

Javier Alberto Romero Oviedo	Ingeniero de sistemas
Ejercicio práctico- Postman	Api https://rickandmortyapi.com/documentation

1. Hacer un listado de todos los personajes, y hacer un test con la información de un personaje que aparezca en esa respuesta (Tener en cuenta que deben hacer una búsqueda entre los diferentes objetos que hay embebidos en la respuesta).

Postman interface showing a GET request to the endpoint `{{urlMorty}}/character`. The request is successful (Status: 200 OK). The test results show two passed tests:

- PASS: El código de estado es 200
- PASS: Realiza búsqueda de personaje por nombre y verifica su Species

2. Hacer un listado de personajes por página (en la documentación se explica cómo hacer paginación), y hacer un test por aparición de personaje en esta página.

Postman interface showing a GET request to the endpoint `{{urlMorty}}/character?page={{numPage}}`. The request is successful (Status: 200 OK). The test results show two passed tests:

- PASS: El código de estado es 200
- PASS: Personajes obtenidos por página

Javier Alberto Romero Oviedo	Ingeniero de sistemas
Ejercicio práctico- Postman	Api https://rickandmortyapi.com/documentation

3. Traer un personaje por id, y hacer un test correspondiente

The screenshot shows a Postman interface for a GET request to the endpoint `{{urlMorty}}/character/{{idPersonaje}}`. The request is successful with a status of 200 OK. The tests section shows two tests that have passed:

- Test 1:** `pm.test("El código de estado es 200", function () { pm.response.to.have.status(200); });`
- Test 2:** `pm.test("validar id", ()=>{ pm.expect(true).to.eql(personID); });`

The test results show "PASS" for both tests.

➤ Traer un personajes por id, y hacer un test correspondiente

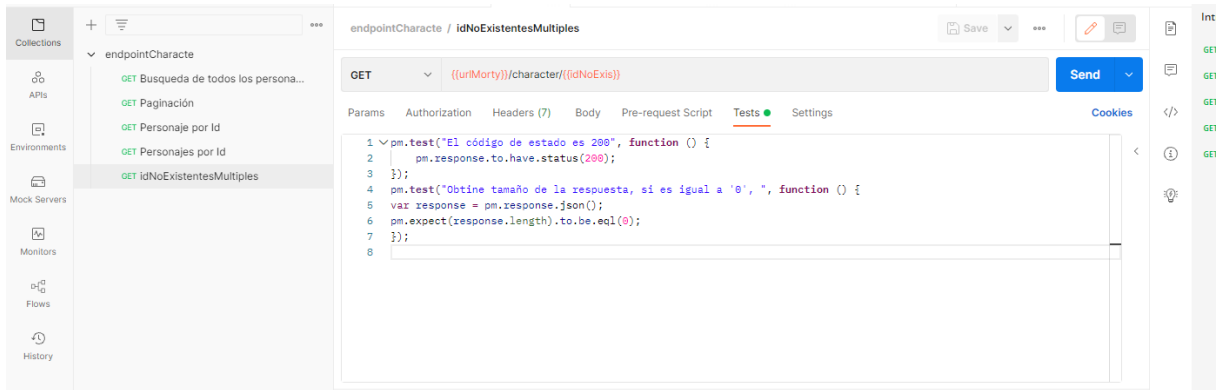
The screenshot shows a Postman interface for a GET request to the endpoint `{{urlMorty}}/character/{{idMul}}`. The request is successful with a status of 200 OK. The tests section shows two tests that have passed:

- Test 1:** `pm.test("Verificar código de estado 200", function () { pm.response.to.have.status(200); });`
- Test 2:** `pm.test("Validar varios ID", function () { var ids= pm.globals.get('idMul').split(',').map(Number); var idsObtenidos= pm.response.json().map(id=>id.id); pm.expect(ids.length).to.eql(idsObtenidos.length); ids.forEach(function(idPer) { pm.expect(ids).to.include(idPer); }); console.log("ID's Obtenidos: "+idsObtenidos); });`

The test results show "PASS" for both tests.

Javier Alberto Romero Oviedo	Ingeniero de sistemas
Ejercicio práctico- Postman	Api https://rickandmortyapi.com/documentation

4. Al momento de buscar múltiples Id, ingresando id's que no existen nos da una respuesta, la cual es un array vacío, hacer un test en donde se valide que se tiene una respuesta vacía



Documentación Postman:

<https://documenter.getpostman.com/view/19965346/Uz5ArJiK>

URL-Ejercicio: <https://www.postman.com/universal-equinox-765712/workspace/rickandmortyapi-com/overview>