

CHALLENGE BACKEND - NodeJs



API –DISNEY -

Documentación

End-Point utilizado para rodo el proyecto:

<https://api.disneyapi.dev/characters>

Herramientas utilizadas : Node.js, Express ,Sequelize,Postgres
,NodeMails.js

Autor : Daniel Perco

1. Listado de Personajes

El listado deberá mostrar:

- Imagen.
- Nombre.

The screenshot displays the Postman application interface. The top navigation bar includes the Postman logo, menu items (File, Edit, View, Help), and a search bar. The main workspace is titled "My Workspace" and shows a list of collections on the left sidebar, including "disney", "dog_temper", "expres", "food", "food_recetas", and "videogames". The "videogames" collection is selected, and a request is configured for the endpoint `http://localhost:3001/characters` using the GET method. The request body is set to JSON. The response is displayed in the "Body" tab, showing a JSON array of three character objects. The status bar at the bottom indicates a successful response (200 OK) with a time of 2.94 s and a size of 21.27 KB.

Postman

File Edit View Help

Home Workspaces API Network Reports Explore

Search Postman

Upgrade

My Workspace

New Import

videogames / http://localhost:3001/videogames

Save

GET http://localhost:3001/characters

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Body Cookies Headers (11) Test Results

Status: 200 OK Time: 2.94 s Size: 21.27 KB Save Response

```
1 {
2   {
3     "image": "https://static.wikia.nocookie.net/disney/images/6/61/Olu_main.png",
4     "name": "'Olu Mel"
5   },
6   {
7     "image": "https://static.wikia.nocookie.net/disney/images/5/51/Giffany.png",
8     "name": ".GIFfany"
9   },
10  {
11    "image": "https://static.wikia.nocookie.net/disney/images/8/80/Profile_-_627.png",
12    "name": "627"
13  },
14  {
15    "image": "https://static.wikia.nocookie.net/disney/images/7/77/9-eye.png"
```

Creación, Edición y Eliminación de Personajes (CRUD)

Creación

The screenshot displays a REST client interface with the following details:

- Method:** POST
- URL:** http://localhost:3001/characters
- Body:** A JSON object representing a character:

```
1 {  
2   "image": "https://i.pinimg.com/236x/ce/5f/53/  
   ce5f53437e291c48705428721406985c--cartoons-characters-looney-tunes-characters.jpg",  
3   "id": "0",  
4   "name": "tuwty",  
5   "age": "60",  
6   "weight": "2",  
7   "history": "antigua",  
8   "pic_asoc": "todas",  
9   "tv_asoc": "series"  
10 }
```
- Response:** A JSON object indicating success:

```
1 {  
2   "message": "personaje creado"  
3 }
```
- Status:** 200 OK
- Time:** 117 ms
- Size:** 498 B
- Buttons:** Send, Beautify, Save Response, Copy, Search.

Tabla de Personajes

pgAdmin

File Object Tools Help

Browser

- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (6)
 - character_movie
 - characters
 - genres
 - genre_movie
 - movie_genre
 - movies
 - Trigger Functions

Dashboard Properties SQL Statistics Dependencies Dependents public.charact... public.charact... public.charact...

disneydb/postgres@PostgreSQL 14

Query Editor Query History

```
1 SELECT * FROM public.characters
2 ORDER BY iddb ASC
```

Data Output Explain Messages Notifications

	iddb [PK] uuid	image text	id integer	name character varying (255)	age character varying (255)	weight character varying (255)	history charac
1	1db872d7-1520-4adf...	https://media.revistagq.com/photos/5d5130eef1043a00092...	1	simpson	50	78	inmen
2	4ba936e0-4173-4f60...	https://i.pinimg.com/236x/ce/5f/53/ce5f53437e291c48705...	0	tuwty	60	2	antigu
3	8bdaf804-3557-4e1f...	https://static.wikia.nocookie.net/disney/images/5/51/Giffan...	2	pato	25	230	prueba
4	b8dca165-b5e8-450...	https://previews.123rf.com/images/hermandesign2015/her...	3	mapache	5	7	de dis

Modificar un Personaje

The screenshot shows the Postman application interface. The left sidebar contains the 'My Workspace' section with a list of collections: 'disney', 'dog_temper', 'expres', 'food', 'food_recetas', and 'videogames'. The 'videogames' collection is expanded, showing a 'GET http://localhost:3001/videogames' request. The main workspace displays a 'PUT' request to 'http://localhost:3001/characters/0'. The request body is a JSON object with the following fields: 'image', 'id', 'name', 'age', 'weight', and 'history'. The 'age' field is highlighted with a red arrow. The response is a JSON object with the field 'message' set to 'personaje modificado'.

Postman

File Edit View Help

Home Workspaces API Network Reports Explore

Search Postman

Invite

Upgrade

My Workspace

New Import

collections

APIs

Environments

Mock Servers

Monitors

Flows

History

videogames / http://localhost:3001/videogames

Save

PUT http://localhost:3001/characters/0

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "image": "https://i.pinimg.com/236x/ce/5f/53/ce5f53437e291c48705428721406985c--cartoons-characters-looney-tunes-characters.jpg",
3   "id": "0",
4   "name": "tuwty",
5   "age": "65",
6   "weight": "2",
7   "history": "antigua".
}
```

Body Cookies Headers (11) Test Results

Status: 200 OK Time: 134 ms Size: 502 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "personaje modificado"
3 }
```

pgAdmin 4

pgAdmin File Object Tools Help

Browser

- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (6)
 - character_movie
 - characters
 - genres
 - genre_movie
 - movie_genre
 - movies
 - Trigger Functions

Dashboard Properties SQL Statistics Dependencies Dependents public.charact... public.charact... public.charact...

disneydb/postgres@PostgreSQL 14

Query Editor Query History

```
1 SELECT * FROM public.characters
2 ORDER BY iddb ASC
```

Data Output Explain Messages Notifications

	iddb [PK] uuid	image text	id integer	name character varying (255)	age character varying (255)	weight character va
1	1db872d7-1520-4adf-a33d-5ce...	https://media.revistagq.com/photos/5d5130eef1043a000920d7de/16:9...	1	simpson	50	78
2	4ba936e0-4173-4f60-b2ad-cc5...	https://i.pinimg.com/236x/ce/5f/53/ce5f53437e291c48705428721406...	0	tuwty	65	2
3	8bdaf804-3557-4e1f-a1e5-d68...	https://static.wikia.nocookie.net/disney/images/5/51/Giffany.png	2	pato	25	230
4	b8dca165-b5e8-4506-916d-81...	https://previews.123rf.com/images/hermandesign2015/hermandesign...	3	mapache	5	7

pgAdmin 4

pgAdmin File Object Tools Help

Browser

- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (6)
 - character_movie
 - characters
 - genres
 - genre_movie
 - movie_genre
 - movies
 - Trigger Functions

Dashboard Properties SQL Statistics Dependencies Dependents public.charact... public.charact... public.charact...

disneydb/postgres@PostgreSQL 14

Query Editor Query History

```
1 SELECT * FROM public.characters
2 ORDER BY iddb ASC
```

Data Output Explain Messages Notifications

	iddb [PK] uuid	image text	id integer	name character varying (255)	age character varying (255)	weight character varying (255)
1	1db872d7-1520-4adf-a33d-5ce...	https://media.revistagq.com/photos/5d5130eef1043a000920d7de/16:9...	1	simpson	50	78
2	4ba936e0-4173-4f60-b2ad-cc5...	https://i.pinimg.com/236x/ce/5f/53/ce5f53437e291c48705428721406...	0	tuwty	65	2
3	8bdaf804-3557-4e1f-a1e5-d68...	https://static.wikia.nocookie.net/disney/images/5/51/Giffany.png	2	pato	25	230
4	b8dca165-b5e8-4506-916d-81...	https://previews.123rf.com/images/hermandesign2015/hermandesign...	3	mapache	5	7

Eliminar un Personaje

The screenshot shows the Postman application interface. The top navigation bar includes the Postman logo, menu items (File, Edit, View, Help), and tabs for Home, Workspaces, API Network, Reports, and Explore. A search bar labeled "Search Postman" is present. On the right side of the top bar, there are icons for a green checkmark, an "Invite" button, a refresh icon, a settings gear, a bell, and a "Upgrade" button.

The left sidebar is titled "My Workspace" and contains a "New" button, an "Import" button, and a list of collections: "disney", "dog_temper", "expres", "food", "food_recetas", and "videogames". The "videogames" collection is expanded, showing a "GET http://localhost:3001/videogames" request.

The main workspace area displays a "DELETE" request to the endpoint "http://localhost:3001/characters/2". The request is configured with the following options: Params, Authorization, Headers (8), Body (selected), Pre-request Script, Tests, and Settings. The body is set to "raw" and "JSON".

The request body is a JSON object with the following fields:

```
1 {
2   "image": "https://i.pinimg.com/236x/ce/5f/53/
3     ce5f53437e291c48705428721406985c--cartoons-characters-looney-tunes-characters.jpg",
4   "id": "0",
5   "name": "tuwty",
6   "age": "65",
7   "weight": "2",
8   "history": "antigua".
9 }
```

The response section at the bottom shows the "Body" tab selected, displaying a JSON object with a single field:

```
1 {
2   "message": "personaje eliminado"
3 }
```

The status bar at the bottom right indicates "Status: 200 OK", "Time: 437 ms", "Size: 501 B", and a "Save Response" button.

pgAdmin 4

File Object Tools Help

Browser

- Aggregates
- Collations
- Domains
- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (6)
 - character_movie
 - characters
 - genres
 - genre_movie
 - movie_genre
 - movies
- Trigger Functions
- Types
- Views
- Subscriptions
- dogs

Dashboard Properties SQL Statistics Dependencies Dependents public.charact... public.charact... public.charact...

Query Editor Query History

```
1 SELECT * FROM public.characters
2 ORDER BY iddb ASC
```

Data Output Explain Messages Notifications

	iddb [PK] uuid	image text	id integer	name character varying (255)	age character varying (255)	weight charac
1	1db872d7-1520-4adf-a3...	https://media.revistagq.com/photos/5d5130eef1043a000920d7ce/16:9/w_2560%2...	1	simpson	50	78
2	4ba936e0-4173-4f60-b2...	https://i.pinimg.com/236x/ce/5f/53/ce5f53437e291c48705428721406985c--cartoo...	0	tuwty	65	2
3	b8dca165-b5e8-4506-9...	https://previews.123rf.com/images/hermandesign2015/hermandesign20151606/he...	3	mapache	5	7

2 eliminado

Búsqueda Query por name

The screenshot shows a REST client interface. At the top, a GET request is configured to `http://localhost:3001/characters?name=Ada`, with a red arrow pointing to the query parameter. Below the URL bar, tabs for Params, Authorization, Headers (9), Body, Pre-request Script, Tests, and Settings are visible. The 'Body' tab is selected, and the content type is set to 'JSON'. A 'Send' button is on the right. Below the tabs, a status bar shows 'Status: 200 OK', 'Time: 3.11 s', and 'Size: 706 B'. The response body is displayed in 'Pretty' format as a JSON object. A red arrow points to the `"name": "Ada"` field in the response.

```
GET http://localhost:3001/characters?name=Ada
```

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON


1

Body Cookies Headers (11) Test Results Status: 200 OK Time: 3.11 s Size: 706 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "iddb": "7081b3e2-1889-4579-b1df-0b0c9c60aaf9",
4     "image": "https://cdn6.dibujos.net/images/listas/129/campanilla.jpg",
5     "id": 5,
6     "name": "Ada",
7     "age": "25",
8     "weight": "7",
9     "history": "magica",
10    "pic_asoc": "Tron : Uprising",
11    "tv_asoc": "Tron : Uprising"
12  }
13 ]
```


Búsqueda Query por edad



GET ⌵ http://localhost:3001/characters?age=25 

Params ● Authorization Headers (9) Body ● Pre-request Script Tests Settings Cookies



● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON ⌵ Beautify

1 5

Body Cookies Headers (11) Test Results  Status: 200 OK Time: 3.44 s Size: 1.04 KB [Save Response](#) ⌵

Pretty Raw Preview Visualize JSON ⌵  

```
6      "name": "Ada",
7      "age": "25",
8      "weight": "7",
9      "history": "magica",
10     "pic_asoc": "Tron : Uprising",
11     "tv_asoc": "Tron : Uprising"
12   },
13   {
14     "iddb": "29abfb20-c8cd-4064-bcaf-46988fa89ab8",
15     "image": "https://r1.abcmg.es/resizer/resizer.php?imagen=https%3A%2F%2Fstatic2.abc.
16               es%2Fmedia%2Fseries%2F0000%2F003%2F242%2Fk-c-agente-especial-2.jpg&nuevoancho=620&medio=abc",
17     "id": 6,
18     "name": "Candente Adams",
19     "age": "25",
```



Búsqueda Query por movies asociadas

The screenshot displays a REST client interface with a GET request to `http://localhost:3001/characters?movies=6`. The response body is `1 AC and DC`. Below the REST client, a database schema is shown with tables including `movies`. A data output table lists movie details, with the row for `Big Hero 6: The Series` (id 6) highlighted, corresponding to the `movies=6` query parameter.

REST Client Request:

- Method: GET
- URL: `http://localhost:3001/characters?movies=6`
- Body: none
- Status: 200 OK
- Time: 3.36 s
- Size: 468 B

REST Client Response:

```
1 AC and DC
```

Database Schema:

- Tables (8):
 - character_movie
 - character_user
 - characters
 - genres
 - genre_movie
 - movie_genre
 - movies**
 - users

Data Output Table:

	id [PK] integer	image text	title character varying (255)	date timestamp with time zone	rating double precision	char_asoc character varying (255)	createdAt timestamp
1	1	https://i0.wp.com/contraste.info/de...	K.C. Undercover	2018-11-22 21:00:00-03	4	Candence Adams	2022-05-20
2	2	https://static.wikia.nocookie.net/dh...	Tron : Uprising	2010-11-22 21:00:00-03	4	Ada	2022-05-20
3	3	https://prod-ripcut-delivery.disney-pl...	Saludos Amigos	2000-11-22 21:00:00-03	4	Guffy	2022-05-20
4	4	https://i.ytimg.com/vi/DfTACfocw1...	Aladdin films	2019-11-22 21:00:00-03	3	Prince Achmed	2022-05-20
5	5	https://pics.filmaffinity.com/lilo_stit...	Lilo & Stich	2017-10-22 21:00:00-03	3	Ace	2022-05-20
6	6	https://www.cinemascomics.com/...	Big Hero 6: The Series	2021-10-22 21:00:00-03	3	AC and DC	2022-05-20

MOVIES

Listado

k ▾

Reports

Explore

Search Postman

Invite

Upgrade

Import

<

GET http ●

GET con: ●

GET http ●

GET http ●

GET http ●

GET http ●

GET http ●

>

+

...

No Environment ▾

...

http://localhost:3001/movies

Save ▾

GET ▾

http://localhost:3001/movies

Send ▾

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

● none

● form-data

● x-www-form-urlencoded

● raw

● binary

● GraphQL

JSON ▾

Beautify

1

Body

Cookies

Headers (11)

Test Results

🌐

Status: 200 OK

Time: 3.02 s

Size: 29.26 KB

Save Response ▾

Pretty

Raw

Preview

Visualize

JSON ▾

1

2

3

4

5

6

7

8

9

10

11

12

13

{

"image": "https://static.wikia.nocookie.net/disney/images/6/61/0lu_main.png",

"pict_asoc": [],

"tv_asoc": []

},

{

"image": "https://static.wikia.nocookie.net/disney/images/5/51/Giffany.png",

"pict_asoc": [],

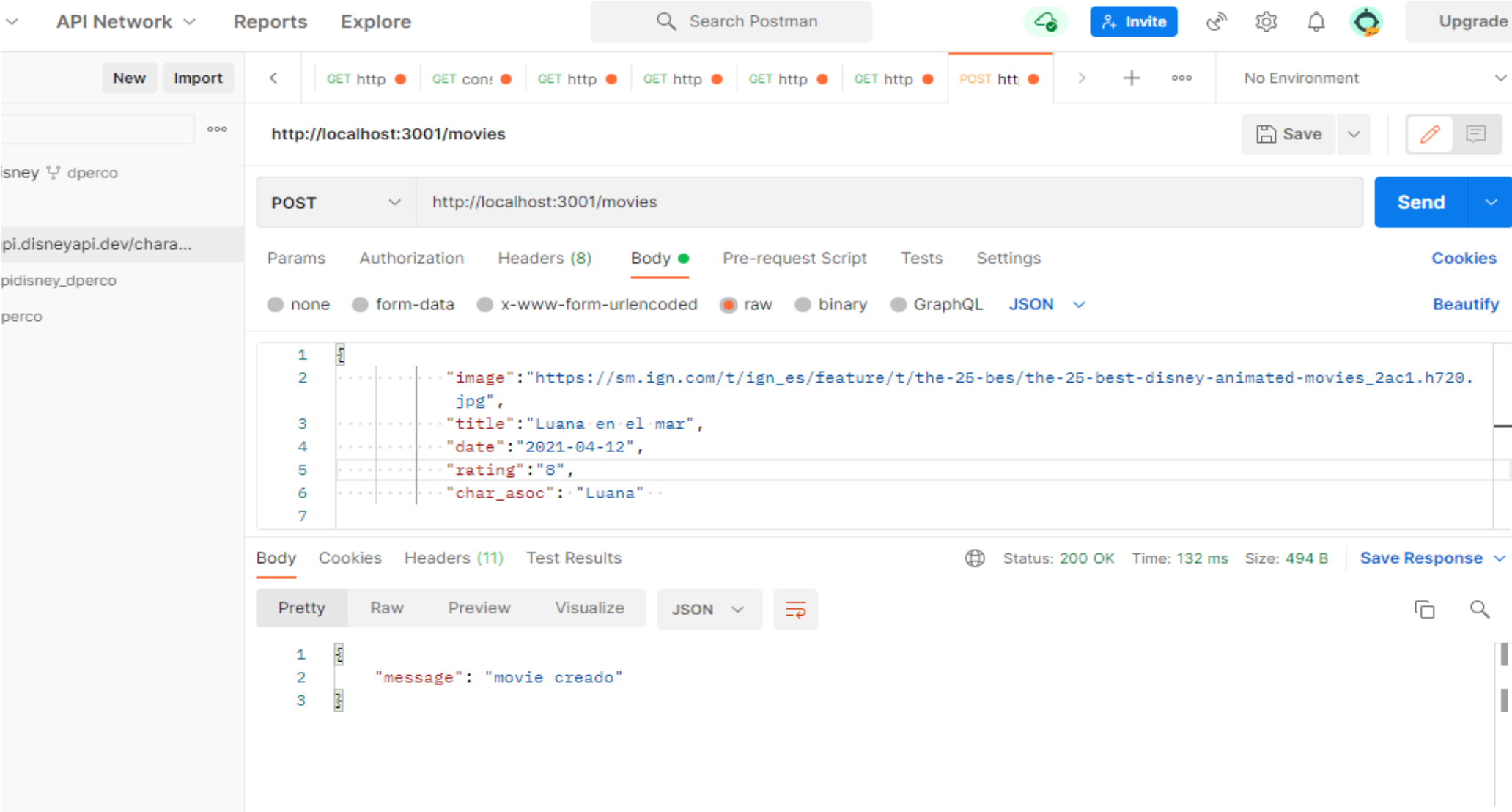
"tv_asoc": [

"Gravity Falls"

]

},

Detalle de Películas



Crear una Película

API Network Reports Explore Search Postman

New Import < GET http GET con GET http GET http GET http GET http POST htt > + ... No Environment

http://localhost:3001/movies Save

POST http://localhost:3001/movies Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON

```
2 ..... "id": "3",
3 ..... "image": "https://hips.hearstapps.com/hmg-prod.s3.amazonaws.com/images/peter-pan-1548255593.jpg",
4 ..... "title": "Peter Pan contra Garfio",
5 ..... "date": "2020-02-26",
6 ..... "rating": "7",
7 ..... "char_asoc": "Cptan Garfio , Peter Pan"
8
9
```

Body Cookies Headers (11) Test Results Status: 200 OK Time: 21 ms Size: 494 B Save Response

Pretty Raw Preview Visualize JSON

```
1
2 "message": "movie creado"
3
```

min

File ▾Object ▾Tools ▾Help ▾

🗄️

📊

🔍

🔎

➤

DashboardPropertiesSQLStatisticsDependenciesDependentspublic.charact...public.movies/...public.movies/disneydb/p...

🗄️

📁

💾

⌵

🗄️

🔍

⌵

📄

⌵

🗑️

✎

⌵

🔎

⌵

🔒

▶

⌵

👤

📄

⌵

🔄

🔄

🏠

⌵

📥

⌵

disneydb/postgres@PostgreSQL 14

Query EditorQuery History

1SELECT * FROM public.movies

2ORDER BY id ASC

Data OutputExplainMessagesNotifications

	id [PK] integer	image text	title character varying (255)	date timestamp with time zone	rating double precision
1	2	https://media.vandalsports.com/i/640x360/10-2018/20181010182952_1.jpg	la bestia y ella	2021-05-22 21:00:00-03	6
2	3	https://hips.hearstapps.com/hmg-prod.s3.amazonaws.com/images/peter-pan-1548255593.jpg	Peter Pan contra Garfio	2020-02-25 21:00:00-03	7

FTS Configurations

FTS Dictionaries

AaFTS Parsers

FTS Templates

Foreign Tables

Functions

Materialized Views

Operators

Procedures

1.3 Sequences

Tables (6)

character_movie

characters

genres

genre_movie

movie_genre

movies

Trigger Functions

Types

Views

Subscriptions

dogs

movies

postgres

videogames

rs
ates
bles

d Views

s
s

er_movie
ers

movie
genre

ctions

Query Editor Query History

```
1 SELECT * FROM public.movies
2 ORDER BY id ASC
```

Data Output Explain Messages Notifications

	id [PK] integer	image text	title character varying (255)	date timestamp with time zone	rating double precision	char_asoc character varying (255)	createdAt timestamp
1	2	https://media.vandalsports.com/i/640x360/10-2018/2...	la bestia y ella	2021-05-22 21:00:00-03	6	Lobo	2022-04-30
2	3	https://hips.hearstapps.com/hmg-prod.s3.amazonaws...	Peter Pan contra Garfio	2020-02-25 21:00:00-03	7	Cptan Garfio , Peter Pan	2022-04-30
3	4	https://media.timeout.com/images/100175529/750/4...	Ratatuil	2020-06-25 21:00:00-03	7	Rat y goodie	2022-04-30

PUT http://localhost:3001/movies/4

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
2 {
3   "id": "4",
4   "image": "https://media.timeout.com/images/100175529/750/422/image.jpg",
5   "title": "Ratatouille",
6   "date": "2020-06-26",
7   "rating": "8",
8   "char_asoc": "Rat y goodie"
9 }
```

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "personaje modificado"
3 }
```

Modificar una Película

- Templates
- Ign Tables
- tions
- erialized Views
- ators
- edures
- iences
- es (6)
- aracter_movie
- aracters
- ners
- hre_movie
- ovie_genre
- ovies
- er Functions
- s
- 's
- ons

Query Editor Query History

```
1 SELECT * FROM public.movies
2 ORDER BY id ASC
```

Data Output Explain Messages Notifications

	id [PK] integer	image text	title character varying (255)	date timestamp with time zone	rating double precision	char_asoc character varying (255)	created timesta
1	2	https://media.vandalSports.com/i/640x360/10-2018/2...	la bestia y ella	2021-05-22 21:00:00-03	6	Lobo	2022-04
2	3	https://hips.hearstapps.com/hmg-prod.s3.amazonaws...	Peter Pan contra Garfio	2020-02-25 21:00:00-03	7	Cptan Garfio , Peter Pan	2022-04
3	4	https://media.timeout.com/images/100175529/750/4...	Ratatouille	2020-06-25 21:00:00-03	8	Rat y goodie	2022-04

Borrar una Película

Postman interface showing a DELETE request to `http://localhost:3001/movies/4`.

Request Details:

- Method: DELETE
- URL: `http://localhost:3001/movies/4`
- Body: `"message": "personaje eliminado"`
- Format: JSON

Response Details:

- Status: 200 OK
- Time: 382 ms
- Size: 501 B
- Body: `"message": "personaje eliminado"`









The interface includes a sidebar with API collections, a top navigation bar with 'kspaces', 'API Network', 'Reports', and 'Explore', and a search bar labeled 'Search Postman'.

Data Output

Explain

Messages

Notifications

	 id [PK] integer 	image text	 title character varying (255) 	 date timestamp with time zone 	 rating double precision 
1	2	https://media.vandalsports.com/i/640x360/10-2018/20181010182952_1.jpg	la bestia y ella	2021-05-22 21:00:00-03	6
2	3	https://hips.hearstapps.com/hmg-prod.s3.amazonaws.com/images/peter-pan-1548255593.jpg	Peter Pan contra Garfio	2020-02-25 21:00:00-03	7

Buscar por Query una película por titulo

The screenshot shows a REST client interface. At the top, a GET request is configured with the URL `http://localhost:3001/movies?title=Saludos Amigos`. A red arrow points to this URL. Below the URL bar, tabs for Params, Authorization, Headers (9), Body, Pre-request Script, Tests, and Settings are visible. The 'Body' tab is selected, and the content type is set to 'JSON'. A 'Send' button is on the right. Below the tabs, a status bar shows 'Status: 200 OK', 'Time: 260 ms', and 'Size: 832 B'. The 'Body' tab is expanded, showing a JSON array with one object. A red arrow points to the 'title' field in the response object. The response is formatted as 'Pretty' JSON.

```
1  [
2
3    "id": 3,
4    "image": "https://prod-ripcut-delivery.disney-plus.net/v1/variant/disney/76A9863FABCCBBA27BFD5E961B8A0B343119A91CD1C521D1408E19113B75A596/scale?width=1200&aspectRatio=1.78&format=jpeg",
5    "title": "Saludos Amigos",
6    "date": "2000-11-23T00:00:00.000Z",
7    "rating": 4,
8    "char_asoc": "Guffy",
9    "createdAt": "2022-05-20T23:31:36.540Z",
10   "updatedAt": "2022-05-20T23:31:36.540Z"
11 ]
12
```

Buscar por Query una película por genero

The screenshot shows a REST client interface with a GET request to `http://localhost:3001/movies?genre=Accion`. The response is a JSON object representing a movie of the 'Accion' genre.

Request:

- Method: GET
- URL: `http://localhost:3001/movies?genre=Accion`
- Body: none

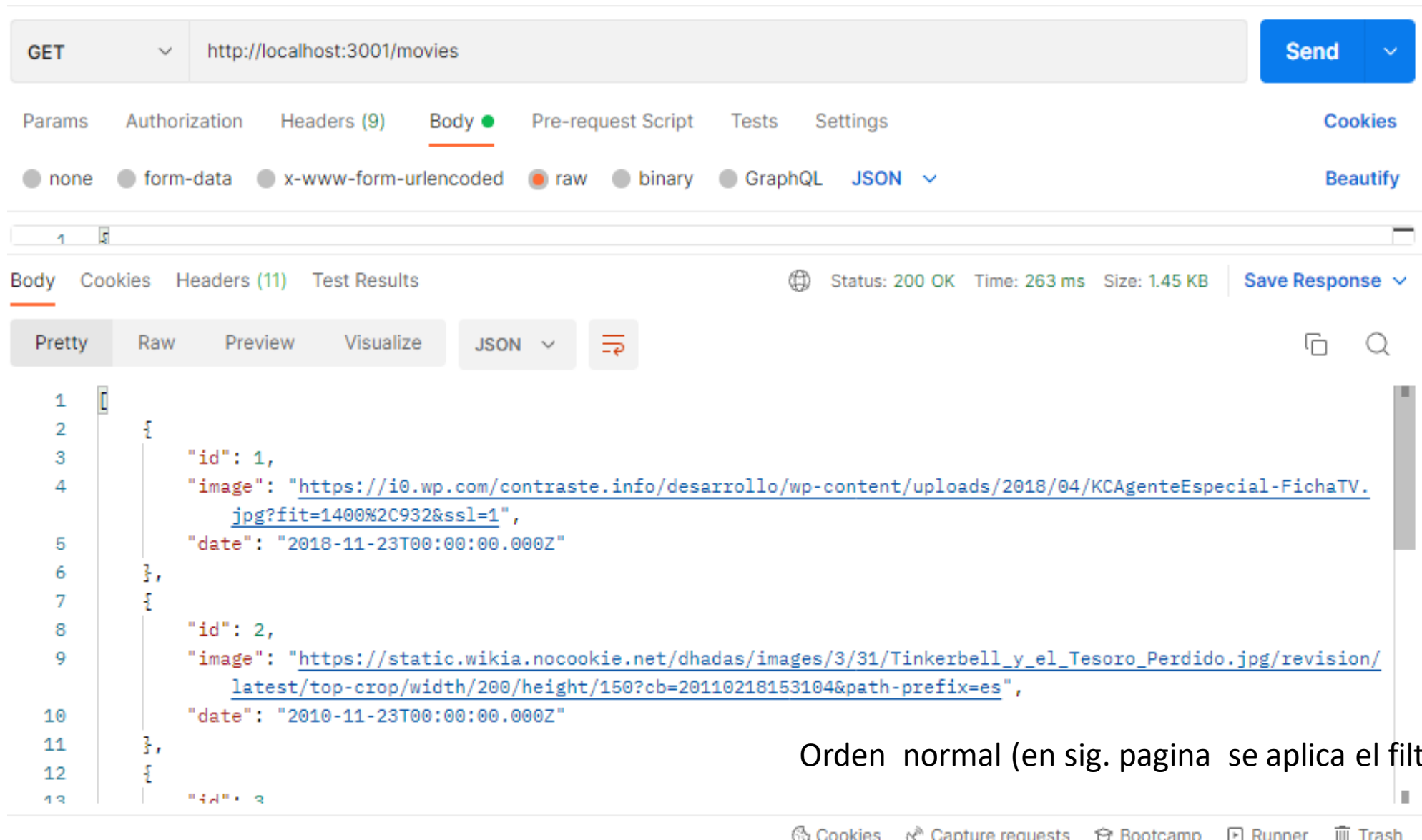
Response:

- Status: 200 OK
- Time: 259 ms
- Size: 735 B

JSON Response:

```
{
  "id": "15bb6ffd-1770-44c1-adb7-7bdfa457a3ac",
  "name": "Accion",
  "image": "https://cdn.hobbyconsolas.com/sites/navi.axelspringer.es/public/styles/hc_280x437/public/media/image/2016/12/aladdin.jpg?itok=UVjaTpMs",
  "pict_asociat": "Aladdin films",
  "tv_asoc": "Aladdin films"
}
```

Filtro por Query Ordenar películas Asc y Desc por fecha



GET ⌵ http://localhost:3001/movies Send ⌵

Params Authorization Headers (9) **Body** ● Pre-request Script Tests Settings Cookies

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ⌵ Beautify

1 5

Body Cookies Headers (11) Test Results 🌐 Status: 200 OK Time: 263 ms Size: 1.45 KB Save Response ⌵

Pretty Raw Preview Visualize **JSON** ⌵ 📄 🔍

```
1 {
2   {
3     "id": 1,
4     "image": "https://i0.wp.com/contraste.info/desarrollo/wp-content/uploads/2018/04/KCAgenteEspecial-FichaTV.jpg?fit=1400%2C932&ssl=1",
5     "date": "2018-11-23T00:00:00.000Z"
6   },
7   {
8     "id": 2,
9     "image": "https://static.wikia.nocookie.net/dhadas/images/3/31/Tinkerbell_y_el_Tesoro_Perdido.jpg/revision/latest/top-crop/width/200/height/150?cb=20110218153104&path-prefix=es",
10    "date": "2010-11-23T00:00:00.000Z"
11  },
12  {
13    "id": 3
```

Orden normal (en sig. pagina se aplica el filtro)

🍪 Cookies 🔍 Capture requests 🏠 Bootcamp 🏃 Runner 🗑️ Trash

GET

http://localhost:3001/movies?term=DESC

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Body

Cookies

Headers (11)

Test Results



Status: 200 OK

Time: 362 ms

Size: 2.26 KB

Save Response

Pretty

Raw

Preview

Visualize

JSON



```
2  {
3    "id": 6,
4    "image": "https://www.cinemascomics.com/wp-content/uploads/2021/02/big-hero-6-marvel-disney-768x448.jpg.webp",
5    "title": "Big Hero 6: The Series",
6    "date": "2021-10-23T00:00:00.000Z",
7    "rating": 3,
8    "char_asoc": "AC and DC",
9    "createdAt": "2022-05-20T23:37:11.896Z",
10   "updatedAt": "2022-05-20T23:37:11.896Z"
11 },
12 {
13   "id": 4,
14   "image": "https://i.ytimg.com/vi/DfTACfocw1M/maxresdefault.jpg",
15   "title": "Aladdin films",
16   "date": "2019-11-23T00:00:00.000Z".
```


Genero

Listado de géneros

<

GET http ●

GET con: ●

GET http ●

GET http ●

GET http ●

GET http ●

GET http ●

>

+

...

No Environment

http://localhost:3001/genres

Save

GET

▼

http://localhost:3001/genres

Send

ParamsAuthorizationHeaders (8)Body ●Pre-request ScriptTestsSettings

Cookie

● none

● form-data

● x-www-form-urlencoded

● raw

● binary

● GraphQL

JSON ▼

Beautif

1

BodyCookiesHeaders (11)Test Results

Status: 200 OKTime: 44 msSize: 1019 B

Save Response

PrettyRawPreviewVisualizeJSON ▼

```
1 [
2   {
3     "id": "e42af144-01db-414e-b1c9-2dd00218aa19",
4     "name": "romantica",
5     "image": "https://elcomercio.pe/resizer/a2tJZilH_69SkLLyeWbSL-tDH0k=/580x330/smart/filters:format(jpeg):quality(75)/cloudfront-us-east-1.images.arcpublishing.com/elcomercio/YK7E3CUGRRC4TNI2QDBRVID6BA.jpg",
6     "pict_asociat": "Encanto",
7     "tv_asoc": "Encanto"
8   },
9   {
10    "id": "e75f1e36-67da-452b-b59b-3717b6897c61",
11    "name": "aventura",
12    "image": "https://i0.wp.com/vamosadisneyland.com/wp-content/uploads/2021/02/PortadaPelisDisneyAmorAmistad.jpg?fit=1200%2C800&ssl=1",
13    "pict_asociat": "Dino"
```

Crear un genero

NewImport

<GET httpGET con:GET httpGET httpGET httpGET httpPOST htt>+...No Environment

pi-backend-disney dperco
isney
GET https://api.disneyapi.dev/chara...
isney API apidisney_dperco
isney API dperco
og_temper
xpres
iod
iod_recetas
deogames

http://localhost:3001/genres

Save

Send

POSThttp://localhost:3001/genres

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettingsCookies

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSONBeautify

```
1 {  
2   "name": "aventura",  
3   "image": "https://www.ecartelera.com/images/sets/22600/22669.jpg",  
4   "pict_asoc": "el fonfo del mar",  
5   "tv_asoc": "ariel"  
6 }  
7
```

BodyCookiesHeaders (11)Test ResultsStatus: 200 OKTime: 61 msSize: 505 BSave Response

PrettyRawPreviewVisualizeJSON

```
1 {  
2   "message": "Producto se ha recibido"  
3 }
```

Configurations

Databases

Users

Templates

Tables

Views

Materialized Views

Sequences

Indexes

Roles

Groups

Character_movie

Characters

Characters

Characters

Character_movie

Character_genre

Characters

Functions

Functions

Functions

disneydb/postgres@PostgreSQL 14

Query Editor Query History

```
1 SELECT * FROM public.generos
2 ORDER BY id ASC
```

Data Output Explain Messages Notifications

	id [PK] uuid	name character varying (255)	image character varying (255)
1	e42af144-01db-414e-b1c9-2dd00218aa19	romantica	https://elcomercio.pe/resizer/a2tJZiIH_69SkLLyeWbSL-tDH0k=/580x330/smart/filters:for
2	e75f1e36-67da-452b-b59b-3717b6897c61	aventura	https://i0.wp.com/vamosadisneyland.com/wp-content/uploads/2021/02/PortadaPelisDis

Registrar un usuario

POST

http://localhost:3001/auth/register?username=danielperco&email=danielperco4@gmail.com&password=123456

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

```
1 {
2   "username": "danielperco",
3   "email": "danielperco4@gmail.com",
4   "password": "123456"
5 }
6
7
```

Body

Cookies

Headers (11)

Test Results

Status: 200 OK

Time: 101 ms

Size: 480 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1 { "user creado" }
```

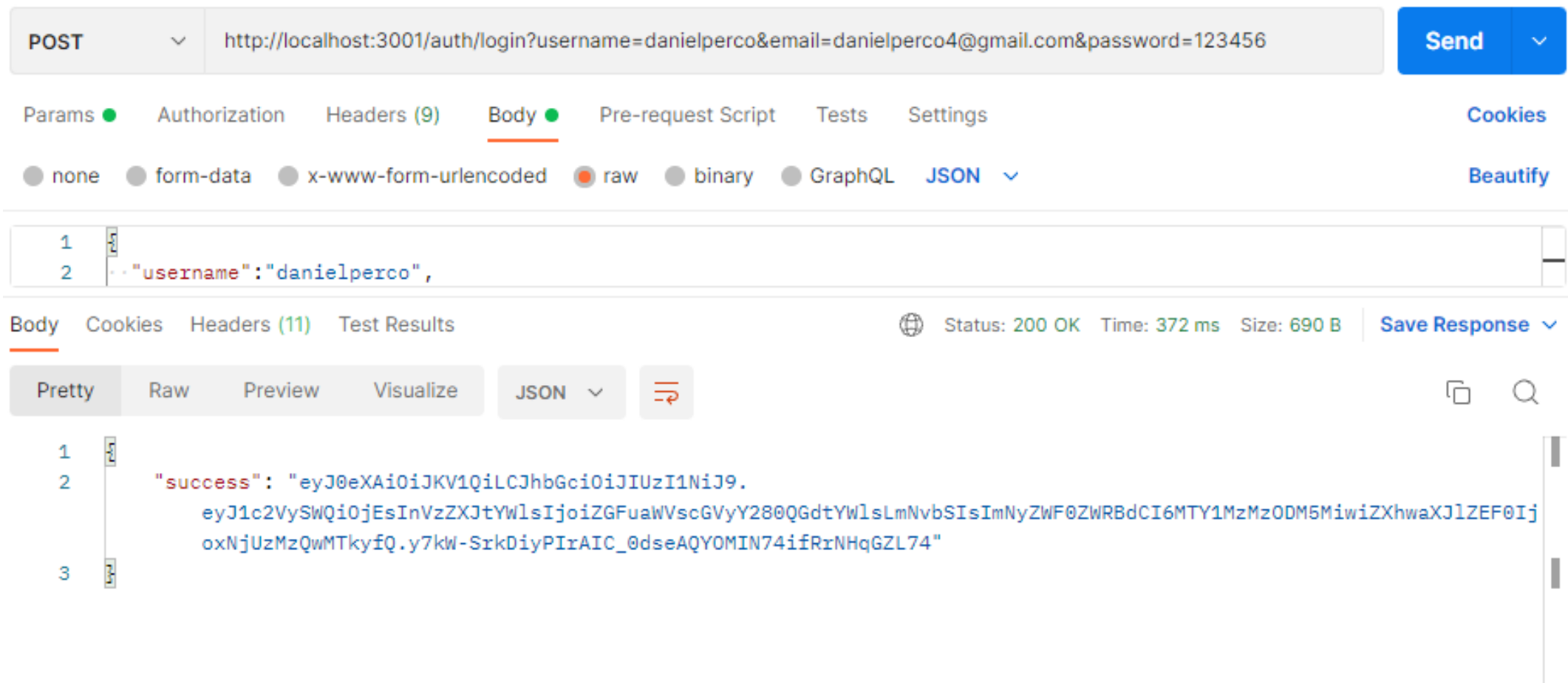
- > Aa FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > 1..3 Sequences
- ▼ Tables (8)
 - > character_movie
 - > character_user
 - > characters
 - > genres
 - > genre_movie
 - > movie_genre
 - > movies
 - > users
- > Trigger Functions
- ▼ Types
- > Views

```
1 SELECT * FROM public.users
2 ORDER BY id ASC
```

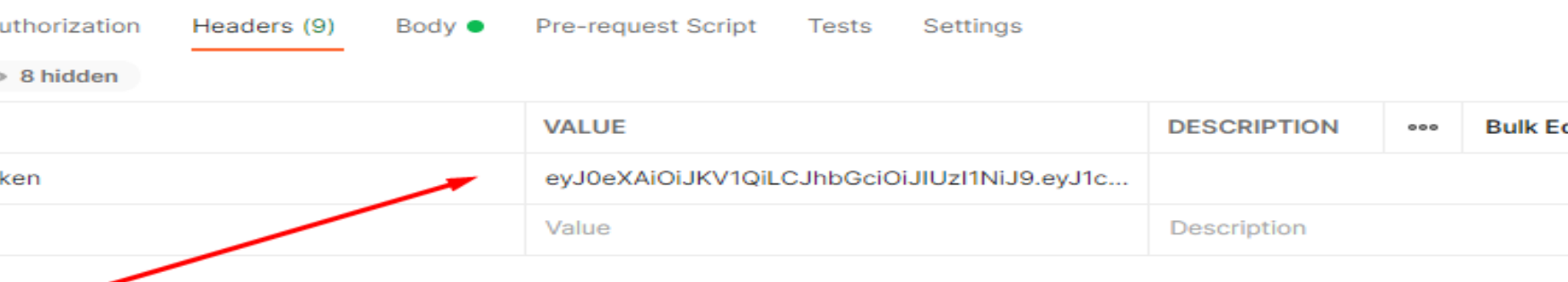
Data Output Explain Messages Notifications

	id [PK] integer	username character varying (255)	email character varying (255)	password character varying (255)	
1	1	danielperco	danielperco4@gmail.com	\$2b\$10\$Km66xfJNe9AS9oMiZB9o6eYtsBEC7l3zhkYyaiEzN2uQtjtKDwsVq	

Logearse y generar el Token de autenticación



Consultar Api con Token de autenticacion



The screenshot shows the Postman interface. At the top, a GET request is made to `http://localhost:3001/characters`. The 'Headers' tab is active, displaying a table with one header: 'usertoken' with a value starting with 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c...'.


KEY	VALUE	DESCRIPTION
usertoken	eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c...	
Key	Value	Description

The 'Body' tab is also visible, showing a JSON response. A red arrow points to the 'image' field in the JSON object.


```
{
  "image": "https://static.wikia.nocookie.net/disney/images/6/61/Olu_main.png",
  "id": 6,
  "name": "'Olu Mel",
  "pic_asoc": [],
  "tv_asoc": []
}
```


```
hola
emial enviado 250 2.0.0 OK 1653338562 j77-20020acaeb50000000b0032aea9c2c23sm4377880oih.40 - gsmtp
[
  character {
    dataValues: {
      iddb: '14b4241e-2026-464d-a9c0-59676baec327',
      image: 'https://static.wikia.nocookie.net/disney/images/7/75/Profile_-_AC_and_DC.jpg/revision/latest/scale-to-width-down/516?cb=20190509220539',
      id: 1,
      name: 'AC and DC',
      age: '20',
    }
  }
]
```


Mail de bienvenida a la Api ,después de loguearse con el Token


 Gmail


Buscar en el correo electrónico


 Redactar


 Recibidos


 Destacados







 Pospuestos

 Importantes











 Enviados


 Borradores 32


 Categorías


-  Social 1,073
-  Notificaciones 2,560
-  Foros 1
-  Promociones 2,614
-  pdf 73
-  Más


Meet

server Api  Recibidos x

 **server Api** <danielperco4@gmail.com>
para mí ▾
Bienvenido a Api Disney

 Responder

 Reenviar