API testing

Ngô Thị Kim Duyên

31/10/2024

HTTP status

The response contains one of the following statuses:

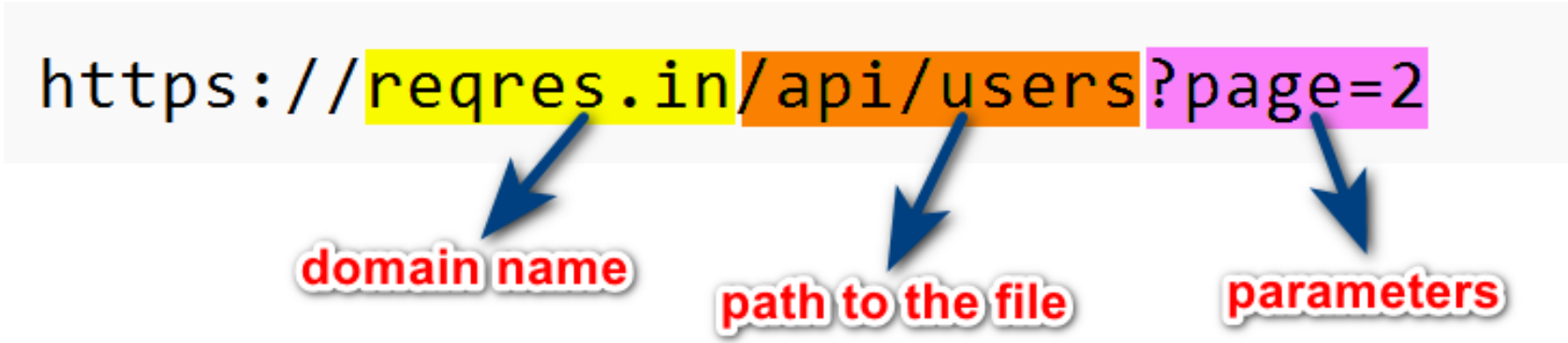
- **200**: Get data from the server successfully
- **201**: Successfully created and pushed data to the server
- **400(bad request):** The server cannot process the request from the user who likes inputting invalid data.
- **404(not found):** The server cannot find the requested resource. For example, the URL is incorrect or the resource does not exist.
- **500**: The error occurred from the server side

Use Postman

Manage multiple API and run test

- Postman app is a powerful feature that allows you to organize, share, and automate your API requests.
- Postman supports all methods including:
  1. GET: get data from a server
  2. POST: send data to a server
  3. PUT: update data to a server
  4. PATCH: update data to a server
  5. DELETE: delete data to a server

What is URL ?


https://reqres.in/api/users?page=2
domain name
path to the file
parameters

Create a new URL object

```
1  const url = new URL('https://reqres.in/api/users?page=2');
2  console.log(url);
```

Result:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

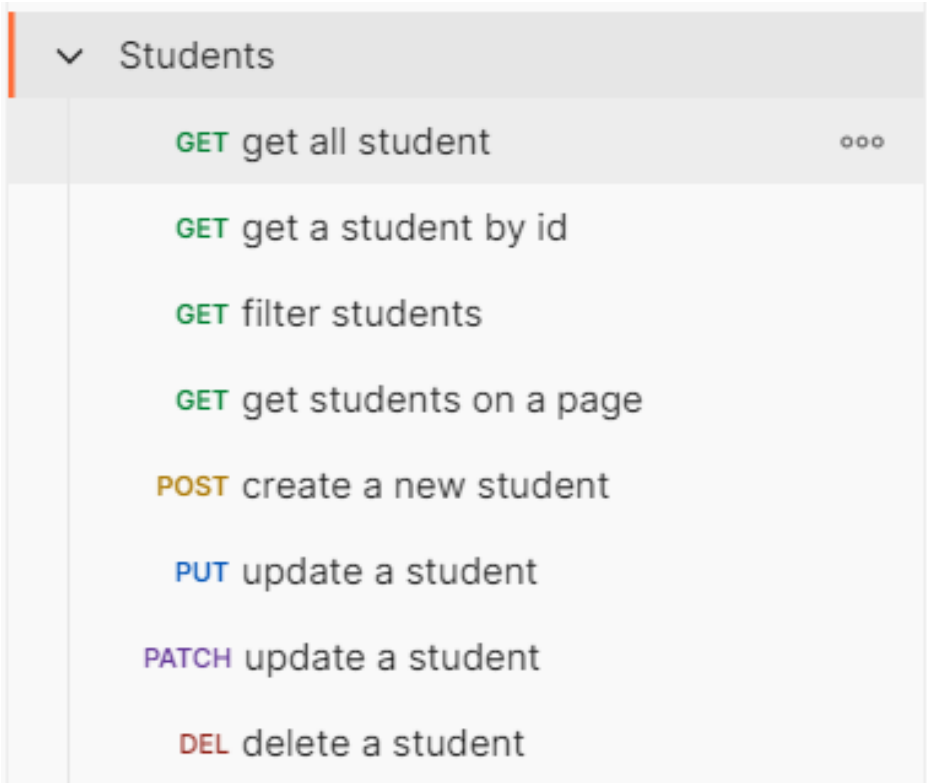
```
PS C:\Users\Admin\Documents\Learning24\web\nodejs> node index.js
URL {
  href: 'https://reqres.in/api/users?page=2',
  origin: 'https://reqres.in',
  protocol: 'https:',
  username: '',
  password: '',
  host: 'reqres.in',
  hostname: 'reqres.in',
  port: '',
  pathname: '/api/users',
  search: '?page=2',
  searchParams: URLSearchParams { 'page' => '2' },
  hash: ''
}
```

Methods API

Student APIs

Method	Resource	Has payload	Desc
GET	/students	NO	Get all students
GET	/students/:studentId	NO	Get one student by id
POST	/students	YES	Add new student
PUT	/students/:studentId	YES	Full update of a student
PATCH	/students/:studentId	YES	Paritial update of a student
DELETE	/students/:studentId	NO	Remove a student by id

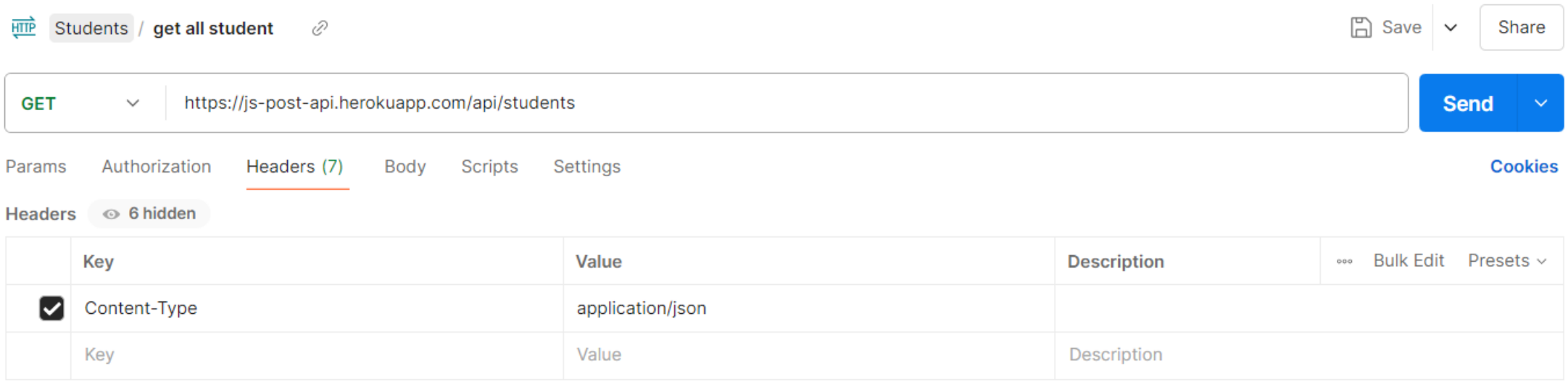
Use Postman


Students
GET get all student
GET get a student by id
GET filter students
GET get students on a page
POST create a new student
PUT update a student
PATCH update a student
DEL delete a student

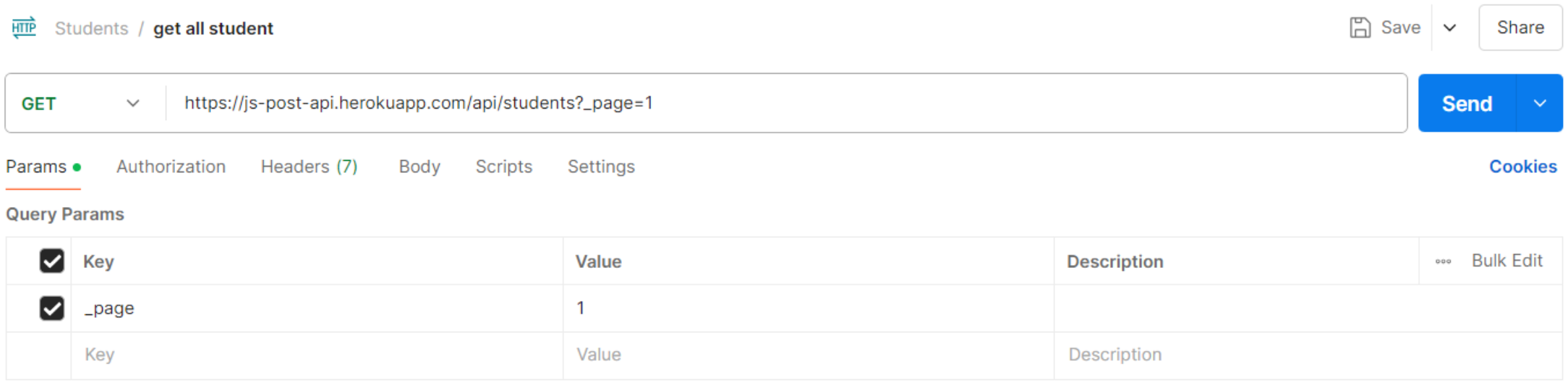
Get all students

Method GET

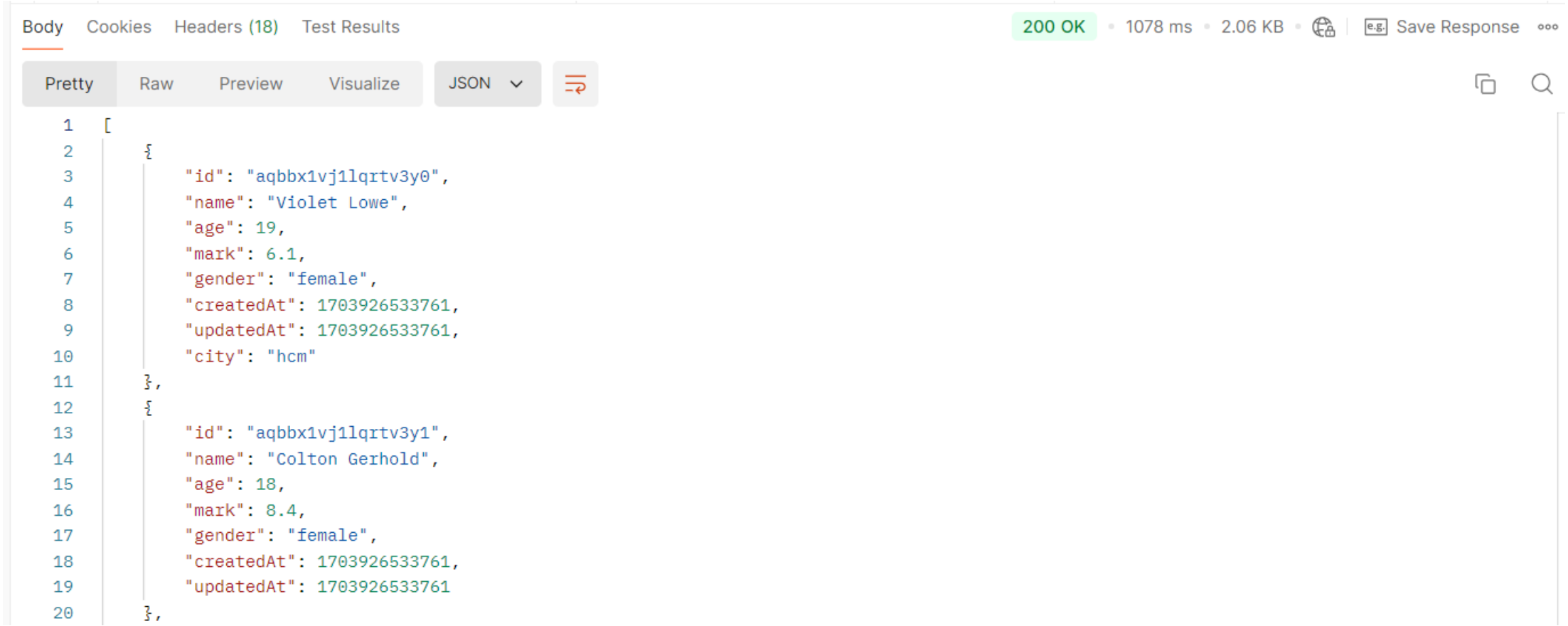
Header


Students / get all student
Save
Share
GET
https://js-post-api.herokuapp.com/api/students
Send
Params
Authorization
Headers (7)
Body
Scripts
Settings
Cookies
Headers
6 hidden
Key
Value
Description
Bulk Edit
Presets
Content-Type
application/json
Key
Value
Description

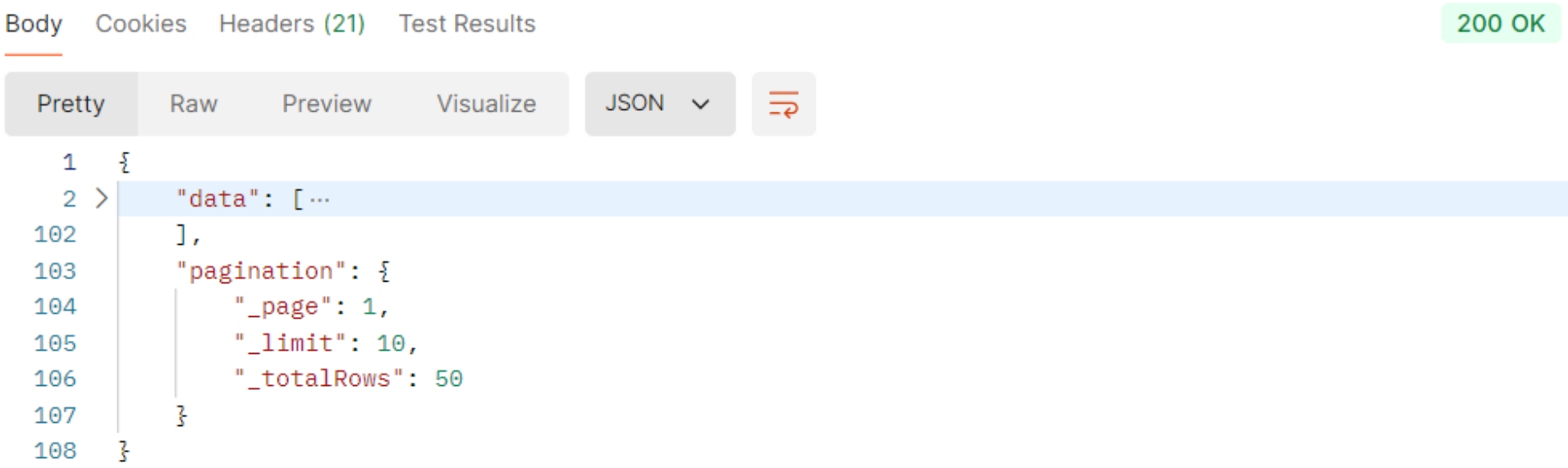
Params


Students / get all student
Save
Share
GET
https://js-post-api.herokuapp.com/api/students?_page=1
Send
Params
Authorization
Headers (7)
Body
Scripts
Settings
Cookies
Query Params
Key
Value
Description
Bulk Edit
_page
1
Key
Value
Description

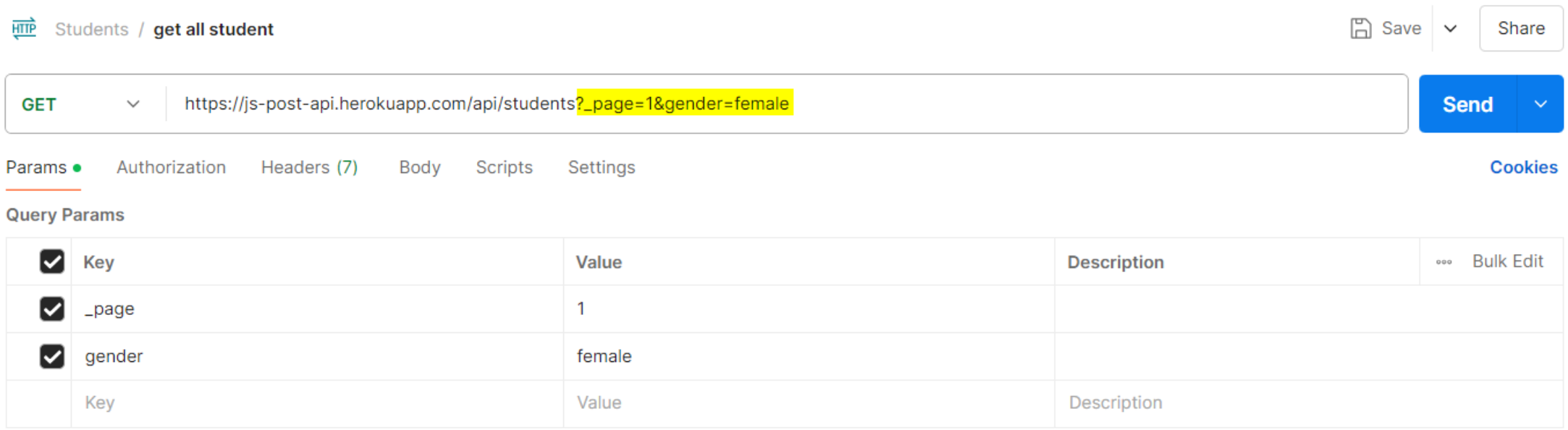
Show all data



Show data in a page

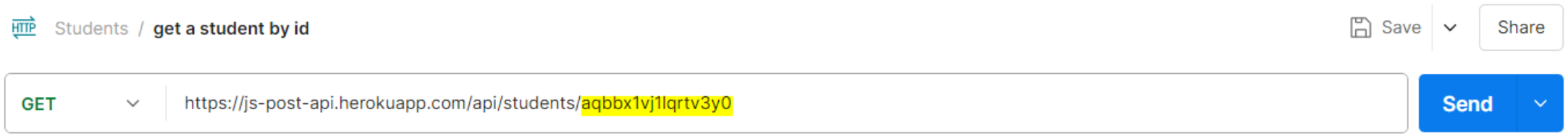


Filler attribute

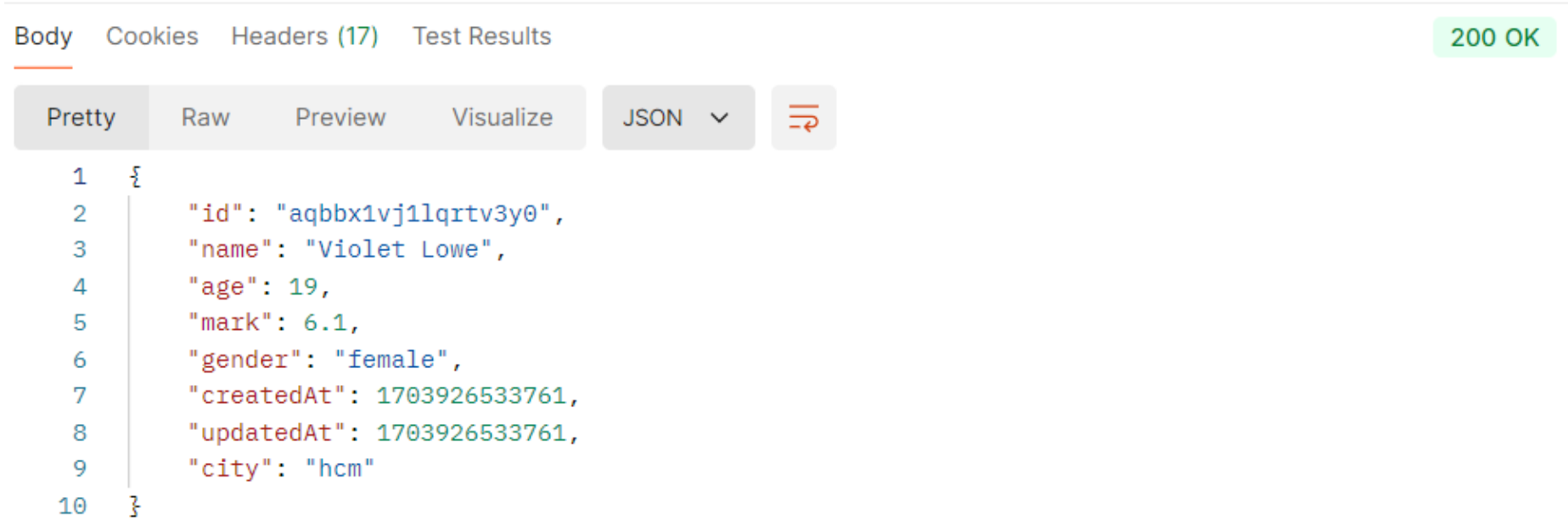


Get a student by id

Method GET /students/:studentId



Show data

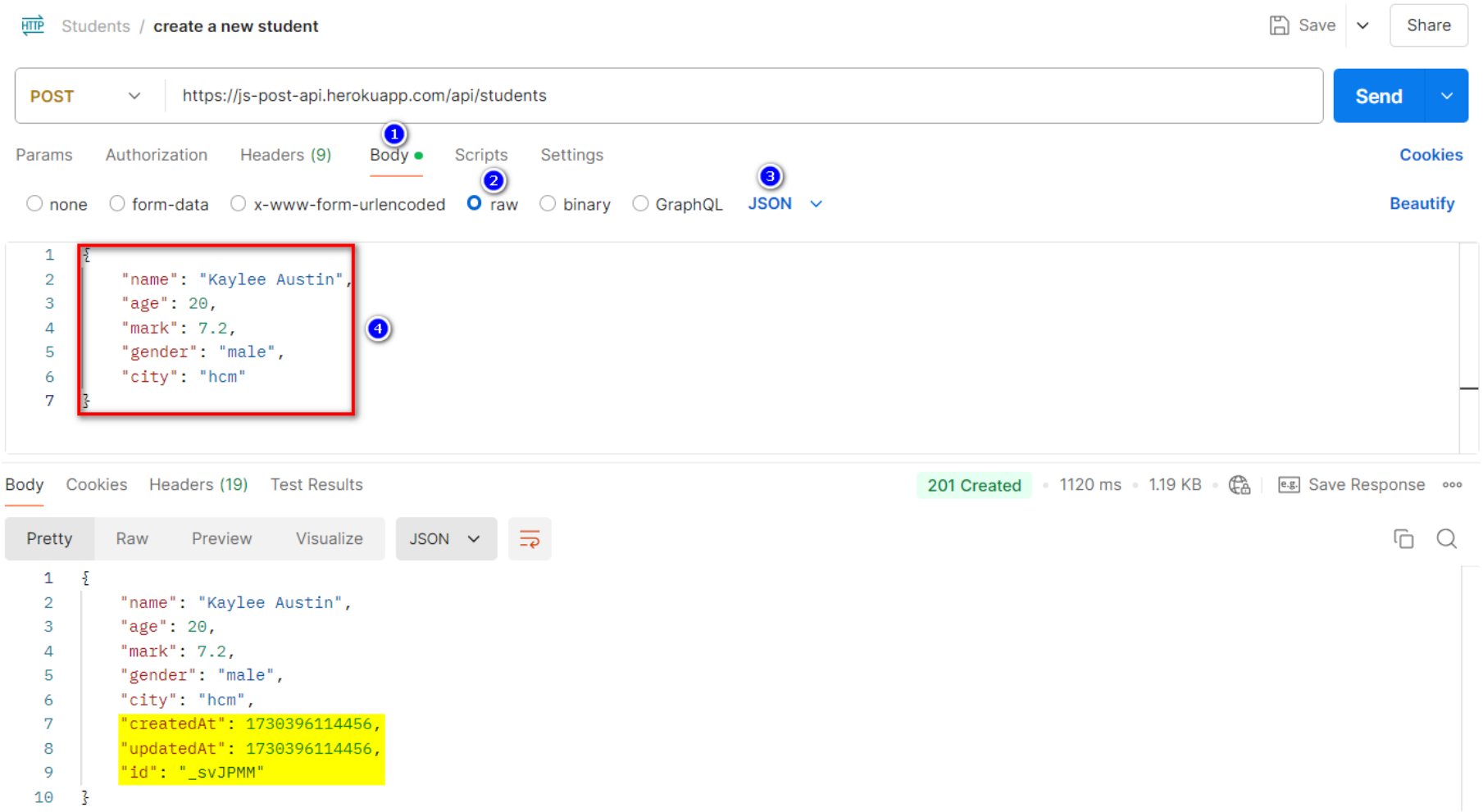

Body
Cookies
Headers (17)
Test Results
200 OK
Pretty
Raw
Preview
Visualize
JSON
{
"id": "aqbbx1vj1lqrtv3y0",
"name": "Violet Lowe",
"age": 19,
"mark": 6.1,
"gender": "female",
"createdAt": 1703926533761,
"updatedAt": 1703926533761,
"city": "hcm"
}

Create a new student

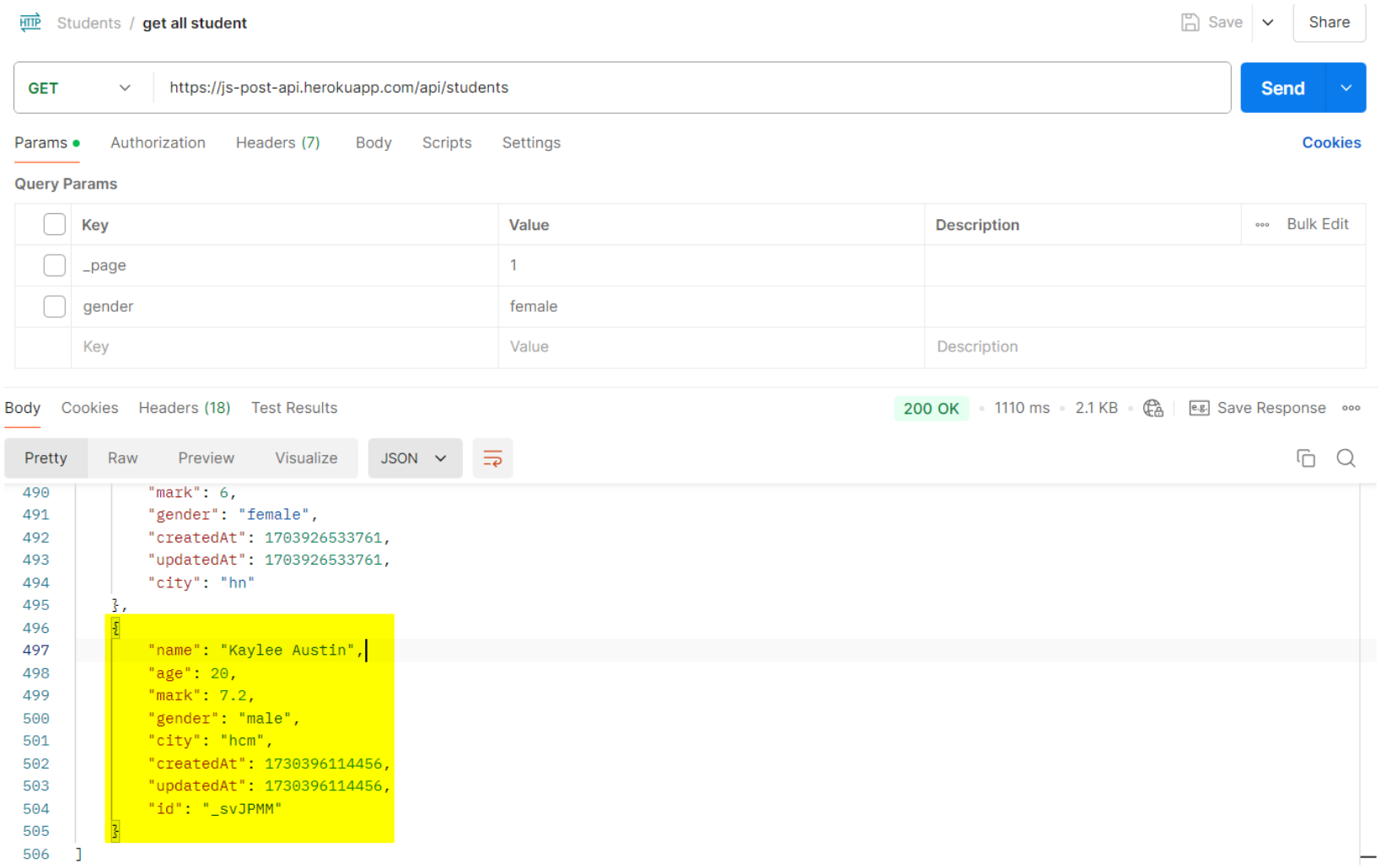
Method POST - /students

- Choose Body --> raw --> JSON

Method POST - /students



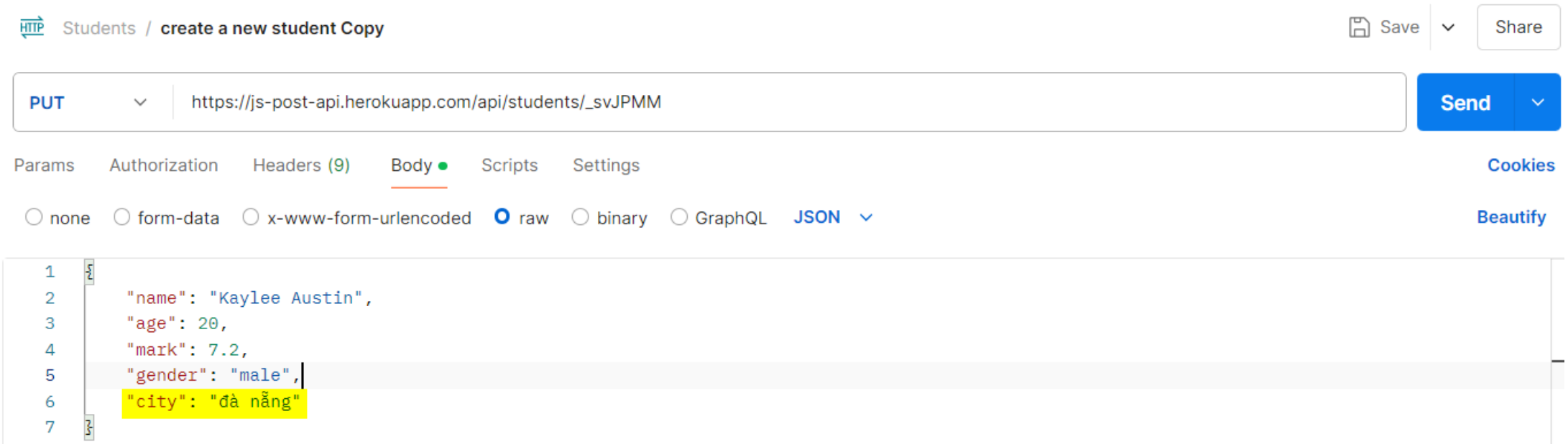
Check Result



Update a student

Method PUT

- A full update of a student



Result method PUT /students/:studentId

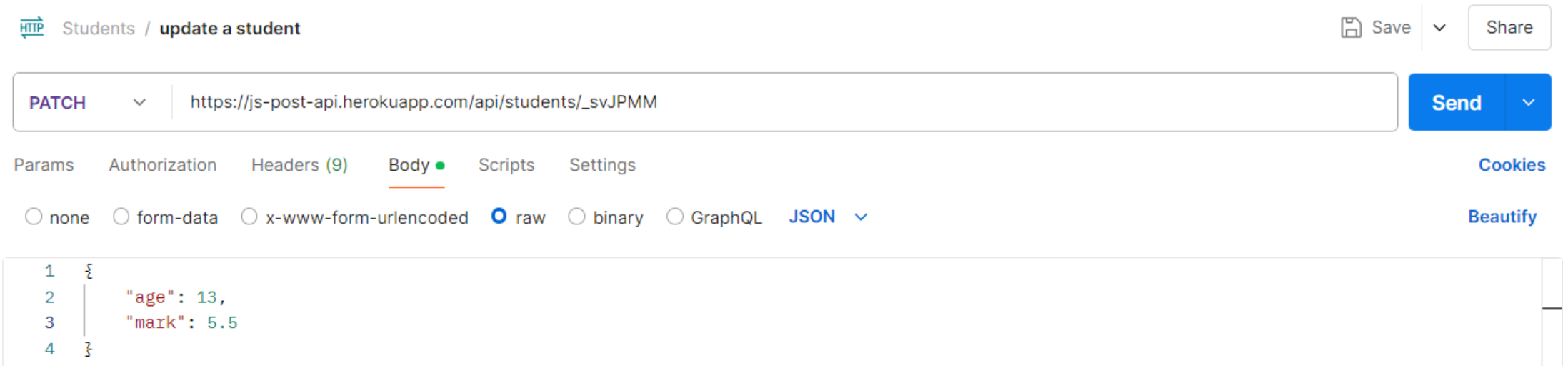
Body Cookies Headers (17) Test Results 200 OK

Pretty Raw Preview Visualize JSON

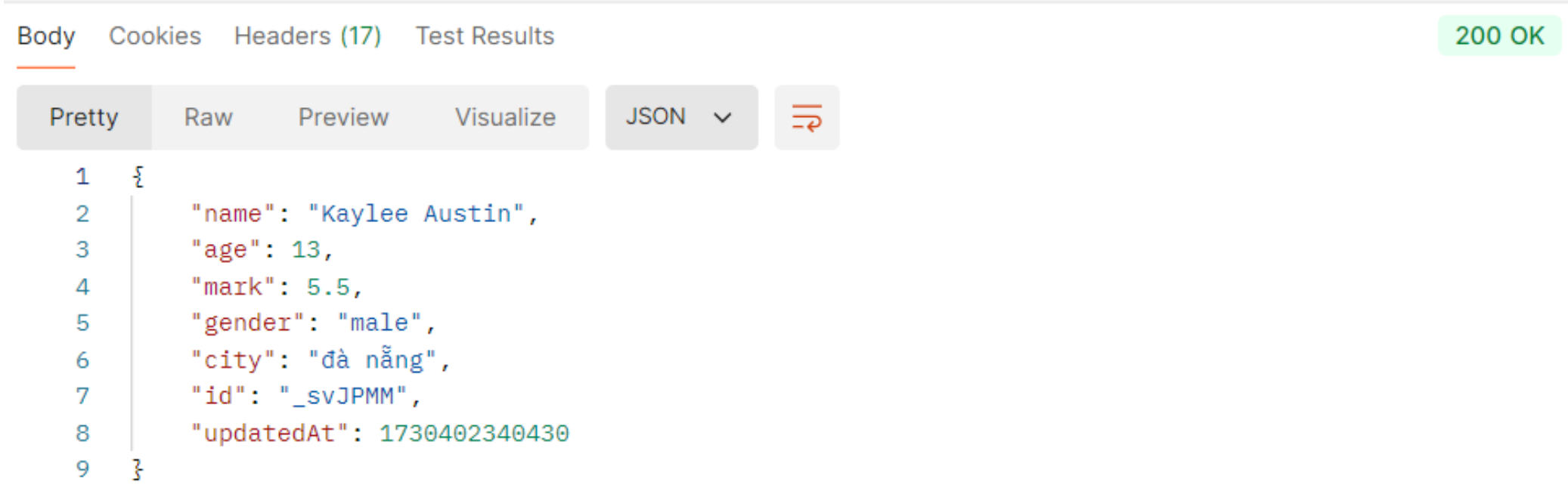

```
{
    "name": "Kaylee Austin",
    "age": 20,
    "mark": 7.2,
    "gender": "male",
    "city": "đà nẵng",
    "id": "_svJPMM"
}
```

Method **PATCH** /students/:studentId

- Paritial update of a student



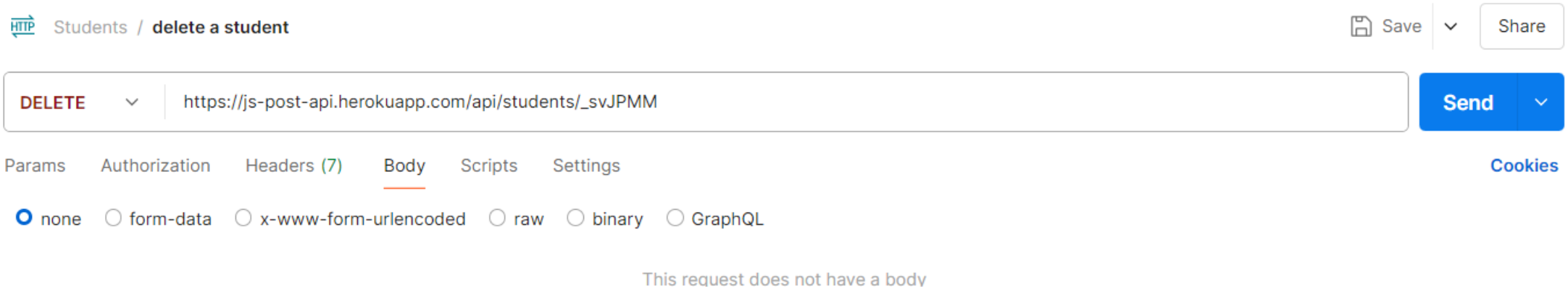
Result method PATCH



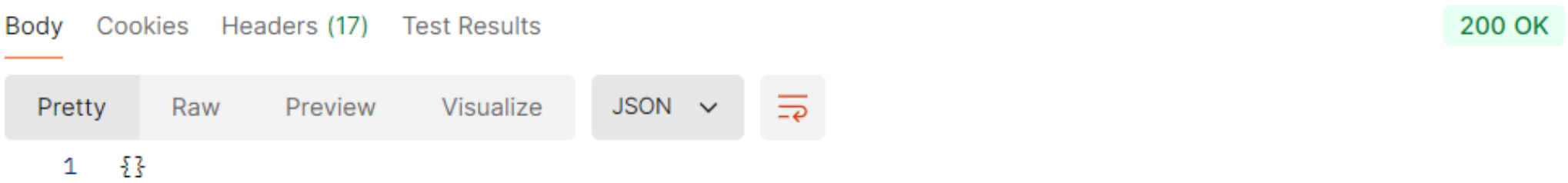
Delete a student by id

Method DELETE /students/:studentId

Body → none

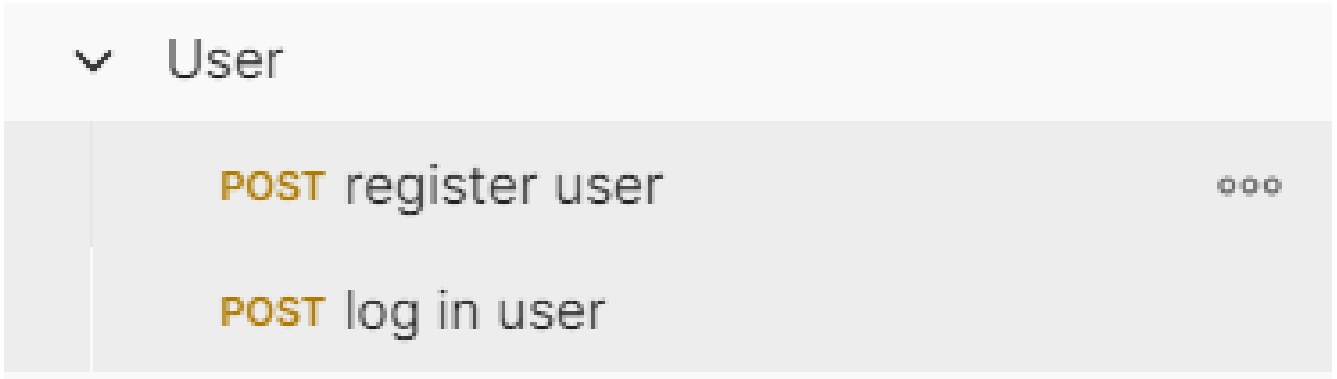


Result method Delete /students/:studentId



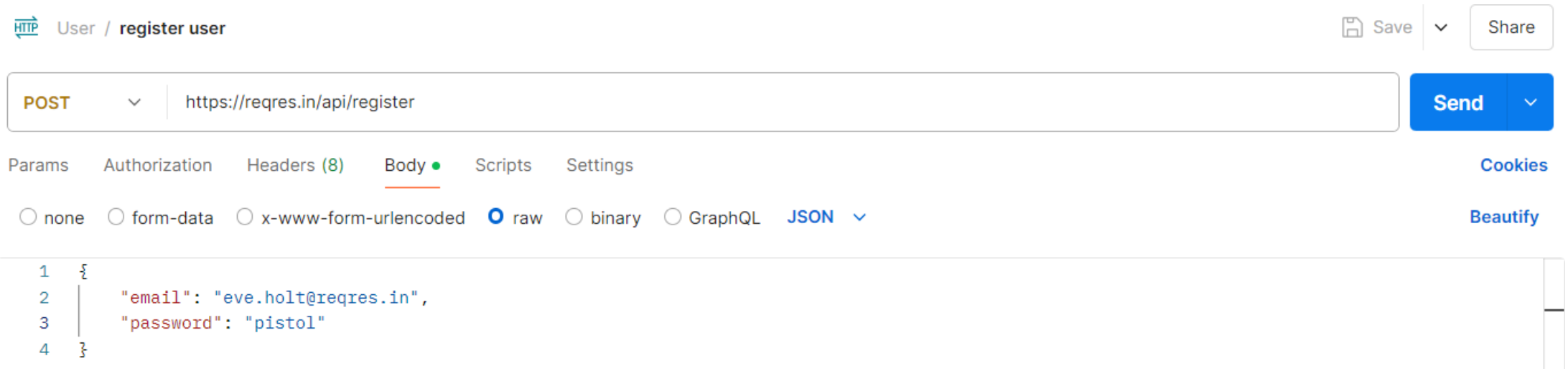
User authentication

Use Postman



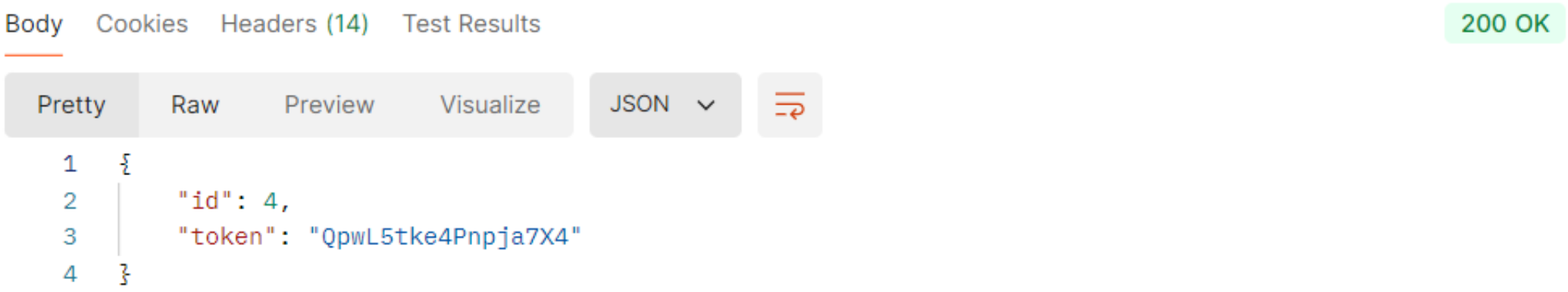
Register user

Method POST


User / register user
Save
Share
POST
https://reqres.in/api/register
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"email": "eve.holt@reqres.in",
"password": "pistol"
}

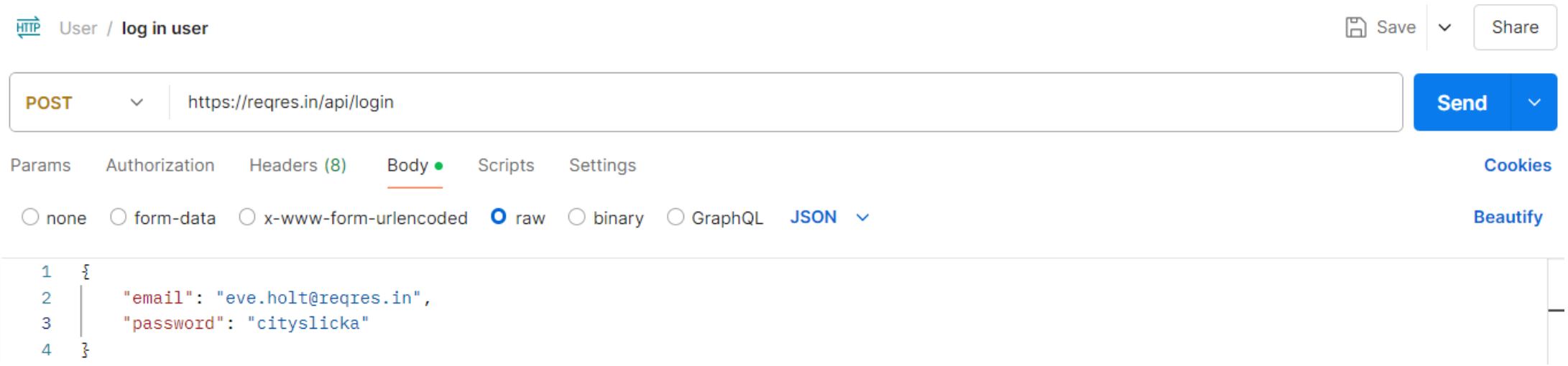
Result

- Return a token (jwt)



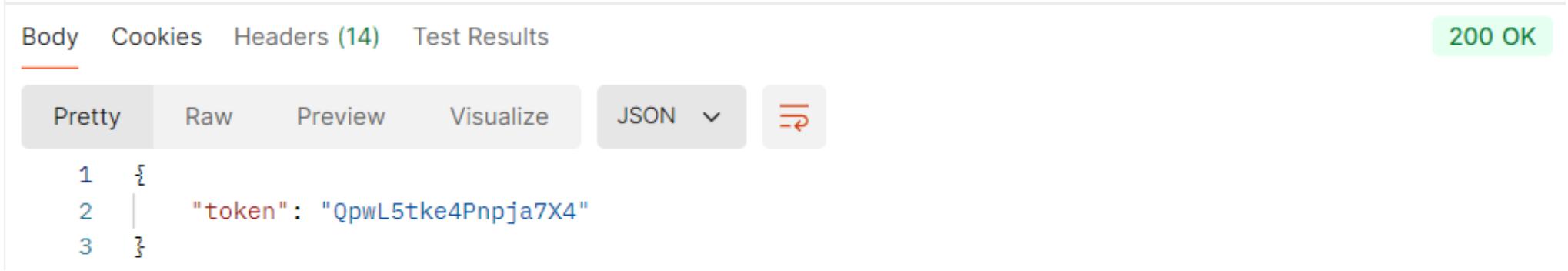
Log in user

Method POST


User / log in user
Save
Share
POST
https://reqres.in/api/login
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"email": "eve.holt@reqres.in",
"password": "cityslicka"
}

Result

- Return a token(jwt)



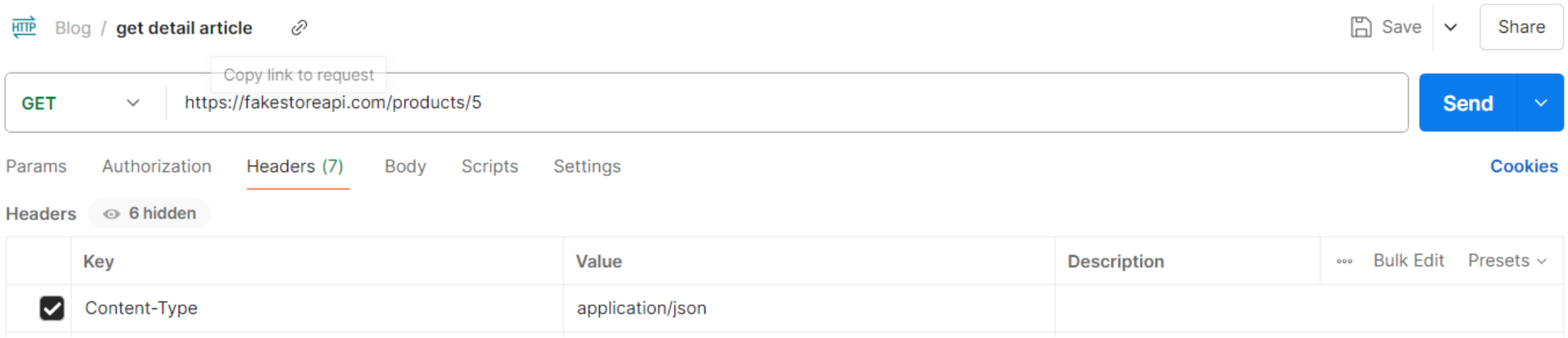
Blog

Get all articles

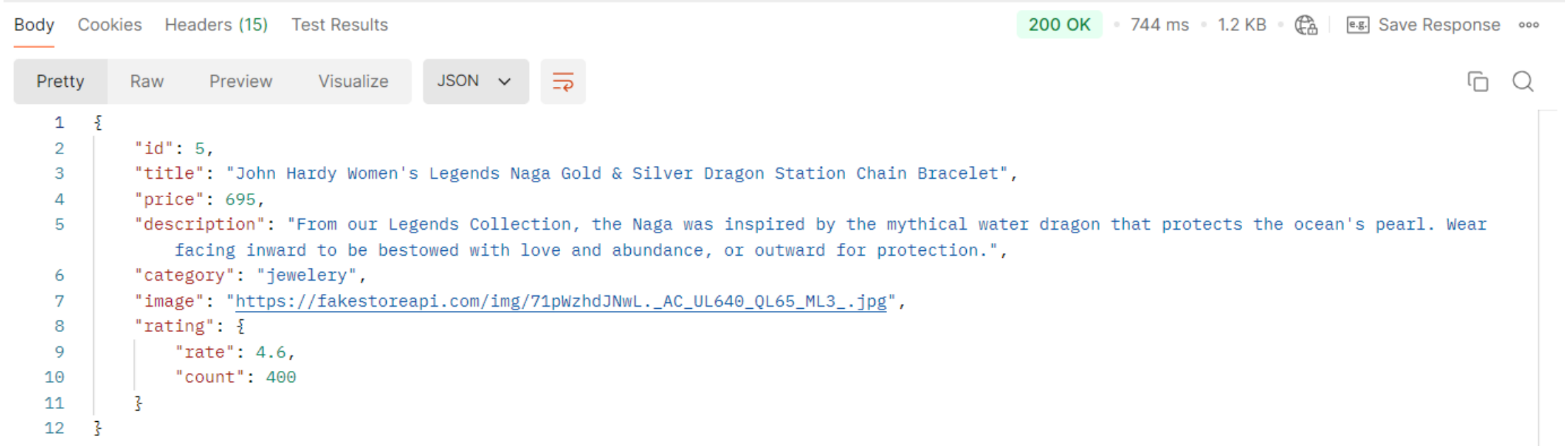
Result

```
[
    {
        "userId": 1,
        "id": 1,
        "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
        "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"
    },
    {
        "userId": 1,
        "id": 2,
        "title": "qui est esse",
        "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"
    },
    {
        "userId": 1,
        "id": 3,
        "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
        "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut"
```

Get a article by id

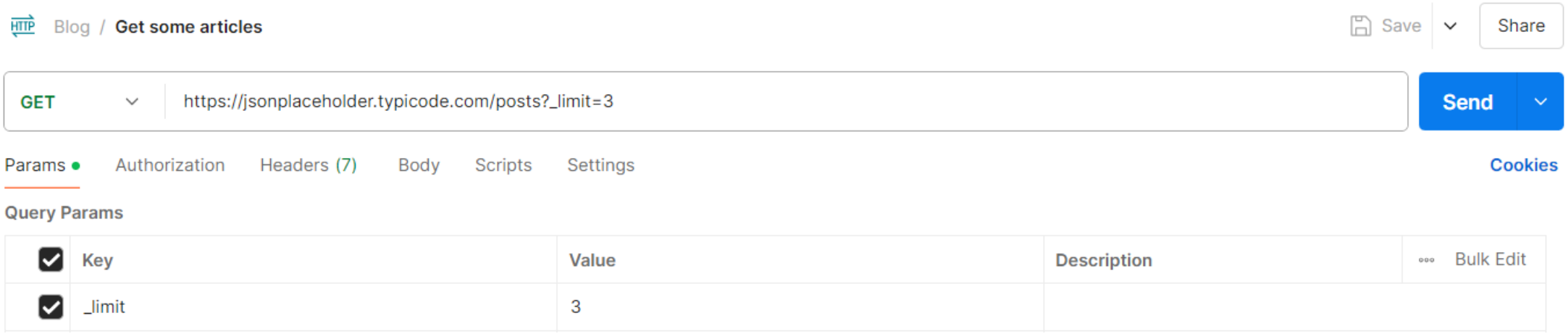


Result


Body
Cookies
Headers (15)
Test Results
200 OK
744 ms
1.2 KB
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"id": 5,
"title": "John Hardy Women's Legends Naga Gold & Silver Dragon Station Chain Bracelet",
"price": 695,
"description": "From our Legends Collection, the Naga was inspired by the mythical water dragon that protects the ocean's pearl. Wear facing inward to be bestowed with love and abundance, or outward for protection.",
"category": "jewelery",
"image": "https://fakestoreapi.com/img/71pWzhdJNwL._AC_UL640_QL65_ML3_.jpg",
"rating": {
"rate": 4.6,
"count": 400
}
}

Params

- Limit the number of articles



Params

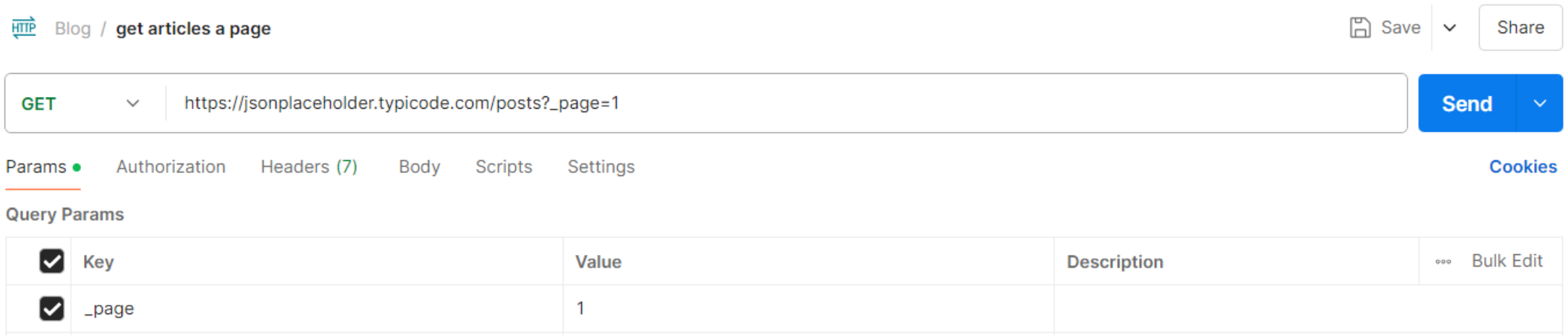
- Limit number of articles

Pretty-print ☑


```
[
  {
    "userId": 1,
    "id": 1,
    "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
    "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"
  },
  {
    "userId": 1,
    "id": 2,
    "title": "qui est esse",
    "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque
nisi nulla"
  },
  {
    "userId": 1,
    "id": 3,
    "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
    "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut"
  }
]
```

Params

- Get all articles on a page



Result

Pretty-print ☑


```
[
  {
    "userId": 1,
    "id": 1,
    "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
    "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"
  },
  {
    "userId": 1,
    "id": 2,
    "title": "qui est esse",
    "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque
nisi nulla"
  },
  {
    "userId": 1,
    "id": 3,
    "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
    "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut"
  },
  {
    "userId": 1,
    "id": 4,
    "title": "eum et est occaecati",
    "body": "ullam et saepe reiciendis voluptatem adipisci\nsit amet autem assumenda provident rerum culpa\nquis hic commodi nesciunt rem tenetur doloremque ipsam iure\nquis sunt voluptatem rerum illo velit"
  },
  {
    "userId": 1,
    "id": 5,
    "title": "nesciunt quas odio",
    "body": "repudiandae veniam quaerat sunt sed\nalias aut fugiat sit autem sed est\nvoluptatem omnis possimus esse voluptatibus quis\nest aut tenetur dolor neque"
  },
  {
    "userId": 1,
    "id": 6,
    "title": "dolorem eum magni eos aperiam quia",
    "body": "ut aspernatur corporis harum nihil quis provident sequi\nmollitia nobis aliquid molestiae\nperspiciatis et ea nemo ab reprehenderit accusantium quas\nvoluptate dolores velit et doloremque molestiae"
  },
  {
    "userId": 1,
    "id": 7,
    "title": "magnam facilis autem",
    "body": "dolore placeat quibusdam ea quo vitae\nmagni quis enim qui quis quo nemo aut saepe\nquidem repellat excepturi ut quia\nsunt ut sequi eos ea sed quas"
  },
  {
    "userId": 1,
    "id": 8,
    "title": "dolorem dolore est ipsam",
    "body": "dignissimos aperiam dolorem qui eum\nfacilis quibusdam animi sint suscipit qui sint possimus cum\nquaerat magni maiores excepturi\nipsam ut commodi dolor voluptatum modi aut vitae"
  },
  {
    "userId": 1,
    "id": 9,
    "title": "nesciunt iure omnis dolorem tempora et accusantium",
    "body": "consectetur animi nesciunt iure dolore\nenim quia ad\nveniam autem ut quam aut nobis\net est aut quod aut provident voluptas autem voluptas"
  },
  {
    "userId": 1,
    "id": 10,
    "title": "optio molestias id quia eum",
    "body": "quo et expedita modi cum officia vel magni\ndoloribus qui repudiandae\nvero nisi sit\nquos veniam quod sed accusamus veritatis error"
  }
]
```

Reference API

- CRUD students: https://js-post-api.herokuapp.com/api/students
- Register and login: https://reqres.in/
- Blog: https://jsonplaceholder.typicode.com/posts