**PRE-REQUEST SCRIPTS**

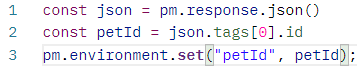
Можем писать функции на JS. Они выполняются до того как запрос отправится на сервер

**TESTS**

Позвоялет писать функции на JS. Они выполняются после того как получен ответ от сервера

Можно использовать готовые Snippets, или использовать свои

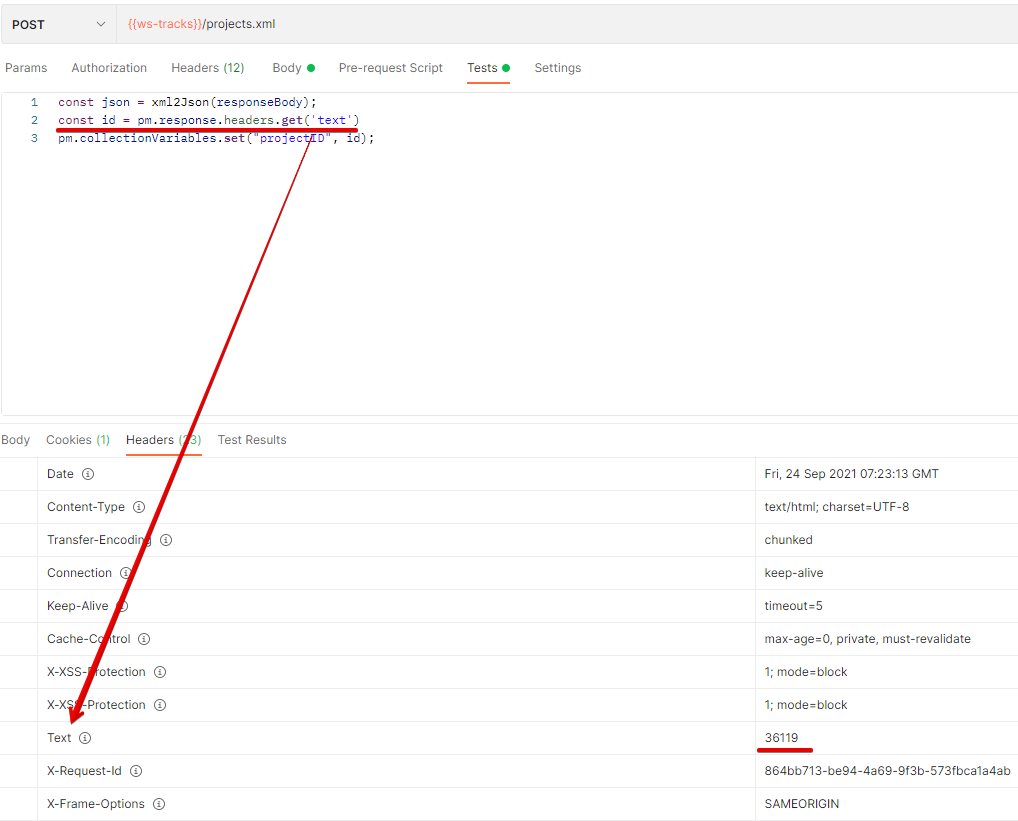
Назначаем переменную – получаем JSON от сервера, а после записываем нужное значение в переменную окружения. Это нужно для того, что бы можно было выполнить подряд несколько запросов, если второй запрос требует данные предыдущего, не вводя изменения в каждый из них мануально



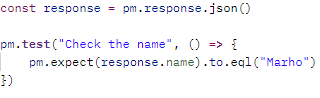
1. Записываем полученный JSON в JS переменную
2. Записываем нужное значение в отдельную JS переменную. Если нужно получить доступ к массиву, то указываем в квадратных скобках, к какому элементу в массиве получаем доступ, а после само значение. Спускаемся по дереву вложенности через точку
3. После записываем значение в переменную окружения - pm.environment.set("variable\_key", "variable\_value");

В качестве ключа пишем любое значение, а в качестве значения – вставляем JS переменную

Так же можно забрать значение из Header и поместить в переменную



Проверка что Response body содержит какое то значение



Явно указываем какой запрос будет выполняться следующим (даже если в Collection Runner мы поставим следующим другой запрос, выполнится тот, который был указан явно в скрипте)



**Что бы проверить, что в response body есть строка**

Используем здесь одинарные кавычки, для обозначения какую строку нужно искать, так как сама строка будет содержать двойные кавычки и POSTMAN будет считать, что двойные кавычки – это кавычки не самой строки, а кавычки для обозначения в коде, что щас будет строка

pm.**test**("Body matches string", **function** () {

    pm.expect(pm.response.text()).to.include('"results":[{"object":"page","id"')

    });

**Что бы проверить несколько элементов в body ответа, что бы был совпадал хотя бы 1**

pm.test("Testing multiple - simple", **function** () {

**var** arr = ["string A", "string B", "string C"];

responseText = "I have string D inside me";

pm.expect(responseText).to.contain.oneOf(arr);

});