

## Post → Create a character

http://localhost:8080/api/create

GET http://localhost:8080/api/create Send

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

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

```
1 {
2   "id": 3,
3   "strength_id": 4,
4   "name": "Kaido of the Beasts",
5   "bounty": "4,611,000,000",
6   "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
7   "crew": "Beast Pirates",
8   "position": "Captain"
9 }
```

Body Cookies Headers (5) Test Results 201 Created 37 ms 344 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 3,
3   "strength_id": 4,
4   "name": "Kaido of the Beasts",
5   "bounty": "4,611,000,000",
6   "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
7   "crew": "Beast Pirates",
8   "position": "Captain"
9 }
```

## Get → Grab character by ID

GET http://localhost:8080/api/1 Send

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

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

```
1 {
2   "id": 3,
3   "strength_id": 4,
4   "name": "Kaido of the Beasts",
5   "bounty": "4,611,000,000",
6   "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
7   "crew": "Beast Pirates",
8   "position": "Captain"
9 }
```

Body Cookies Headers (5) Test Results 200 OK 43 ms 309 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "strength_id": 1,
4   "name": "Gol.D.Roger",
5   "bounty": "5,564,800,000",
6   "devil_fruit": "no devil_fruit",
7   "crew": "Roger Pirates",
8   "position": "Captain"
9 }
```

Get → Grab all characters by ID

GET

http://localhost:8080/api/getAll

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "id": 1,
3   "strength_id": 1,
4   "name": "Gol.D.Roger",
5   "bounty": "6,864,000,000",
6   "devil_fruit": "no devil_fruit",
7   "crew": "Roger Pirates",
8   "position": "Captain"
9 }
10
11 {
12   "id": 2,
13   "strength_id": 2,
14   "name": "Edward Newgate",
15   "bounty": "6,846,000,000",
16   "devil_fruit": "Gura-Gura no Mi (Tremor-Tremor Fruit)",
17   "crew": "Whitebeard Pirates",
18   "position": "Captain"
19 }
20
21 {
22   "id": 3,
23   "strength_id": 4,
24   "name": "Kaido of the Beasts",
25   "bounty": "4,611,000,000",
26   "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
27   "crew": "Beast Pirates",
28   "position": "Captain"
29 }
```

Put → Grab character by ID and Update Character

PUT

http://localhost:8080/api/update/3

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

2

3

4

5

6

7

8

9

10

```
1 {
2   "id": 3,
3   "strength_id": 4,
4   "name": "Kaido of the Beasts",
5   "bounty": "4,611,000,000",
6   "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
7   "crew": "Beast Pirates",
8   "position": "Yonko Captain"
9 }
```

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "id": 3,
3   "strength_id": 4,
4   "name": "Kaido of the Beasts",
5   "bounty": "4,611,000,000",
6   "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
7   "crew": "Beast Pirates",
8   "position": "Captain"
9 }
```

Shows change

PUT

http://localhost:8080/api/update/3

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

2

3

4

5

6

7

8

9

10

11

1

2

3

4

5

6

7

8

9

10

11

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

6

7

8

9

1

2

3

4

5

6

7

8

9

```
{
  "id": 3,
  "strength_id": 4,
  "name": "Kaido of the Beasts",
  "bounty": "4,611,000,000",
  "devil_fruit": "Fish-Fish fruit, Model: Azure Dragon",
  "crew": "Beast Pirates",
  "position": "Yonko Captain"
}
```

Get → Grab characters by attribute “Captain”

GET

http://localhost:8080/api/position/captain

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

2

3

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17
















18

19

20

```
[
  {
    "id": 1,
    "strength_id": 1,
    "name": "Gol.D.Roger",
    "bounty": "6,564,000,000",
    "devil_fruit": null,
    "crew": "Roger Pirates",
    "position": "Captain"
  },
  {
    "id": 2,
    "strength_id": 2,
    "name": "Edward Newgate",
    "bounty": "6,046,000,000",
    "devil_fruit": "Gura-Gura no Mi (Tremor-Tremor Fruit)",
    "crew": "Whitebeard Pirates",
    "position": "Captain"
  }
]
```

## Coverage Overall

Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
▼ OnePieceAPI	 91.8 %	435	39	474
▼ src/main/java	 86.1 %	217	35	252
▼ com.example.demo	 37.5 %	3	5	8
> OnePieceApiApplication.java	 37.5 %	3	5	8
▼ com.example.demo.controllers	 92.3 %	60	5	65
> API_Controller.java	 92.3 %	60	5	65
▼ com.example.demo.exceptions	 0.0 %	0	23	23
> APINotFoundException.java	 0.0 %	0	23	23
▼ com.example.demo.model	 100.0 %	76	0	76
> API.java	 100.0 %	76	0	76
▼ com.example.demo.service	 97.5 %	78	2	80
> API_Service.java	 97.5 %	78	2	80
▼ src/test/java	 98.2 %	218	4	222
> com.example.demo	 0.0 %	0	4	4
> com.example.demo.controllers	 100.0 %	218	0	218

**Risk Assessment:**

	Insignificant	Negligible	Moderate	Extensive	Significant
Highly Probable	5	7	8	9	10
Probable	4	6	7	8	9
Potential	3	4	6	7	8
Unprobeable	2	3	4	6	7
Circumstantial	1	2	3	4	5

Risk	Statement	Response	Objective	Likelihood	Impact	Risk Level
Injuries	External factors or back pain from posture	Limiting exercise, correcting posture every 5 minutes	Improve overall productivity and ergonomic environment	Probable	Moderate	7
Computer Malfunction	Software/ Hardware failure	Update system/ cloud backups	Make project agile and undisturbed	Unprobeable	Significant	7
Compile / runtime error	Code doesn't work or mistakes in formatting	Debug code to identify error, peer access	Clean and responsive code	Probable	Insignificant	4
Procrastination	Slow progress for long period of time. Distractions	Take regular breaks and also move the body to stay awake	Moderate a interactive environment where it forces you to keep upbeat	Highly Probable	Negligible	7
Bad Internet Connection	Wi-Fi bandwidth not working for the day, loss of repo	Make sure project can be accessed locally	More time on project, less time connecting to internet	Potential	Significant	8

Jira RoadMap and Backlog

Projects / OnePieceAPI

Backlog

Q

🔍

👤

👤

Epic

Label

Issues without epic

PROJ1:MODEL

PROJ2:CONTROLLER

PROJ3:SERVICE

PROJ4:REPO

PROJ5:EXCEPTION

PROJ6:TESTING

+ Create Epic

ON Sprint 1

ON-7 As a Client, I want a One Piece Character API with private variables. PROJ1:MODEL

+ Create issue

Backlog (13 issues)

ON-8 As a Client, I want a One Piece Character API that allows me to create characters. PROJ2:CONTROLLER

ON-10 As a Client, I want a One Piece Character API that allows me to get All characters. PROJ2:CONTROLLER

ON-11 As a Client, I want a One Piece Character API that allows me to get a specific item in the database using the id variable. PROJ2:CONTROLLER

ON-13 As a Client, I want a One Piece Character API that allows me to update my characters using the put request. PROJ2:CONTROLLER

ON-14 As a Client, I want a One Piece Character API that allows me to delete my characters using the delete request. PROJ2:CONTROLLER

ON-15 As a Programmer, I need to create a service that adds characters and saves to a repo. PROJ3:SERVICE

ON-16 As a Programmer, I need a service that can save all the inputted data to a repo. PROJ3:SERVICE

ON-18 As a Programmer, I need a service that can use the inputted ID to collect all data associated with that ID. PROJ3:SERVICE

ON-19 As a Programmer, I need a service that can alter the input data in the database. PROJ3:SERVICE

ON-20 As a Programmer, I need a service that can check whether or not it is present and actually delete the data. PROJ3:SERVICE

ON-21 As a Programmer, I need a repo that can be used as a saving point. PROJ4:REPO

ON-23 As a Programmer, I need an exception file that can be used in-between the services encase they don't work. PROJ5:EXCEPTION

ON-25 As a Programmer, I need to set up integration testing for each function between the service file and controller file. PROJ6:TESTING

+ Create issue

125

Create sprint

PROJ6:TESTING / ON-25

As a Programmer, I need to set up integration testing for each function between the service file and controller file.

To Do

Description

Make sure all code runs smoothly for a live environment.

Acceptance Criteria:

Given the client + owner is using the API, when they /function(), it will always work due to it being tested in a live environment and through integration testing

Details

Assignee

Zakariya Nasser

Labels

None

Sprint

None

Story point estimate

25

Development

Branch

Reporter

Zakariya Nasser

Created 10 seconds ago

Updated 10 seconds ago

Configure

Activity

Show

All

Comments

History

Newest first 17

25

Add a comment...

Pin this comment

Uncomment

Roadmap

