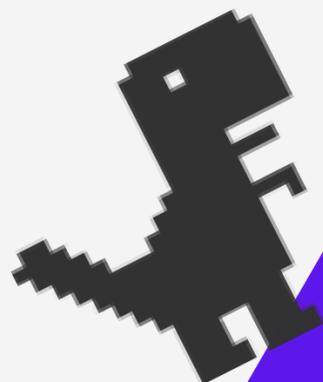


КОВЧЕГ +

# MAKE.COM

14 занятие

[WWW.KV-AI.RU/OBUCHENIE-PO-MAKE](http://WWW.KV-AI.RU/OBUCHENIE-PO-MAKE)



КОВЧЕГ +

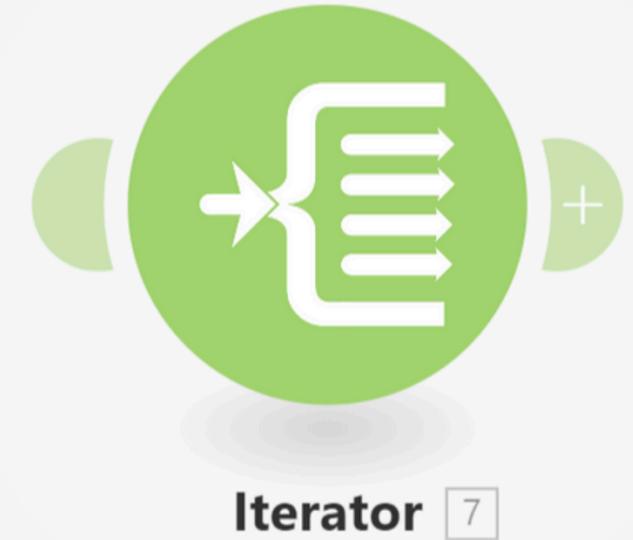
# TOOLS

знакомимся с Iterator ,  
Aggregator и многое другоу

[WWW.KV-AI.RU/OBUCHENIE-PO-MAKE](http://WWW.KV-AI.RU/OBUCHENIE-PO-MAKE)



# ITERATOR

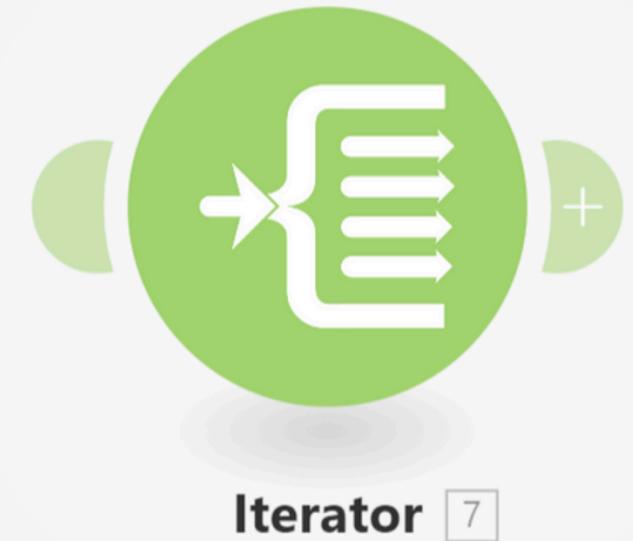


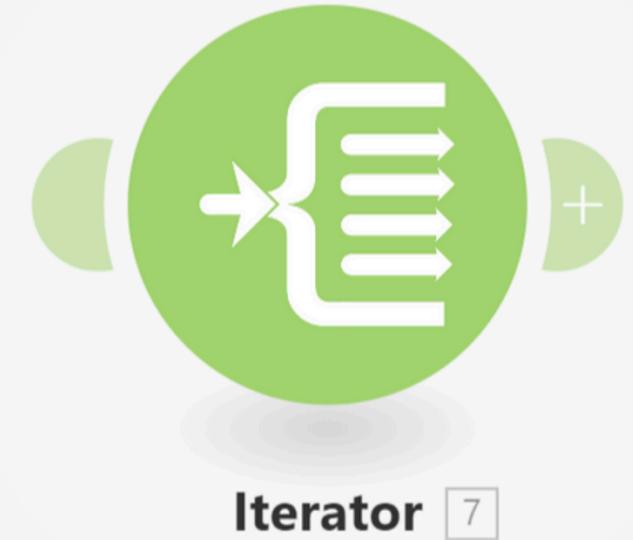
## Что такое Iterator ?

Iterator — это специальный инструмент (модуль) в Make.com, который позволяет разбивать массив данных (список) на отдельные элементы-бандлы для их поочередной обработки. Другими словами, если у вас есть массив с несколькими элементами — например, список заказов, контактов или строк таблицы — Iterator возьмёт этот список и «итеративно» (поштучно) пропустит каждый элемент через последующие модули в сценарии.

## Для чего это нужно?

- Поштучная обработка элементов массива: Без итератора, если у вас есть список данных, вам было бы сложно применить одно и то же действие к каждому элементу списка. Iterator автоматически создаёт независимую последовательность операций для каждого элемента.
- Гибкая автоматизация: Итератор позволяет строить сложные сценарии, где, например, вы получаете массив контактов и отправляете каждому индивидуальное письмо или обрабатываете заказы по одному, обновляя данные о каждом товаре по отдельности.





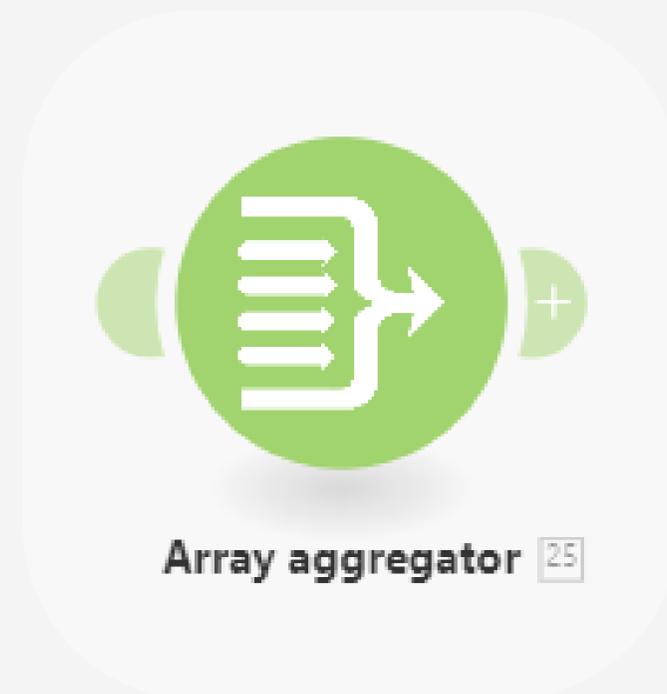
## Основные функции и возможности итератора:

Разбиение массива на отдельные элементы: Вы указываете массив (например, данные из Google Sheets, Airtable, или результат предыдущего запроса), и Iterator создаёт поток задач для каждого элемента массива.

Настройка выходных данных: Iterator на выходе возвращает бандлы по одному элементу. Это позволяет последующим модулям «видеть» каждый элемент отдельно.

Гибкое применение в любых сценариях: Итератор не «знает», что за данные он получает. Он работает с любым массивом — массив строк, массив объектов, массив чисел.

# ARRAY AGGREGATOR



## Что такое Array Aggregator ?

Array Aggregator — это модуль , который позволяет собрать несколько отдельных бандлов данных в один массив. Если Iterator помогает «разбить» данные на элементы, то Array Aggregator работает в обратном направлении: он берёт отдельные выходные данные из предыдущих модулей (или повторений модулей) и формирует из них единый массив. Это особенно полезно, когда вы хотите аккумулировать отдельные данные, поступающие по одному, в цельный список, который потом можно будет использовать для дальнейшей обработки.

## Основные функции Array Aggregator:

### Сбор отдельных данных в массив:

Допустим, у вас есть несколько контактов, товаров или заказов, которые поочерёдно выдает сценарий. Array Aggregator соберёт их все в одном массиве.

### Гибкое формирование структуры массива:

Вы можете указать, какие поля попадут в результирующий массив. Это даёт возможность делать аккуратную выборку данных для итогового массива.

### Поддержка сложных сценариев:

Если ваш сценарий Make выполняется в цикле или запускается несколько раз по разным условиям, Array Aggregator может объединить результаты всех этих повторных запусков (бандлов) в единый массив.

### Удобство для последующей обработки:

Получив массив, вы можете отправить его куда-то одним «пакетом» — в базу данных, в Google Sheets, в API другого сервиса или использовать внутри Make для дальнейших операций (например, для отчётности или массовой обработки).



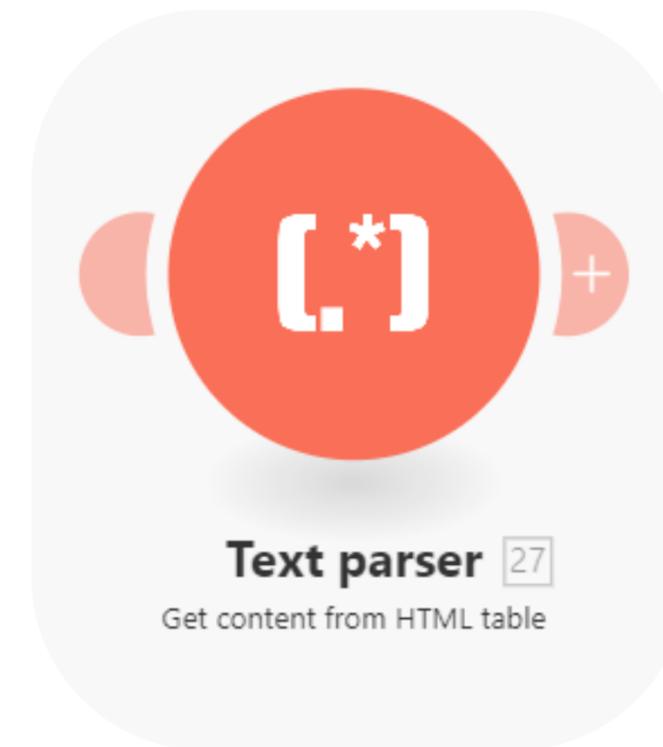
Array aggregator 25

# Get content from HTML table

Модуль Get content from HTML table в Make.com (ранее Integromat) предназначен для извлечения содержимого из HTML-таблиц и преобразования его в удобный формат для дальнейшей обработки. Если у вас есть веб-страница или HTML-код, содержащий таблицу с данными, этот модуль поможет «достать» эти данные и преобразовать их в массив объектов или строк, которые легко использовать в сценариях автоматизации.

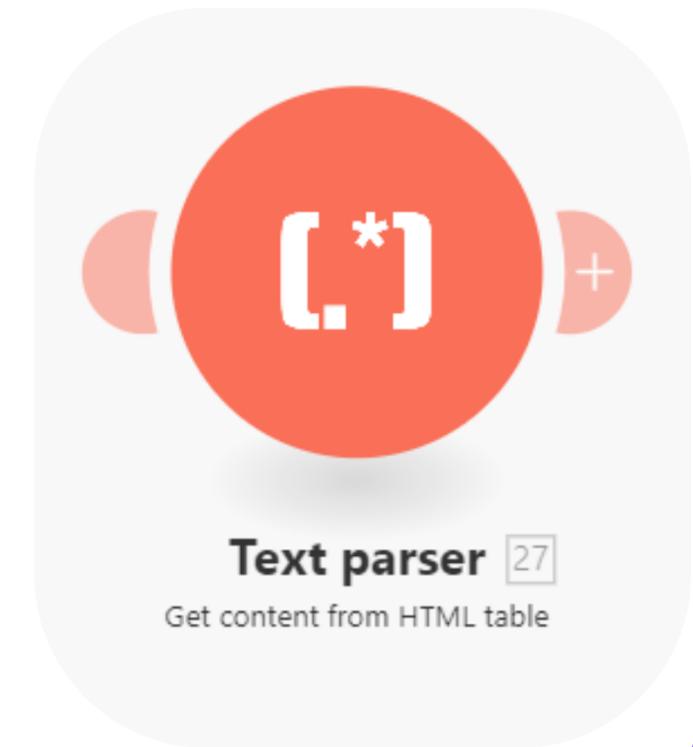
Основные возможности модуля:

1. Парсинг HTML-кода с таблицами:
2. Вы указываете модулю HTML-код или ссылку на ресурс с HTML (например, используя модуль HTTP для получения страницы). Модуль затем ищет в этом HTML-коде таблицу.
3. Извлечение данных по строкам и столбцам:
4. Модуль сканирует каждый `<tr>` (табличную строку) и `<td>` (ячейку) в таблице, извлекает текст или содержимое, и формирует удобную структуру данных.
5. Формирование массива объектов:
6. Результат работы модуля — это, как правило, массив, где каждый элемент соответствует одной строке таблицы. Внутри каждого элемента — ключи и значения, соответствующие столбцам таблицы. Это упрощает дальнейшую работу с данными, например, передачу их в Google Sheets, преобразование, фильтрацию или отправку в другие сервисы.
7. Обработка как статических, так и динамических таблиц:
8. Независимо от того, получили ли вы статический HTML-код или динамически сгенерированный при запросе к API, модуль сможет извлечь табличные данные.



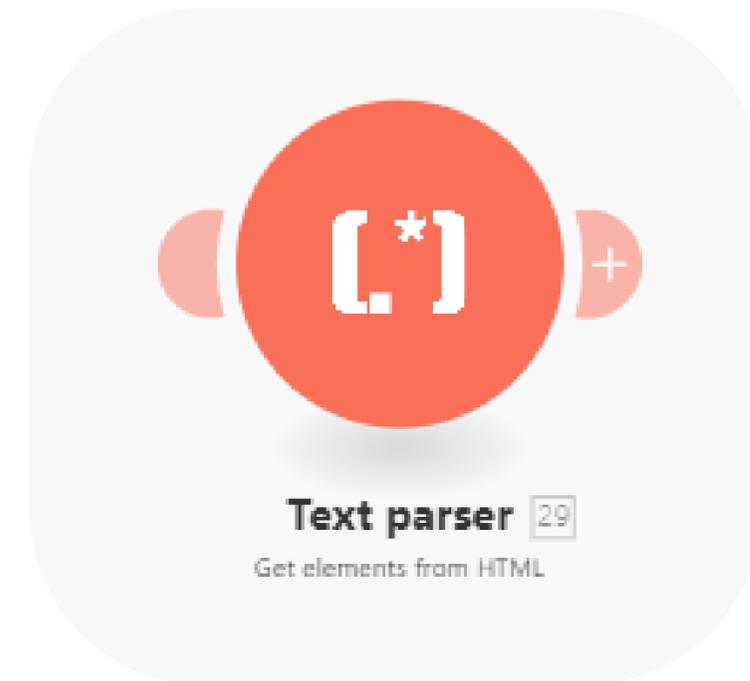
# Как применять модуль **Get content from HTML table**

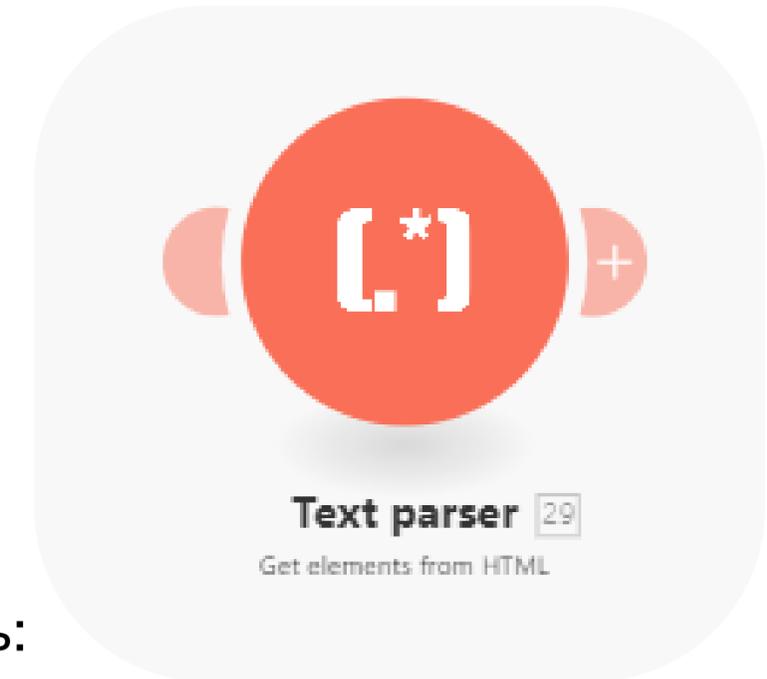
- Получите HTML-код, содержащий таблицу:
- Обычно перед модулем **Get content from HTML table** вы используете модуль HTTP или другой модуль для получения HTML-страницы.
- Например, модуль HTTP (Make) делает GET запрос к веб-странице и возвращает ее HTML-код.
- Подключите модуль **Get content from HTML table**:
- Добавьте модуль **Get content from HTML table** после модуля, возвращающего HTML.
- В настройках модуля укажите, откуда брать HTML-код (обычно это выходные данные предыдущего шага).
- Настройте параметры выборки таблицы:
- Некоторые модули позволяют указать селектор (CSS или XPath) для выбора конкретной таблицы, если на странице их несколько.
- Укажите, как модуль должен интерпретировать шапку таблицы (thead/tr/th) — это позволяет правильно именовать столбцы. Если в таблице есть заголовок столбцов (<th>), модуль может автоматически использовать их как названия полей в результирующем массиве.
- Если заголовков нет, вы можете указать, чтобы столбцы назывались "Column1", "Column2", и т.д.
- Результат — массив строк (объектов) таблицы:
- На выходе вы получите массив, где каждый объект соответствует строке таблицы. Поля объекта — это данные из ячеек столбцов.



## Text parser (Get elements from HTML)

Этот модуль предназначен для извлечения определённых HTML-элементов из переданного HTML-кода. Он позволяет выбрать типы элементов, такие как картинки, ссылки или iframe, и получить их атрибуты и содержимое. После извлечения данные можно использовать в последующих шагах сценария.





# Text parser (Get elements from HTML)

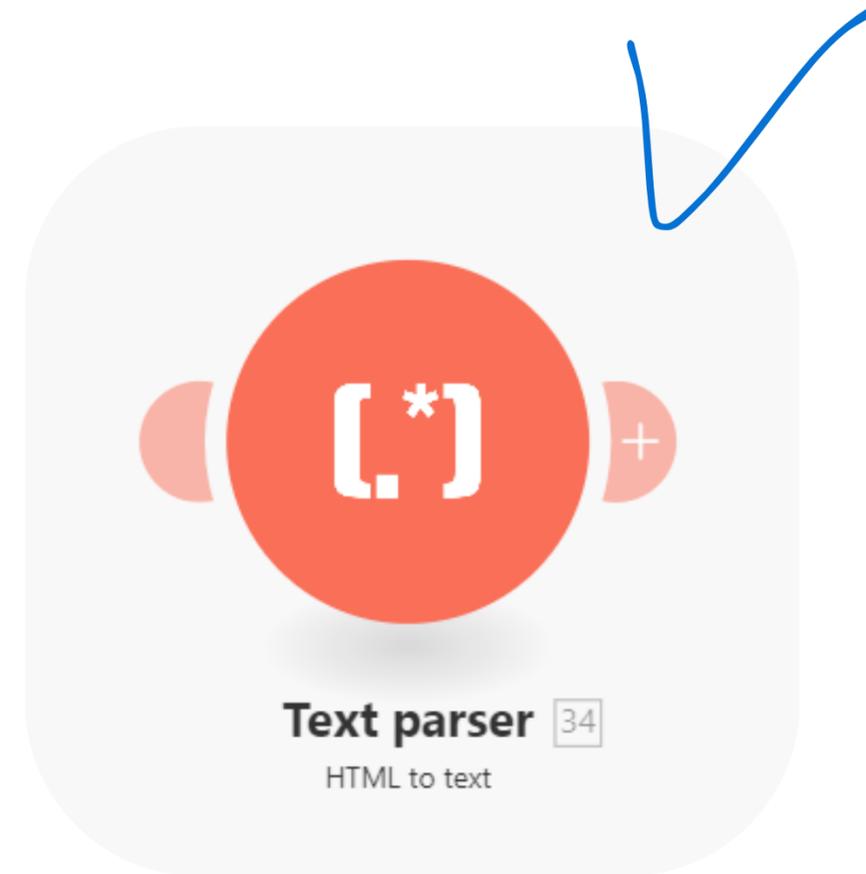
## Выбор типа элемента (Element type):

В модуле доступен выбор типа HTML-элемента, который нужно извлечь:

- **Image (img):** Позволяет получить все теги `<img>` со страницы, вытащить атрибут `src` (ссылку на изображение), а также другие доступные атрибуты (например, `alt`).
- **Link (a):** Извлекает все теги `<a>`, позволяя получить `href` (URL ссылки) и текст ссылки.
- **Iframe:** Находит все теги `<iframe>` и даёт возможность получить адрес встраиваемого контента (`src`) или другие атрибуты.

## HTML to text

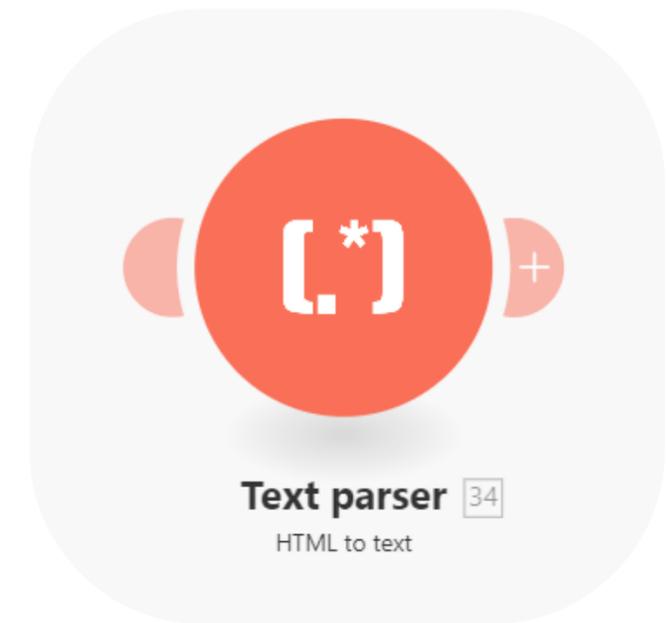
Модуль HTML to text предназначен для преобразования HTML-кода в обычный текст. Часто бывает нужно извлечь содержимое веб-страницы или сообщения электронной почты, представленное в формате HTML, и получить «чистый» текст без тегов и разметки.



# HTML to text

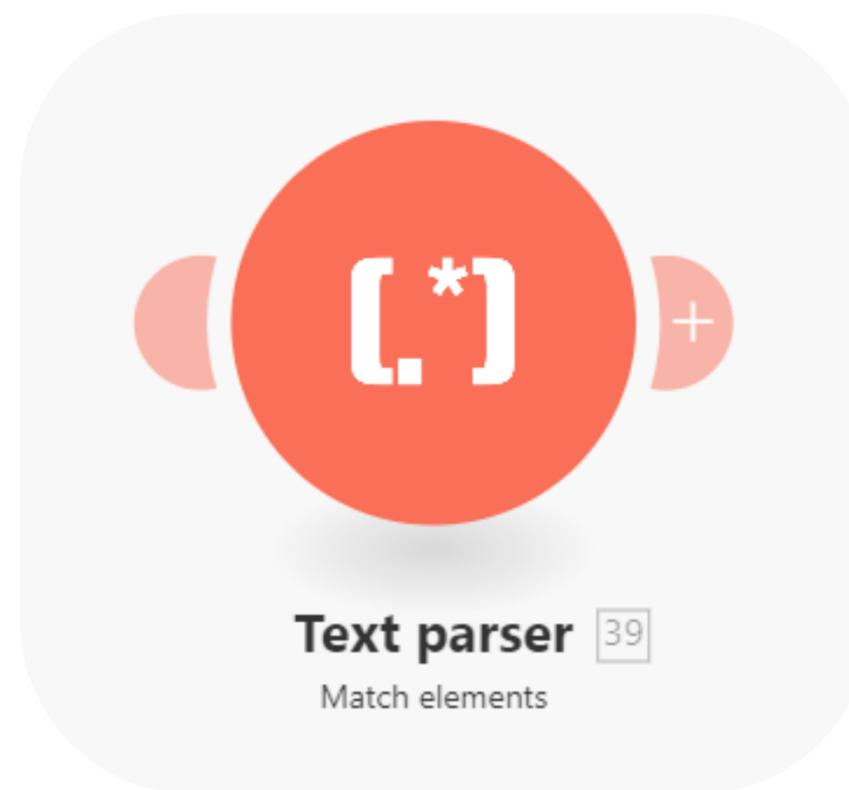
Основные возможности:

1. Удаление HTML-тегов:
2. Модуль берёт входной HTML-код (который может содержать теги `<p>`, `<div>`, `<a>`, `<strong>` и др.) и преобразует его в простой текст, убирая разметку.
3. Сохранение структуры текста:
4. Хотя HTML-теги исчезают, модуль, как правило, пытается сохранить читабельность текста. Например, переносы строк или пробелы между отдельными частями текста могут быть сохранены, чтобы итоговый текст был понятным.
5. Универсальность применения:
6. Можно применять для:
  - Извлечения текстового контента из HTML-писем.
  - Парсинга веб-страниц, чтобы получить именно текстовую информацию.
  - Упрощения анализа данных (например, дальнейшего анализа текста, поиска по содержимому, отправки текстовой версии куда-либо).
7. Чистый результат для других модулей:
8. После преобразования вы можете использовать полученный чистый текст в других модулях Make.com: отправить его в SMS, сохранить в Google Sheets, применить фильтры, выполнить анализ или использовать для генерации каких-то уведомлений.



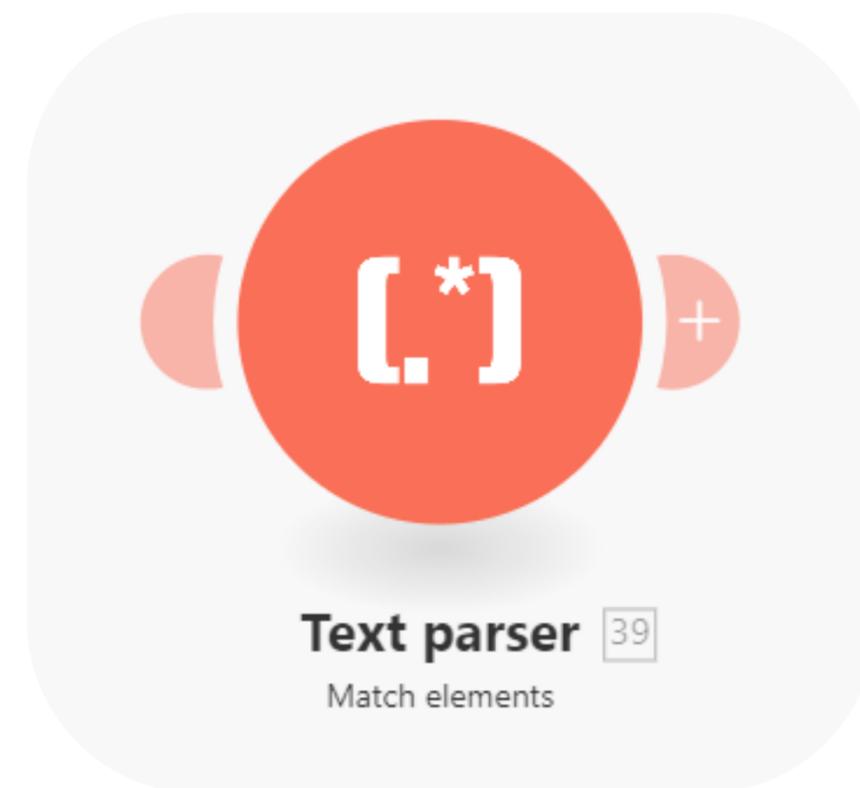
## Match elements

Модуль Match elements предназначен для поиска и извлечения из текстового контента набора элементов, соответствующих определённым паттернам. Он облегчает процесс парсинга неструктурированного текста, выделяя нужные фрагменты данных — такие как ссылки, email-адреса, телефонные номера или пары «ключ:значение» — без необходимости вручную разрабатывать сложные регулярные выражения.



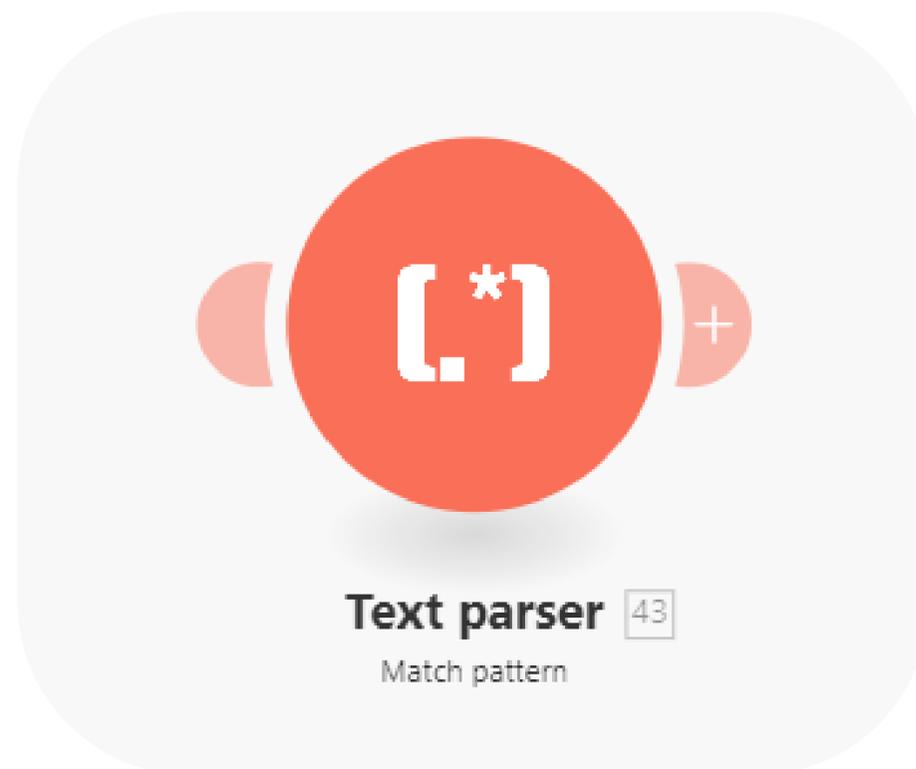
# Match elements

- Email: Ищет адреса электронной почты в тексте. Например: john.doe@example.com
- HTML tag: Извлекает HTML-теги, например <div>, <a href="..."> и т.д.
- HTTP address: Ищет ссылки, начинающиеся с http:// или https:// — например: https://example.com
- FTP address: Ищет адреса для протокола FTP, например: ftp://server.com/folder
- Phone numbers: Находит телефонные номера по заданному формату.
- IPv4 address: Находит IPv4-адреса, например: 192.168.0.1
- IPv6 address: Ищет IPv6-адреса, например: 2001:0db8:85a3:0000:0000:8a2e:0370:7334
- Emoji: Ищет эмодзи в тексте.
- Pair of key:value: Находит пары «ключ:значение», например: name:John Doe

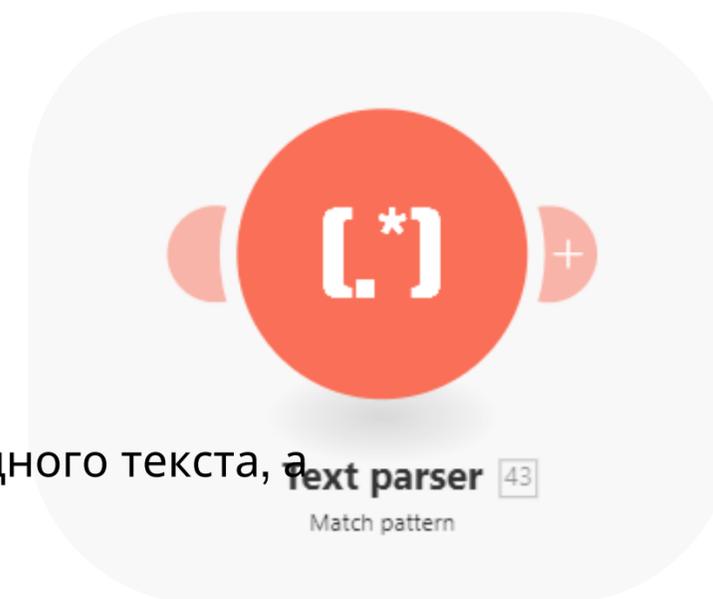


# Match pattern

Модуль Match pattern предназначен для проверки текста на соответствие заданному шаблону (паттерну) и извлечения отдельных частей данных, если паттерн совпадает. В отличие от модулей с готовыми предустановленными шаблонами, Match pattern чаще всего используется, когда вам нужно применить собственное регулярное выражение или кастомный паттерн для более точного или специфичного извлечения информации из строки.



# Match pattern



## Multiline (Многострочный режим)

- Опции: Yes / No
- Если Yes, метасимволы начала `^` и конца `$` строки будут применяться к каждой строке входного текста, а не только к началу или концу всего текста целиком.
- Если No, `^` и `$` соответствуют только началу и концу всего текста.

## Singleline (Однострочный режим)

- Опции: Yes / No
- Если Yes, метасимвол `.` (точка) будет совпадать в том числе и с символами переноса строки `\n`. Это значит, что точка будет «растянута» на несколько строк.
- Если No, `.` не будет совпадать с переводом строки, и многострочный текст будет разбиваться логически.

## Continue the execution of the route even if the module finds no matches (Продолжать ли выполнение при отсутствии совпадений)

- Опции: Yes / No
- Если Yes, сценарий продолжится, даже если модуль не найдёт совпадений. В таком случае модуль просто не выведет результат или выведет пустой результат.
- Если No, при отсутствии совпадений модуль завершится с ошибкой и остановит дальнейшее выполнение сценария.

## Text (Текст для анализа)

- В это поле вставляется исходный текст, к которому будет применён указанный паттерн.
- Это может быть переменная, результат другого модуля (например, HTTP-запрос) или статический текст.

# Text parser Replace

Модуль Text parser Replace предназначен для поиска определённого текста или шаблона в заданной строке и замены его на новый фрагмент. Он особенно полезен, когда вам нужно массово заменить определённый фрагмент текста, почистить данные или стандартизировать их формат.

