SimbirSoft

ООО "СимбирСофт", 432071,
г. Ульяновск, пр-кт Нариманова, 1, стр. 2
+7 (8422) 44-66-91 I info@simbirsoft.com
simbirsoft.com

Мануал по работе с Postman

Версия 1.0 от 24.07.2022

24.07.2022 г.

SimbirSoft

Оглавление

SimbirSoft

История изменений

Версия	Дата	ФИО	Изменения
1.0	24.07.2022	Астуков Е.Ю.	Инициализация документа

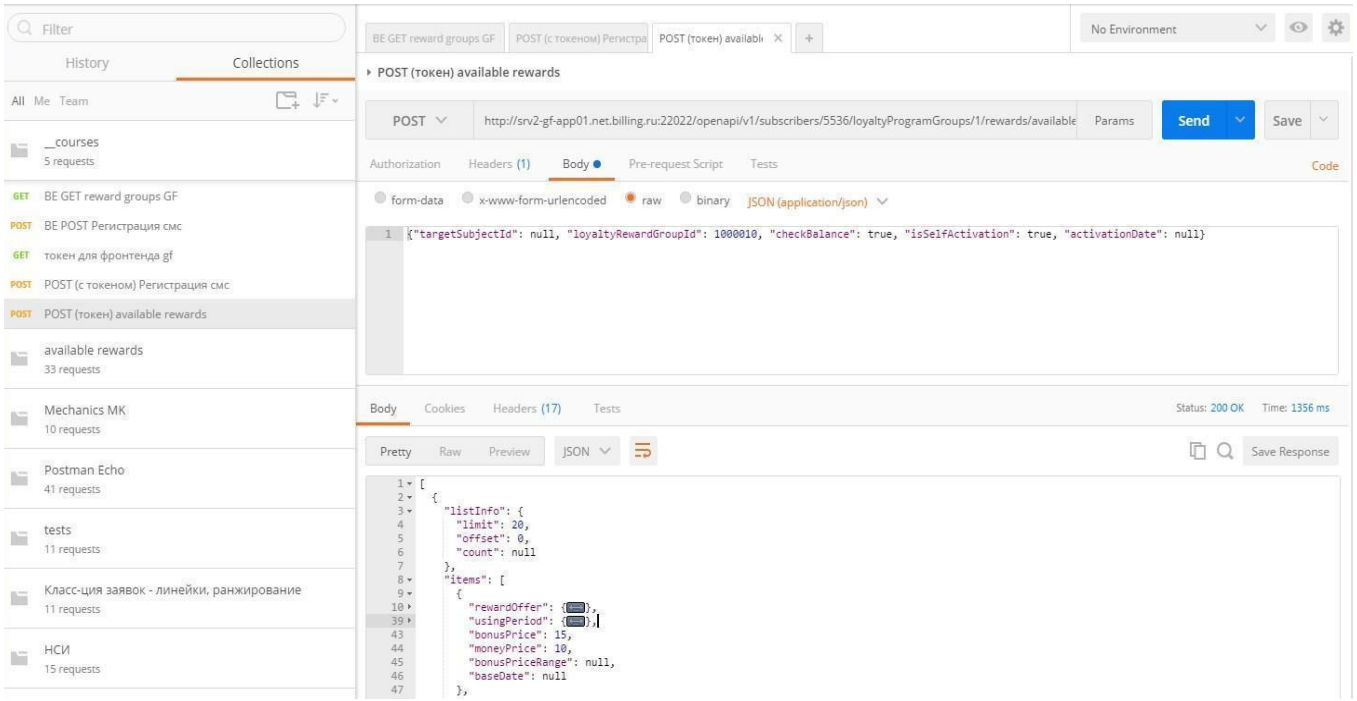
SimbirSoft

Postman

Postman – инструмент для тестирования API-запросов, в частности REST API.
Скачать
Официальная документация

Для выполнения простейшего запроса через Postman нужно:

1. Выбрать тип запроса (GET/POST или др.).
2. Указать строку запроса вместе с хостом в соответствующее поле.
3. Нажать кнопку «Send».



Если все сделано правильно, то в нижней части окна отобразится результат выполненного запроса. Обратите также внимание на http-код ответа над результатом справа.
В зависимости от типа запроса также может понадобиться:

1. Параметры, передающиеся в строке запроса, можно также указать в строке в виде ?param1=value1&param2=value2 или открыть меню Params и заполнить через форму.
2. Для POST/PUT и других методов, предполагающих наличие тела запроса, указать его на вкладке Body. Если у вас есть готовое тело запроса в json/xml, удобнее всего на вкладке Body выбрать режим “Raw” и вставить тело целиком.
3. Указать на вкладке Headers заголовки запроса стандартные или специфичные для API вашей системы
4. Если ваше приложение предполагает авторизацию, может потребоваться выполнить дополнительные действия в зависимости от конкретной реализации. Например, может потребоваться указать значение для header’a Authorization,

или специфичного параметра запроса, или же использование одного из методов авторизации, указанных на вкладке Authorization

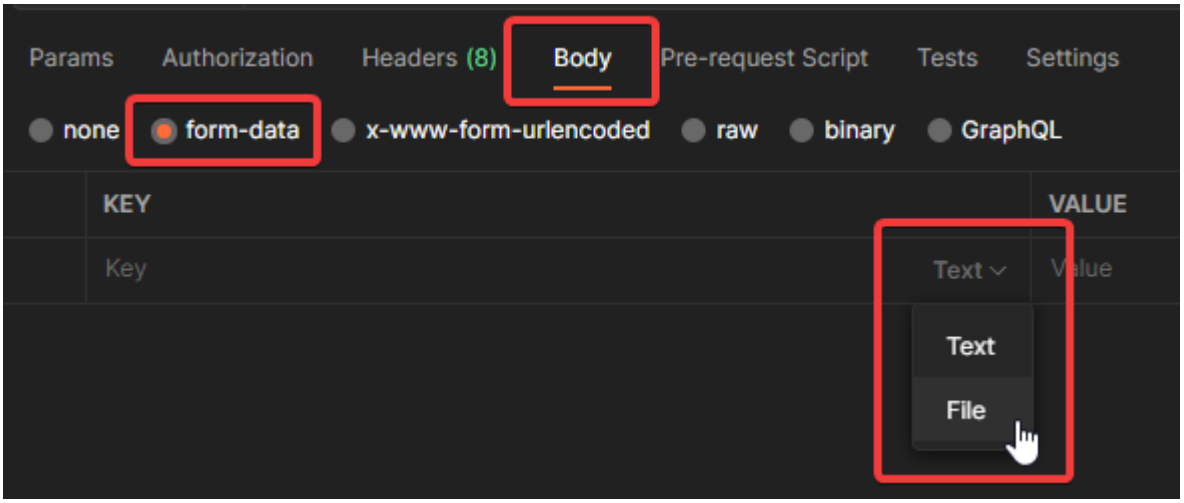
Postman позволяет:

- видеть историю запросов;
- отправлять запросы и получать ответы;
- сохранять запросы в папки и коллекции;
- выполнять все тесты из коллекции;
- изменять параметры запросов;
- изменять окружения (dev, test, production);
- выполнять автотесты, используя Collections runner, в том числе по расписанию;
- импортировать и экспортировать коллекции запросов и наборы тестов, чтобы обмениваться данными с коллегами.

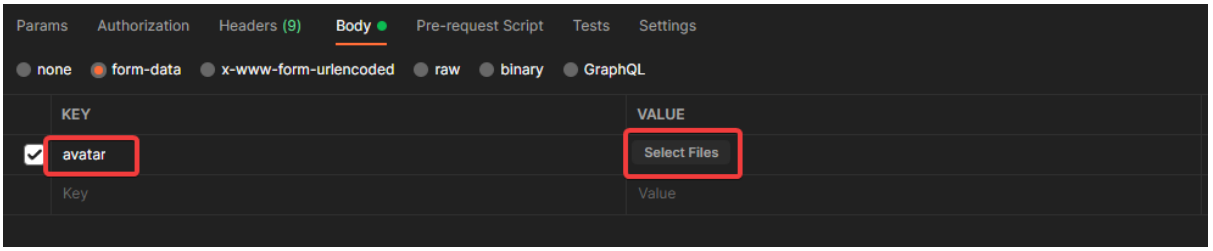
Отправка файлов

Вариант 1:

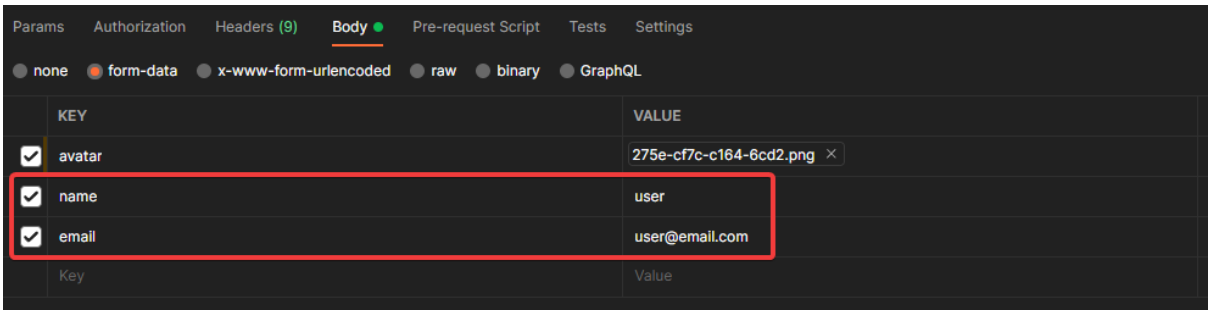
1. На вкладке **Body** выбрать **form-data**
2. Выбрать **file** вместо **text**



3. Задать название ключа (уточнить в документации)
4. Прикрепить файл

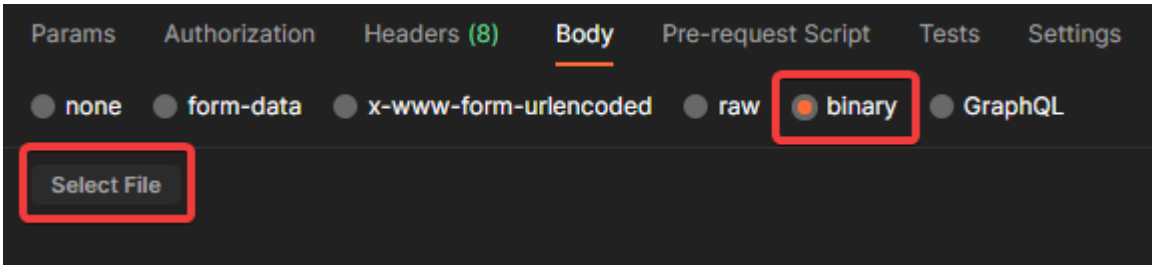


5. Помимо файла в запросе могут быть переданы и другие параметры, если это необходимо



Вариант 2:

1. На вкладке **Body** выбрать **binary**
2. Прикрепить необходимый файл

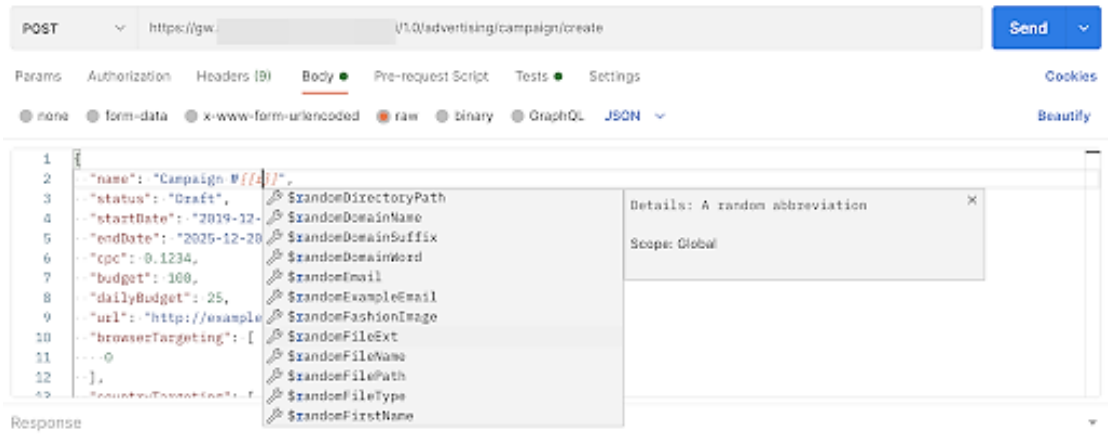


3. В данном случае в теле запроса будет отправлен только сам файл, в отличии от варианта 1

Необходимо уточнить в документации, каким образом содержимое файла должно передаваться в запросе, исходя из этого выбрать необходимый вариант.

Переменные в Postman

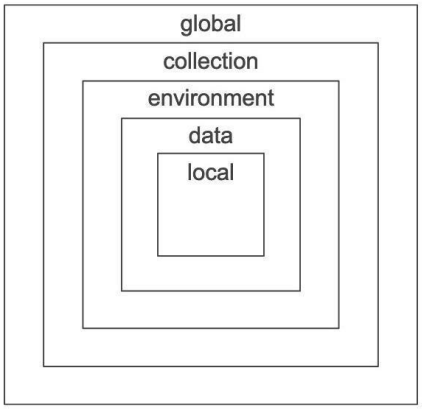
В ходе тестирования удобно использовать переменные.
Допустим, недавно почистили базу и для тестирования нам нужно ее заполнить - создать несколько рекламных кампаний с разными именами. Чтобы не делать это вручную, можно использовать динамические рандомные переменные.
Рандомных переменных в Postman много. Если при написании кода начать вводить парные фигурные скобки, Postman сам подскажет, какие из них доступны.



Подробнее про переменные можно почитать здесь.

Области видимости

Postman поддерживает несколько видов переменных, в зависимости от пространств и областей видимости. Идею хорошо иллюстрирует картинка из документации:



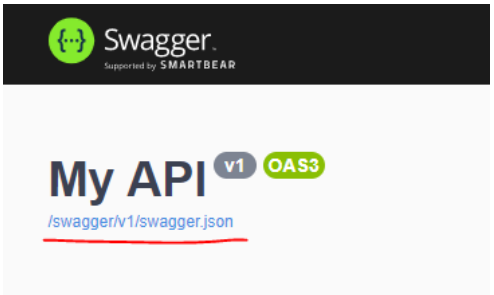
Поддерживаются следующие типы переменных:

- Глобальные - переменные, которые не относятся ни к какому окружению, они доступны во всем рабочем пространстве, из всех окружений. Глобальные переменные могут использоваться для передачи данных между коллекциями, запросами и окружениями.
- Переменные коллекции доступны во всех запросах внутри одной коллекции.
- Переменные окружения изменяются в зависимости от выбранного окружения.
- Локальные переменные являются временными. Они доступны только внутри запроса и используются, когда нужно переписать переменные коллекции или какие-то глобальные значения.
- Переменные данных - файловые переменные.

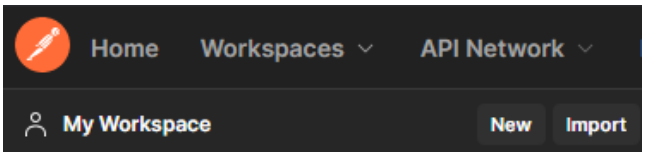
Глобальные переменные не могут иметь дубликаты. А вот локальные переменные могут иметь одни и те же имена, но только если они находятся в разных окружениях. Если Postman встречается с двумя переменными с одинаковыми именами, высший приоритет будет у локальной переменной (она затрет глобальную).

Импорт API Swagger в Postman

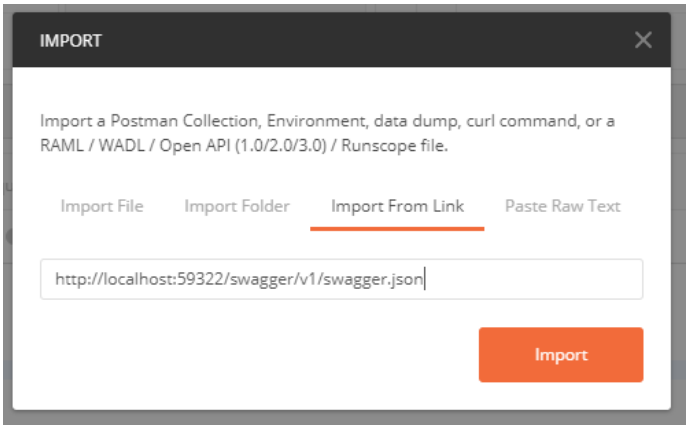
Находим URL-адрес JSON на своей странице Swagger:



Щелкаем эту ссылку и копируем URL страницы. Затем переходим в Postman и нажимаем Import:



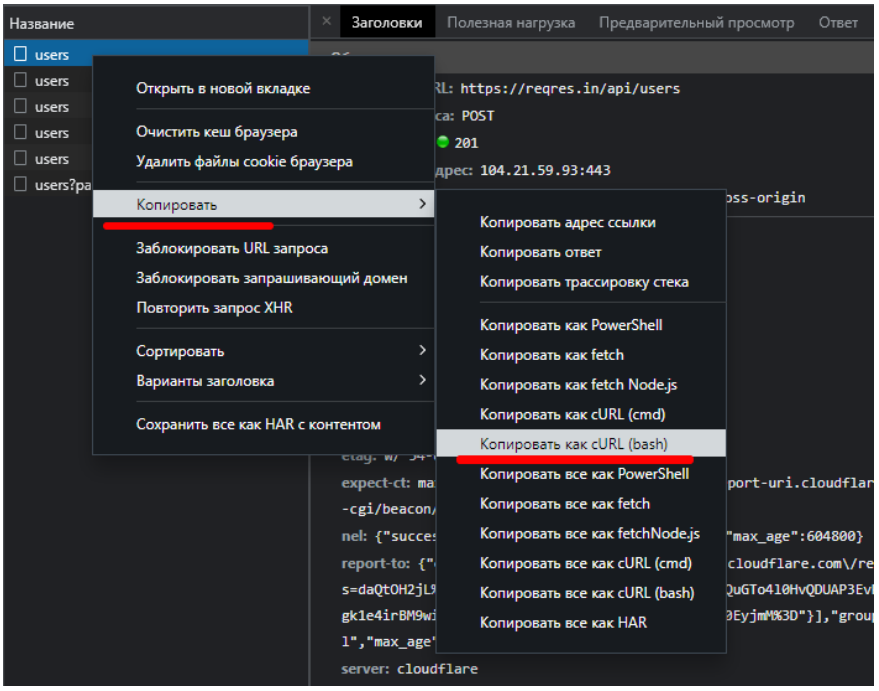
В следующей вкладке вставляем сохраненный URL:



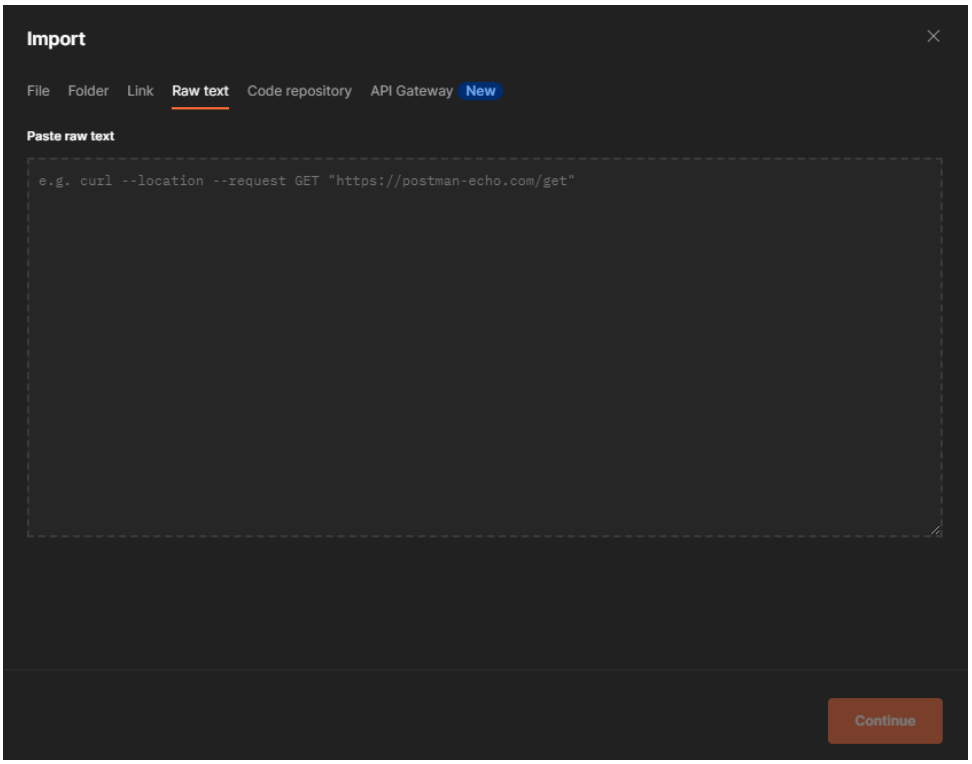
Выбираем необходимые настройки и импортируем.

Копировать запрос cURL в Postman

Для того, чтобы скопировать запрос из браузера в Postman, нужно отловить его в консоли браузера, ПКМ > Копировать > Копировать как cURL



В Postman нажать Import > Raw text > Вставить сохраненный cURL > Кнопка Continue



После импорта команды cURL в Postman, его можно повторно отправить и / или изменить запрос.

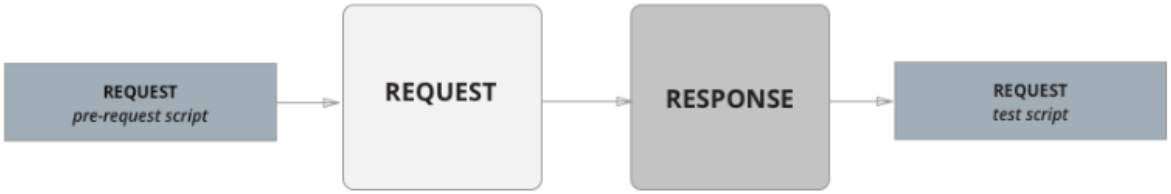
Тесты в Postman

Postman имеет мощную среду выполнения на основе Node.js, которая позволяет добавлять динамическое поведение к запросам и коллекциям. Это дает возможность писать тесты API, создавать запросы, которые могут содержать динамические параметры, передавать данные между запросами и многое другое. Вы можете добавить код JavaScript для выполнения двух событий:

- Перед отправкой запроса на сервер в виде сценария предварительного запроса на вкладке Pre-request Script.
- После получения ответа в виде тестового скрипта на вкладке Tests.

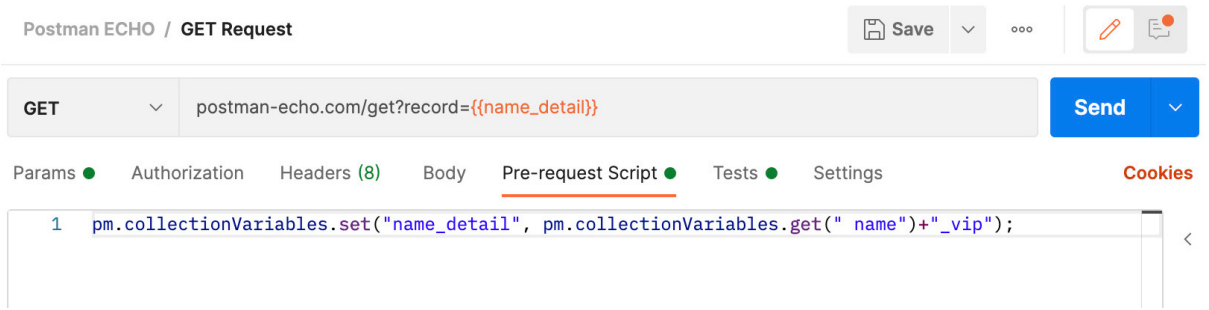
Порядок выполнения скриптов

В Postman порядок выполнения скрипта для одного запроса выглядит так: сценарий предварительного запроса будет выполняться до отправки запроса. А тестовый сценарий будет выполняться после отправки запроса.



Pre-request Script

Вы можете использовать сценарии предварительного запроса в Postman перед выполнением запроса. Открыв вкладку Pre-request Script для запроса, коллекции или папки, вы можете выполнить предварительную обработку, такую как установка значений переменных, параметров, заголовков и данных тела. Вы также можете использовать сценарии предварительного запроса для отладки кода, например, выводя что-то на консоль.



Пример использования сценария предварительного запроса может быть следующим:

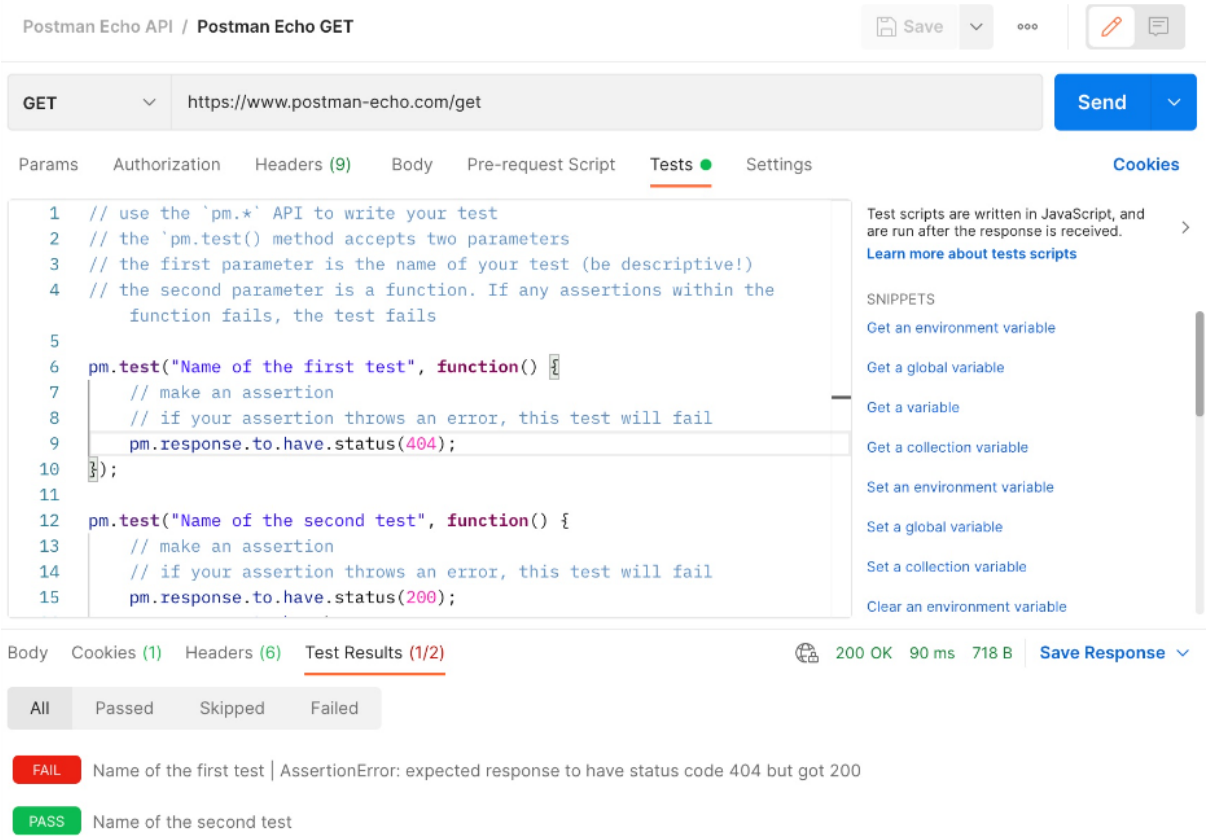
- У вас есть серия запросов в коллекции, и вы запускаете их в определенной последовательности, например, при использовании средства запуска коллекции.
- Второй запрос зависит от значения, возвращенного из первого запроса.
- Значение необходимо обработать, прежде чем передать его второму запросу.
- Первый запрос устанавливает значение данных из поля ответа в переменную в сценарии Tests.
- Второй запрос извлекает значение и обрабатывает его в своем сценарии предварительного запроса, затем устанавливает обработанное значение в переменную (на которую ссылается второй запрос, например в его параметрах).

Tests

Тесты подтверждают, что ваш API работает должным образом, интеграция между службами работает надежно, а новые разработки не нарушают существующие функции. Вы можете написать тестовые скрипты для ваших запросов Postman API на JavaScript.
Вы также можете использовать тестовый код, чтобы облегчить процесс отладки, когда что-то пойдет не так с вашим проектом API. Например, вы можете написать тест для проверки обработки ошибок вашего API, отправив запрос с неполными данными или неверными параметрами.

Вы можете добавлять тесты к отдельным запросам, коллекциям и папкам в коллекции. Postman включает в себя фрагменты кода, которые вы добавляете, а затем модифицируете в соответствии с логикой вашего теста.

Чтобы добавить тесты в запрос, откройте запрос и пишите свой код на вкладке Tests. Тесты будут выполняться после выполнения запроса. Вы сможете увидеть выходные данные на вкладке Test Results вместе с данными ответа.



Тесты выполняются после получения ответа. Когда вы нажимаете Send, Postman запускает ваш тестовый сценарий после того, как данные ответа возвращаются из API.

Если вам нужно выполнить код перед выполнением запроса, используйте вместо этого сценарии предварительного запроса (Pre-request Script).

Использование сниппетов (Snippets).

Справа от редактора тестов - в Snippets есть подборка наиболее часто используемых фрагментов тестового кода. Выберите один, и он вставит его в ваш редактор. Сниппеты могут ускорить процесс начала работы со сценариями. Вы можете редактировать их после добавления в соответствии с вашими требованиями к тестированию.

Взято из официальной документации Postman, подробнее здесь.