

Sample API testing plan.

- Design test cases/ test suites based on given specification.
- Automate test suites using Postman (pre-request and post-request scripts on JavaScript).
- Run automated tests.
- Analyse and record test results.

1. Design test cases/ test suites based on given specification.

/add/?value1=&value2= -Endpoint:

Positive tests:

- Execute API call with valid required parameters.
Double and Integer passed to request parameters.
- Validate response status code.
Code 200 expected.
- Assert result equals to expected value.
Result must be equal to sum of passed number parameters.
- Optionally performance and payload can be validated.

Negative tests:

- Execute API call with invalid required parameters.
Passing a String to one of the parameters and to both.
Passing "null" to one of the parameters or to both.
- Validate response status code.
Code 200 expected.
- Assert result equals to expected value.
Expected result is invalid input advice.
- Optionally performance and payload can be validated.

/validate/?value1= – Endpoint:

Positive tests:

- Execute API call with valid required parameters.
Integer or String values passed to request parameters according to given specification ranges.
- Validate response status code.
Code 200 expected.
- Assert result equals to expected value.
Result must be equal to expected value in given API specification.
- Optionally performance and payload can be validated.

Negative tests:

- Execute API call with valid required parameters.
Passing empty String ("") to request parameter.
Passing "null" to request parameter.
- Validate response status code.
Code 200 expected.
- Assert result equals to expected value.
Expected result is invalid input advice.
- Optionally performance and payload can be validated.

2. Automate test suites using Postman.

- Write Pre-request and post-request scripts (JavaScript) in Postman for test suites automation.
- Set Postman environment variables to use them dynamically in request URL.
- Setup assertion functions/ test cases to validate the response.
- Setup test running environment (Number of iterations, variables initial values etc)

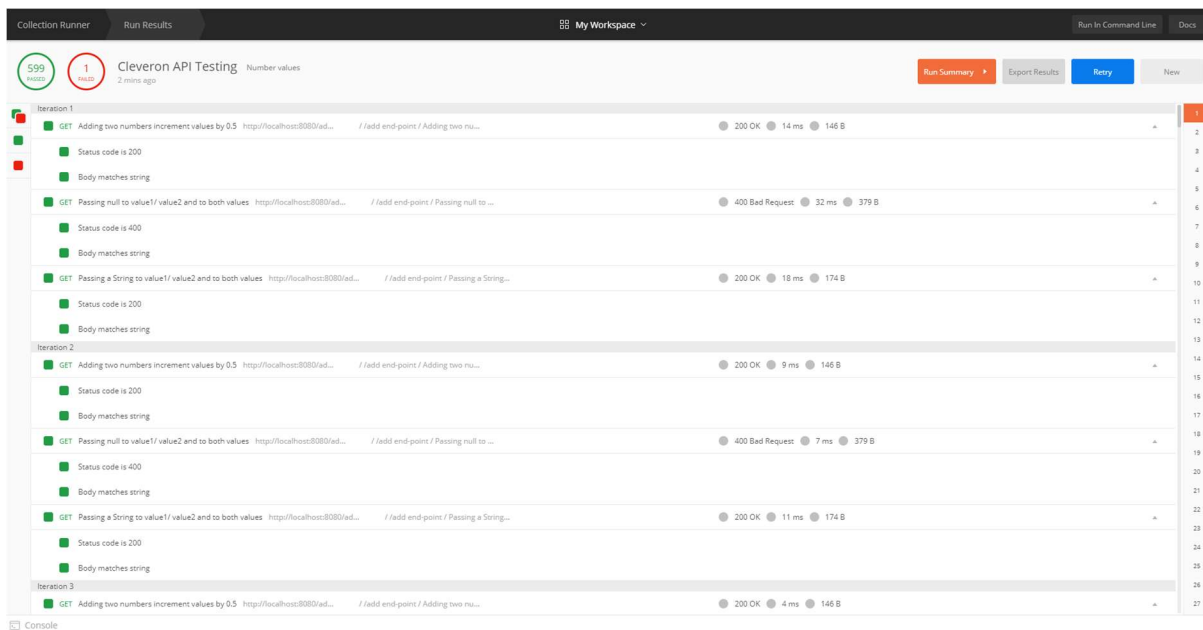
3. Run automated tests. Analyse and record results.

- Analyse test results (Failed test cases)
- Record found bugs.
- Prepare test results for export in Postman. (JSON format)

**Test results and important remark on the next page.*

The screenshot displays the Postman Collection Runner interface. At the top, the 'Run Results' tab is active, showing a collection named 'Cleveron API Testing' with 49 tests and 1 failed. The 'Run Summary' button is highlighted in orange. Below the summary, a table lists the test results for each iteration (40 to 47). Each iteration includes a 'GET' request for 'Spring validation' and a 'Testing multiple values' script. The 'GET' requests all passed with a 200 OK status. However, the 'Testing multiple values' script in iteration 44 failed with an 'AssertionError expected \'bug\' to be one of [Array(5)]'. The error message is displayed in a red box. The table also shows the response status, time, and size for each request.

Iteration	Test Name	Status	Response	Time	Size
40	GET Spring validation	200 OK	13 ms	159 B	
41	GET Spring validation	200 OK	6 ms	163 B	
42	GET Spring validation	200 OK	6 ms	157 B	
43	GET Spring validation	200 OK	6 ms	159 B	
44	GET Spring validation	200 OK	5 ms	149 B	
44	Testing multiple values	Failed	AssertionError expected 'bug' to be one of [Array(5)]		
45	GET Spring validation	200 OK	3 ms	159 B	
46	GET Spring validation	200 OK	3 ms	163 B	
47	GET Spring validation	200 OK	5 ms	157 B	



Bugs found:

- One bug found in `/add/?value1=&value2=` Endpoint.
 -Result in response is not as expected value.
 -To reproduce – pass 8 to value1 parameter and 8 to value2 parameter (or any other numbers which give in sum 16).
 -Value received – “bug”.
 -Value expected – 16.
- One bug found in `/validate/?value=` Endpoint.
 - Result in response is not as expected value.
 -To reproduce – pass a String with length of 27 characters.
 -Value received – “bug”.
 -Value expected – “very long string”.

Potentially more bugs could be found depending how many iterations of the test suites are made. For `/add` endpoint 50 iterations performed, for `/validate` endpoint 100 iterations performed.

