Certified Tech Developer
The Ultimate Degree

API Testing - Criação de testes (JS) com Postman

Passos

1. Vamos começar com alguns conceitos para ter em mente:

- Para codificar os testes com Postman, devemos conhecer um pouco sobre a API que ele nos oferece. Cada um dos testes é executado com o objeto **pm** e especificamente com o método **.test( )**. Assim, para cada um, teremos a seguinte estrutura:

```
{ }
pm. test("Descrição da funcionalidade a ser testada", function( ) {
        // Código que valida o teste de test
});
```

- Para acessar o conteúdo da resposta das requisições, temos o objeto **pm.response** e seu método **.json( )**, que nos permitirá acessar os elementos da resposta em JSON.

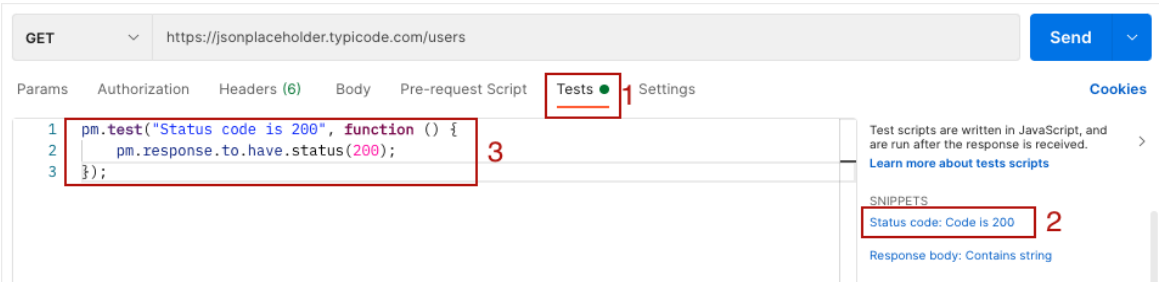
```
{ }
pm. test("Obtendo conteúdo da resposta", function( ) {
        pm.response.json( );
});
```

- Outro método importante é aquele que nos permite fazer uma verificação de conteúdo, o **pm.expect**.

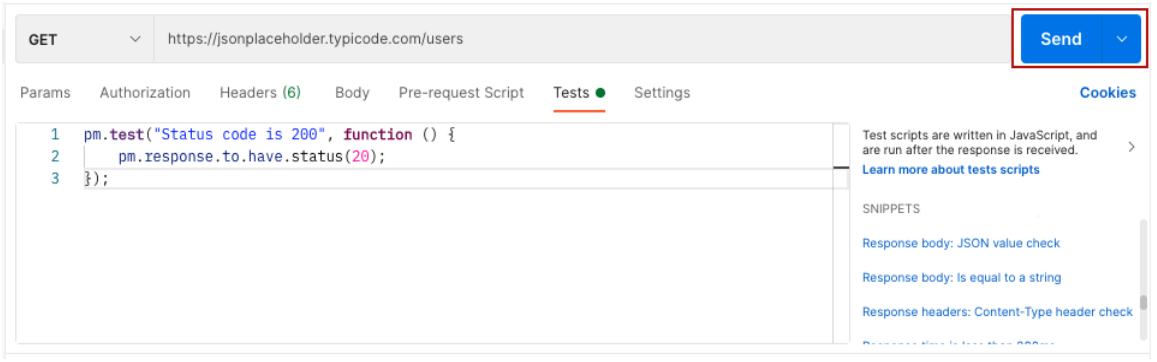
```
{ }
pm. test("Comparando o valor retornado com o esperado", function( ) {
        pm.expect(valor).to.equal("Valor esperado")
});
```

2. Considerando os conceitos já definidos, veremos dois dos testes mais utilizados em API Testing. O Postman nos fornece uma série de fragmentos padrões que nos orientam ao construir nossos testes:

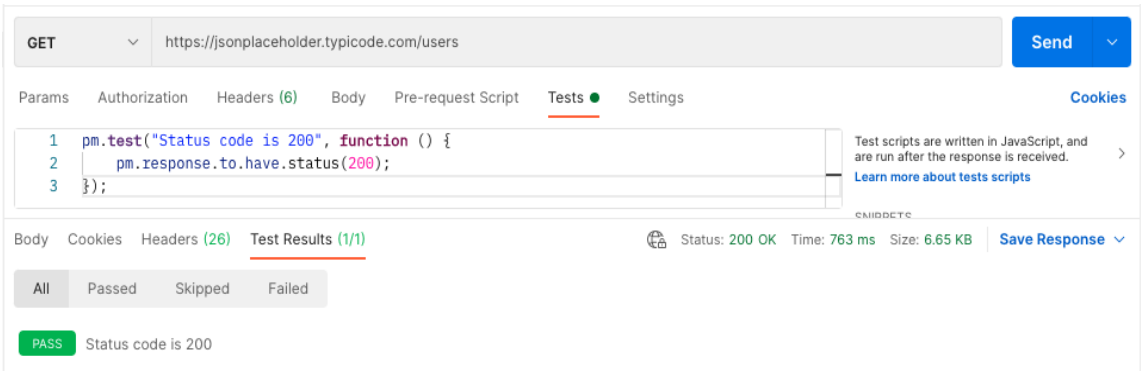
- Para começar, vamos para a requisição GET que criamos anteriormente e selecionamos a guia **Tests** (1). Nesta seção, escreveremos nosso conjunto de testes relacionados a essa API. Na subseção de fragmentos, devemos clicar em **"Status code: Code is 200"** (2) para gerar um dos testes padrão. O script será concluído automaticamente (3).



- Ao clicar em **Send**, o resultado do teste será exibido.

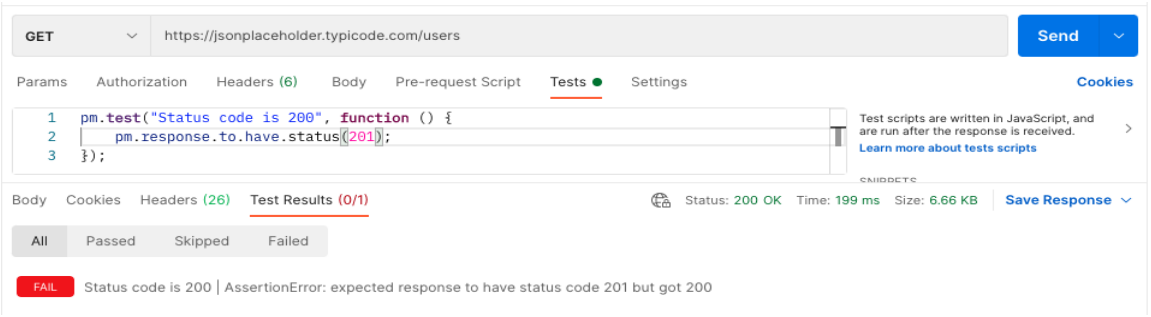


- Com este teste, estamos validando que o código de resposta da API é 200. Se este estiver correto, o teste retornará **PASS**: o que significa que o serviço está respondendo conforme o esperado.



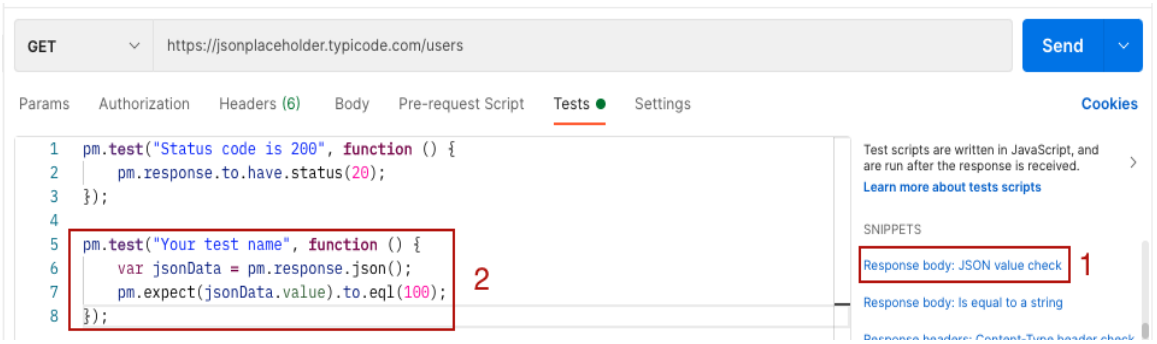
- Se o serviço falhar, será retornado o status **FAIL** e o código de erro relacionado a este status.

Neste teste, estamos reutilizando trechos de código que o Postman nos fornece para validar se a solicitação foi bem sucedida. Podemos editar esta consulta ao nosso gosto usando o código javascript.



3. Vamos adicionar mais um dos testes mais usados. Neste, iremos comparar o resultado esperado com o resultado real.

- Para isso, na subseção de fragmentos, devemos clicar em **“Response body: JSON value check”** (1). O script será concluído automaticamente (2).

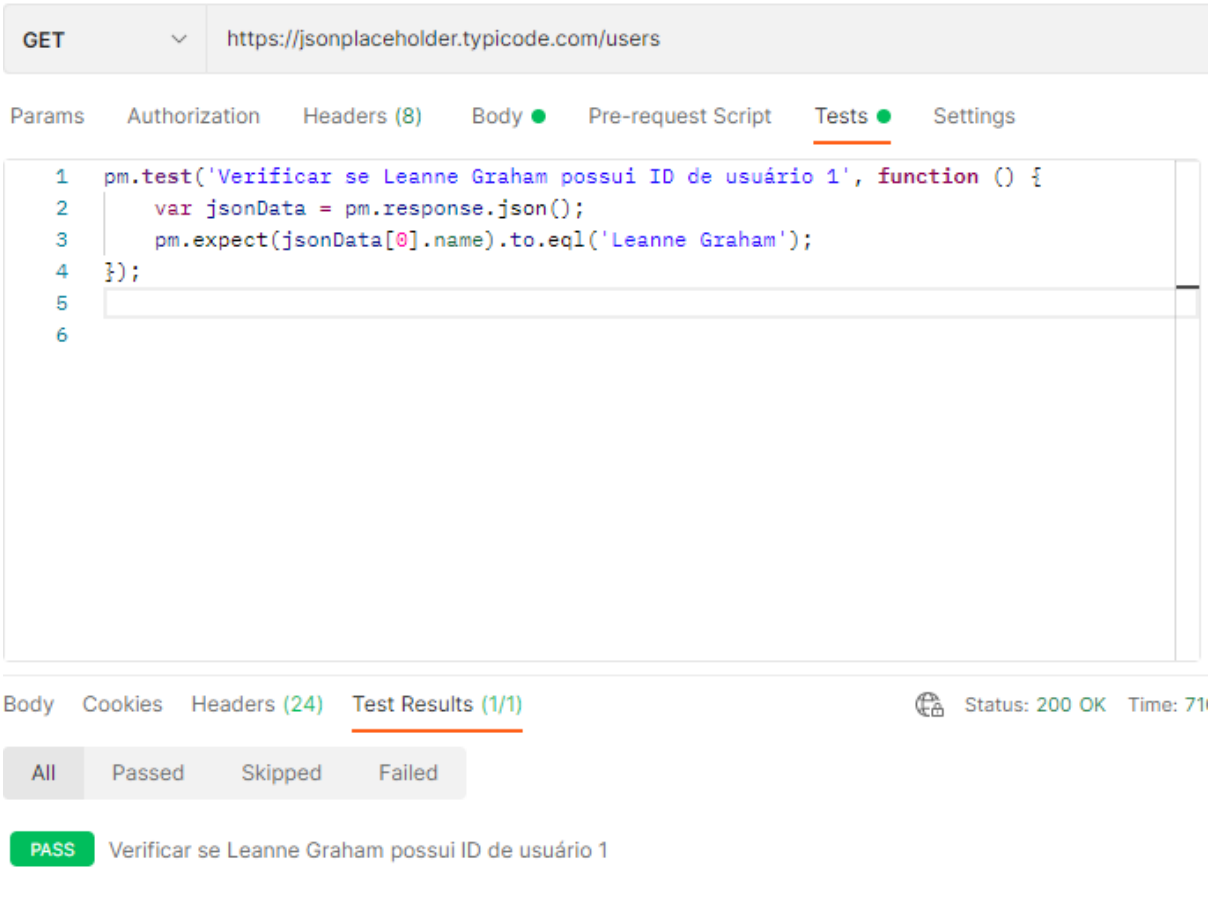


- É possível alterar o nome do teste padrão. Neste caso, substituímos por **“Verificar se Leanne Graham possui ID de usuário 1”**, já que este é o primeiro usuário na lista retornada pela API. Também devemos atualizar o corpo da função substituindo **jsonData.value** por **jsonData[0].name**; assim, obteremos o primeiro elemento da lista.

```
pm.test('Verificar se Leanne Graham possui ID de usuário 1', function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData[0].name).to.eql('Leanne Graham');
});
```

- Ao clicar em **Send**, o resultado do teste será exibido.

- Observa-se que o nosso teste nos retornou o status **PASS**. Desta forma, validamos que o conteúdo da resposta é o esperado. Assim, podemos validar dados diferentes e ver se nossa solicitação retorna os dados desejados.



- Por fim, observa-se que quando a solicitação é enviada, todos os testes relacionados a ela são executados. Desta forma, podemos criar um conjunto de testes vinculado a cada solicitação e verificar rapidamente seu status.


GET
https://jsonplaceholder.typicode.com/users
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
1 pm.test('Verificar se Leanne Graham possui ID de usuário 1', function () {
2 var jsonData = pm.response.json();
3 pm.expect(jsonData[0].name).to.eql('Leanne Graham');
4 });
5
6 pm.test("Status code is 200", function () {
7 pm.response.to.have.status(200);
8 });
Body
Cookies
Headers (25)
Test Results (2/2)
Status: 200 OK
All
Passed
Skipped
Failed
PASS
Verificar se Leanne Graham possui ID de usuário 1
PASS
Status code is 200