Bornfight®

Testing APIs with Postman

Common challenges & solutions

About me

Rudolf Jurišić

Software Developer Team Lead

Bornfight

Why

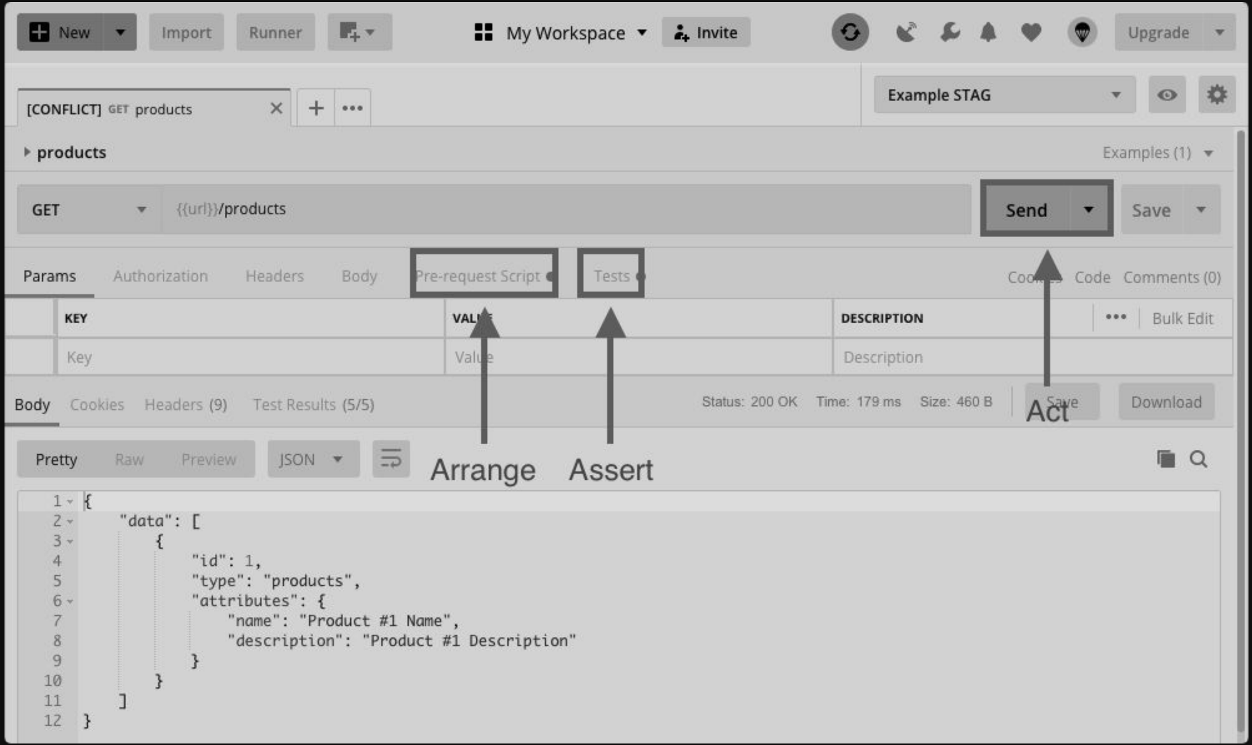
Bornfight®

What

Arrange, Act, Assert

- Arrange all necessary preconditions and inputs.
- Act on the object or method under test.
- Assert that the expected results have occurred.

- Pre-request Script tab
- Send button
- Tests tab



Example STAG

products

GET {{url}}/products

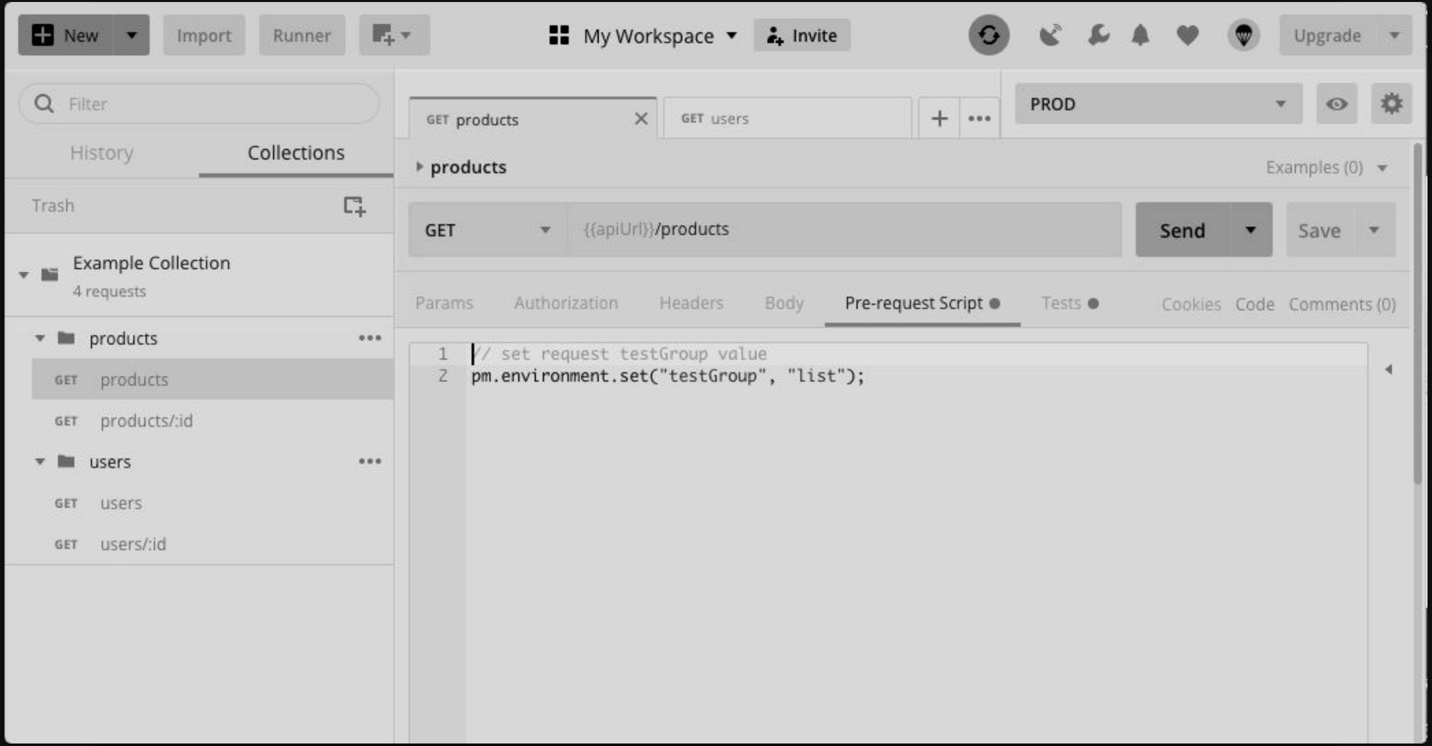
Status: 200 OK Time: 179 ms Size: 460 B

```
{
    "data": [
        {
            "id": 1,
            "type": "products",
            "attributes": {
                "name": "Product #1 Name",
                "description": "Product #1 Description"
            }
        }
    ]
}
```

Grouping Requests

- Requests
- Folders
- Collections

Bornfight.



GET {{apiUrl}}/products

```
// set request testGroup value
pm.environment.set("testGroup", "list");
```

- Requests (Arrange, Assert)
- Folders (Arrange, Assert)
- Collections (Arrange, Assert)

Running tests

- Requests (Send button, Collection Runner)
- Folders (Collection Runner)
- Collections (Collection Runner)

Bornfight®

Challenges

#1: Reusing tests

#2: Grouping tests

#3: Skipping requests

#4: (Re)using custom functions

Bornfight®

#5: Simplifying tests grouping

#6: Using external libraries

#7: Validating against a schema

#8: HTTP requesting

#9: Debugging

Bornfight®

#10: Integrating tests with CI

Thank you