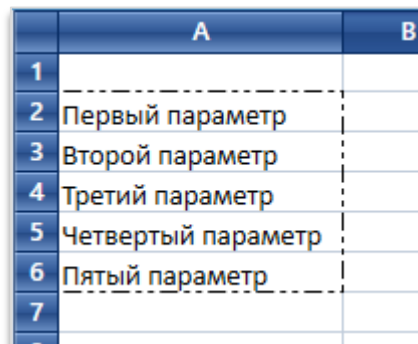
Исходный вид блока текста представлен на рисунке 2.5.77:



	A	B	
1			
2	Первый параметр		
3	Второй параметр		
4	Третий параметр		
5	Четвертый параметр		
6	Пятый параметр		
7			

Рис. 2.5.77. Пример исходных данных

Результат работы представлен на рисунке 2.5.78:



	A	B	
1			
2	Первый параметр		
3	Второй параметр		
4	Третий параметр		
5	Четвертый параметр		
6	Пятый параметр		
7			

Рис. 2.5.78. Результат выполнения действия «Границы»

Глава 6. Модуль «Закрыть Excel-файл»

Excel-модуль работает с файлами следующим образом – как только он начинает работу с некоторым файлом, то он открывает его (в фоновом, незаметном для пользователя, режиме), и держит открытым до конца работы робота. Это было сделано для ускорения работы модуля Excel – если модулей, использующих один и тот же файл, к примеру, несколько, пришлось бы несколько раз открывать и сохранять файл столько раз, сколько он используется что отразилось бы на быстродействии программы. В связи с этим, файл открывается при первом его использовании в модуле Excel и сохраняется только один раз – в самом конце. Но! Так как файл является открытым, с самим файлом нельзя ничего делать – переносить, архивировать, удалять за ненадобностью (например, провели расчеты в новом файле, взяли данные и затем его удалили), так как он занят. Для таких задач существует модуль «Закрыть Excel-файл» - он закрывает и сохраняет заданные ему файлы. Если эти файлы впоследствии снова будут использоваться – они опять откроются и, в следующий раз, будут закрыты либо этим же модулем, либо в конце работы робота.

Интерфейс модуля состоит из одного поля и списка путей к файлам, которые необходимо закрыть.

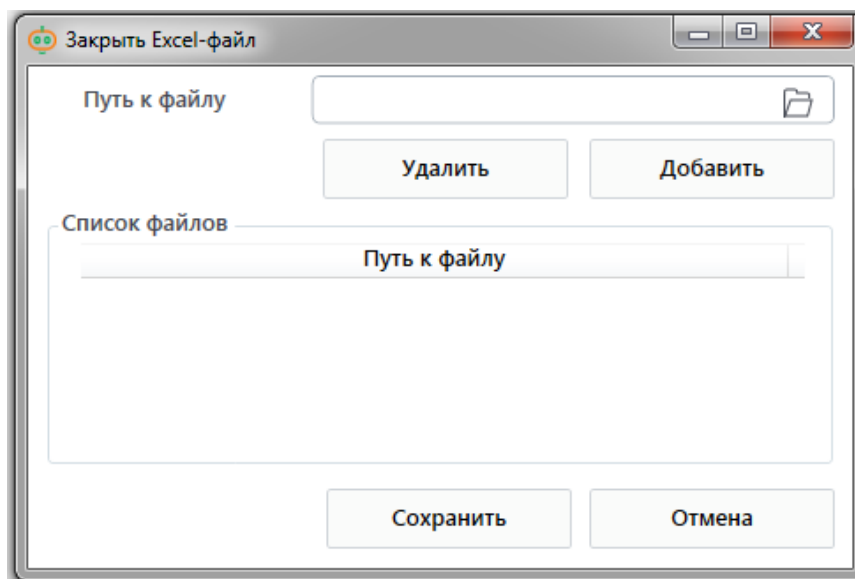


Рис. 2.6.1. Окно «Закрыть Excel-файл»

Поле «Путь к файлу» заполняется путем до файла, который необходимо закрыть. Он может содержаться в переменной, или можно выбрать его вручную, нажав на кнопку с изображением папки, находящуюся в поле для ввода.

По кнопке «Добавить» указанный в поле «Путь к файлу» путь будет занесен в «Список файлов».

По кнопке «Удалить» можно удалить выбранный путь из «Списка файлов».

ЧАСТЬ III. МОДУЛЬ «КЛИКЕР» И МОДУЛИ ПОИСКА ИЗОБРАЖЕНИЙ

Глава 7. Модуль «Кликер»

Модуль «Кликер» предназначен для записи последовательности действий, выполняемых пользователем, а именно кликов по определенным местам экрана и нажатий кнопок на клавиатуре или мыши. С помощью этого модуля можно производить интеграции между самыми разными программами, так как он точно имитирует действия пользователя, то есть, если вы сможете это сделать, то это сможет сделать и «Кликер».

Использование роботов-кликеров рекомендуется на компьютерах с таким же разрешением, как и тот, на котором был написан данный робот.

Создание макроса

При открытии модуля «Кликер» появится диалоговое окно «Запись действий макроса», состоящее из трех элементов – кнопки «Начать запись», области, в которой будут отображены записанные действия, и кнопки «Сохранить».

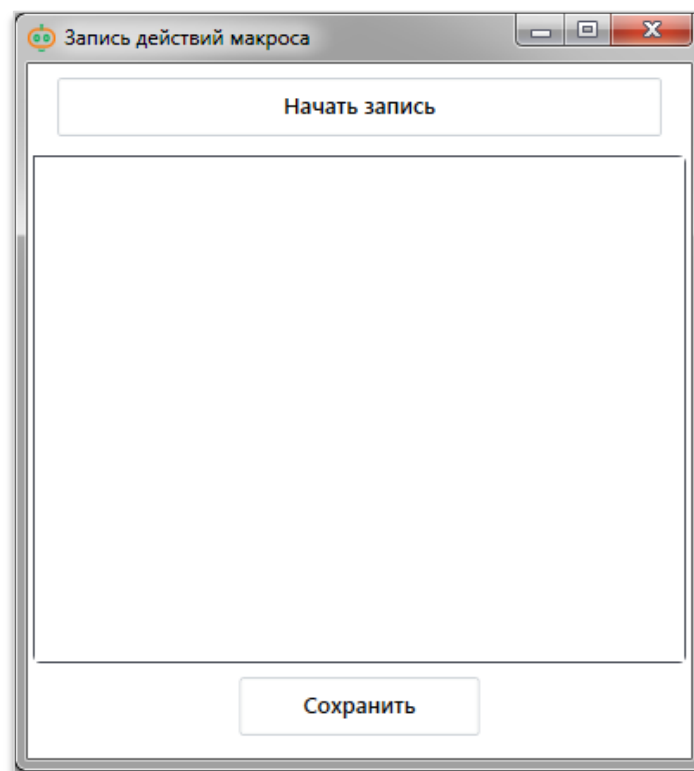


Рис. 3.7.1. Запись действий макроса

По нажатию кнопки «Начать запись» программа переходит в режим *ожидания нажатия кнопки «TAB»*, по которой открывается окно создания действия (команды).

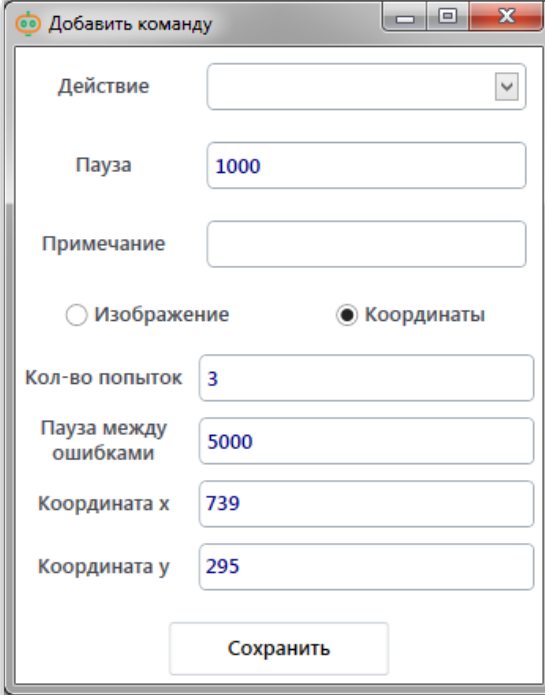


Рис. 3.7.2. Создание команды

В открывшемся окне происходит настройка действия (описание всех возможных действий будет дано ниже). По завершению создания команды и нажатию кнопки сохранить, новое действие будет добавлено в макрос.

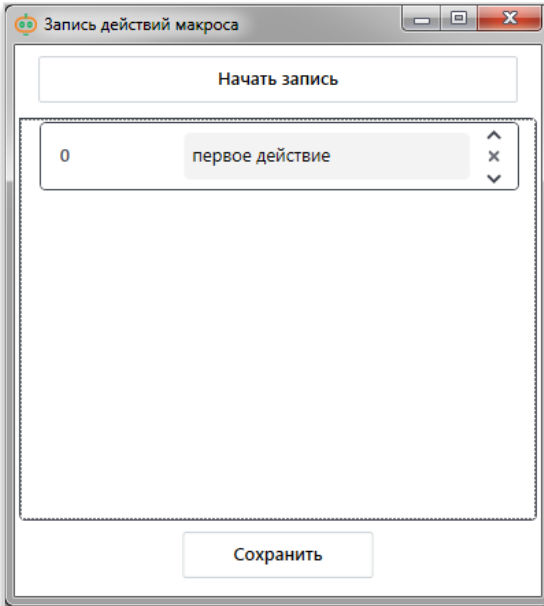


Рис. 3.7.3. Добавление команды

Изменить положение действия в списке команд можно с помощью кнопок с изображением стрелок, расположенных в правой части блока.

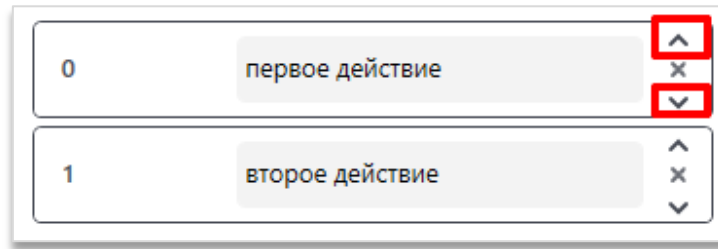


Рис. 3.7.4. Изменение порядка команды

Кнопка с изображением крестика, расположенная между стрелками, предназначена для удаления действия из списка команд.

Для сохранения макроса необходимо нажать на кнопку «Сохранить».

Создание команды

Окно создания команды состоит из статичных и изменяющихся полей. Статичными являются такие поля, как «Действие», «Пауза» и «Примечание».

Статичные поля

Действие

Поле «Действие» состоит из перечня доступных действий для записи макроса – нажатие кнопок мыши и клавиатуры, наиболее часто встречающиеся сочетания кнопок (скопировать, вставить). Далее будут приведены все действия, доступные в модуле.

Пауза

Поле «Пауза» предназначено для создания эффекта ожидания после выполняемой команды. Значение указывается в миллисекундах.

Например, после нажатия на иконку браузера роботу следует подождать несколько секунд, пока браузер запустится и будет готов к работе – в данном примере в команде открытия браузера (кликнуть мышкой по его иконке) в поле «Пауза» следует написать некоторое значение, например, 5000, т.е. 5 секунд.

Примечание

Поле «Примечание» позволяет писать комментарии, чтобы не запутаться в командах.

Поля, появляющиеся только при определенных действиях

Переключатели «Изображение» и «Координаты»

При создании действий, связанных с кликом мыши, вам предлагается выбрать, куда кликать – по определенным координатам или по изображению. Если программа, с которой

вы работаете, статичная и не меняет расположение своих кнопок, то можно воспользоваться кликом по координатам. Тогда возникает вопрос, как рассчитать координаты? Все очень просто – при нажатии кнопки «ТАВ» нужно располагать мышь ровно в том месте, куда следует нажать. В диалоговом окне создания команды координаты уже будут заполнены.

Если программа может менять расположение её элементов, то настоятельно рекомендуем использовать клик по изображению. Для этого нажмите на переключатель «Изображение».

Рис. 3.7.5. Создание команды

При изменении переключателя изменится нижняя часть окна – появятся кнопки «Выбрать изображение», «Выделить изображение» и поле для ввода «Процент совпадения».

Изображение, по которому потребуется кликнуть для выполнения действия, можно настроить двумя способами:

- 1) загрузить имеющееся изображение, нажав на кнопку «Выбрать изображение»
- 2) выбрать фрагмент снимка окна, в котором будет воспроизводиться действие, с помощью кнопки «Выделить изображение». После нажатия на кнопку студия сделает снимок экрана и предложит вам выбрать фрагмент получившегося изображения.

При использовании клика по изображению, **обратите внимание** на то, что, если на экране таких изображений будет несколько, действие будет применено к **первому** найденному!

Поле «Процент совпадения» необходимо для случаев, когда искомое изображение может немного отличаться от того, что может быть в реальности (например, наличие выделения вокруг кнопки, если она выбиралась ранее, и отсутствие такого выделения). В подобных случаях уменьшайте процент совпадения до тех пор, пока изображение не будет находиться и в том, и в другом случае. Однако если процент совпадения уменьшить слишком сильно, модуль может найти то, что не планировали искать.



Рис. 3.7.6. Пример отличия искомого изображения от действительного

Поля «Количество попыток» и «Пауза между ошибками» предназначены для случаев, когда программа не смогла найти заданное ей изображение. Вы можете указать, сколько раз следует пытаться искать требуемое изображение и сколько секунд ждать между попытками. Например, после запуска браузера нужно нажать на кнопку с иконкой «Плюсик» – создание новой вкладки. Если браузер будет загружаться слишком долго, больше 5-ти секунд (это время указано в паузе команды открытия браузера), тогда программа начнёт искать «плюсик», которого еще нет. Затем программа подождет указанное ей количество секунд между попытками и попробует снова. Как только изображение будет найдено, программа подождет то количество секунд, которое указано в поле «Пауза» данной команды, и перейдет к следующей.

Прокрутка мыши

Данное поле появляется при действии «Прокрутка мыши». В него вводится значение в пикселях, на сколько необходимо прокрутить колесико мыши. Если прокрутка вверх, то значение должно быть положительным, если вниз, то отрицательным.

Во всех полях окна создания команды **возможно использование переменных**.

Описание действий, доступных в модуле

Действие «Кликнуть мышкой»

Основное действие модуля – нажатие левой кнопкой мыши по указанным координатам или изображению.

Действие «Кликнуть мышкой дважды»

Двойной клик по указанным координатам или изображению, например, для открытия приложения с рабочего стола.

Действие «Кликнуть правой кнопкой мыши»

Клик правой кнопкой мыши, например, по приложению на рабочем столе, чтобы открыть его контекстное меню.

Действие «Кликнуть колесиком мыши»

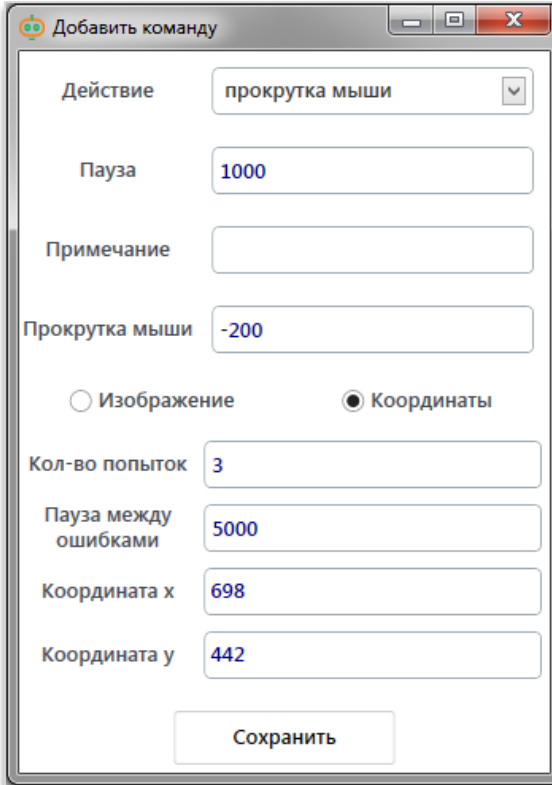
В некоторых программах есть возможность использовать клик колесика мыши для исполнения некоторых действий, например, закрыть вкладку в браузере.

Действие «Прокрутка мыши»

Прокрутка колесом мыши вниз или вверх по странице. Здесь координаты играют роль того, где будет находиться курсор при прокрутке колеса, так как ползунков на экране может быть несколько. Для прокрутки экрана вниз в поле «Прокрутка мыши» должно быть введено отрицательное значение (-150), вверх - положительное (150).

ПРИМЕР.

Для прокрутки вниз конкретного ползунка необходимо навести на него курсор, нажать клавишу «ТАВ» на клавиатуре, выбрать действие «прокрутка мыши» и заполнить поле «Прокрутка мыши», как показано на рисунке 3.7.7.:



Действие	прокрутка мыши
Пауза	1000
Примечание	
Прокрутка мыши	-200
<input type="radio"/> Изображение	<input checked="" type="radio"/> Координаты
Кол-во попыток	3
Пауза между ошибками	5000
Координата x	698
Координата y	442
<input type="button" value="Сохранить"/>	

Рис. 3.7.7. Пример настройки команды

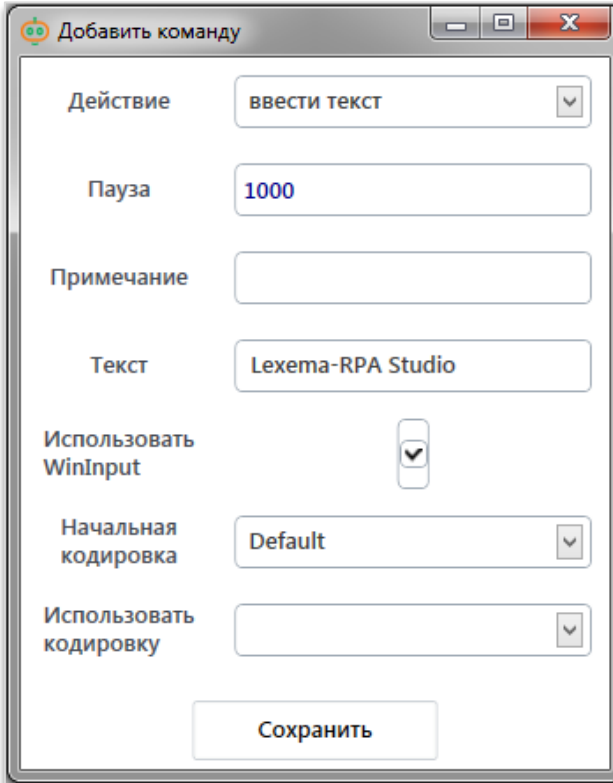
Действие «Ввести текст»

Данное действие вводит текст, который записан в поле «Текст». Поле, куда вводится текст, должно быть выделено (должен быть фокус, стоять каретка).

В некоторых программах необходим специфичный ввод текста – в конкретной кодировке. В подобных случаях следует выбрать кодировку из предложенных в выпадающем списке «Использовать кодировку», или указать свою. Текст может лежать в переменной и быть скопированным из какой-либо программы, использующей конкретную кодировку. В таком случае, при вставке скопированного текста в другую программу, необходимо сначала считать его в кодировке исходной программы. Кодировка исходной программы вводится или выбирается в поле «Начальная кодировка». Заполнение этих полей **необязательно**.

Флажок «Использовать WinInput» позволяет включить режим более корректного и быстрого ввода текста, но он может не поддерживаться некоторыми программами. Его использование **необязательно**.

Пример заполнения действия:

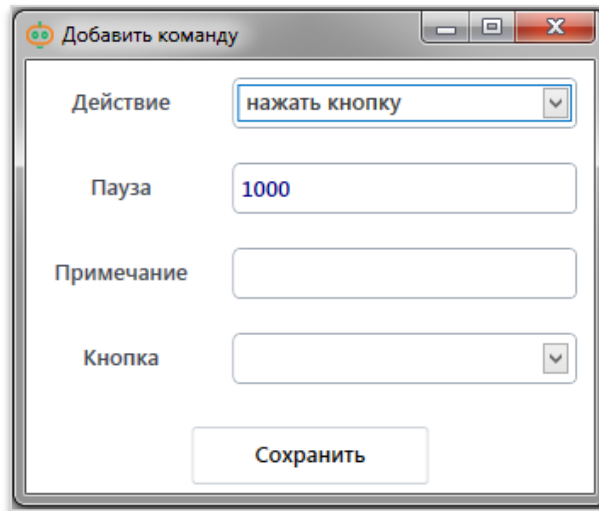


Действие	ввести текст
Пауза	1000
Примечание	
Текст	Lexema-RPA Studio
Использовать WinInput	<input checked="" type="checkbox"/>
Начальная кодировка	Default
Использовать кодировку	
Сохранить	

Рис. 3.7.8. Пример заполнения

Действие «Нажать кнопку»

Имитирует нажатие клавиши клавиатуры. Необходимую клавишу следует выбрать в выпадающем списке «Кнопка». В списке предложены все возможные клавиши, которые могут присутствовать как на простых, так и на расширенных клавиатурах (изменение громкости звука, запуск/остановка музыки и другие нестандартные кнопки).



The screenshot shows a dialog box titled "Добавить команду" (Add command). It contains the following fields:

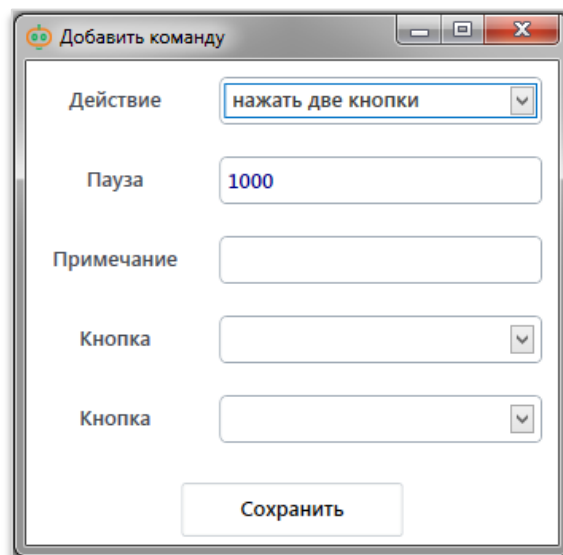
- Действие** (Action): A dropdown menu with "нажать кнопку" (press button) selected.
- Пауза** (Pause): A text input field containing "1000".
- Примечание** (Note): An empty text input field.
- Кнопка** (Button): A dropdown menu that is currently empty.

At the bottom of the dialog is a "Сохранить" (Save) button.

Рис. 3.7.9. Действие «Нажать кнопку»

Действие «Нажать две кнопки»

Имитирует нажатие сочетания двух клавиш клавиатуры, например CTRL+A – выделение всего текста в поле. Необходимые кнопки выбираются в двух выпадающих списках «Кнопка». Список кнопок тот же, что и в действии «Нажать кнопку».



The screenshot shows a dialog box titled "Добавить команду" (Add command). It contains the following fields:

- Действие** (Action): A dropdown menu with "нажать две кнопки" (press two buttons) selected.
- Пауза** (Pause): A text input field containing "1000".
- Примечание** (Note): An empty text input field.
- Кнопка** (Button): Two empty dropdown menus for selecting two different keys.

At the bottom of the dialog is a "Сохранить" (Save) button.

Рис. 3.7.10. Действие «Нажать две кнопки»

Действие «Скопировать»

Имитирует нажатие сочетания клавиш клавиатуры CTRL+C. Копирует выделенное заранее значение в буфер обмена и записывает его, при необходимости, в переменную, указанную в поле «Записать в». Данное поле **не обязательно** к заполнению. Ввод переменной начинается с символа «@».

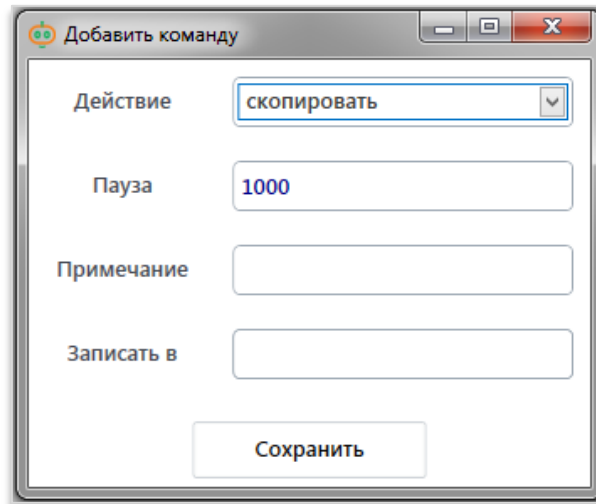


Рис. 3.7.11. Действие «Скопировать»

Действие «Вставить»

Имитирует нажатие сочетания клавиш клавиатуры CTRL+V. Вставляет в выбранное поле значение из буфера обмена.

Глава 8. Модуль «Поиск изображения»

Модуль «Поиск изображения», в основном, используется в связке с модулем «Кликер». Его задача – определить, существует ли на текущем экране заданное изображение, и если да, то вернуть координаты его центра.

Интерфейс модуля

Окно модуля выглядит следующим образом:

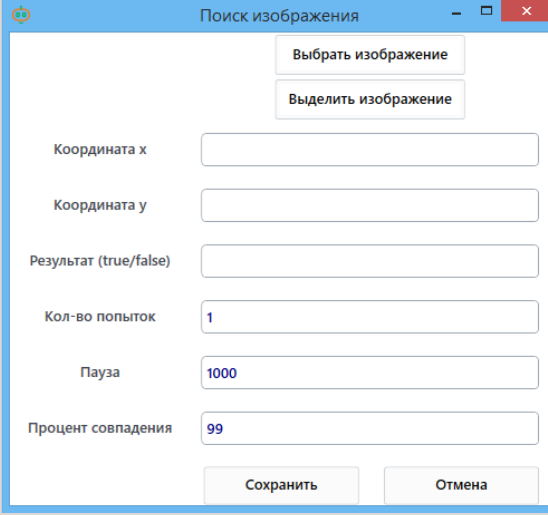


Рис. 3.8.1. Окно модуля

Кнопки «Выбрать изображение» и «Выделить изображение» позволяют выбрать изображение для поиска.

Поле «Координата x». В поле вводится *переменная* (начиная с символа «@»), в которую будет записана координата x искомого изображения, если оно найдется.

Поле «Координата y». Аналогично полю «Координата x», в данном поле указывается переменная, в которую будет записана координата y.

Поле «Результат». В поле вводится название переменной, в которой будет итоговый результат поиска изображения (после использования всех попыток его найти). Результирующее значение будет представлено в виде строки: «true» - найдено, «false» - не найдено.

Поля «Кол-во попыток», «Пауза» и «Процент совпадения» аналогичны полям модуля «Кликер».

Поле «Кол-во попыток». В поле указывается количество попыток поиска изображения.

Поле «Пауза» предназначено для указания паузы между попытками поиска в миллисекундах.

Поле «Процент совпадения» - насколько сильно совпадает изображение, которое будет искоматься, с фрагментом изображения на текущем экране. Вводится число от 0 до 100, чем больше, тем больше должно быть сходства между изображениями.

Глава 9. Модуль «Поиск изображений»

Модуль «Поиск изображений» аналогичен модулю «Поиск изображения» с одним лишь отличием в том, что этот модуль возвращает список всех найденных изображений, а не одного.

Интерфейс модуля

Интерфейс этого модуля очень похож на интерфейс модуля «Поиск изображения», но в нем вместо полей «Координата X», «Координата Y» и «Результат» только одно поле – «Переменная», а также появилось поле «Минимальная дистанция».

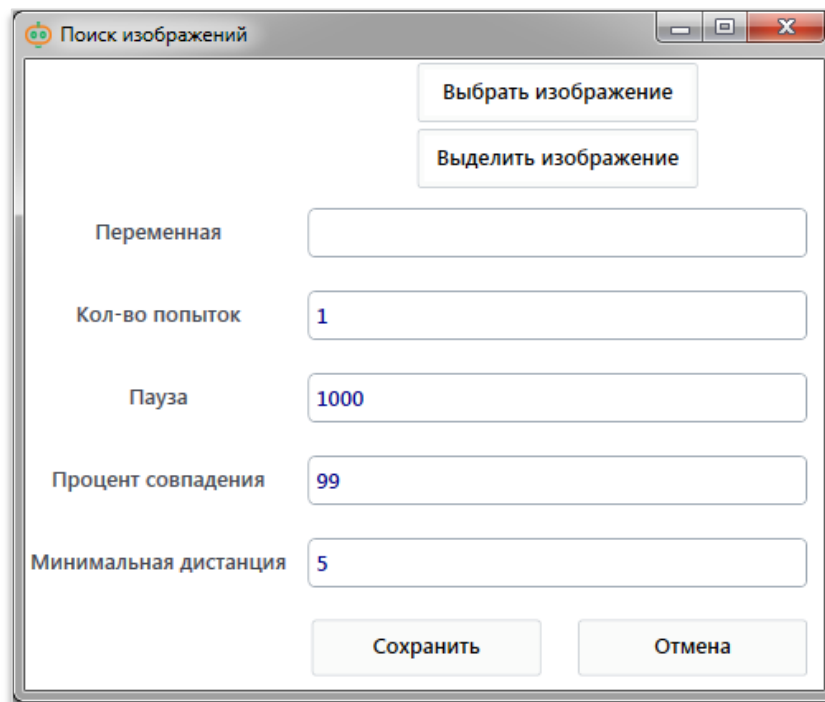


Рис. 3.9.1. Окно модуля

В поле «Переменная» вводится название переменной с символа «@», в которую запишется результат поиска.

Поле «Кол-во попыток» заполняется числом, указывающим, сколько раз пробовать искать изображения, если с первого раза ни одного найти не удалось.

Поле «Пауза» предназначено для указания паузы между попытками поиска в миллисекундах.

В поле «Процент совпадения» вводится значение от 0 до 100, указывающее, насколько сильно совпадают изображение, которое будет искаться, с фрагментом изображения на текущем экране. Чем больше число, тем сильнее они должны совпадать.

Поле «Минимальная дистанция» заполняется числом, указывающим, какое минимальное количество пикселей должно быть между двумя одинаковыми изображениями.

В некоторых задачах может быть потребоваться отсекаать изображения, которые расположены достаточно близко друг к другу и получать только первое из этих изображений (начиная с верхнего левого угла). **Обратите внимание**, если это расстояние поставить совсем маленьким (не менее 5 пикселей) или равным 0, то одно изображение может быть найдено несколько раз!

Выходные данные

В указанную переменную будет записан список объектов, количество которых будет равно количеству найденных изображений на экране. Каждый объект соответствует найденному изображению. Для обращения к конкретному изображению списка необходимо указывать его номер (нумерация с 0) в квадратных скобках после названия переменной, например @listImages[0].

Каждый объект содержит три поля: «x», «y» и «similarity». В поле «x» содержится координата x выбранного изображения, в поле «y» - координата y, а в поле «similarity» - процент совпадения найденного изображения с искомым.

ПРИМЕР.

Пусть мы хотим с помощью робота найти количество одинаковых изображений, например, непочитанных писем с Яндекс-почты, которые выделены небольшим синим кружочком около письма:

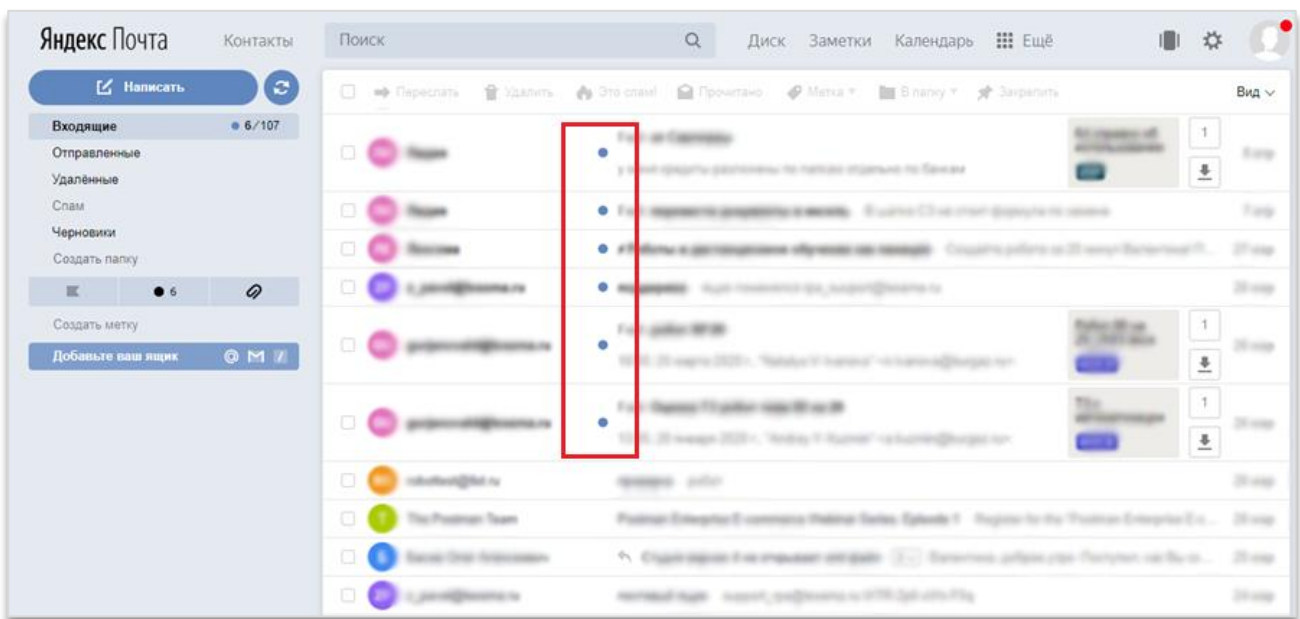


Рис. 3.9.2. Пример

Для этого создадим действие «Поиск изображений» и выберем в нем в качестве искомого изображения эту синенькую точку:

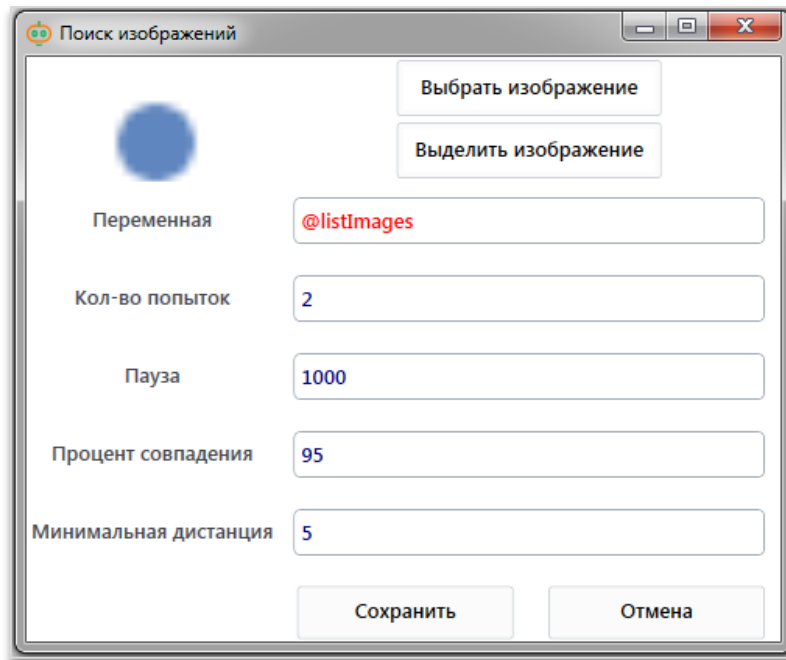


Рис. 3.9.3. Настройка действия

Количество найденных изображений, а значит непрочитанных писем, будет `@listImages[%]`.

Для нажатия на каждую такую точку и прочтения сообщений, необходимо добавить цикл по количеству изображений, и в каждом шаге цикла нажимать на координаты `@listImages[@i].x` и `@listImages[@i].y`, где `@i` – текущий шаг цикла.

ЧАСТЬ IV. МОДУЛЬ «РАБОТА С ФАЙЛАМИ»

Глава 10. Основные моменты модуля

Модуль «Работа с файлами» предназначен для изменения расположения файлов, их копирования, архивирования, создания папки и переноса файлов в неё, в общем, для манипуляции с файлами и директориями.

Интерфейс модуля

Окно модуля объединяет три группы элементов.

Первая группа элементов – это набор полей и кнопок для создания команды. Вторая – непосредственно список команд. И третья – это область предварительного просмотра текстовых файлов.

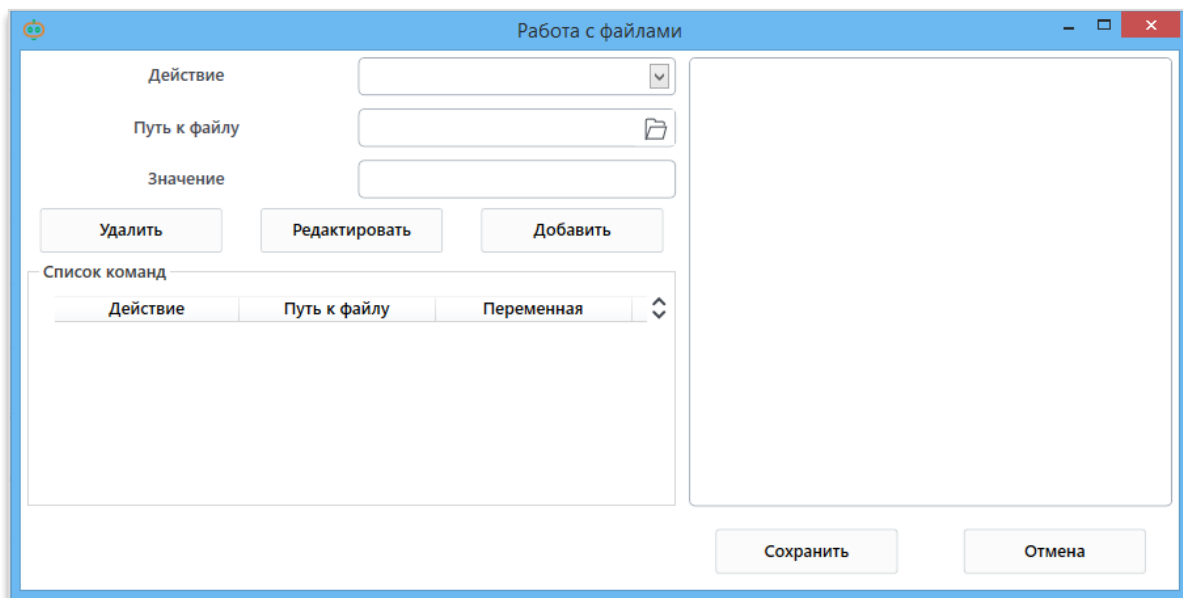


Рис. 4.10.1. Окно модуля

Набор полей для заполнения меняется в зависимости от выбранного действия, подробное описание всех полей будет дано в главе 11.

Функциональность модуля

Создание команды

Команда создается следующим образом – выбирается необходимое действие, заполняются остальные поля – нажимается кнопка «Добавить», по которой команда добавляется в «Список команд».

Редактирование команды

Для того чтобы отредактировать команду, выделите её в списке команд щелчком левой кнопкой мыши, после чего все поля в левой верхней части окна «Работа с файлами» заполнятся в соответствии с выбранной командой. Измените те поля, которые требуется отредактировать, и, убедившись, что в списке команд все еще выбрано нужное действие, нажмите на кнопку «Редактировать».

Удаление команды

Для удаления команды необходимо выбрать её в списке команд и нажать на кнопку «Удалить».

Отключение/включение команды

Созданные команды можно отключать и обратно включать. Отключенные команды не будут выполняться. Сделать это можно, нажав на соответствующую отключаемой команде строку правой кнопкой мыши и выбрав «Отключить/включить».

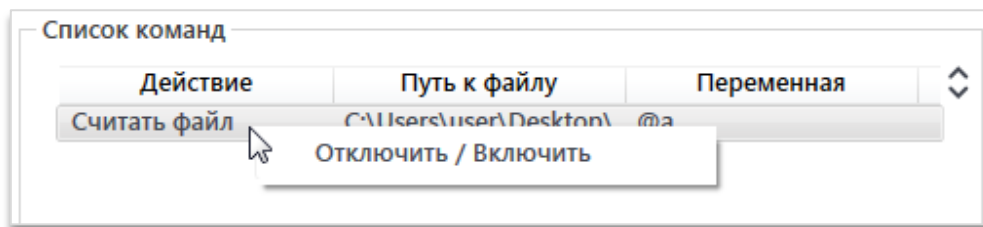



Рис. 4.10.2. Контекстное меню команды

Выбор файла/папки

Кнопка  в заполняемых полях открывает стандартный диалог выбора файла или папки, в зависимости от выбранного действия.

Глава 11. Описание действий, доступных в модуле

Модуль содержит 14 действий, позволяющих работать с папками и файлами, каждое из которых состоит из своего набора полей для настройки. Ниже описаны все действия модуля.

Перечень и описание действий

Действие «Сохранить в файл»

Действие перезаписывает существующий текстовый файл, записывая в него строку, указанную в поле «Значение».

Действие	Сохранить в файл
Путь к файлу	C:\Users\user\Desktop\локализац
Значение	Тестовый файл

Рис. 4.11.1. Пример настройки действия «Сохранить в файл»

Действие «Считать файл»

Действие считывает весь текст из выбранного файла и записывает его в указанную переменную.

Действие	Считать файл
Путь к файлу	C:\Users\user\Desktop\локализац
Переменная	@text

Рис. 4.11.2. Пример настройки действия «Считать файл»

Действие «Получить список файлов в папке»

Действие создает переменную-список, в которой лежат пути до каждого файла в папке.

Действие	Получить список файлов в папке
Путь к папке	C:\Users\user\Desktop\робот
Значение	@files

Рис. 4.11.3. Пример настройки действия «Получение списка файлов»

ПРИМЕР.

Пусть имеется папка «Робот», в которой находится 3 Excel-файла. Тогда в модуле Excel мы можем работать с каждым из файлов, например, чтобы работать с первым, нужно записывать в поле «Имя файла» @files[0].

Переменная после выполнения действия будет выглядеть как на рисунке 4.11.4:

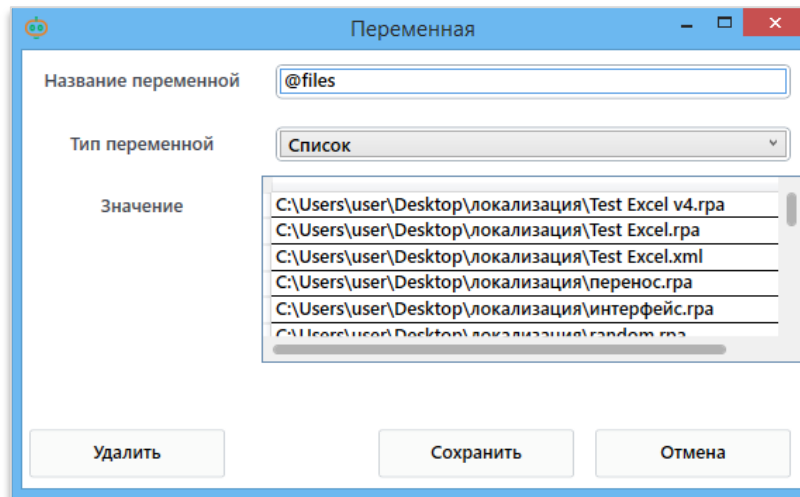


Рис. 4.11.4. Пример переменной, в которой содержится список файлов

Действие «Переместить файл»

Действие перемещает выбранный файл в указанную папку.

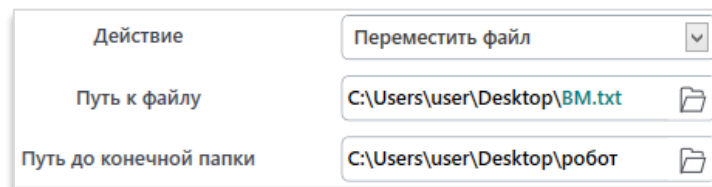


Рис. 4.11.5. Пример настройки действия «Перемещение файла»

Действие «Переместить папку»

Действие перемещает папку, указанную во втором поле, в директорию, указанную в третьем поле.

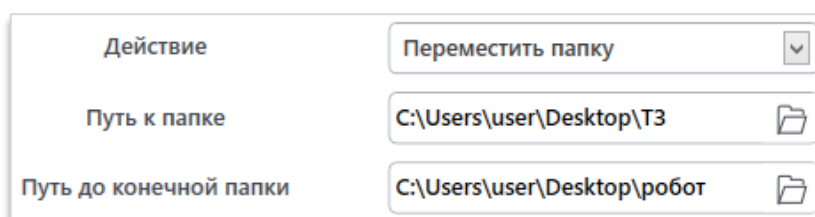


Рис. 4.11.6. Пример настройки действия «Перемещение папки»

Действие «Заархивировать папку»

Действие создает ZIP-архив с указанным именем, добавляя в него содержимое выбранной папки.

Действие	Заархивировать папку
Путь к папке	C:\Users\user\Desktop\робот
Имя архива	Robot

Рис. 4.11.7. Пример настройки действия «Заархивировать папку»

Действие «Распаковать архив»

Действие распаковывает ZIP-архив в указанную папку.

Действие	Распаковать архив
Путь к файлу	C:\Users\user\Desktop\робот3.zip
Путь до конечной папки	C:\Users\user\Desktop

Рис. 4.11.8. Пример настройки действия «Распаковать архив»

Действие «Создать папку»

Действие создает папку с указанным названием в выбранной директории.

Действие	Создать папку
Путь к папке	C:\Users\user\Desktop
Имя папки	Роботы

Рис. 4.11.9. Пример настройки действия «Создание папки»

Действие «Удалить папку»

Действие удаляет выбранную папку.

Действие	Удалить папку
Путь к папке	C:\Users\user\Desktop\робот3

Рис. 4.11.10. Пример настройки действия «Удалить папку»

Действие «Создать файл txt»

Действие создает новый файл с расширением .txt.

Действие	Создать файл txt
Путь к папке	C:\Users\user\Desktop
Имя файла	Тест

Рис. 4.11.11. Пример настройки действия «Создать файл txt»

Действие «Удалить файл»

Действие удаляет выбранный файл.

Действие	Удалить файл
Путь к файлу	C:\Users\user\Desktop\sendTo1C.s

Рис. 4.11.12. Пример настройки действия «Удалить файл»

Действие «Скопировать файл»

Действие копирует выбранный файл в указанную директорию.

В поле «Путь до конечного файла» необходимо ввести путь до папки, в которую файл должен скопироваться, и его название с расширением так, как будто файл уже есть. Новое название может отличаться от исходного.

Действие	Скопировать файл
Путь к файлу	C:\Users\user\Desktop\sendTo1C.sql
Путь до конечного файла	C:\Users\user\Desktop\робот\sendTo1C.sql

Рис. 4.11.13. Пример настройки действия «Скопировать файл»

Действие «Проверить существование файла»

Действие проверяет существование указанного файла и записывает результат в переменную.

«True» - если существует, и иначе «False».

Действие	Проверить существование файла
Путь к файлу	C:\Users\user\Desktop\sendTo1C.sql
Переменная	@result

Рис. 4.11.14. Пример настройки действия «Проверить существование файла»

Действие «Проверить существование папки»

Действие проверяет существование указанной папки и записывает результат в переменную.

«True» - если существует, и иначе «False».

Действие	Проверить существование папки
Путь к папке	C:\Users\user\Desktop\projects
Переменная	@result

Рис. 4.11.15. Пример настройки действия «Проверить существование папки»

ЧАСТЬ V. МОДУЛИ «ЦИКЛ», «УСЛОВИЕ», «WHILE», «BREAK» И «RETURN»

Глава 12. Модуль «Условие»

Описываемые в этой части модули сами по себе ничего не делают, их следует использовать вместе с любыми другими. У описываемых модулей (кроме модуля «Break»), помимо привычного «Примечания», есть еще несколько параметров, обязательных для заполнения, но нет отдельного окна с настройками.

Модуль «Условие» позволяет в зависимости от указанного условия выполнять те или иные действия. Условием может быть любое выражение, которое приводится к значению «True» или «False».

Интерфейс модуля

Модуль состоит из полей «Примечание» и «Условие». Действия, которые должны выполняться при значении условия «True», добавляются в раздел «THEN», при значении «False» - в раздел «ELSE». Действия добавляются в модуль перетаскиванием их из меню или другого места рабочей области.

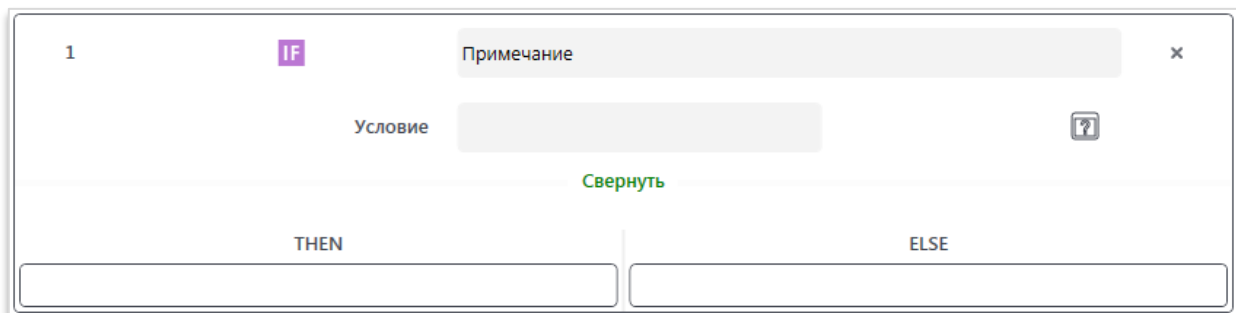



Рис. 5.12.1. Условный модуль

Поле «Примечание» предназначено для внесения комментария от разработчика робота.

Поле «Условие» заполняется по правилам языка программирования C#. Для пользователей, незнакомых с этим языком, предлагается помощь в виде «Конструктора условий», доступная по кнопке справа от условия - .

Конструктор условий

Конструктор условий позволяет составлять условия в соответствии с синтаксисом языка C#.

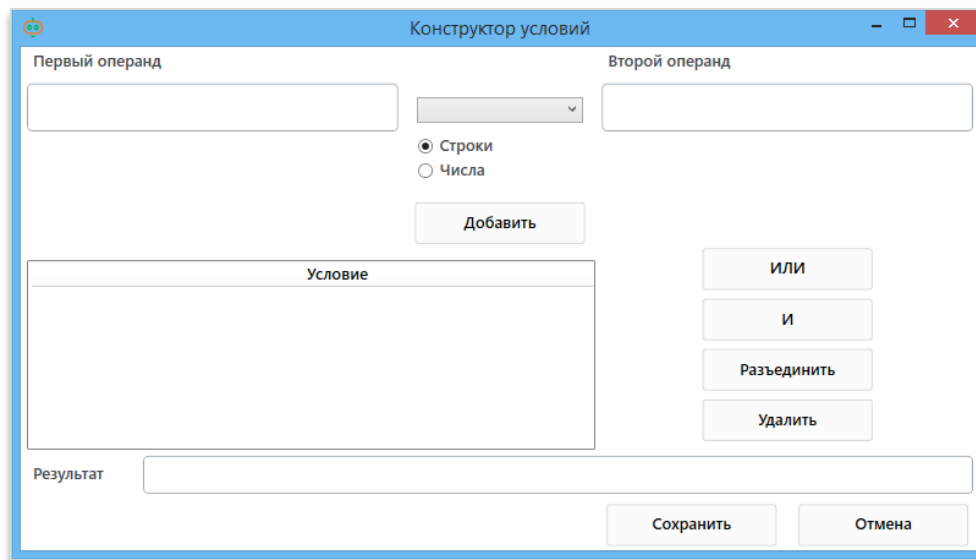


Рис. 5.12.2. Конструктор условий

Добавление

Для того чтобы составить условие, необходимо заполнить оба операнда, выбрать тип операндов (строки или числа), выбрать оператор между ними из предложенного и нажать на кнопку «Добавить»:

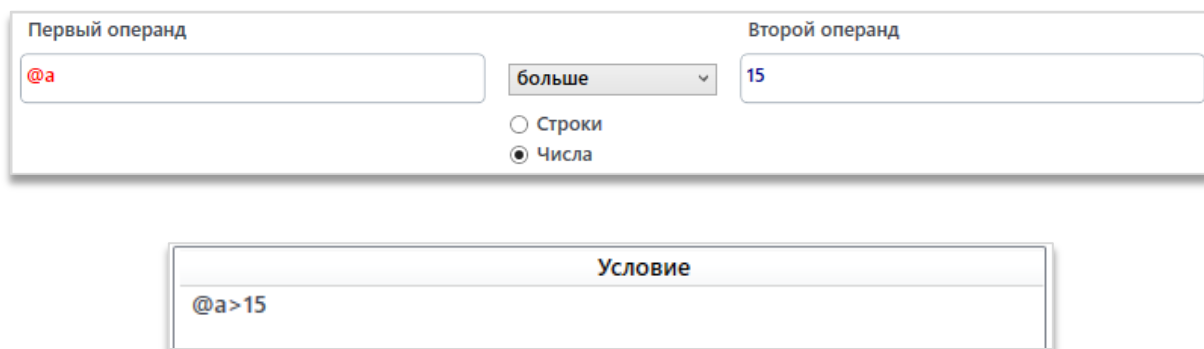


Рис. 5.12.3. Добавление условий

Список операций для строк и для чисел отличается, поэтому сначала выберите тип операндов, а лишь затем операцию.

Связывание

Несколько условий можно связать с помощью логического «И» либо логического «ИЛИ».

Делается это следующим образом – удерживая клавишу Shift, выбираются несколько условий, которые должны быть соединены одним оператором, затем нажимается кнопка с соответствующим логическим оператором:

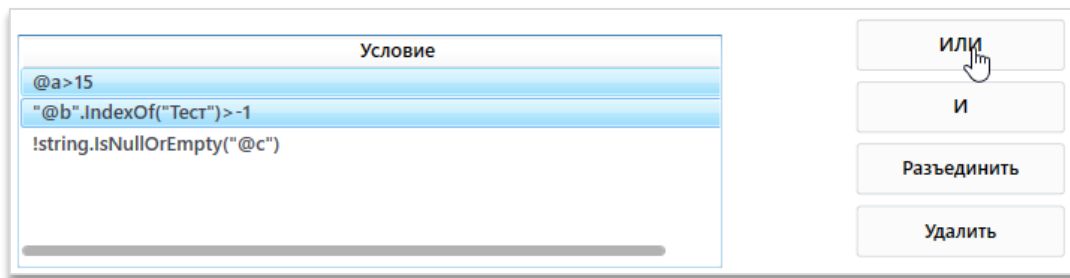


Рис. 5.12.4. Связывание условий

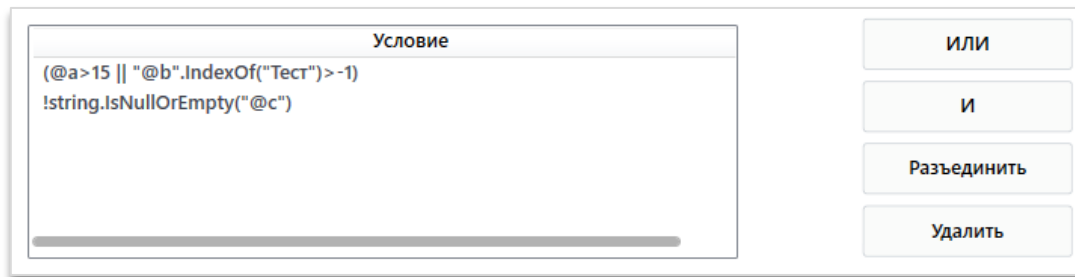


Рис. 5.12.5. Результат связывания условий

Символ «||» означает логическое ИЛИ, символ «&&» - логическое И.

Разъединение

Условия, состоящие из нескольких простых условий, то есть те, которые используют логическое И или ИЛИ, можно разъединить, нажав на соответствующую кнопку. Разъединение происходит по логическим операторам.

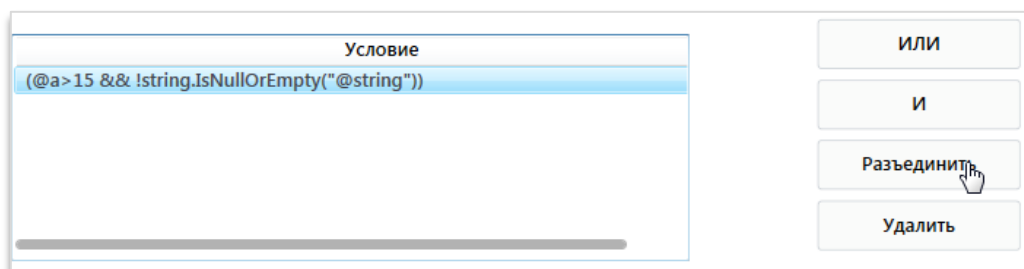


Рис. 5.12.6. Разъединение условий

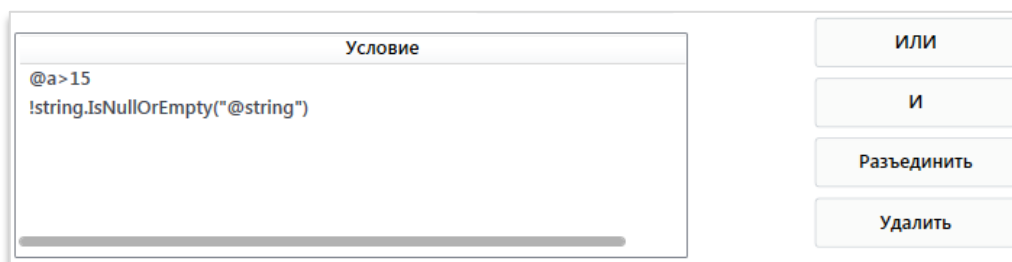


Рис. 5.12.7. Результат разъединения условий

Сохранение

Для сохранения условия необходимо нажать на кнопку «Сохранить». Получившиеся условие вставится в поле «Условие» условного модуля:

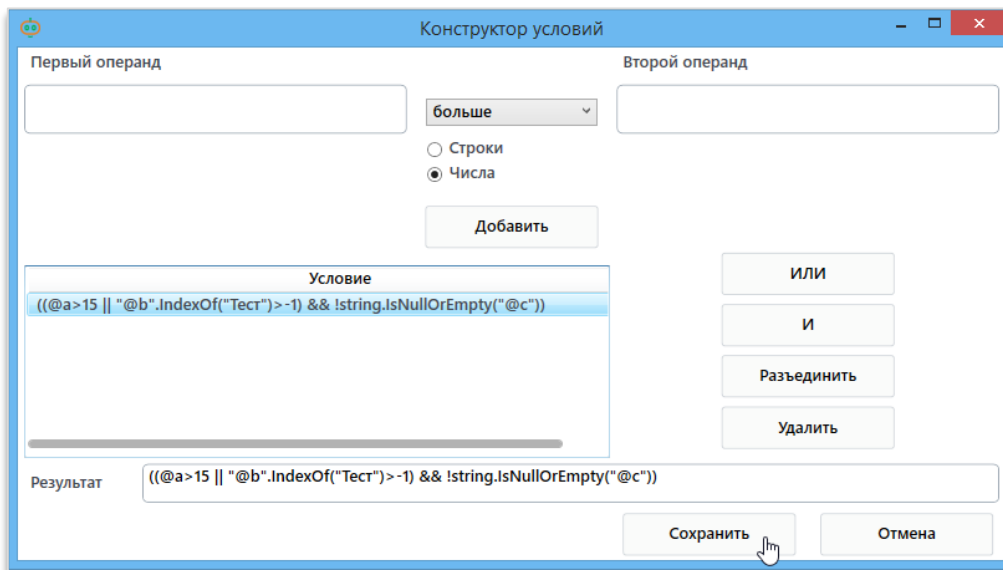


Рис. 5.12.8. Сохранение условия

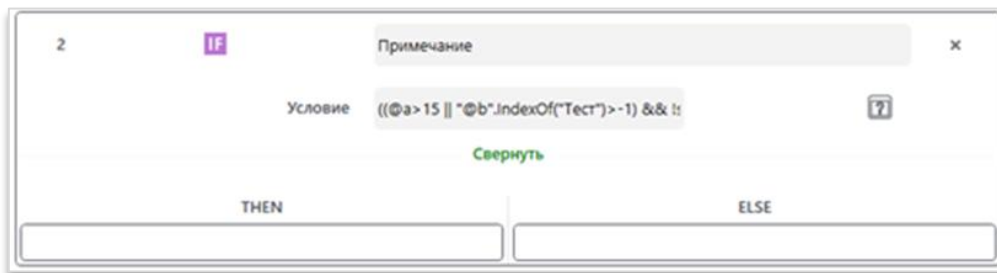


Рис. 5.12.9. Отображение условия

Законченный модуль может выглядеть следующим образом:

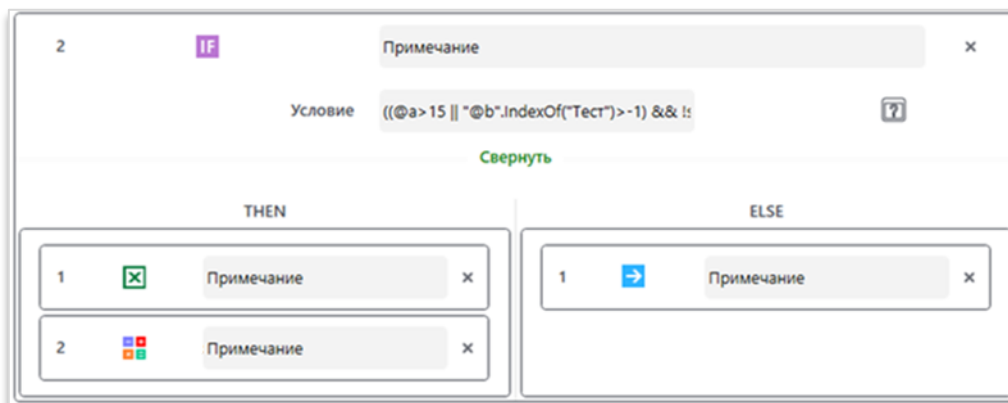


Рис. 5.12.10. Отображение условия

Сворачивание

Для уменьшения занимаемого места в рабочей области проекта, можно свернуть входящие в данный модуль действия по кнопке «Свернуть».

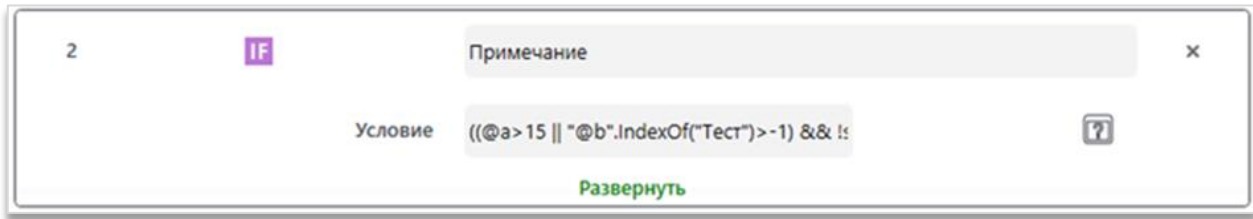


Рис. 5.12.11. Свернутый вид модуля

Для удобства использования также можно менять размеры веток «THEN» и «ELSE» вплоть до полного сворачивания одной из них, если та, к примеру, не используется.

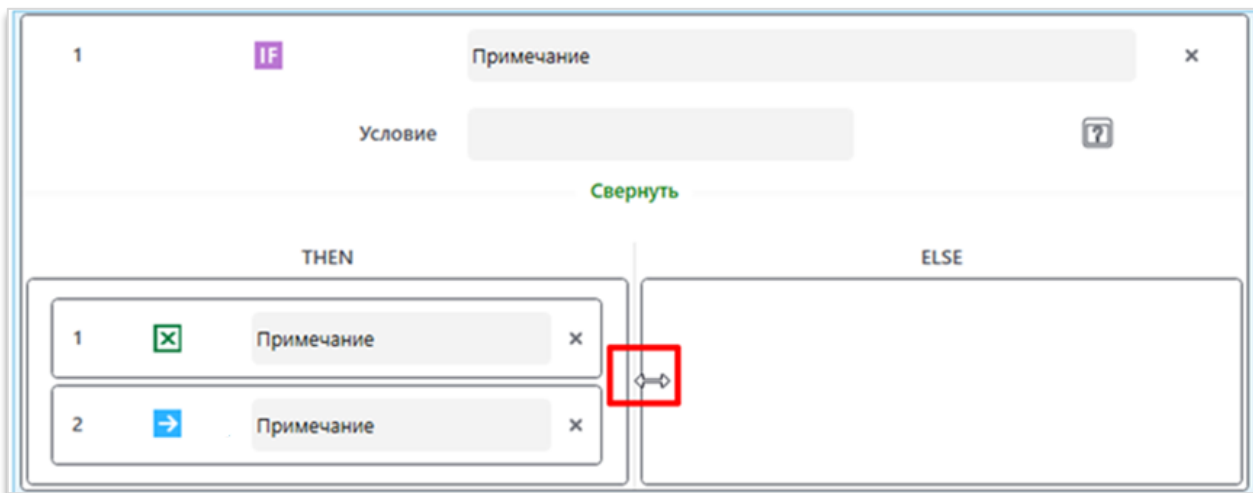


Рис. 5.12.12. Изменение размера веток

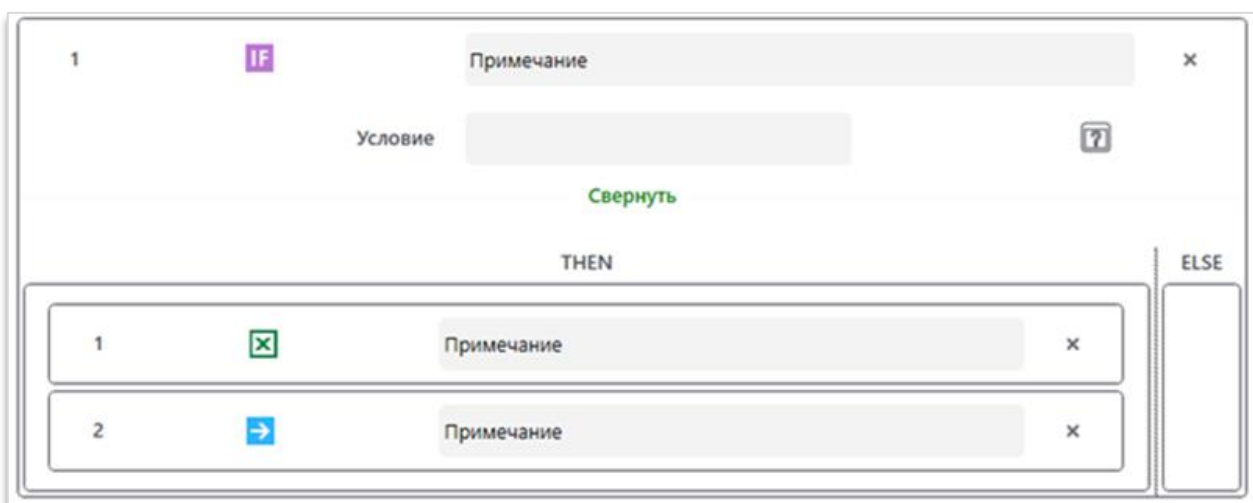


Рис. 5.12.13. Развернутый вид модуля

Глава 13. Модуль «Цикл»

Модуль «Цикл» применяется для неоднократного (циклического) выполнения набора действий. Действия, добавленные в модуль, будут выполняться до тех пор, пока не пройдет указанное в настройках количество итераций.

Интерфейс модуля

Модуль состоит из полей «Примечание», «Количество итераций» и «Переменная». Действия, которые должны выполняться в цикле, добавляются в блок, расположенный в нижней части модуля. Действия добавляются в модуль перетаскиванием их из меню или другого места рабочей области.

Модуль выглядит следующим образом:

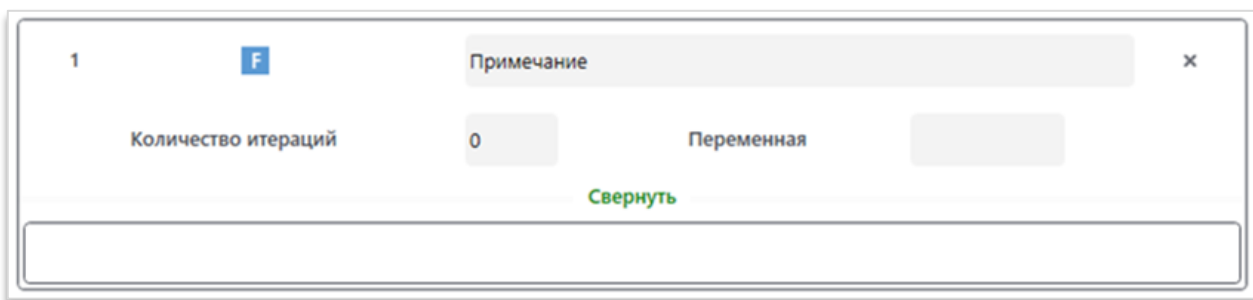


Рис. 5.13.1. Окно модуля

Поле «Примечание», как и во всех других модулях, служит для внесения комментария от разработчика робота.

Остальные поля присущи только этому модулю.

Поле «Количество итераций» предназначено для ввода числа, означающего, сколько раз необходимо выполнить действия, содержащиеся в модуле.

Поле «Переменная» - переменная, в которой будет храниться число – текущий шаг цикла. Цикл начинается с единицы.

Пример заполненного цикла представлен на рисунке 5.13.2.

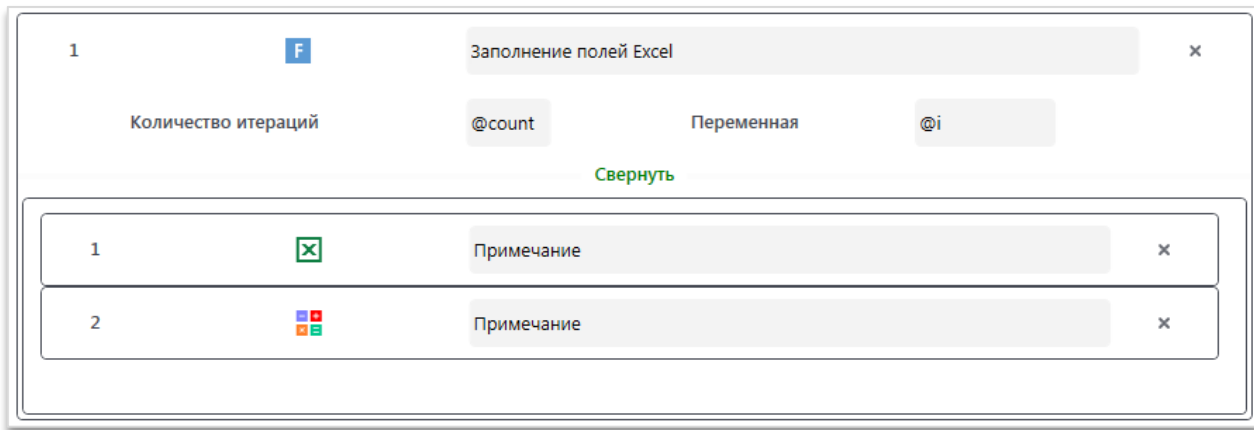


Рис. 5.13.2. Пример заполнения полей в модуле

По кнопке «Свернуть» можно свернуть действия цикла для более компактного отображения команды.

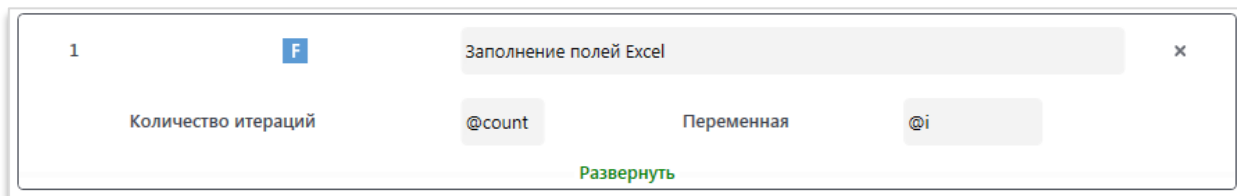


Рис. 5.13.3. Свернутый вид модуля

Глава 14. Модуль «While»

Модуль «While» (цикл с условием) выполняет добавленные в него действия до тех пор, пока указанное условие возвращает значение «True».

Интерфейс модуля

В окне модуля расположены поля «Примечание» и «Условие». В блоке, расположенном в нижней части модуля, следует располагать действия, которые требуется выполнять в цикле.

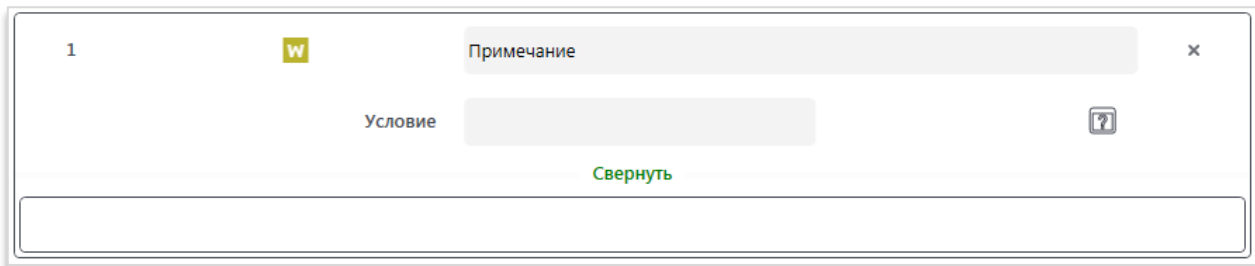


Рис. 5.14.1. Окно модуля

Поле «Условие». В поле вводится условие, как и в модуле «Условие», которое необходимо писать в соответствии с синтаксисом языка C#, поэтому здесь присутствует кнопка, открывающая «Конструктор условий» (см. главу 11).

Добавление действий в модуль происходит с помощью перетаскивания из меню модулей или из рабочей области проекта.

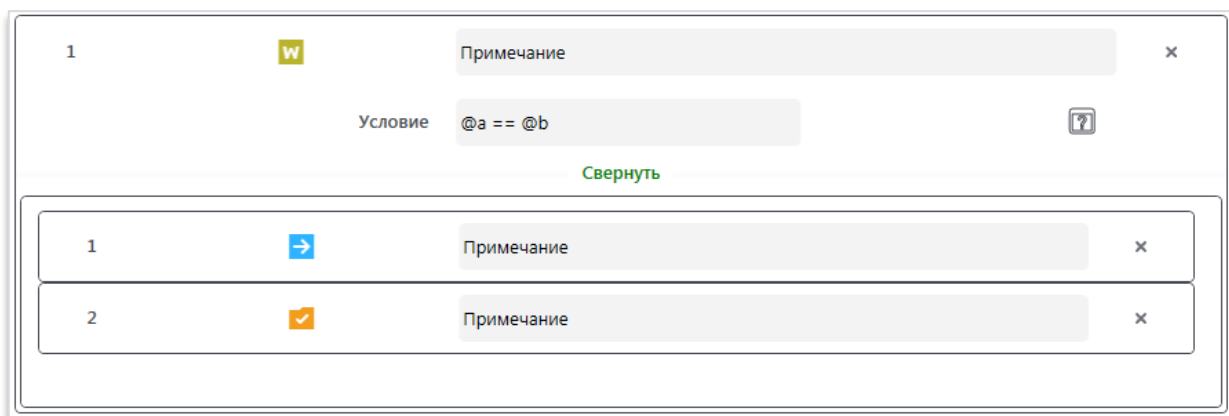


Рис. 5.14.2. Добавление действий

При нажатии на кнопку «Свернуть» действия, описанные в цикле, сворачиваются для более компактного отображения команды.

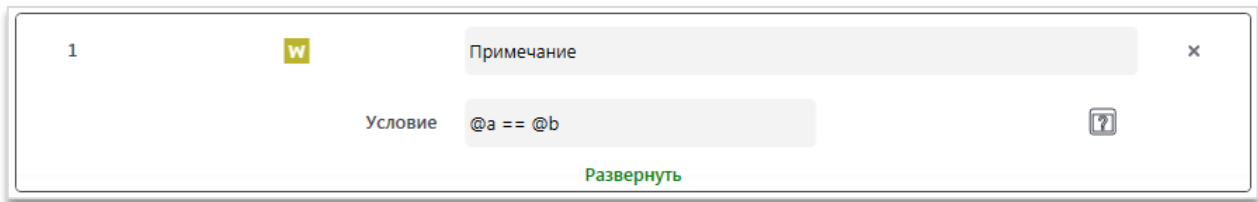


Рис. 5.14.3. Свернутый вид модуля

Глава 15. Модуль «Break»

Для «аварийного» выхода из простого цикла используют модуль «Break». Например, в цикле ведется поиск конкретного изображения, и если такое изображение найдётся, следует нажать на него, считать какие-либо данные и перейти к следующему действию, иначе – выйти из цикла и не проигрывать следующее действие. Для такого сценария необходимо создать цикл, в котором будет модуль «Условие», в одной из веток которого будет находиться модуль «Break»:

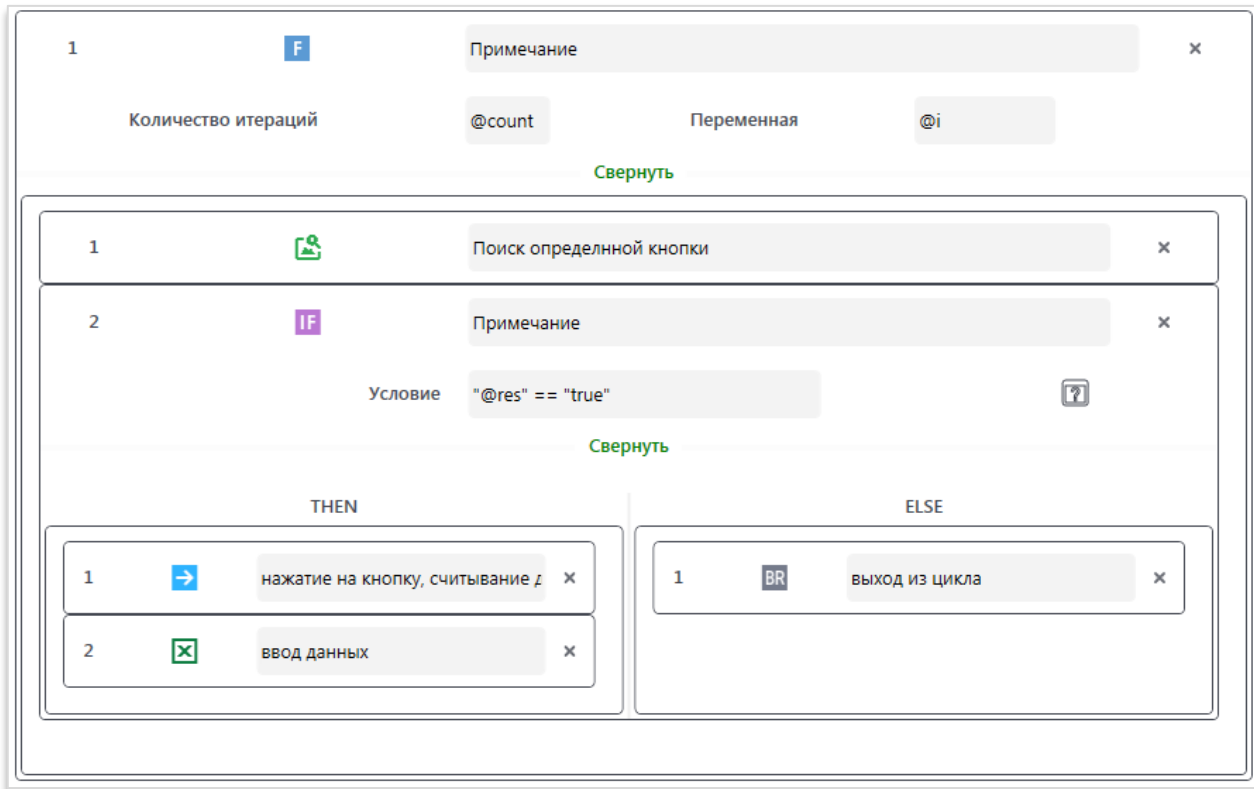


Рис. 5.15.1. Окно модуля

Модуль «Break» не требует настройки, но может применяться только внутри модуля «Цикл».

Глава 16. Модуль «Return»

Бывают ситуации, когда при некотором условии необходимо завершить работу робота, например, нужно обработать письмо, которое должно прийти на почту, а оно не пришло. Для таких целей существует модуль «Return».

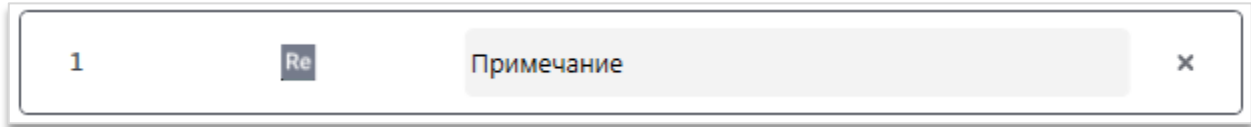


Рис. 5.16.1. Внешний вид модуля

Данный модуль заканчивает работу всего робота и не требует настройки. Может использоваться в любом месте.

ПРИМЕР.

На рисунке ниже представлен пример использования модуля: пусть с помощью модуля «Считать письма» считываются все письма за определенный промежуток времени. Все темы письма и их содержания необходимо внести в Excel-файл. Но что, если этих писем нет? Для этого проверяем наличие писем, и, если их нет, робот прекратит работу, иначе – продолжит.

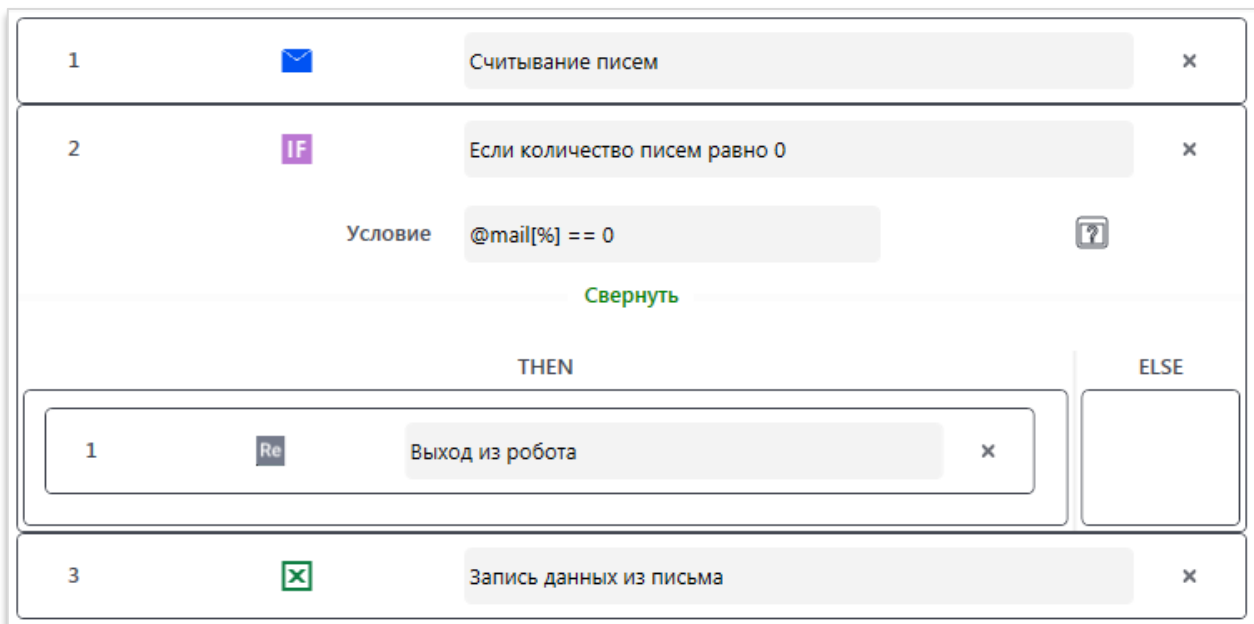


Рис. 5.16.2. Пример использования модуля «Return»

Благодаря этому модулю можно писать робота в общей рабочей области, а не в ветках модуля «Условие». Так робот выглядит более аккуратным и наглядным.

ЧАСТЬ VI. МОДУЛИ «ПОЧТОВЫЕ АГЕНТЫ»

Глава 17. Модуль «Отправить письмо»

Модуль «Отправить письмо» предназначен для отправки письма одному или нескольким получателям.

Интерфейс модуля

Окно модуля состоит из нескольких вкладок: «Настройки», «Получатель», «Письмо». Рассмотрим по порядку все вкладки. Начнем знакомство с рассмотрением вкладки «Настройки».

Вкладка «Настройки»

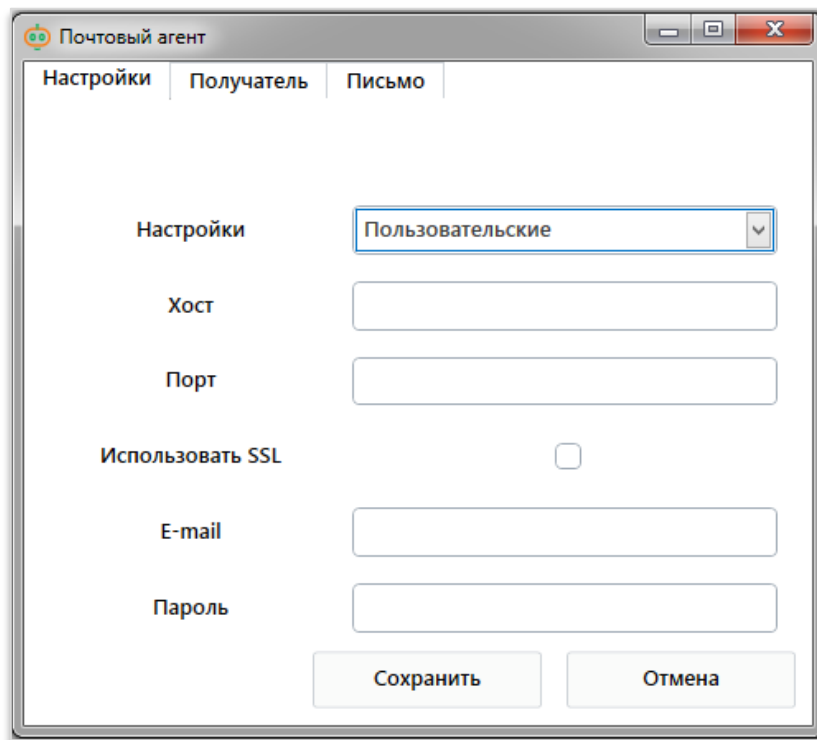
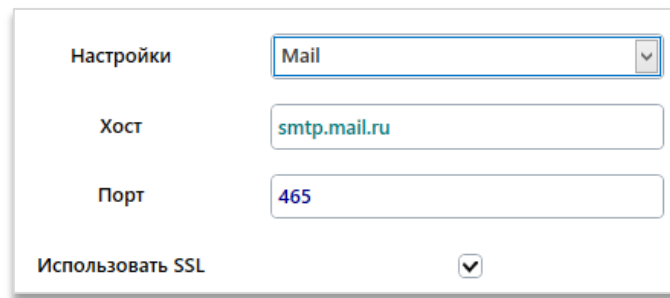


Рис. 6.17.1. Окно модуля

Поле «Настройки» – выпадающий список, в котором предложены несколько почтовых служб. При выборе почтовой службы будут заполнены поля «Хост», «Порт» и установлен флажок «Использовать SSL» согласно её настройкам. Пункт "Пользовательские" предоставляет возможность самостоятельной настройки параметров доступа к почтовой службе.



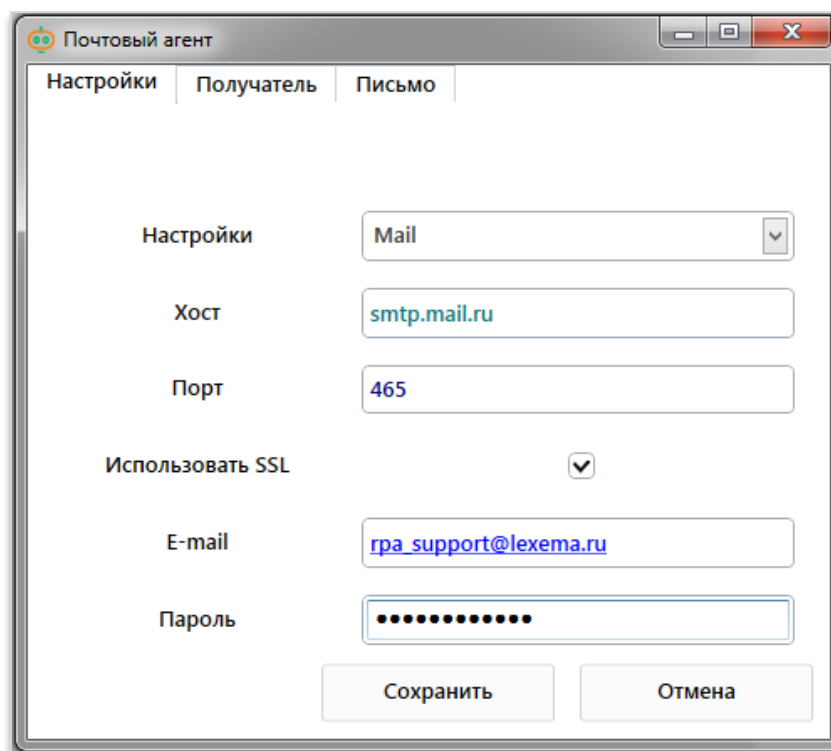
Настройки	Mail
Хост	smtp.mail.ru
Порт	465
Использовать SSL	<input checked="" type="checkbox"/>

Рис. 6.17.2. Пример настройки почтовой службы «Mail»

Поля «Хост», «Порт» и флажок «Использовать SSL» необходимы к заполнению только при выборе пользовательских настроек. Представляют собой набор обязательных полей для подключения к почтовым сервисам.

Поле «E-mail» предназначено для указания адреса электронной почты, с которой будут отправлены сообщения, в формате [name@mail.ru](mailto:lexema@lexema.ru).

Поле «Пароль» предназначено для ввода пароля от почты, указанной в поле «E-mail».



Настройки	Mail
Хост	smtp.mail.ru
Порт	465
Использовать SSL	<input checked="" type="checkbox"/>
E-mail	rpa_support@lexema.ru
Пароль

Сохранить Отмена

Рис. 6.17.3. Пример заполненной вкладки «Настройки»

Вкладка «Получатель»

На данной вкладке заполняется информация о лицах, которым будет отправлено письмо.

Модуль предоставляет различные способы указания получателей, а именно:

- ручной ввод списка получателей (имена и e-mail'ы);
- из переменной-элемента, в виде одного e-mail'а;
- из переменной-списка, в виде нескольких e-mail'ов.

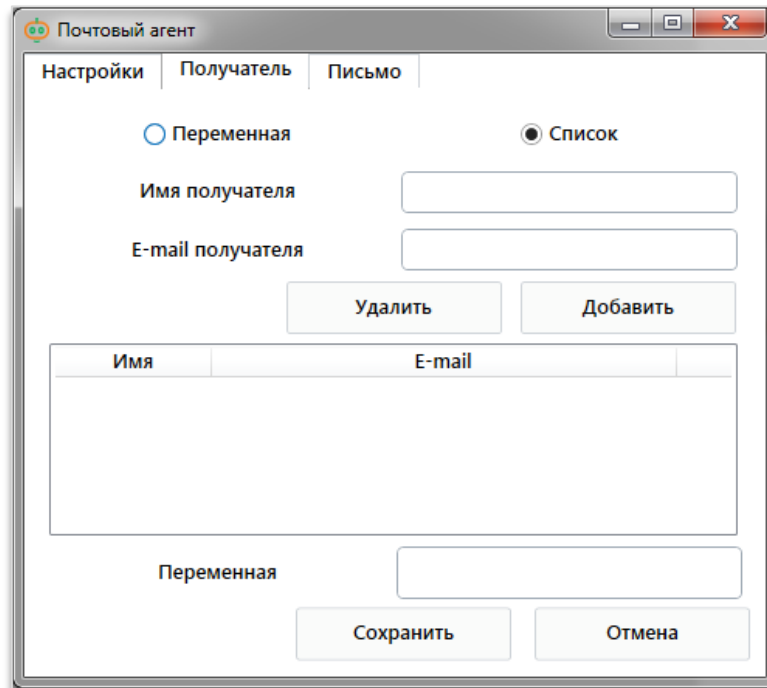


Рис. 6.17.4. Вкладка «Получатель»

Если выбран переключатель «Список», то поле «Переменная» недоступно для ввода, иначе – наоборот, доступно поле «Переменная», но недоступны другие поля.

Для ввода получателей методом «Список» необходимо заполнить поля «Имя получателя» и «E-mail получателя», затем нажать на кнопку под полями «Добавить». После этого введенные значения появятся в таблице в центре окна. Если необходимо удалить какого-либо получателя из таблицы, необходимо выбрать строку с ним и нажать на кнопку «Удалить».

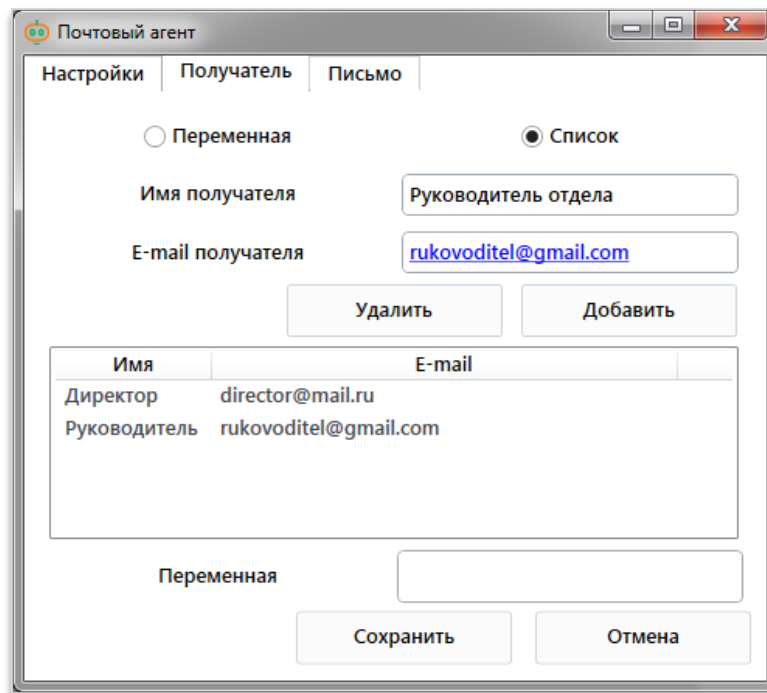


Рис. 6.17.5. Пример заполнения таблицы получателей

Если используется метод выбора получателей через переменную, то поля «Имя получателя», «E-mail получателя» и таблица становятся недоступными. В поле «Переменная» необходимо ввести название переменной, начиная с символа «@», из которой будут взяты email'ы получателей.

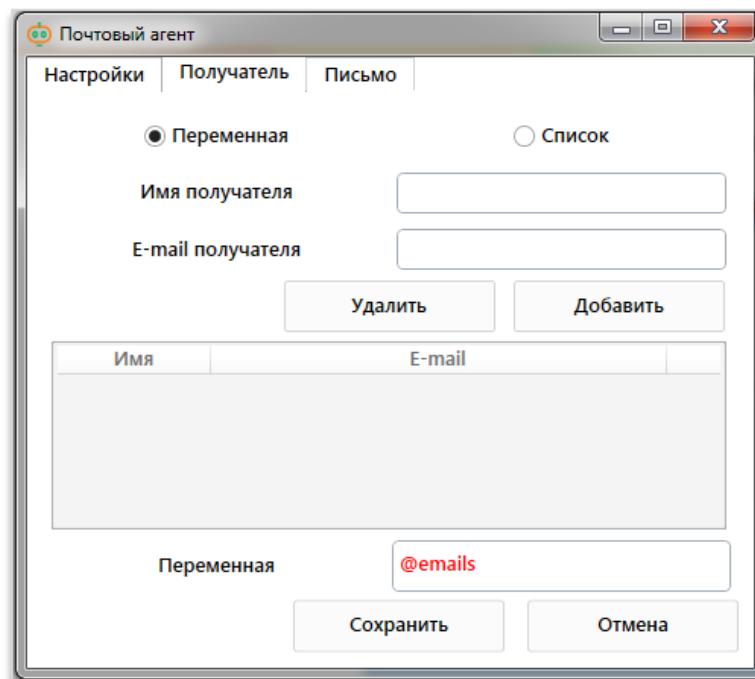
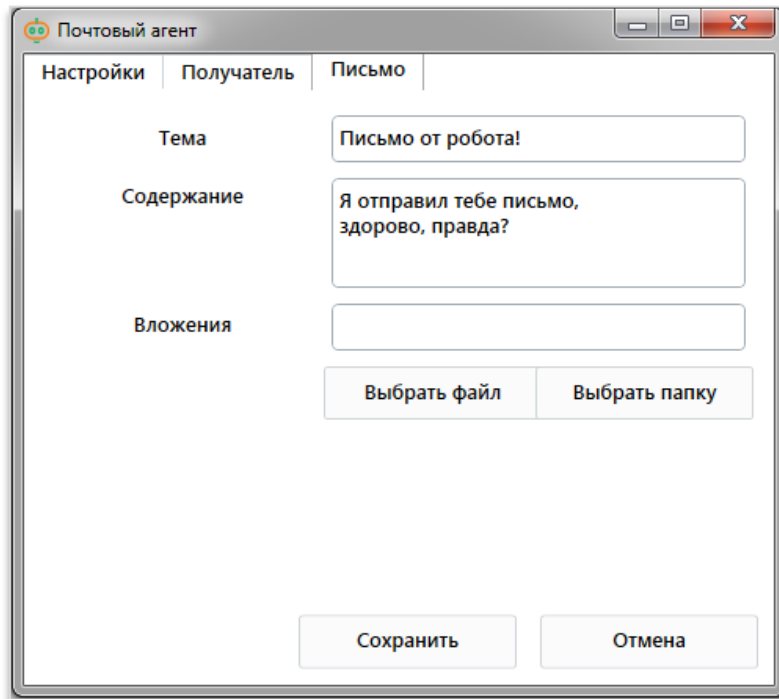


Рис. 6.17.6. Вкладка «Получатель»

Вкладка «Письмо»

На данной вкладке настраивается письмо, которое будет отправлено получателем, указанным на вкладке «Получатель». Все поля необязательны для заполнения, если не заполнить ни одного из них будет отправлено пустое письмо с темой «<Без темы>».



Настройки	Получатель	Письмо
		Тема: Письмо от робота!
		Содержание: Я отправил тебе письмо, здорово, правда?
		Вложения: [Выбрать файл] [Выбрать папку]
		[Сохранить] [Отмена]

Рис. 6.17.7. Вкладка «Письмо»

Для прикрепления одного файла во вложение следует нажать на кнопку «Выбрать файл» и в открывшемся диалоговом окне выбрать требуемый файл.

Для прикрепления сразу нескольких файлов необходимо нажать на кнопку «Выбрать папку» и в открывшемся диалоговом окне выбрать папку со всеми требуемыми файлами.

Каждое из полей формы может быть заполнено переменной.

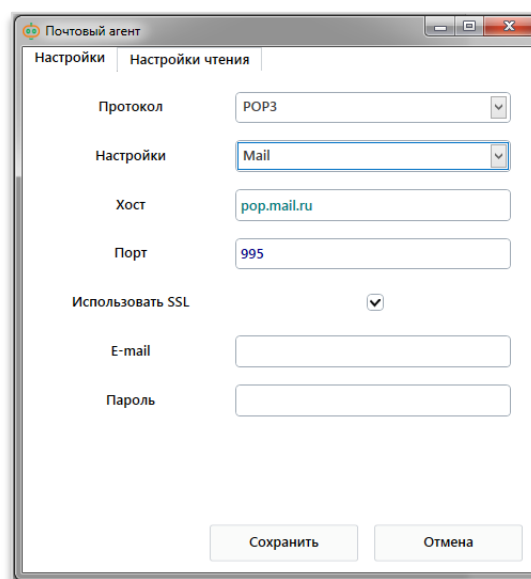
Глава 18. Модуль «Считать письма»

Модуль «Считать письма» предназначен для получения списка писем с почты для дальнейшей их обработки или просто для скачивания вложений.

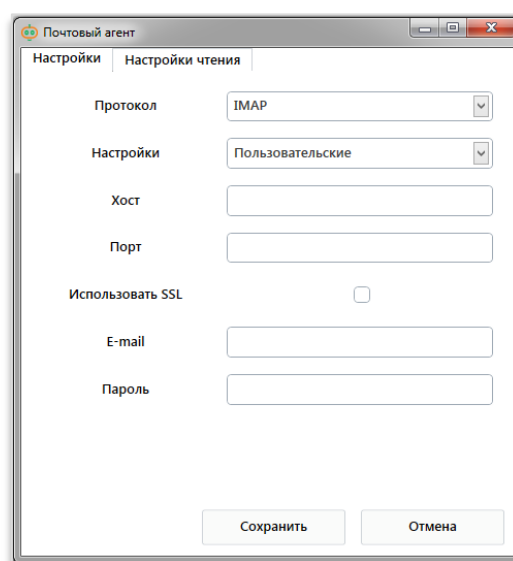
Интерфейс модуля

Модуль состоит из двух вкладок: «Настройки» и «Настройки чтения».

Вкладка «Настройки» аналогична одноименной вкладке в модуле «Отправить письмо», с одним лишь отличием - в данном модуле предоставляется выбор протокола отправки: IMAP или POP3. При выборе соответствующего протокола и его настроек, поля «Хост», «Порт» и «Использовать SSL» заполняются автоматически (кроме режима «Пользовательские настройки»).



The screenshot shows the 'Почтовый агент' (Mail Agent) settings window, 'Настройки' (Settings) tab. The 'Протокол' (Protocol) is set to 'POP3'. The 'Настройки' (Settings) dropdown is set to 'Mail'. The 'Хост' (Host) field is filled with 'pop.mail.ru'. The 'Порт' (Port) field is filled with '995'. The 'Использовать SSL' (Use SSL) checkbox is checked. The 'E-mail' and 'Пароль' (Password) fields are empty. At the bottom, there are 'Сохранить' (Save) and 'Отмена' (Cancel) buttons.



The screenshot shows the 'Почтовый агент' (Mail Agent) settings window, 'Настройки' (Settings) tab. The 'Протокол' (Protocol) is set to 'IMAP'. The 'Настройки' (Settings) dropdown is set to 'Пользовательские' (User-defined). The 'Хост' (Host) and 'Порт' (Port) fields are empty. The 'Использовать SSL' (Use SSL) checkbox is unchecked. The 'E-mail' and 'Пароль' (Password) fields are empty. At the bottom, there are 'Сохранить' (Save) and 'Отмена' (Cancel) buttons.

Рис. 6.18.1. Окно модуля

Вкладка «Настройки чтения»

Вкладка «Настройки чтения» представляет собой набор полей, которые заполняются в зависимости от того, какие письма следует считать.

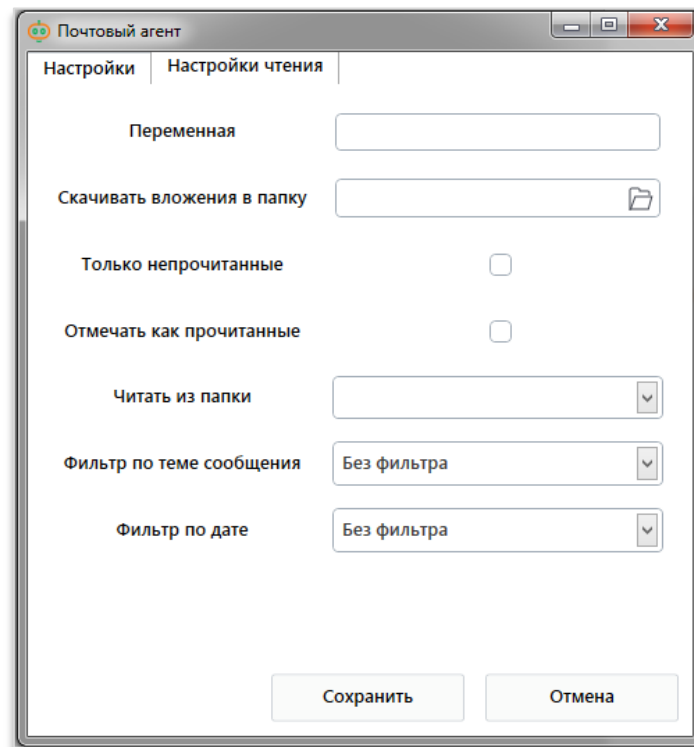


Рис. 6.18.2. Вкладка «Настройка чтения»

Поле «Переменная». В поле ожидается ввод названия переменной, в которую будет внесена информация о считанных письмах. В программе это поле будет представлено списком объектов, имеющих четыре поля:

- 1) Subject – тема письма;
- 2) Body – тело письма, его основной текст;
- 3) Sender – отправитель;
- 4) Attachments – список путей до скачанных вложений.

Примеры работы с переменными

- @mail[1].Attachments[0] – получение пути первого вложения второго считанного письма:

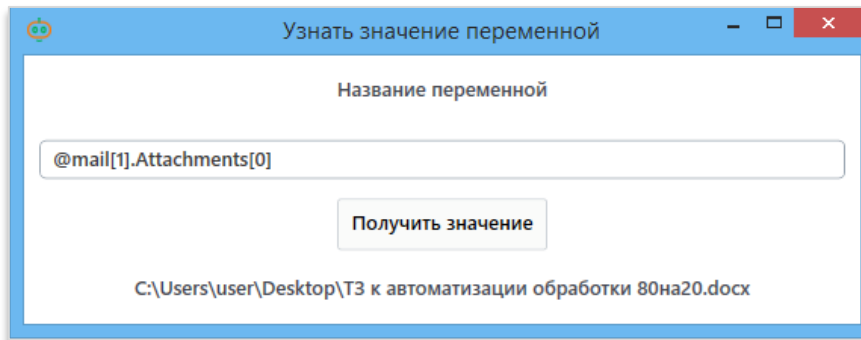


Рис. 6.18.3. Пример получения пути до первого вложения второго письма

- @mail[0].Subject – получение темы первого скачанного сообщения:

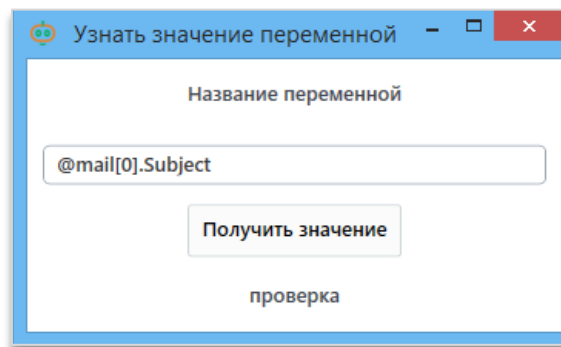


Рис. 6.18.4. Пример получения темы первого письма

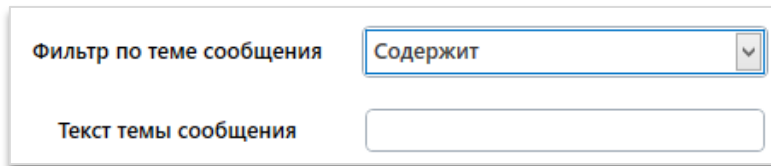
Поле «Скачивать вложения в папку» предназначено для ввода пути до папки, в которую загрузятся все вложения считанных писем. Если оставить поле пустым, то файлы будут загружаться во временную директорию компьютера.

Флажок «Только непрочитанные» позволяет настроить считывание только новых писем. *Доступно только для протокола IMAP.*

Флажок «Отмечать как прочитанные» позволяет после считывания устанавливать пометку о прочтении письма. *Доступно только для протокола IMAP.*

Поле «Читать из папки» предназначено для настройки конкретной папки в почтовом ящике, из которой следует считывать письма. По умолчанию, письма считываются из папки «Входящие». *Доступно только для протокола IMAP.*

Поле «Фильтр по теме сообщений» предназначено для настройки фильтра по теме сообщений. Если необходимо скачать письма с темой, содержащей или не содержащей определенную строку, то следует выбрать одноименную операцию в выпадающем списке и вписать нужную строку в появившемся поле:



Фильтр по теме сообщения

Текст темы сообщения

Рис. 6.18.5. Фильтр по теме сообщений

Поле «Фильтр по дате» предназначено для настройки фильтрации считывания писем по дате. При указании значения в данном поле, робот будет считывать только те сообщения, которые удовлетворяют заданному условию.

Доступны следующие методы фильтрации:

- за определенную дату – «На дату»;
- начиная с какой-либо даты, включая её – «С даты»;
- все письма, до какой-либо даты, включая её – «До даты»;
- «Между датами».

После выбора метода фильтрации под полем «Фильтр по дате» появится одно или два поля для внесения в них нужных значений в формате ДД.ММ.ГГГГ.

ЧАСТЬ VII. МОДУЛЬ «РАБОТА С БРАУЗЕРОМ»

Глава 19. Основные моменты модуля

Модуль «Работа с браузером» построен на базе Selenium WebDriver⁶ и позволяет реализовывать основные методы работы со страницей браузера – открыть вкладку, перейти на вкладку, найти элемент на странице по её ключу и как-то его обработать.

Интерфейс модуля

Окно модуля делится на область создания команды для работы с браузером, группу кнопок управления действиями и таблицу, содержащую список созданных команд. Область создания команды состоит из нескольких полей для заполнения, причем, количество этих полей меняется в зависимости от выбранного действия в поле «Действие». Группа кнопок состоит из трех кнопок – «Добавить», «Редактировать» и «Удалить».

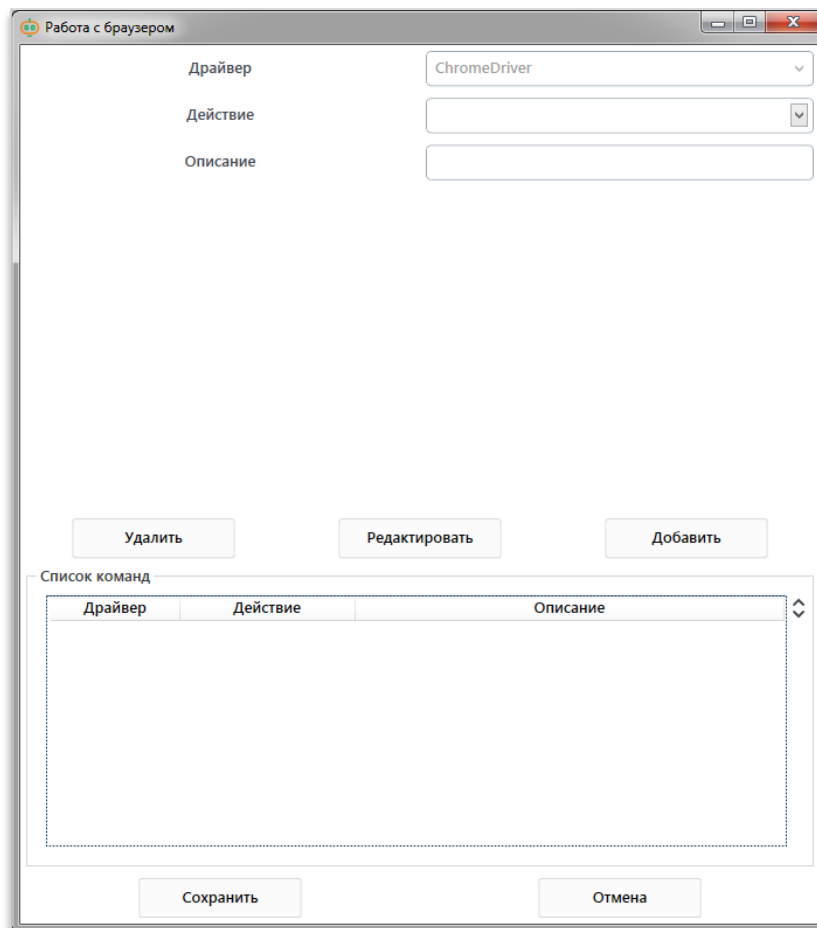


Рисунок 7.19.1. Окно модуля «Работа с браузером»

⁶ Selenium WebDriver – это программная библиотека для управления браузерами.

Создание команды начинается с выбора действия. После его выбора список доступных полей в окне может измениться. Необходимо заполнить поля, требующиеся для выбранного действия и нажать на кнопку «Добавить». Созданное действие отобразится в таблице «Список команд». Для редактирования команды следует выбрать её в списке команд, изменить необходимые поля и нажать «Редактировать». Если при этом нажать на кнопку «Добавить», то будет добавлено новое действие, похожее на то, что было выбрано в списке команд, но с внесенными изменениями. Можно ничего не менять и нажать на кнопку «Добавить», тогда получится полное копирование команды.

Для удаления ненужных команд необходимо выбрать команду в таблице и нажать «Удалить».

Робот будет выполнять все указанные команды последовательно, поэтому для изменения порядка команд в таблице можно воспользоваться стрелочками в правом верхнем углу таблицы. Делается это следующим образом – выбирается строка, положение которой необходимо изменить, а затем нажимаются стрелочки вверх или вниз, в зависимости от того, где эта строка должна находиться.

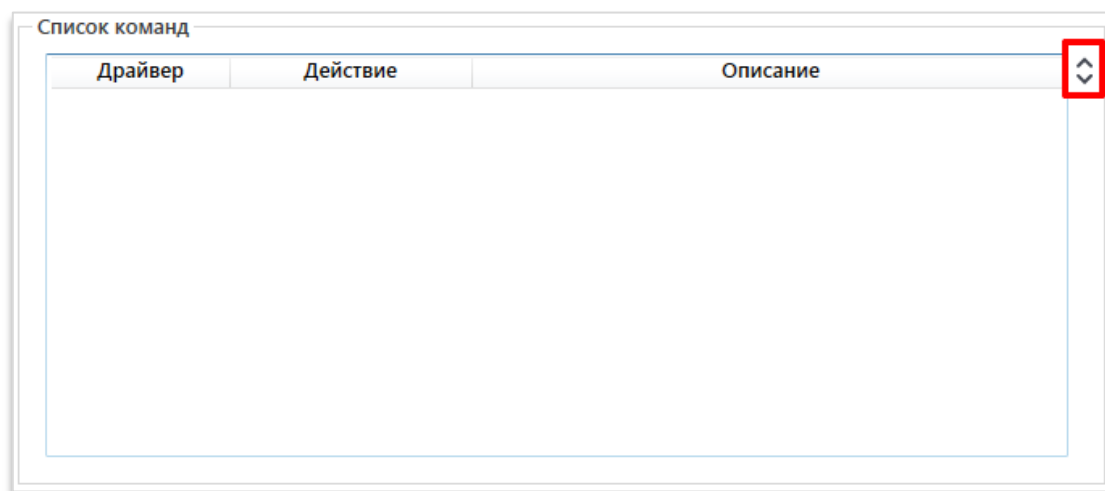


Рис. 7.19.2. Стрелочки для изменения порядка команд

Для всех действий три первых поля остаются неизменными, это «Драйвер», «Действие» и «Примечание».

Поле «Драйвер» на данный момент недоступно для редактирования и в нем по умолчанию выбран драйвер для браузера Chrome, но в обозримом будущем этот список планируется дополнять другими браузерами.

Поле «Действие» — это выпадающий список с перечнем возможных действий этого модуля. Подробнее о них будет рассказываться в следующей главе.

Поле «Примечание» — строка с примечанием к команде.

Глава 20. Описание действий

Модуль предлагает использование 7 основных действий – перейти на страницу, перезагрузить страницу, перейти назад или вперед, работа со страницей, работа со вкладками и закрыть браузер. Каждое действие требует заполнения определенных полей.

Обратите внимание, что закрытие браузера, открытого с помощью модуля «Работа с браузером», следует осуществлять средствами самого модуля, а не самостоятельно, иначе, если вы закроете браузер сами, а затем попытаетесь продолжить работу с ним с помощью этого модуля, то при каждой попытке запуска робота будет возникать ошибка и придется перезапускать программу.

Перечень и описание действий

Действие «Перейти»

С помощью действия «Перейти» происходит переход на заданную страницу. В области создания команды появляется поле – «URL».

Драйвер	ChromeDriver
Действие	Перейти
Примечание	Открытие страницы "Конструктор роботов"
Url	http://lexema.ru/solutions/konstruktor-robotov/

Рис. 7.20.1. Пример настройки действия «Перейти»

Если уже открыта какая-либо страница, то браузер перейдет от неё к странице, указанной в поле «URL», то есть исходная страница будет заменена новой. Если браузер еще не был запущен, то он будет открыт с этой страницы.

Действие «Перезагрузить страницу»

Действие позволяет перезагружать уже открытую страницу. Перед использованием данного действия, убедитесь, что в браузере есть открытая страница.

Действие не имеет никаких дополнительных для настройки полей.

Действие «Назад»

Действие позволяет выполнять переход к странице, которая была открыта в браузере до текущей.

Действие «Вперед»

Действие позволяет выполнять переход к странице, которая была открыта в браузере после текущей, если таковая была.

Действие «Работа со страницей»

Данное действие предоставляет возможность работать с отдельным элементом страницы. Для настройки действия появляются следующие поля: «Тип поиска элемента», «Строка поиска», «Действие на странице», «Переменная», «Ожидание» и «Номер элемента».

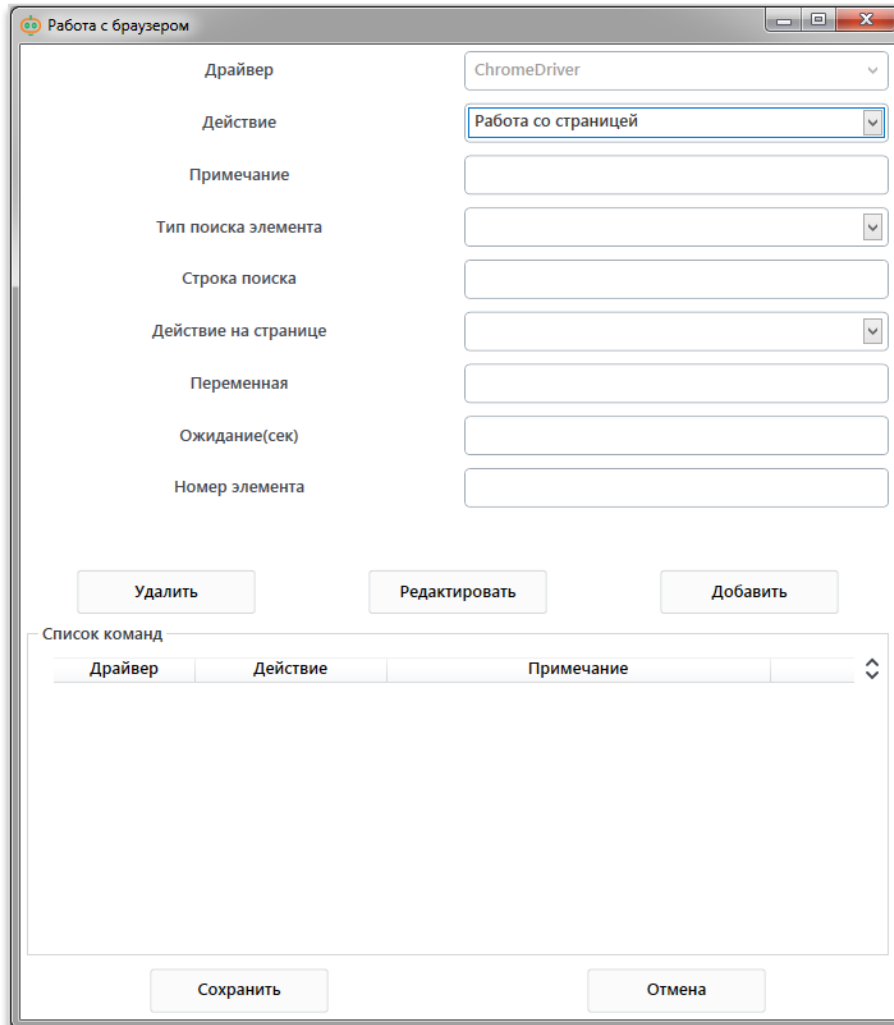


Рис. 7.20.2. Вид окна для действия «Работа со страницей»

Поле «**Тип поиска элемента**» представляет собой выпадающий список с перечнем методов поиска элемента на странице, используется в паре с полем «**Строка поиска**».

Ниже будут перечислены все типы поиска элемента с примерами заполнения поля «**Строка поиска**»:

- GetElementsByName – поиск элементов по их свойству name.

```
<div name="element">
  <p>some content</p>
</div>
```

Листинг. 7.20.1. Пример элемента со свойством name

«Строка поиска» для нахождения элемента div-element.

- GetElementsById – поиск элементов по их свойству id.

```
<div id="element">
  <p>some content</p>
</div>
```

Листинг. 7.20.2. Пример элемента со свойством id.

«Строка поиска» для нахождения элемента div-element.

- GetElementsByClass – поиск элементов по их свойству class.

```
<div id="element">
  <p>some content</p>
</div>
```

Листинг. 7.20.3. Пример элемента со свойством class

«Строка поиска» для нахождения элемента div-myClass.

- GetElementsByCssSelector – поиск элементов по их CSS селектору.⁷

```
<div class="main">
  <p>text</p>
  <p> Another text </p>
</div>
```

Листинг. 7.20.4. Пример элемента div со свойством class

«Строка поиска» для нахождения элемента div-div.main.

- GetElementByLinkText – поиск элементов-ссылок по их тексту. Ищутся только элементы с полным совпадением.

```
<div>
  <a ref=" ...">text</a>
  <a ref=" ...">Another text</a>
</div>
```

Листинг. 7.20.5. Пример элементов – ссылок a

⁷ Селектор — это часть CSS-правила, которая сообщает браузеру, к какому элементу (или элементам) веб-страницы будет применён стиль.

«Строка поиска» нахождения первого элемента а на примере – text.

- `GetElementsByPartialLinkText` - поиск элементов-ссылок по их тексту. Ищутся все элементы, содержащие заданную строку.

```
<div>
  <a ref="...">text</a>
  <a ref="...">Another text</a>
</div>
```

Листинг. 7.20.6. Пример элементов – ссылок а

«Строка поиска» нахождения двух элементов а – text

- `GetElementsByTagName` – поиск элементов по их тегу⁸.

```
<div>
  <a ref="...">text</a>
  <p>Another text</p>
</div>
```

Листинг. 7.20.7. Пример элементов с разными тегами

«Строка поиска» нахождения элемента а – а.

- `GetElementsByXPath` – поиск элементов по языку запросов HTML – xpath.

```
<div class="main">
  <p>text</p>
  <p> Another text </p>
</div>
```

Листинг. 7.20.8. Пример элементов div со свойством class

«Строка поиска» для нахождения элемента div - `//div[@class='main']`.

Поле **«Действие на странице»** предлагает выбор действия из выпадающего списка, которое необходимо произвести над найденными элементами. Доступны следующие действия:

- **Записать** – вводит заданный в поле «Переменная» текст (например, после нажатия на определенный элемент для ввода текста);
- **Получить text** – получает значение свойства «text» всех найденных или только выбранного элемента и возвращает результат в виде списка в переменную;

⁸ Тег - элемент языка разметки.

- Отправить – нажатие клавиши «ВВОД» («ENTER») – применение записанных данных (например, в строке поиска можно ввести какой-либо текст действием «Записать», а затем «Отправить» его);
- Нажать – нажатие левой кнопкой мыши на заданный элемент (обязательно заполнение поля «Номер элемента»);
- Навести мышь – наведение мыши на конкретно заданный элемент (обязательно заполнение поля «Номер элемента»);
- Получить Property – при выборе действия появляется поле «Название Property» - возвращает значение указанного в поле «Название Property» свойства всех найденных элементов или конкретно заданного. Результатом будет являться список;
- Получить CSS свойство – при выборе действия появляется поле «Свойство CSS» - возвращает значение указанного в поле «Свойство CSS» свойства всех найденных элементов или конкретно заданного. Результатом будет являться список;
- Получить Attribute – при выборе действия появляется поле «Свойство Attribute» - возвращает значение указанного в поле «Свойство Attribute» свойства всех найденных элементов или конкретно заданного. Результатом будет являться список;

В поле «Переменная» вводится название переменной, начиная с символа «@», в которую будет помещен результат выполнения команды или, наоборот, из которой будут браться значения для её выполнения.

Поле «Ожидание (сек)». В поле вводится количество секунд, которые необходимо подождать после выполнения команды.

Поле «Номер элемента» предназначено для указания индекса найденного элемента, к которому необходимо применить действие.

Пример заполненного действия:

Работа с браузером

Драйвер: ChromeDriver

Действие: Работа со страницей

Примечание: Вставка данных в поисковик

Тип поиска элемента: GetElementByName

Строка поиска: q

Действие на странице: Записать

Переменная: поиск

Ожидание(сек): 10

Номер элемента: 0

Удалить Редактировать Добавить

Список команд

Драйвер	Действие	Примечание
ChromeDriver	Перейти	Вставка данных в поисковик
ChromeDriver	Работа со страницей	Вставка данных в поисковик

Сохранить Отмена

Рис. 7.20.3. Пример настройки действия «Работа со страницей»

На рисунке 7.20.3 представлен пример действия, настроенного для поиска элемента, имя которого равно «q». После того как элемент найден, в него будет вставлен текст «поиск».

Действие «Работа с вкладками»

Данное действие позволяет манипулировать вкладками в окне браузера. Для настройки действия добавляется как минимум одно поле – «Действие», остальные появляются в зависимости от выбранного в нем действия.

Драйвер	ChromeDriver
Действие	Работа с вкладками
Примечание	
Действие	

Рис. 7.20.4. Поля при выборе действия «Работа со вкладками»

Доступные действия для работы с вкладками:

- Добавить вкладку – создаст новую вкладку и та станет активной;
- Закрыть вкладку – закроет активную вкладку;
- Перейти на вкладку – при выборе действия появляется поле «Номер вкладки» - переходит на вкладку, номер которой указан в поле «Номер вкладки» (начиная с 0);
- Получить количество вкладок – при выборе действия появляется поле «Переменная» - возвращает в указанную переменную количество вкладок в браузере.

Действие «Закрыть браузер»

Закрывает браузер, открытый с помощью модуля, не требует дополнительных настроек.

ЧАСТЬ VIII. МОДУЛЬ «ИНТЕРФЕЙС»

Глава 21. Основные моменты

Данный модуль предназначен для создания диалоговых окон для общения с пользователем робота. Это могут быть окна, позволяющие прикрепить файл, ввести какие-либо данные – дату, имена, почтовые адреса и другое, с чем впоследствии будет работать робот, или наоборот, окна, отображающие результат работы.

Создание окна заключается в добавлении на него контролов – элементов интерфейса таких, как надписи, текстовые поля, кнопки, выпадающие списки.

Основное окно

Окно модуля состоит из нескольких частей:

- Добавить элемент - настройка добавляемого элемента;
- Общие настройки – общие настройки диалогового окна;
- Список элементов – таблица, в которой содержатся все добавленные элементы;
- Окно с предварительным просмотром диалогового окна на текущий момент (правая часть окна).

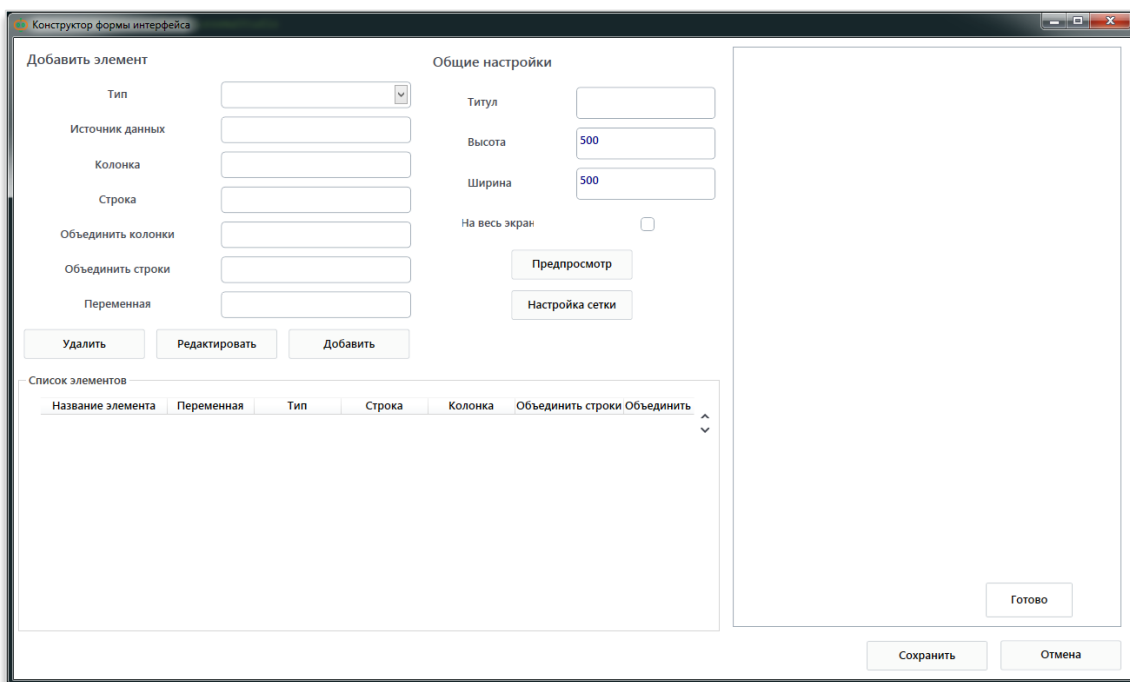


Рис. 8.21.1. Основное окно модуля

Начинать работу с диалоговым окном следует с общих настроек, а именно с настройки сетки, которую можно открыть по кнопке «Настройка сетки».

Глава 22. Общие настройки

К общим настройкам относится название окна – «Титул», его ширина и высота и настройка сетки. При установке флажка «На весь экран» поля «Высота» и «Ширина» не будут учитываться.

По нажатию на кнопку «Настройка сетки» появится окно настроек строк и колонок сетки.

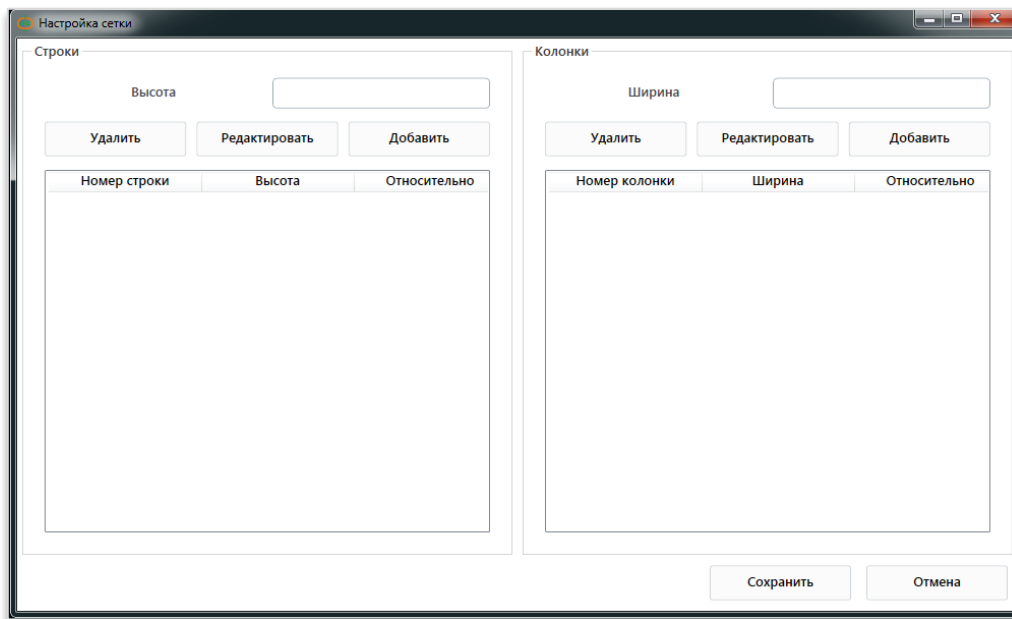


Рис. 8.22.1. Настройка сетки

Сетка – это условная разметка элементов интерфейса, некоторая таблица, в которую будут помещаться контролы. Например, если нужно создать окно, в которое пользователь бы вводил два каких-то параметра – конечную и начальную даты, за которые необходимо скачивать почту, то настроить диалоговое окно для ввода этих дат можно несколькими способами:

- Первый способ

В две строки, 4 элемента интерфейса – на первой строке подпись и поле для ввода начальной даты, на второй строке подпись и поле для ввода конечной даты

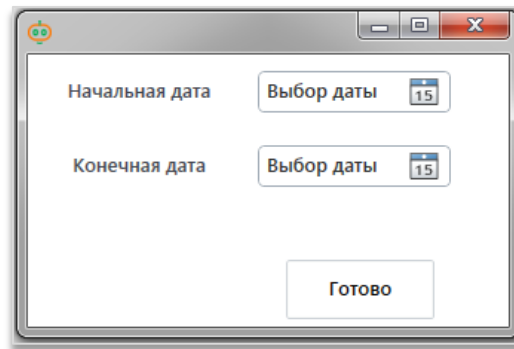


Рис. 8.22.2. Пример окна №1

Для такой настройки сетки необходимо задать две колонки – первая, для надписей, вторая для полей ввода, и две строки – первая для начальной даты, вторая - для конечной.

- Второй способ

В одну строку, 3 элемента интерфейса – подпись, поле для ввода первой даты, поле для ввода второй даты.

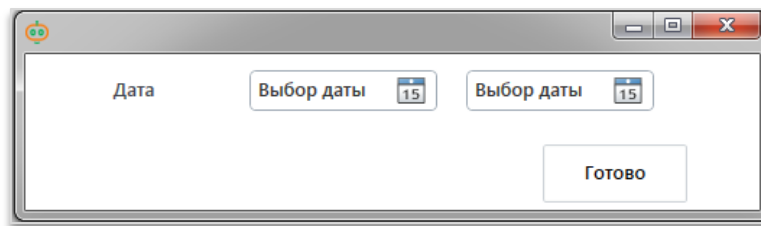


Рис. 8.22.3. Пример окна №2

В таком случае используется три колонки и одна строка.

- Третий способ

В одну строку, 4 элемента интерфейса – два поля для ввода и подписи к каждому из них.

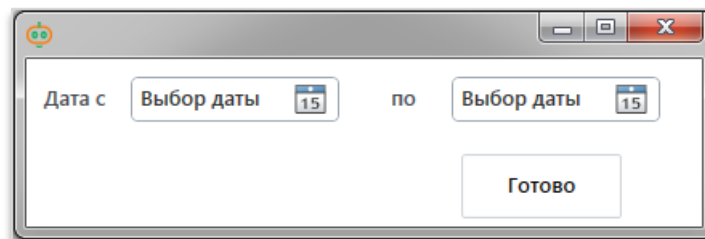


Рис. 8.22.4. Пример окна №3

Здесь используются 4 колонки и 1 строка.

Для каждой колонки можно задать свою ширину, для каждой строки – высоту:

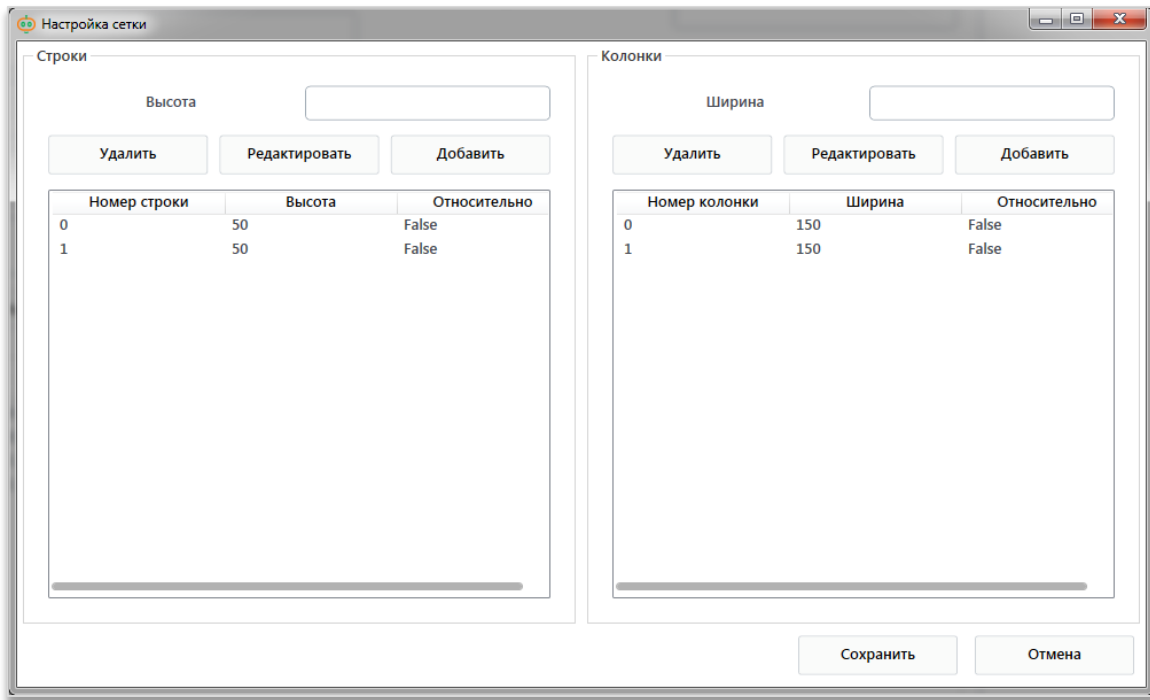


Рис. 8.22.5. Пример настройки сетки

Высота и ширина вводятся в пикселях, но можно использовать относительные размеры окна, например, если одна колонка должна быть больше другой в два раза и их всего две, то в поле «Ширина» можно ввести «1*» для первой колонки, и «2*» для второй:

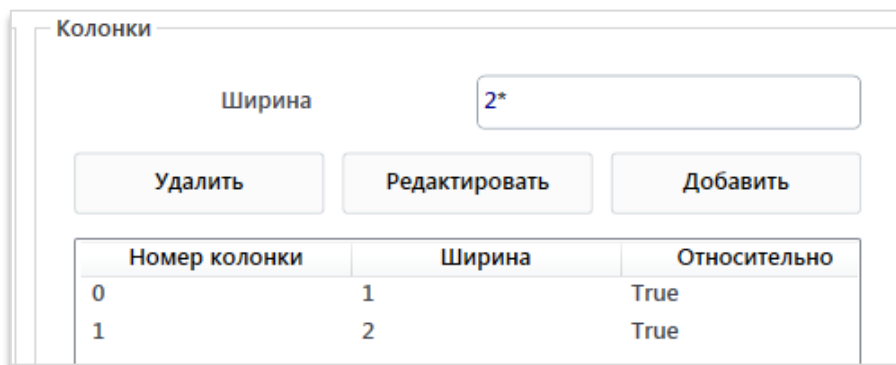


Рис. 8.22.6. Пример использования относительных размеров

Результат такой разметки:



Рис. 8.22.7. Пример разметки

Нумерация строк и колонок начинается с 0. С помощью кнопки «Удалить» можно удалить ненужную строку или колонку, по кнопке «Редактировать» можно изменить высоту или ширину.

После настройки сетки она будет отображена пунктирной линией в окне предварительного просмотра в правой части окна, и затем можно приступить к добавлению контролов.

Глава 23. Добавление элемента интерфейса

Для добавления на окно нового элемента необходимо заполнить поля, представленные на рис. 8.23.1:

Рис. 8.23.1. Добавление нового элемента

Обязательными полями являются все, кроме полей «Объединить колонки/строки».

Поле «Тип»

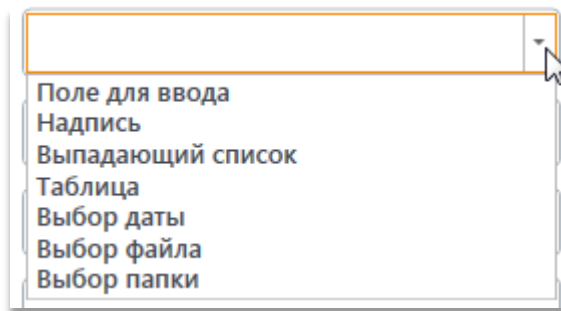


Рисунок 8.23.2. Выпадающий список с выбором типа элемента

Сначала выбирается тип добавляемого элемента интерфейса из предложенных:

- Поле для ввода – поле, в которое можно ввести любую строку;



Рисунок 8.23.3. Элемент «Поле для ввода»

- Надпись – статичная строка текста, которая не может быть отредактирована пользователем;

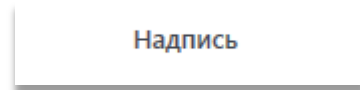


Рисунок 8.23.4. Элемент «Надпись»

- Выпадающий список – поле со стрелочкой сбоку, по нажатию которого открывается список заранее определенных значений;

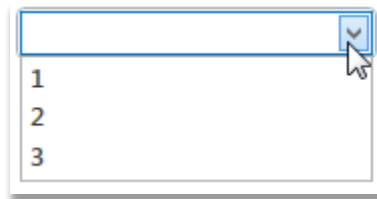


Рисунок 8.23.5. Элемент «Выпадающий список»

- Таблица – отображение в виде таблицы заранее созданной переменной. Возможно редактирование полей таблицы, но невозможно создание новых колонок в ней;

Column1	Column2	Column3
1.1 ячейка	1.2 ячейка	1.3 ячейка
2.1 ячейка	2.2 ячейка	

Рисунок 8.23.6. Элемент «Таблица»

- Выбор даты – поле с иконкой календаря, по нажатию на которую пользователю будет предложено выбрать определенную дату. Возможен ввод значения даты вручную, при вводе других строк поле автоматически подберет наиболее подходящую дату к введенным данным. Формат даты ДД.ММ.ГГГГ;

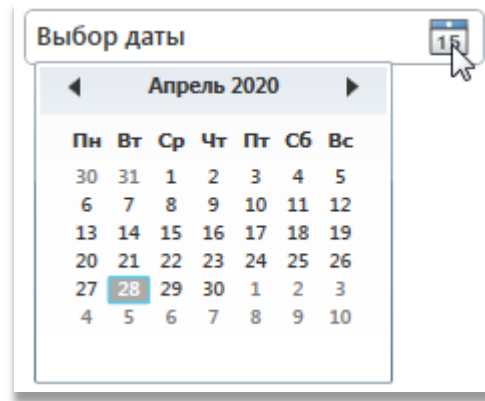


Рисунок 8.23.7. Элемент «Выбор даты»

- Выбор файла – поле с иконкой в виде трех точек, по нажатию на которую пользователю будет предложено выбрать файл. Путь до выбранного файла запишется в это поле;



Рисунок 8.23.8. Элемент «Выбор файла»

- Выбор папки – поле с иконкой в виде трех точек, по нажатию на которую пользователю будет предложено выбрать папку. Путь до выбранной папки запишется в это поле. Выглядит так же, как и «Выбор файла».

Поле «Источник данных»

В это поле вводится название переменной или строка, на основе которой контрол будет заполнен. Немного поподробнее о поле при выборе различных элементов интерфейса:

- Для типа элемента «Надпись» туда вводится значение, которое должно быть отображено в выводимой надписи;
- Для «Поле для ввода», «Выбор даты», «Выбор файла» и «Выбор папки» вводится значение, которое сразу будет указано в поле, если это необходимо. Если поле должно быть пустым, то не надо заполнять «Источник данных»;
- Для «Таблицы» необходимо подать заполненную табличную переменную для её отображения;
- Для элемента интерфейса «Выпадающий список» необходимо передать набор значений, которые пользователю будет предложено выбрать. Делается это через символ «;». Это могут быть строки, переменные-элементы, переменные-списки или табличные переменные, состоящие из одного столбца. Например, введено следующее значение, при приведенных далее примерах заполнения переменных:

Тип	Выпадающий список
Источник данных	EUR;@usd;@currency

Рис. 8.23.9. Пример заполнения поля «Источник данных»

Переменная

Название переменной: @usd

Тип переменной: Элемент

Значение: USD

Константа:

Зашифровать:

Удалить Сохранить Отмена

Переменная

Название переменной: @currency

Тип переменной: Список

Значение:

RUB
GBP
CHF

Удалить Сохранить Отмена

Рис. 8.23.10. Примеры заполнения используемых переменных

Результат будет выглядеть следующим образом:

EUR
USD
RUB
GBP
CHF

Рис. 8.23.11. Пример результирующих данных

Поля «Колонка» и «Строка»

В эти поля необходимо ввести номер колонки и строки сетки (таблицы), в которых должен располагаться добавляемый контрол. Например, в примере с датами (рис. 8.22.2) контрол «Надпись» - «Начальная дата» располагается в 0 колонке и 0 строке, а надпись «Конечная дата» - в 0 колонке и 1 строке.

Поля «Объединить колонки» и «Объединить строки»

В некоторых случаях необходимо выводить поля не в конкретной клеточке сетки, а в объединении нескольких клеток. Например – нужно предложить пользователю выбрать Excel-файл для записи в него результатов и ввести начальную и конечную дату, по которым будут собираться некоторые данные. Хотелось бы объединить это в две строки, но для ввода даты нужно как минимум 2 поля, а поле для выбора файла – одно.

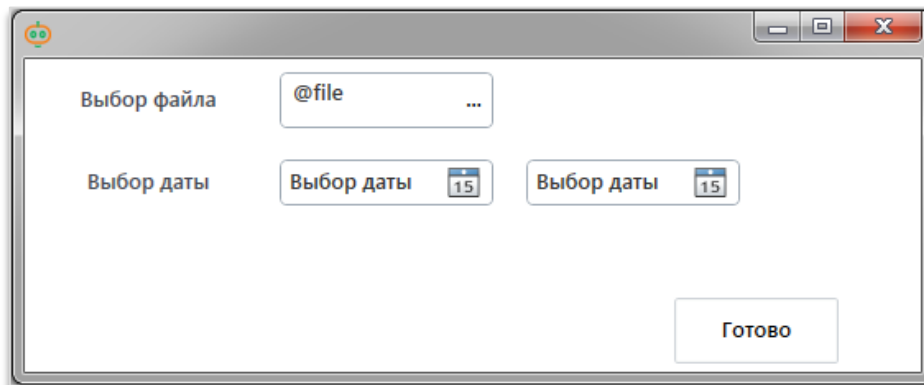


Рис. 8.23.12. Пример окна без объединения колонок

Для того, чтобы растянуть поле для выбора файла по длине полей для ввода даты, нам необходимо разместить его не просто в 1 колонке и 0 строке, но и объединить для него две колонки, чтобы оно занимало сразу обе:

Тип	Выбор файла
Источник данных	@file
Колонка	1
Строка	0
Объединить колонки	2
Объединить строки	

Рис. 8.23.13. Пример заполнения полей элемента интерфейса для настройки объединения колонок

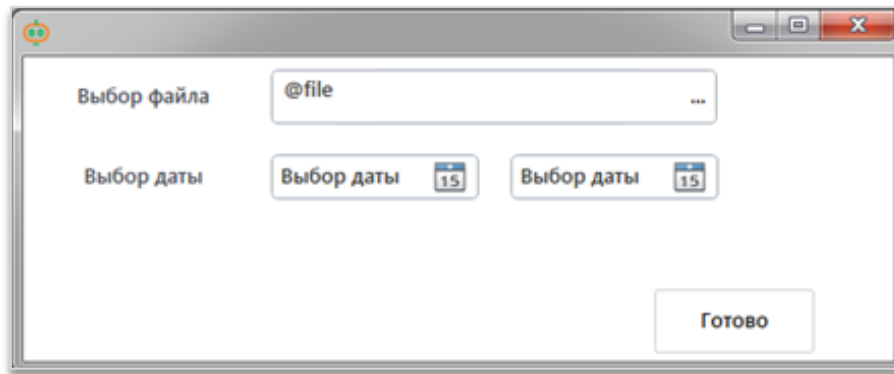


Рис. 8.23.14. Пример окна с объединенными колонками в первой строке сетки

То есть при заполнении поля «Объединить колонки» колонки, начиная с той, что указана в поле «Колонка», будут объединены по горизонтали в одну.

Аналогично с полем «Объединять строки» - указанное количество строк, начиная с той, что указана в поле «Строка», будут объединены в одну:

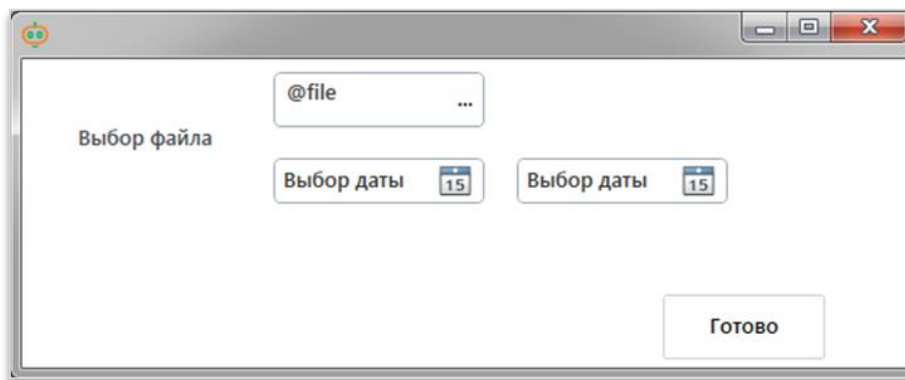


Рис. 8.23.15. Пример окна с объединенными строками в первой колонке сетки

Поле «Переменная»

В это поле вводится название переменной, в которую будет записано значение, введенное (для выпадающего списка – выбранное) пользователем. Является обязательным полем для всех типов контролов, кроме «Надписи».

После заполнения всех полей необходимо нажать на кнопку «Добавить». После нажатия на нее элемент сразу отобразится в окне предпросмотра в правой части окна.

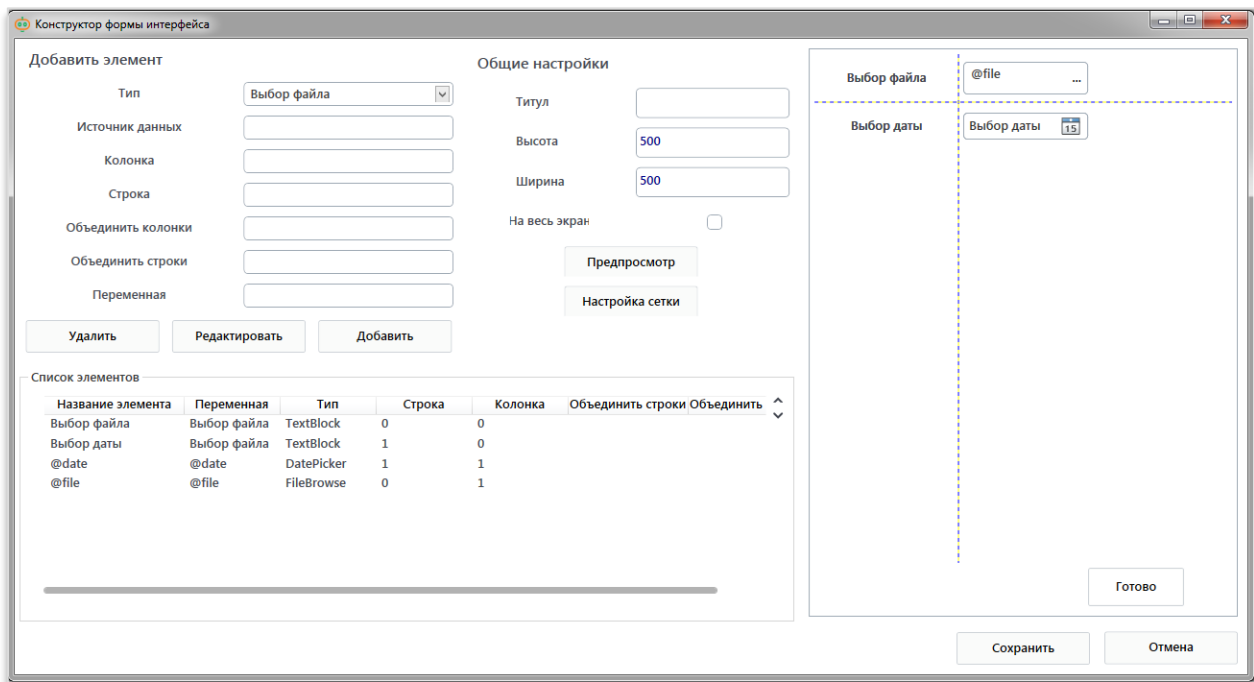


Рис. 8.23.16. Пример заполненного окна модуля «Интерфейс»

По кнопке «Предпросмотр» откроется диалоговое окно в таком виде, в котором оно будет представлено пользователю.

Удалить или отредактировать элемент можно, предварительно выбрав его в области «Список элементов», а затем нажав на соответствующую кнопку, находящуюся над списком с элементами.

После добавления всех необходимых контролов нужно нажать кнопку «Сохранить» в правой нижней части окна.

ЧАСТЬ IX. МОДУЛИ ДЛЯ РАБОТЫ С ТЕКСТОМ

Глава 24. Модуль «Обработка текста»

Модуль «Обработка текста» предлагает несколько методов сравнения текста. В некоторых задачах требуется знание о том, как сильно различен текст между эталонным и указанным. Это задачи, относящиеся к компьютерной лингвистике и искусственному интеллекту.

Интерфейс модуля

Окно модуля состоит из части составления команды, кнопок управления командами и списком команд в виде таблицы. Часть составления команды состоит из следующих полей: выпадающий список «Действие» с доступным набором методов анализа текста, два поля «Строка 1» и «Строка 2», предназначенные для ввода двух строк текста или переменных, содержащих текст, и поле «Переменная результат» - для названия переменной, в которую будет помещен результат работы модуля.

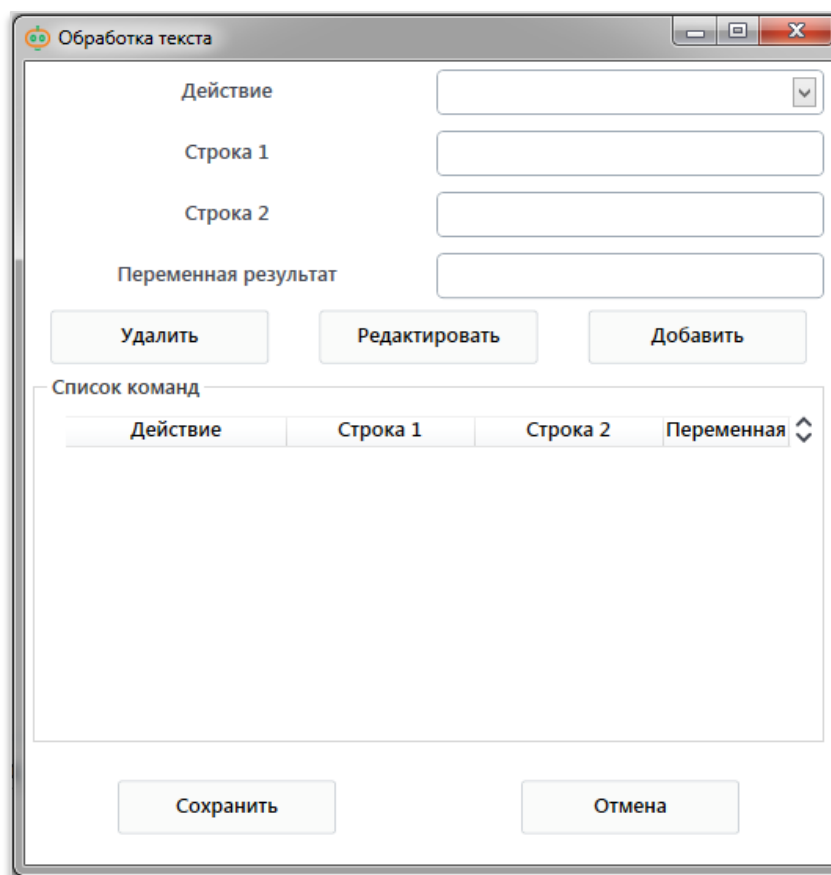


Рис. 9.24.1. Окно модуля

По кнопке «Добавить» созданная команда поместится в таблицу «Список команд». Для редактирования команды из списка нужно выбрать её в таблице, изменить необходимые поля

и нажать на кнопку «Редактировать». Для удаления команды нужно выбрать ее в списке и нажать на кнопку «Удалить». С помощью стрелочек в углу таблицы можно менять команды местами, аналогично тому, как это происходит в модуле Excel (Часть II).

Методы анализа текста

В модуле предложены следующие методы анализа текста:

1. **Расстояние Левенштейна** – рассчитывает разницу между двумя строками. Например – «Lexema RPA» и «Lexema SR» отличаются на 3 символа – слова «Lexema» совпадают полностью, остальные символы различны, то есть результат, записанный в переменную, будет равен 3;
2. **3-граммы** – метод, основанный на работе с n-граммами, в нашем случае $n=3$ – оценивается схожесть каждых 3 символов. Чем больше число (до 1), тем большую схожесть имеют строки. В примере «Lexema RPA» и «Lexema SR» результатом будет число 0,52.
3. **Сходство Джаро-Винклера** – мера схожести строк для измерения расстояния между двумя последовательностями символов. Чем меньше расстояние Джаро-Винклера для двух строк, тем больше сходства имеют эти строки друг с другом. Для примера «Lexema RPA» и «Lexema SR» результатом будет число 0,5.

Глава 25. Модуль «Распознавание текста»

Модуль «Распознавание текста» предназначен для считывания текста с указанного изображения и представления полученного текста в формате XML.

Интерфейс модуля

Модуль состоит из двух полей – «Путь к файлу» и «Переменная», и флажка «Скриншот».

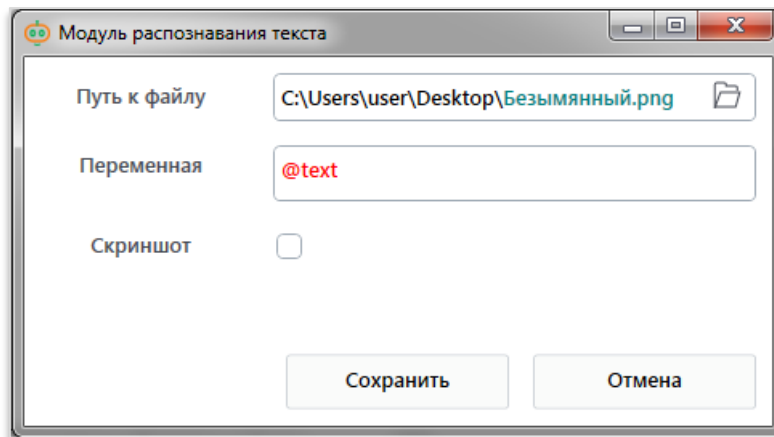


Рис. 9.25.1. Окно модуля

Поле «Путь к файлу». Заполняется путем до файла, который необходимо распознать. Это может быть картинка или PDF-документ.

Поле «Переменная». Заполняется названием переменной, начиная с символа «@», в которую будет помещен результат распознавания.

Флажок «Скриншот». Флажок ставится, когда необходимо распознать скриншот экрана в момент работы данного модуля в процессе выполнения работы. В таком случае поле «Путь к файлу» указывать не нужно.

ПРИМЕР.

Пусть имеется следующее изображение формата jpeg. Подадим его в модуль распознавания текста.

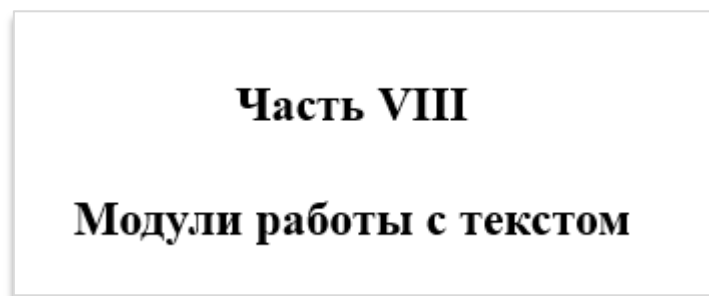


Рис. 10.34.2. Изображение, поданное на распознавание текста

Результат, записанный в переменную при распознавании изображения:

```
<OCRPages><Page><number>1</number><word><value>Часть</value><x>160</x><y>63</y></word><word><value>VIII</value><x>221</x><y>63</y></word><word><value>Модули</value><x>89</x><y>123</y></word><word><value>работы</value><x>177</x><y>123</y></word><word><value>с</value><x>229</x><y>123</y></word><word><value>текстом</value><x>283</x><y>123</y></word></Page></OCRPages>
```

Результат содержит информацию о страницах, координате и значении каждого слова изображения.

Глава 26. Модуль «Распознавание текста АБВУУ»

Модуль «Распознавание текста АБВУУ» работает, в основном, с PDF-файлами. В отличие от предыдущего модуля имеет более высокую точность и скорость работы.

Результат распознавания записывается в переменную-объект, в которой можно обратиться к конкретной строке, таблице, слову.

Интерфейс модуля

Данный модуль состоит из двух полей – «Путь к PDF-файлу» и «Переменная».

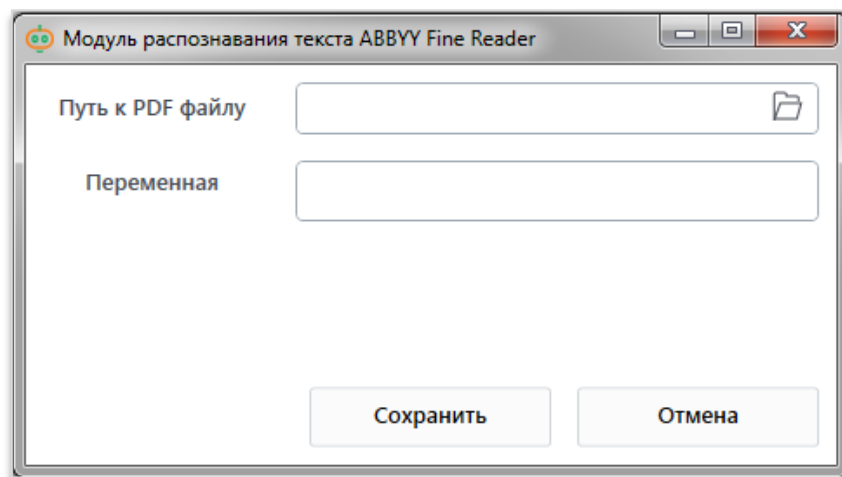


Рис. 9.26.1. Окно модуля

Поле «Путь к PDF файлу». В поле указывается путь до распознаваемого файла/картинки.

Поле «Переменная». В поле вводится имя переменной, начиная с символа «@», в которую будет занесен результат распознавания.

PDF файл может состоять из нескольких страниц, поэтому переменная будет являться списком страниц. Например, @text[0] – первая распознанная страница файла.

Весь распознанный текст делится на две части – основной текст и табличная часть.

Основной текст

Для обращения к тексту, необходимо обращаться к полю `rows_word[0]` (в данной записи [0] означает, что идет обращение к первой текстовой части, в других версиях АБВУУ возможно использование большего разбиения текста). Затем указывается, как отдельное поле, номер строки и, через точку, номер слова в строке - `@text[0].rows_word[0].2.4`. Затем, после выбора нужного слова, можно получить его значение, обратившись к полю `value` - `@text[0].rows_word[0].2.4.value` – такая строка выдаст значение слова, находящегося на первой странице распознанного документа, во второй строке и является четвертой слева.

Табличная часть

Для обращения к таблице, необходимо обращаться к полю **tables[0]** ([0] нужен для корректной работы, его смысловая часть заложена в продукте АВВУУ). Затем отдельным полем вводится номер таблицы, так как на странице их может быть несколько - **@text[0].tables[0].0**. После этого вводится номер ячейки в таблице в виде одного числа. Если вы хотите найти ячейку и знаете её номер колонки и строки, то можно получить индекс рассматриваемой ячейки переменной в виде «строка,колонка», обратившись к полю **index** - **@text[0].tables[0].0.5.index** (5 – пятая ячейка в массиве распознанных ячеек). Для получения значения из ячейки нужно обратиться к её значению, а затем либо собрать всю строку, записанную в ячейку, либо обратиться к конкретному слову по его порядковому номеру - **@text[0].tables[0].0.5.value.1.value** – значение второго слова из пятой ячейки первой таблицы с первой страницы распознанного текста.

Для получения количества ячеек в таблице используйте следующую конструкцию - **@text[0].tables[0].0[%]**, то есть указывайте набор символов «[%]» после того поля, после которого идет поле, количество элементов которого необходимо узнать.

Для получения количества слов в ячейке используйте **@text[0].tables[0].0.5[%]**.

ЧАСТЬ X. ДРУГИЕ МОДУЛИ

Глава 27. Модуль «Вычисления»

Модуль «Вычисления» является связующим модулем или модулем пред- и постобработки данных из других модулей. Данный модуль необходим для создания, вычисления и изменения значений переменных.

Интерфейс модуля

Окно модуля очень простое и состоит всего лишь из поля для ввода и кнопок «Сохранить» и «Отмена».

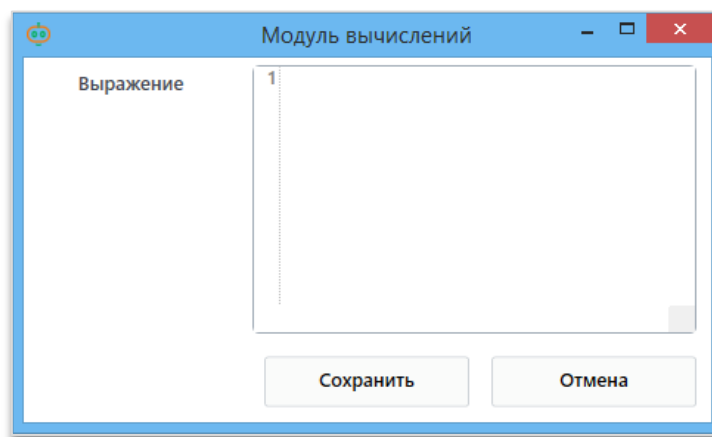


Рис. 10.27.1. Окно модуля

Поле «Выражение». В поле построчно вводятся операции, которые необходимо провести с переменными. Это может быть сложение, вычитание, умножение или вычисление функции.

Поддерживаемые операции

- + (сложение);
- - (вычитание);
- * (умножение);
- / (деление);
- % (остаток от деления);
- \ (целое от деления);
- ^ (возведение в степень).

Все строки должны быть вида «Переменная = Операция;», операция в свою очередь может состоять как из одиночного числа или строки, так и из нескольких операндов, связанных вышеуказанными операторами, либо быть функцией.

Каждая строка должна быть отделена от следующей символом «;».

Пример работы с модулем

Предположим, необходимо получить последний день предыдущего месяца. Сделаем это с помощью ввода следующего кода:

```
@now = #GetDate();
@day = #GetDay(@now);
@day = @day + 1;
@lastDay = #DateAdd(day, -@day,
```

Листинг 10.26.1. Получение последнего дня предыдущего месяца

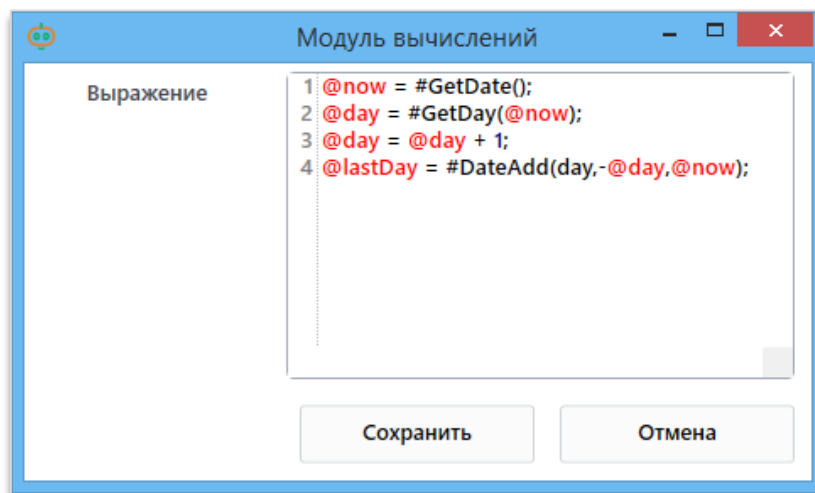


Рис. 10.26.2. Пример заполнения модуля

Что делает написанный выше код? А вот что – получаем сегодняшний день, например, сегодня 25 марта, вычитаем из полученной даты на один день больше, чем на самом деле есть, то есть 26 дней и оказываемся в последнем дне предыдущего месяца – 29 февраля (если год високосный). Более подробно:

1 строка – использование функции GetDate, которая возвращает сегодняшнюю дату, и сохранение её результата в переменную @now (@now = 25.03.2020);

2 строка – использование функции GetDay, которая принимает один параметр – дату, и возвращает из поданной в неё даты только день. Подадим в функцию в качестве параметра сегодняшнюю дату, которая лежит в переменной @now. После этого запишем результат работы функции в переменную @day (@day = 25);

3 строка – увеличиваем переменную @day на единицу (@day = 26);

4 строка – использование функции DateAdd. Функция принимает три параметра: 1) day, month или year – компонент даты, с которым работаем, 2) – количество, которое прибавляется к выбранному компоненту даты, 3) – дата, с которой проводятся манипуляции. Вычтем из сегодняшней даты 26 дней – используем DateAdd с параметрами:

1. day – вычитаем дни;
2. -@day – так как функция по умолчанию складывает, добавим знак минус;
3. @now – сегодняшняя дата.

В итоге получается, что мы вычитаем из 25.03.2020 26 дней и оказываемся в 29.02.2020.

Глава 28. Модуль «Запустить процесс»

Модуль «Запустить процесс» предназначен для запуска каких-либо приложений. Это может быть браузер, MS Word, 1С и другие.

Интерфейс модуля

Окно модуля состоит из полей «Путь к файлу», «Параметры», флажок «От имени администратора», «Пауза», «Переменная с именем процесса» и «Переменная с id процесса», а также кнопок «Сохранить» и «Отмена».

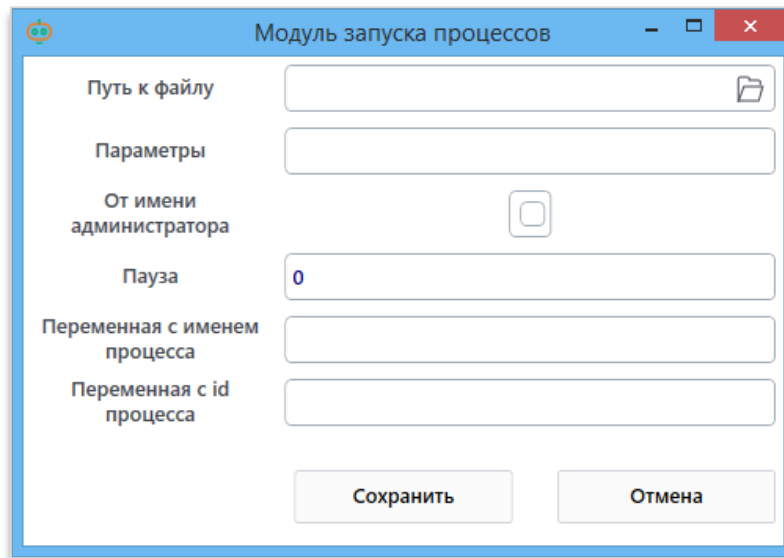


Рис. 10.28.1. Окно модуля

Поле «Путь к файлу». В данное поле вводится название стандартного процесса Windows (например, notepad – блокнот), либо путь до корневого расположения программы, например, стандартный путь до браузера Google Chrome - C:\Program Files (x86)\Google\Chrome\Application\chrome.exe.

Поле «Параметры» является необязательным полем, в него вводятся аргументы вызываемого приложения. Например, для открытия нового блокнота или чистого листа Excel-аргументы не нужны, но если необходимо открыть конкретный файл, то в аргументы передается название файла, а если открывается браузер, то аргументом может быть ссылка на сайт.

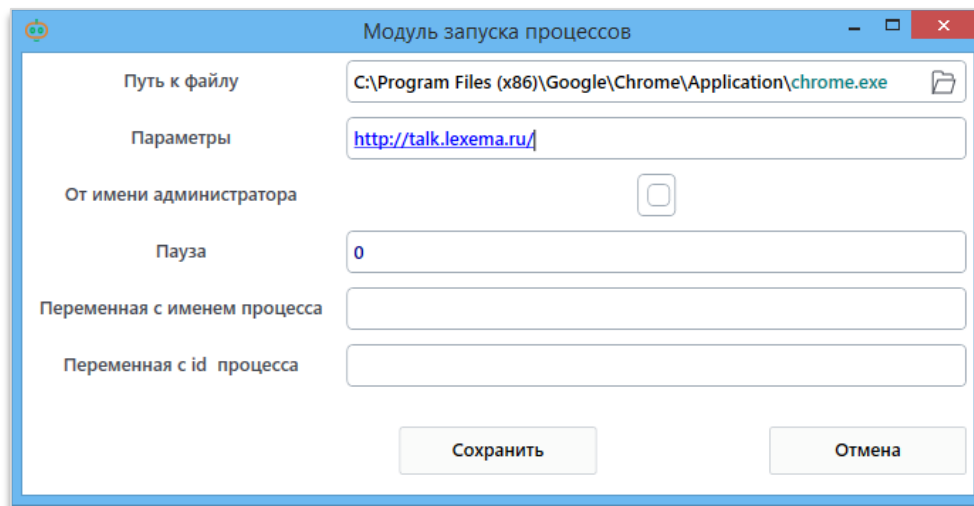


Рис. 10.28.2. Пример заполнения

Флажок «От имени администратора» запускает процесс с правами, позволяющими запускаемому процессу вносить изменения в системные файлы компьютера. *Если вы не доверяете приложению, то не стоит ставить этот флажок.*

Поле «Пауза» устанавливает время ожидания после начала запуска процесса.

Поле «Переменная с именем/id процесса». В данное поле вводятся переменные, в которые будут помещены имя и идентификатор (id) запускаемого процесса соответственно. Данные параметры нужны, чтобы в дальнейшем можно было завершить (закрыть) указанные процессы. *Если у вас задача только запустить процессы, то можете оставить эти поля пустыми.*

Глава 29. Модуль «Завершить процесс»

Модуль «Завершить процесс» предназначен для завершения ранее запущенных процессов по их имени и идентификатору.

Интерфейс модуля

Окно состоит из двух полей – имени и идентификатора (id) завершаемого процесса.

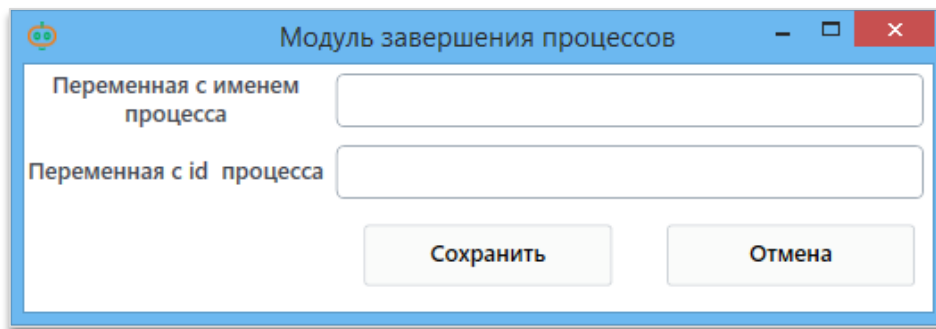


Рис. 10.29.1. Окно модуля

Если завершаемый процесс запускался через модуль «Запустить процесс», то в эти поля передаются переменные, указанные в одноименных полях окна запуска процессов. В противном случае, если известно имя процесса, который необходимо завершить, достаточно ввести только его, но (!) в таком случае **завершатся все процессы с подобным именем**. Например, если вы захотите закрыть excel-файл и напишете просто «Excel», то закроются все excel-файлы, которые были открыты.

Глава 30. Модуль «Код С#»

Модуль «Код С#» предназначен для использования сложных вычислений, отсутствующих в студии.

В модуле допускается использование переменных студии.

Интерфейс модуля

Данный модуль состоит из двух полей: «Переменная» и «Код».

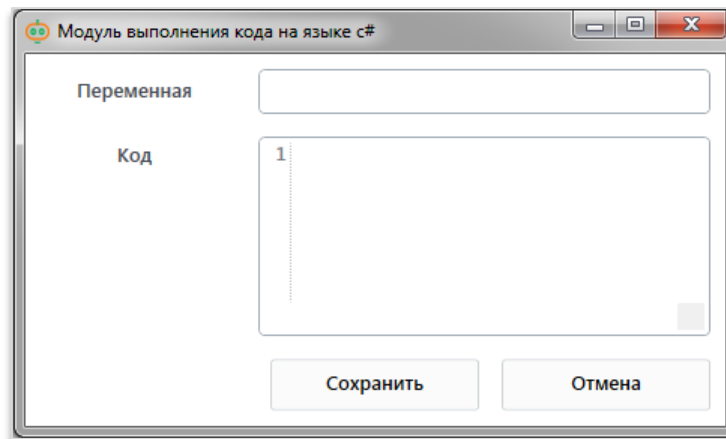


Рис. 10.30.1. Окно модуля

Поле «Переменная». В поле указывается переменная, в которую будет записан результат.

Поле «Код». В поле необходимо ввести часть кода в соответствии с синтаксисом языка С# без описания используемых библиотек, как если бы это была функция внутри существующего проекта, у которой главное условие – *она обязательно должна что-то возвращать*, то есть иметь слово *«return»*.

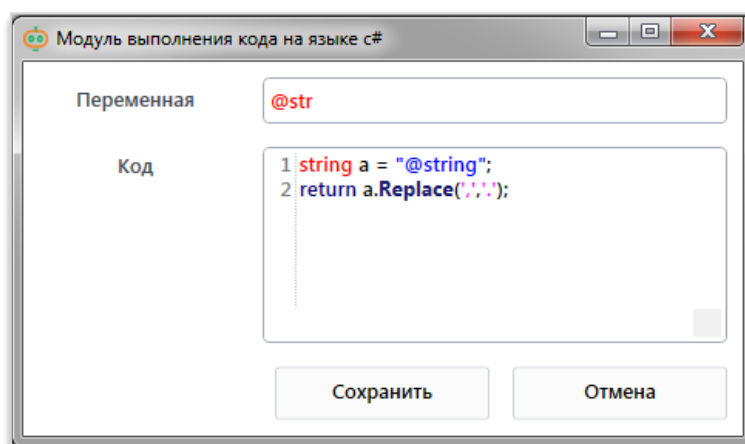


Рис. 10.30.2. Пример заполнения

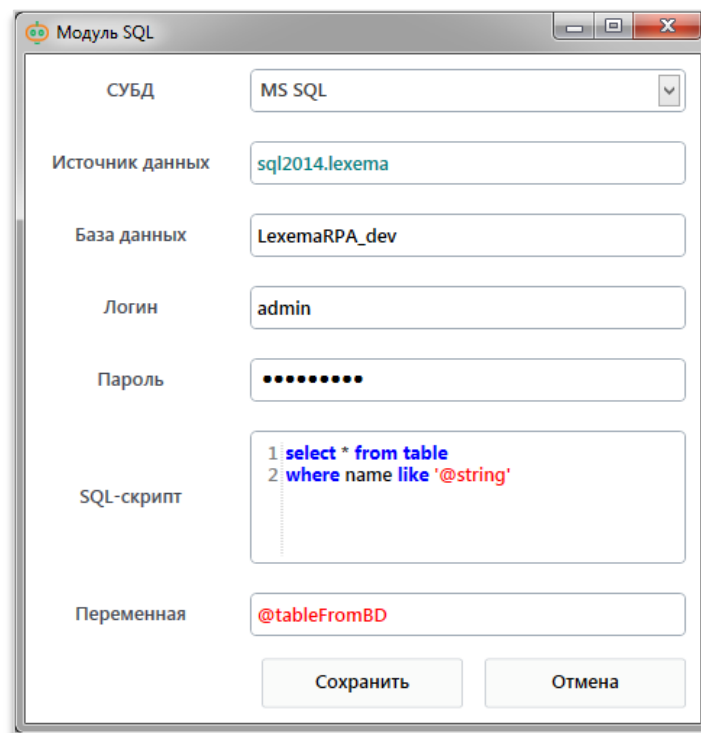
Глава 31. Модуль «SQL»

Модуль «SQL» обеспечивает работу с базами данных. Поддерживаемые СУБД: MS SQL и PostgreSQL.

Интерфейс модуля

Данный модуль обеспечивает работу с базами данных. Поддерживаются такие СУБД, как MS SQL и PostgreSQL. Для работы с БД **необходимо заполнить все поля** окна «Модуль SQL».

Окно состоит из следующих полей: «СУБД», «Источник данных», «База данных», «Логин», «Пароль», «SQL-скрипт» и «Переменная».



Модуль SQL

СУБД	MS SQL
Источник данных	sql2014.lexema
База данных	LexemaRPA_dev
Логин	admin
Пароль	••••••••
SQL-скрипт	1 select * from table 2 where name like '@string'
Переменная	@tableFromBD

Сохранить Отмена

Рис. 10.31.1. Пример заполнения модуля

Поля «СУБД», «Источник данных», «База данных», «Логин» и «Пароль» обеспечивают подключение к БД.

Поле «SQL-скрипт». В данное поле вводится скрипт на языке SQL, поддерживается ввод переменных студии (как показано на рис. 10.31.1).

Поле «Переменная». В поле вводится название переменной, в которую будет записан ответ от базы данных в виде таблицы. *В одном SQL-модуле может возвращаться только одна таблица, последующие будут проигнорированы.*

Глава 32. Модуль «HTTP_GET»

Http-запросы – запросы для передачи информации между сайтами (или серверами) и клиентами (например, нами).

Модуль «HTTP_GET» предназначен для отправки Get-запросов по протоколу *http*.⁹

Для работы с полученными ответами можно использовать модуль «Парсинг JSON» (если, конечно, запрашиваемый формат данных был таковым).

Интерфейс модуля

Данный модуль состоит из полей: «Ссылка», «Логин», «Пароль» и «Переменная».

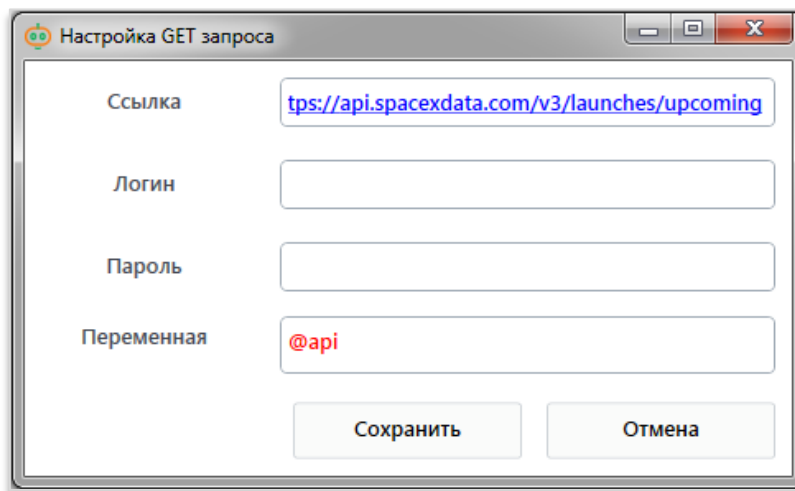


Рис. 10.32.1. Окно GET запроса

Поле «Ссылка». В поле вводится необходимый URL-адрес.

Поле «Переменная». В поле вводится название переменной, в которую будет помещен ответ.

Поля «Логин» и «Пароль». Предназначены для ввода логина и пароля, так как некоторые сайты используют защиту своей информации и запрашивают авторизацию.

⁹ HTTP (англ. HyperText Transfer Protocol — «протокол передачи гипертекста») — протокол прикладного уровня передачи данных изначально — в виде гипертекстовых документов в формате «HTML», в настоящий момент используется для передачи произвольных данных.

Глава 33. Модуль «HTTP_POST»

Модуль «HTTP_POST» предназначен для отправки Post-запросов по протоколу *http*.

Интерфейс модуля

Форма для отправки Post-запроса аналогична форме для отправки Get-запроса, с одним лишь отличием - присутствует поле для ввода самого запроса.

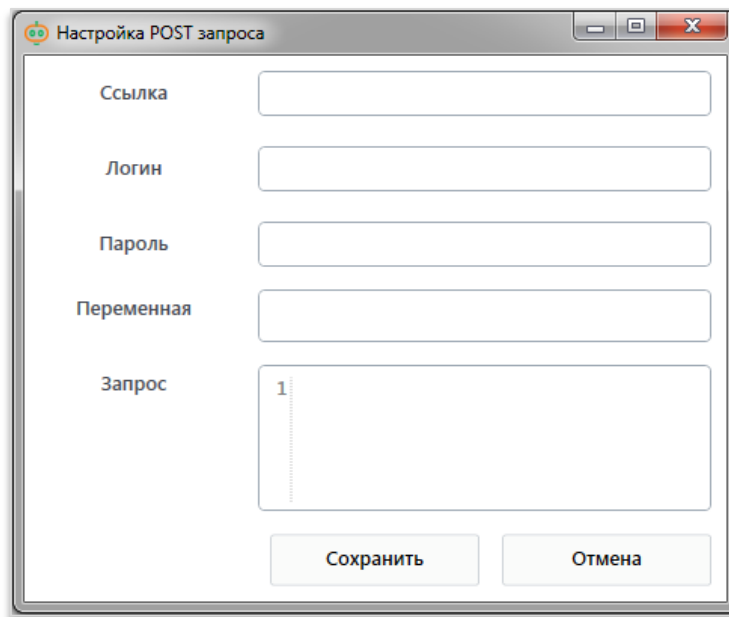


Рис. 10.33.1. Окно POST запроса

Поле «Ссылка». В поле вводится необходимый URL-адрес.

Поле «Переменная». В поле вводится название переменной, в которую будет помещен ответ.

Поля «Логин» и «Пароль». Предназначены для ввода логина и пароля, так как некоторые сайты используют защиту своей информации и запрашивают авторизацию.

Поле «Запрос». В поле вводится HTTP-сообщение, согласно общим правилам его структуры.

Глава 34. Модуль «Парсинг JSON»

Данный модуль предназначен для обработки данных в формате JSON – парсинга JSON-объектов.

Интерфейс модуля

В окне модуля расположены поля для ввода значений: «Переменная» и «JSON», и кнопки «Сохранить» и «Отмена».

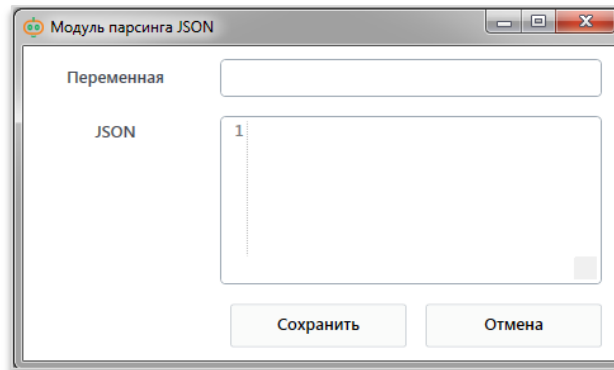


Рис. 10.34.1. Окно модуля

Поле «Переменная». В поле «Переменная» вводится название переменной, которая будет преобразована в объектную переменную с полями, имеющимися в JSON-объекте.

Поле «JSON». В поле необходимо описать объект в формате JSON, в виде пар «ключ»: «значение», обрамлённых в фигурные скобки. JSON-объект может быть описан непосредственно текстом в поле «JSON» либо быть созданным в процессе выполнения других модулей. В случае формирования JSON-объекта в других модулях, в поле необходимо ввести название переменной, в которой лежит необходимый объект.

ПРИМЕР.

Пусть у нас есть текст в формате JSON. Введем его в поле JSON модуля, а результат пусть запишется в переменную @a.

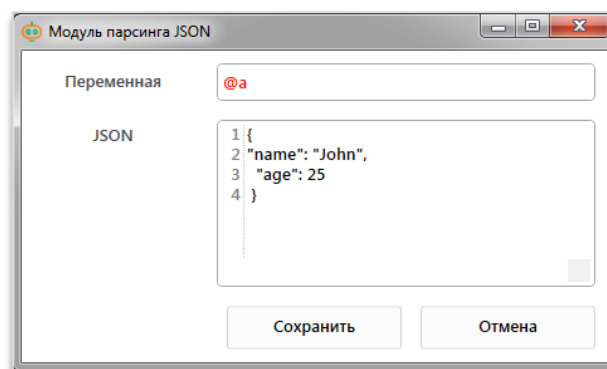



Рис. 10.34.2. Пример заполнения

В результате переменная @a станет объектом с полями «name» и «age». К полям объекта можно обратиться, написав название поля через точку после переменной. Проверить значение переменной можно с помощью функционала «Узнать значение переменной», доступного по кнопке  в обозревателе решений.

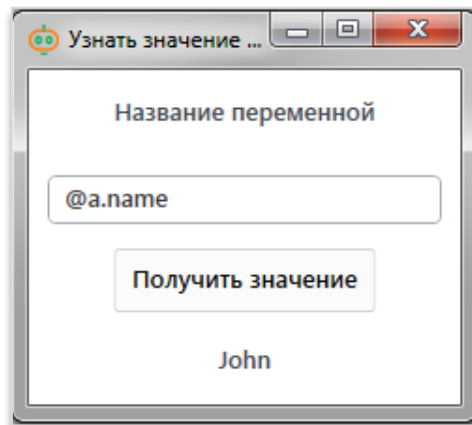


Рис. 10.34.3. Получение значения из поля «name» переменной «a»

На рис. 10.34.3 видно, что на основе текста формата JSON создавалась переменная с указанными в нем полями и заполненными значениями.

Глава 35. Модуль «Пауза»

Модуль «Пауза» предназначен для создания некоторой паузы между модулями.

Вся настройка модуля заключается в заполнении поля «Пауза» - количество секунд, которые необходимо подождать. Допускается использование переменной.

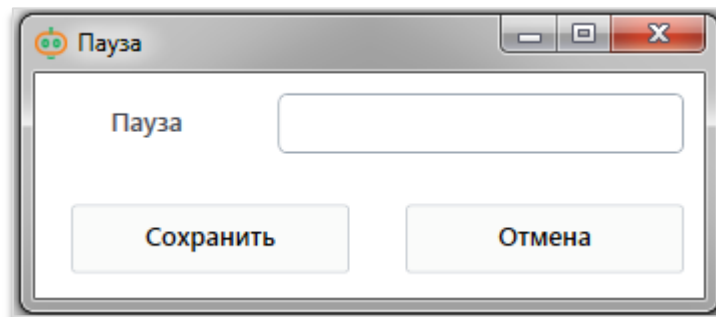


Рис. 10.35.1. Окно модуля

Глава 36. Модуль «Прокси»

Модуль «Прокси» предназначен для создания подключения к прокси-серверу. Для использования модуля необходимо заполнить адрес прокси-сервера.

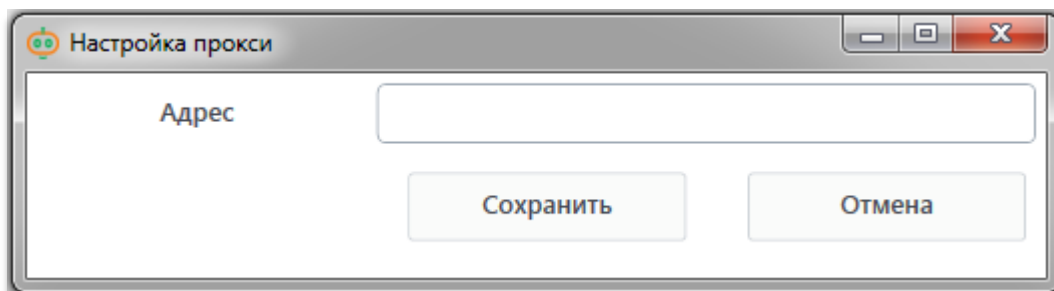


Рис. 10.36.1. Окно модуля