

ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»

**Программное обеспечение
МойОфис Комплект Средств Разработки (SDK)**

3.6

**Информация о новой версии программного обеспечения
(Release notes)**

На 13 листах

Версия документа: 1

Дата публикации: 25.11.2025

**Москва
2025**



Все упомянутые в этом документе названия продуктов, логотипы, торговые марки и товарные знаки принадлежат их владельцам.

Товарные знаки «МойОфис» и «MyOffice» принадлежат ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ».

Ни при каких обстоятельствах нельзя истолковывать любое содержимое настоящего документа как прямое или косвенное предоставление лицензии или права на использование товарных знаков, логотипов или знаков обслуживания, приведенных в нем. Любое несанкционированное использование этих товарных знаков, логотипов или знаков обслуживания без письменного разрешения их правообладателя строго запрещено.



СОДЕРЖАНИЕ

1 Надстройки	4
1.1 Условное форматирование	4
1.2 Работа с ячейками	5
1.3 Работа с фильтрами	5
1.4 Интеграция Python скриптов	5
2 Макрокоманды Lua	6
2.1 Условное форматирование	6
2.2 Работа с ячейками	7
2.3 Работа с фильтрами	7
2.4 VBA интерпретатор (БЕТА)	7
3 Document API C++	8
3.1 Условное форматирование	8
3.2 Работа с ячейками	9
3.3 Работа с фильтрами	9
3.4 Генерация HTML документов	9
4 Document API C#	10
4.1 Условное форматирование	10
4.2 Работа с ячейками	11
4.3 Работа с фильтрами	11
4.4 Генерация HTML документов	11
5 Document API Python	12
5.1 Условное форматирование	12
5.2 Работа с ячейками	13
5.3 Работа с фильтрами	13
5.4 Генерация HTML документов	13



1 НАДСТРОЙКИ

1.1 Условное форматирование

Добавлена возможность взаимодействия с правилами условного форматирования в табличном документе. Теперь вы можете создавать, получать, редактировать и удалять правила следующих типов: Обычное выделение, Цветовая шкала, Гистограмма и Значки.

Новые методы `Document:getConditionalFormatRules` и `Table:getConditionalFormatRules` позволяют получить коллекции правил условного форматирования для всего документа (`ConditionalFormatDocumentRules`) или отдельного листа (`ConditionalFormatTableRules`). Элементы этих коллекций представлены объектами `ConditionalFormatRuleProxy`.

Вы можете удалить все правила в документе (`ConditionalFormatDocumentRules:removeAllRules`), все правила на листе (`ConditionalFormatTableRules:removeAllRules`), правила из определенного диапазона (`ConditionalFormatTableRules:removeRulesFromRange`) или определенное правило (`ConditionalFormatRuleProxy:remove`).

Для добавления нового правила условного форматирования используется метод `ConditionalFormatTableRules:addRule`, который принимает объект `ConditionalFormatRule`. Данный объект должен содержать следующие настройки:

- Диапазон ячеек, к которому применяется правило (`CellRangePosition` или `CellRangePositions`).
- Оператор условного форматирования, который определяет тип правила и настройки применения форматирования.
- Настройки форматирования ячеек `ConditionalFormatCellStyle` (только для правил с обычным выделением).

Доступны следующие операторы условного форматирования:

- `AboveAverageConditionalFormatOperator` – Выше и ниже среднего
- `BinaryConditionalFormatOperator` – Между и не между
- `ColorScaleConditionalFormatOperator` – Цветовая шкала
- `DataBarConditionalFormatOperator` – Гистограмма
- `IconSetConditionalFormatOperator` – Значки
- `NullaryConditionalFormatOperator` – Оператор без аргументов
- `TextConditionalFormatOperator` – Оператор для текстовых значений
- `TopBottomConditionalFormatOperator` – Наибольшие и наименьшие значения



- `UnaryConditionalFormatOperator` – Больше, меньше, равно, не равно, формулы
- `UniquenessConditionalFormatOperator` – Уникальные и повторяющиеся значения

1.2 Работа с ячейками

- Добавлен метод `CellRange:textToColumns`, который позволяет разделить текст в ячейках диапазона на отдельные горизонтальные ячейки. Используйте новый класс `TextToColumnsSettings` для задания параметров разделения и символов-разделителей.
- Новый метод `CellRange:setArrayFormula` позволяет поместить формулу массива в диапазон ячеек.

1.3 Работа с фильтрами

- Теперь вы можете получить фильтры, которые находятся в диапазоне фильтрации, с помощью метода `FiltersRange:getFilters`.
- Новый метод `TableFilters:getFilterType` позволяет тип фильтра, установленного в столбце: по значению (`ValuesTableFilter`) или по условию (`ConditionalTableFilter`).
- Добавлены методы `TableFilters:getAsValueFilter` и `TableFilters:getAsConditionalFilter` для получения фильтров по значению и по условию из коллекции.
- Для удаления значения из фильтра, добавлен метод `ValuesTableFilter:remove`.

1.4 Интеграция Python скриптов

Добавлена возможность интегрировать и вызывать функции, написанные на языке Python, непосредственно из Lua кода надстроек. Для этого используется Lua модуль **pyxm**, который поставляется вместе с новой версией редакторов МойОфис. Данный модуль позволяет вызывать Python функции, передавать параметры и получать результаты обратно в Lua код.



2 МАКРОКОМАНДЫ LUA

2.1 Условное форматирование

Добавлена возможность взаимодействия с правилами условного форматирования в табличном документе. Теперь вы можете создавать, получать, редактировать и удалять правила следующих типов: Обычное выделение, Цветовая шкала, Гистограмма и Значки.

Новые методы Document:getConditionalFormatRules и Table:getConditionalFormatRules позволяют получить коллекции правил условного форматирования для всего документа (ConditionalFormatDocumentRules) или отдельного листа (ConditionalFormatTableRules). Элементы этих коллекций представлены объектами ConditionalFormatRuleProxy.

Вы можете удалить все правила в документе (ConditionalFormatDocumentRules:removeAllRules), все правила на листе (ConditionalFormatTableRules:removeAllRules), правила из определенного диапазона (ConditionalFormatTableRules:removeRulesFromRange) или определенное правило (ConditionalFormatRuleProxy:remove).

Для добавления нового правила условного форматирования используется метод ConditionalFormatTableRules:addRule, который принимает объект ConditionalFormatRule. Данный объект должен содержать следующие настройки:

- Диапазон ячеек, к которому применяется правило (CellRangePosition или CellRangePositions).
- Оператор условного форматирования, который определяет тип правила и настройки применения форматирования.
- Настройки форматирования ячеек ConditionalFormatCellStyle (только для правил с обычным выделением).

Доступны следующие операторы условного форматирования:

- AboveAverageConditionalFormatOperator – Выше и ниже среднего
- BinaryConditionalFormatOperator – Между и не между
- ColorScaleConditionalFormatOperator – Цветовая шкала
- DataBarConditionalFormatOperator – Гистограмма
- IconSetConditionalFormatOperator – Значки
- NullaryConditionalFormatOperator – Оператор без аргументов
- TextConditionalFormatOperator – Оператор для текстовых значений
- TopBottomConditionalFormatOperator – Наибольшие и наименьшие значения



- `UnaryConditionalFormatOperator` – Больше, меньше, равно, не равно, формулы
- `UniquenessConditionalFormatOperator` – Уникальные и повторяющиеся значения

2.2 Работа с ячейками

- Добавлен метод `CellRange:textToColumns`, который позволяет разделить текст в ячейках диапазона на отдельные горизонтальные ячейки. Используйте новый класс `TextToColumnsSettings` для задания параметров разделения и символов-разделителей.
- Новый метод `CellRange:setArrayFormula` позволяет поместить формулу массива в диапазон ячеек.

2.3 Работа с фильтрами

- Теперь вы можете получить фильтры, которые находятся в диапазоне фильтрации, с помощью метода `FiltersRange:getFilters`.
- Новый метод `TableFilters:getFilterType` позволяет тип фильтра, установленного в столбце: по значению (`ValuesTableFilter`) или по условию (`ConditionalTableFilter`).
- Добавлены методы `TableFilters:getAsValueFilter` и `TableFilters:getAsConditionalFilter` для получения фильтров по значению и по условию из коллекции.
- Для удаления значения из фильтра, добавлен метод `ValuesTableFilter:remove`.

2.4 VBA интерпретатор (БЕТА)

В редактор МояТаблица добавлена бета-версия VBA интерпретатора, который поможет вам конвертировать код существующих в ваших документах VBA макрокоманд на язык программирования Lua.



3 DOCUMENT API C++

3.1 Условное форматирование

Добавлена возможность взаимодействия с правилами условного форматирования в табличном документе. Теперь вы можете создавать, получать, редактировать и удалять правила следующих типов: Обычное выделение, Цветовая шкала, Гистограмма и Значки.

Новые методы `Document::getConditionalFormatRules` и `Table::getConditionalFormatRules` позволяют получить коллекции правил условного форматирования для всего документа (`ConditionalFormatDocumentRules`) или отдельного листа (`ConditionalFormatTableRules`). Элементы этих коллекций представлены объектами класса `ConditionalFormatRuleProxy`.

Вы можете удалить все правила в документе (`ConditionalFormatDocumentRules::removeAllRules`), все правила на листе (`ConditionalFormatTableRules::removeAllRules`), правила из определенного диапазона (`ConditionalFormatTableRules::removeRulesFromRange`) или определенное правило (`ConditionalFormatRuleProxy::remove`).

Для добавления нового правила условного форматирования используется метод `ConditionalFormatTableRules::addRule`, который принимает объект `ConditionalFormatRule`. Данный объект должен содержать следующие настройки:

- Диапазон ячеек, к которому применяется правило (`CellRangePosition` или `CellRangePositions`).
- Оператор условного форматирования, который определяет тип правила и настройки применения форматирования.
- Настройки форматирования ячеек `ConditionalFormatCellStyle` (только для правил с обычным выделением).

Доступны следующие операторы условного форматирования:

- `AboveAverageConditionalFormatOperator` – Выше и ниже среднего
- `BinaryConditionalFormatOperator` – Между и не между
- `ColorScaleConditionalFormatOperator` – Цветовая шкала
- `DataBarConditionalFormatOperator` – Гистограмма
- `IconSetConditionalFormatOperator` – Значки
- `NullaryConditionalFormatOperator` – Оператор без аргументов
- `TextConditionalFormatOperator` – Оператор для текстовых значений
- `TopBottomConditionalFormatOperator` – Наибольшие и наименьшие значения



- `UnaryConditionalFormatOperator` – Больше, меньше, равно, не равно, формулы
- `UniquenessConditionalFormatOperator` – Уникальные и повторяющиеся значения

3.2 Работа с ячейками

- Добавлен метод `CellRange::textToColumns`, который позволяет разделить текст в ячейках диапазона на отдельные горизонтальные ячейки. Используйте новый класс `TextToColumnsSettings` для задания параметров разделения и символов-разделителей.
- Новый метод `CellRange::setArrayFormula` позволяет поместить формулу массива в диапазон ячеек.

3.3 Работа с фильтрами

- Теперь вы можете получить фильтры, которые находятся в диапазоне фильтрации, с помощью метода `FiltersRange::getFilters`.
- Новый метод `TableFilters::getFilterType` позволяет тип фильтра, установленного в столбце: по значению (`ValuesTableFilter`) или по условию (`ConditionalTableFilter`).
- Добавлены методы `TableFilters::getAsValueFilter` и `TableFilters::getAsConditionalFilter` для получения фильтров по значению и по условию из коллекции.
- Для удаления значения из фильтра, добавлен метод `ValuesTableFilter::remove`.

3.4 Генерация HTML документов

Добавлен глобальный метод `exportWorksheetToHtml`, который позволяет сгенерировать HTML документ на основе листа табличного документа. Новый класс `WorksheetHtmlExportSettings` содержит настройки генерации HTML. В текущем релизе поддерживается только генерация данных и форматирования ячеек.



4 DOCUMENT API C#

4.1 Условное форматирование

Добавлена возможность взаимодействия с правилами условного форматирования в табличном документе. Теперь вы можете создавать, получать, редактировать и удалять правила следующих типов: Обычное выделение, Цветовая шкала, Гистограмма и Значки.

Новые методы `Document.getConditionalFormatRules` и `Table.getConditionalFormatRules` позволяют получить коллекции правил условного форматирования для всего документа (`ConditionalFormatDocumentRules`) или отдельного листа (`ConditionalFormatTableRules`). Элементы этих коллекций представлены объектами класса `ConditionalFormatRuleProxy`.

Вы можете удалить все правила в документе (`ConditionalFormatDocumentRules.removeAllRules`), все правила на листе (`ConditionalFormatTableRules.removeAllRules`), правила из определенного диапазона (`ConditionalFormatTableRules.removeRulesFromRange`) или определенное правило (`ConditionalFormatRuleProxy.remove`).

Для добавления нового правила условного форматирования используется метод `ConditionalFormatTableRules.addRule`, который принимает объект `ConditionalFormatRule`. Данный объект должен содержать следующие настройки:

- Диапазон ячеек, к которому применяется правило (`CellRangePosition` или `CellRangePositions`).
- Оператор условного форматирования, который определяет тип правила и настройки применения форматирования.
- Настройки форматирования ячеек `ConditionalFormatCellStyle` (только для правил с обычным выделением).

Доступны следующие операторы условного форматирования:

- `AboveAverageConditionalFormatOperator` – Выше и ниже среднего
- `BinaryConditionalFormatOperator` – Между и не между
- `ColorScaleConditionalFormatOperator` – Цветовая шкала
- `DataBarConditionalFormatOperator` – Гистограмма
- `IconSetConditionalFormatOperator` – Значки
- `NullaryConditionalFormatOperator` – Оператор без аргументов
- `TextConditionalFormatOperator` – Оператор для текстовых значений
- `TopBottomConditionalFormatOperator` – Наибольшие и наименьшие значения



- `UnaryConditionalFormatOperator` – Больше, меньше, равно, не равно, формулы
- `UniquenessConditionalFormatOperator` – Уникальные и повторяющиеся значения

4.2 Работа с ячейками

- Добавлен метод `CellRange.textToColumns`, который позволяет разделить текст в ячейках диапазона на отдельные горизонтальные ячейки. Используйте новый класс `TextToColumnsSettings` для задания параметров разделения и символов-разделителей.
- Новый метод `CellRange.setArrayFormula` позволяет поместить формулу массива в диапазон ячеек.

4.3 Работа с фильтрами

- Теперь вы можете получить фильтры, которые находятся в диапазоне фильтрации, с помощью метода `FiltersRange.getFilters`.
- Новый метод `TableFilters.getFilterType` позволяет тип фильтра, установленного в столбце: по значению (`ValuesTableFilter`) или по условию (`ConditionalTableFilter`).
- Добавлены методы `TableFilters.getAsValueFilter` и `TableFilters.getAsConditionalFilter` для получения фильтров по значению и по условию из коллекции.
- Для удаления значения из фильтра, добавлен метод `ValuesTableFilter.remove`.

4.4 Генерация HTML документов

Добавлен глобальный метод `exportWorksheetToHtml`, который позволяет сгенерировать HTML документ на основе листа табличного документа. Новый класс `WorksheetHtmlExportSettings` содержит настройки генерации HTML. В текущем релизе поддерживается только генерация данных и форматирования ячеек.



5 DOCUMENT API PYTHON

5.1 Условное форматирование

Добавлена возможность взаимодействия с правилами условного форматирования в табличном документе. Теперь вы можете создавать, получать, редактировать и удалять правила следующих типов: Обычное выделение, Цветовая шкала, Гистограмма и Значки.

Новые методы `Document.getConditionalFormatRules` и `Table.getConditionalFormatRules` позволяют получить коллекции правил условного форматирования для всего документа (`ConditionalFormatDocumentRules`) или отдельного листа (`ConditionalFormatTableRules`). Элементы этих коллекций представлены объектами класса `ConditionalFormatRuleProxy`.

Вы можете удалить все правила в документе (`ConditionalFormatDocumentRules.removeAllRules`), все правила на листе (`ConditionalFormatTableRules.removeAllRules`), правила из определенного диапазона (`ConditionalFormatTableRules.removeRulesFromRange`) или определенное правило (`ConditionalFormatRuleProxy.remove`).

Для добавления нового правила условного форматирования используется метод `ConditionalFormatTableRules.addRule`, который принимает объект `ConditionalFormatRule`. Данный объект должен содержать следующие настройки:

- Диапазон ячеек, к которому применяется правило (`CellRangePosition` или `CellRangePositions`).
- Оператор условного форматирования, который определяет тип правила и настройки применения форматирования.
- Настройки форматирования ячеек `ConditionalFormatCellStyle` (только для правил с обычным выделением).

Доступны следующие операторы условного форматирования:

- `AboveAverageConditionalFormatOperator` – Выше и ниже среднего
- `BinaryConditionalFormatOperator` – Между и не между
- `ColorScaleConditionalFormatOperator` – Цветовая шкала
- `DataBarConditionalFormatOperator` – Гистограмма
- `IconSetConditionalFormatOperator` – Значки
- `NullaryConditionalFormatOperator` – Оператор без аргументов
- `TextConditionalFormatOperator` – Оператор для текстовых значений
- `TopBottomConditionalFormatOperator` – Наибольшие и наименьшие значения



- `UnaryConditionalFormatOperator` – Больше, меньше, равно, не равно, формулы
- `UniquenessConditionalFormatOperator` – Уникальные и повторяющиеся значения

5.2 Работа с ячейками

- Добавлен метод `CellRange.textToColumns`, который позволяет разделить текст в ячейках диапазона на отдельные горизонтальные ячейки. Используйте новый класс `TextToColumnsSettings` для задания параметров разделения и символов-разделителей.
- Новый метод `CellRange.setArrayFormula` позволяет поместить формулу массива в диапазон ячеек.

5.3 Работа с фильтрами

- Теперь вы можете получить фильтры, которые находятся в диапазоне фильтрации, с помощью метода `FiltersRange.getFilters`.
- Новый метод `TableFilters.getFilterType` позволяет тип фильтра, установленного в столбце: по значению (`ValuesTableFilter`) или по условию (`ConditionalTableFilter`).
- Добавлены методы `TableFilters.getAsValueFilter` и `TableFilters.getAsConditionalFilter` для получения фильтров по значению и по условию из коллекции.
- Для удаления значения из фильтра, добавлен метод `ValuesTableFilter.remove`.

5.4 Генерация HTML документов

Добавлен глобальный метод `exportWorksheetToHtml`, который позволяет сгенерировать HTML документ на основе листа табличного документа. Новый класс `WorksheetHtmlExportSettings` содержит настройки генерации HTML. В текущем релизе поддерживается только генерация данных и форматирования ячеек.