

REST API USING MONGODB (PYTHON FLASK)

this repository consist of order crud services to add order (POST), get all order or by id (GET), update all order(PUT), update item of order(PATCH), and delete order (DELETE).

1. POST

add new order (status : SUCCESS)

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

- Method:** POST
- URL:** http://127.0.0.1:5000/orders
- Body Type:** JSON
- Request Body (JSON):**

```
1 {
2   "transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",
3   "customer_name": "PT. AMARA PRIMATIGA",
4   "customer_code": "1678593",
5   "transaction_amount": "70700",
6   "transaction_discount": "0",
7   "transaction_payment_type": "29",
8   "transaction_additional_field": "",
9   "transaction_state": "UNPAID",
10  "transaction_code": "CGKFT20200715121",
11  "transaction_order": 121,
12  "location_id": "5cecb20b6c49615b174c3e74",
13  "organization_id": 6,
14  "created_at": "2020-07-15T11:11:13+0700"
```
- Response Body (JSON):**

```
1 {
2   "id": "5f861d48bad93599bac66235"
3 }
```
- Status:** 200 OK

2. GET

get/new order list (STATUS : SUCCESS)

GET

▼

http://127.0.0.1:5000/orders

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

BodyCookiesHeaders (4)Test Results

PrettyRawPreviewVisualizeJSON

```
1  [
2    {
3  >   "_id": { ...
4
5     },
6     "transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",
7     "customer_name": "PT. AMARA PRIMATIGA",
8     "customer_code": "1678593",
9     "transaction_amount": "70700",
10    "transaction_discount": "0",
11    "transaction_payment_type": "29",
12    "transaction_additional_field": "",
13    "transaction_state": "UNPAID",
14    "transaction_code": "CGKFT20200715121",
15    "transaction_order": 121,
16    "location_id": "5cecb20b6c49615b174c3e74",
17    "organization_id": 6,
18    "created_at": "2020-07-15T11:11:12+0700",
19    "updated_at": "2020-07-15T11:11:22+0700",
20    "transaction_payment_type_name": "Invoice",
21    "transaction_cash_amount": 0,
22    "transaction_cash_change": 0,
23    "customer_attribute": {
24      "Nama_Sales": "Radit Fitrawikarsa",
25      "TOP": "14 Hari",
26      "Jenis_Pelanggan": "B2B"
27    },
28    "connote": {
29      "connote_id": "f70670b1-c3ef-4caf-bc4f-eefa702092ed",
30      "connote_number": 1,
31      "connote_service": "ECO",
32      "connote_service_price": 70700,
33      "connote_amount": 70700,
34      "connote_code": "4B00100000000000"
```

get by ID (status: SUCCESS)

GET

http://127.0.0.1:5000/orders/5f861d48bad93599bac66235

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Body

Cookies

Headers (4)

Test Results

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "_id": {
3     "$oid": "5f861d48bad93599bac66235"
4   },
5   "transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",
6   "customer_name": "PT. AMARA PRIMATIGA",
7   "customer_code": "1678593",
8   "transaction_amount": "70700",
9   "transaction_discount": "0",
10  "transaction_payment_type": "29",
11  "transaction_additional_field": "",
12  "transaction_state": "UNPAID",
13  "transaction_code": "CGKFT20200715121",
14  "transaction_order": 121,
15  "location_id": "5cecb20b6c49615b174c3e74",
16  "organization_id": 6,
17  "created_at": "2020-07-15T11:11:12+0700",
18  "updated_at": "2020-07-15T11:11:22+0700",
19  "transaction_payment_type_name": "Invoice",
20  "transaction_cash_amount": 0,
21  "transaction_cash_change": 0,
22  "customer_attribute": {
23    "Nama_Sales": "Redit Fitrawikarsa",
24    "TOP": "14 Hari",
25    "Jenis_Pelanggan": "B2B"
26  },
27  "connote": {
28    "connote_id": "f70670b1-c3ef-4caf-bc4f-eefa702092ed",
29    "connote_number": 1,
30    "connote_service": "ECO",
31    "connote_service_price": 70700,
32    "connote_amount": 70700,
33    "connote_code": "AWB00100209082020",
34    "connote_booking_code": "",
35    "connote_order": 326931,
36    "connote_state": "PATN"
```

get by id when id not found (status : SUCCESS)

GET

http://127.0.0.1:5000/orders/5f860ff7c0fd033e686ed228

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON

1

Body

Cookies

Headers (4)

Test Results

Pretty

Raw

Preview

Visualize

HTML

1

order id : 5f860ff7c0fd033e686ed228 not found

3. PUT

update order (status: SUCCESS)

PUT

http://127.0.0.1:5000/orders/5f861d48bad93599bac66235

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON

1

{

2

"transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",

3

"customer_name": "PT. PRIMATIGA",

4

"customer_code": "1678593",

5

"transaction_amount": "70700",

6

"transaction_discount": "0",

7

"transaction_payment_type": "29",

8

"transaction_additional_field": "",

9

"transaction_state": "UNPAID",

10

"transaction_code": "CGKFT20200715121",

11

"transaction_order": 121

Body

Cookies

Headers (4)

Test Results

Pretty

Raw

Preview

Visualize

HTML

1

order id : 5f861d48bad93599bac66235 has been updated

when update failed because of id or item key didn't match (status: SUCCESS)

PUT http://127.0.0.1:5000/orders/5f861d48bad93599bac66235

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

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

```
1 {
2   "transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",
3   "customer_name": "PT. PRIMATIGA",
4   "customer_code": "1678593",
5   "transaction_amount": "70700",
6   "transaction_": "0",
7   "transaction_payment_type": "29",
8   "transaction_additional_field": "",
9   "transaction_state": "UNPAID",
10  "transaction_code": "CGKFT20200715121",
11  "transaction_order": 121
}
```

Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize HTML ▼ ↻

```
1 order id : 5f861d48bad93599bac66235 or item not match
```

4. PATCH

update item dari sebuah order yang terdaftar (status: SUCCESS)

PATCH http://127.0.0.1:5000/orders/5f860ff7c0fd033e686ed228

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

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

```
1 {
2   "transaction_state": "UNPAID"
3 }
```

Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize HTML ▼ ↻

```
1 item id : 5f860ff7c0fd033e686ed228 has been updated
```

GET http://127.0.0.1:5000/orders/5f860ff7c0fd033e686ed228

Params Authorization Headers (8) **Body** Pre-request Script

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary

```
1 {
2   "transaction_state": "UNPAID"
3 }
```

Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize JSON

```
12 "transaction_state": "UNPAID",
```

patch gagal ketika id tidak match (status: SUCCESS)

PATCH http://127.0.0.1:5000/orders/5f860ff7c0fd033e686ed228

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

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL

```
1 {
2   "transaction_state": "UNPAID"
3 }
```

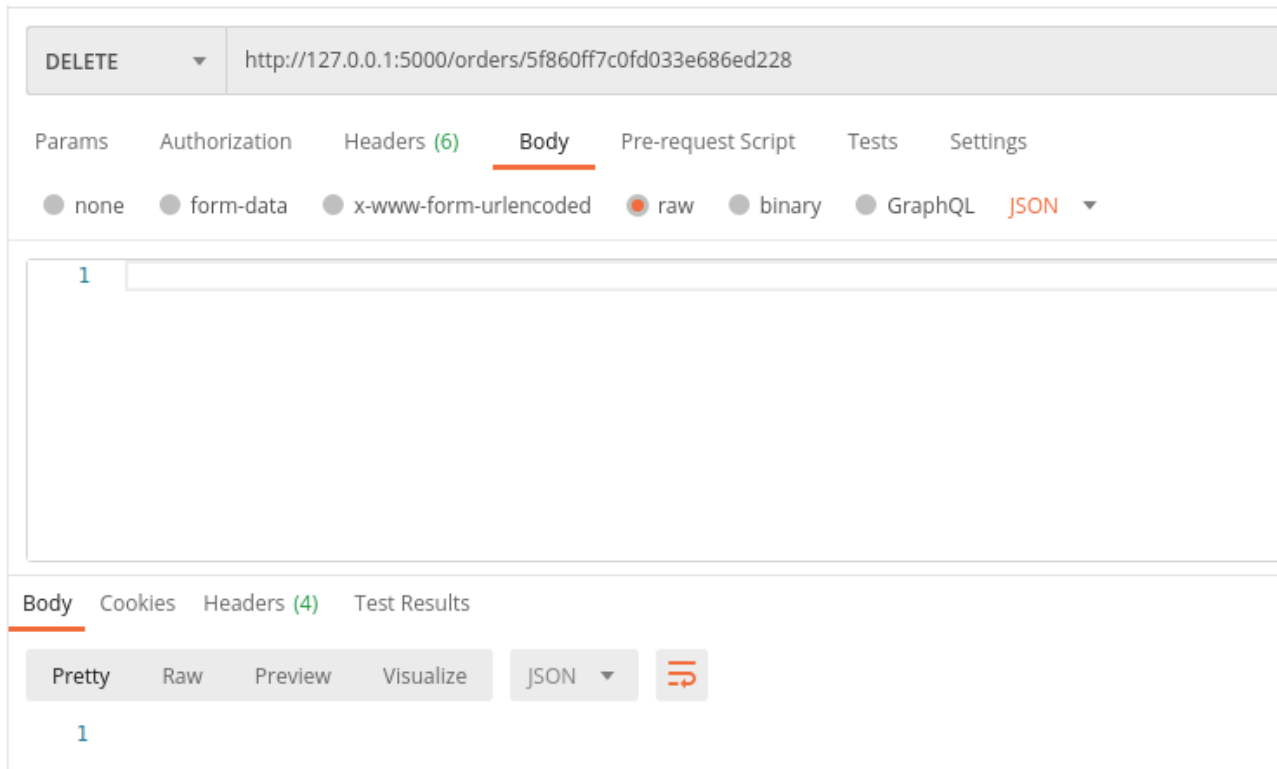
Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize HTML

```
1 item id : 5f860ff7c0fd033e686ed228 has been updated
```

5. DELETE

delete by id (status: SUCCESS)



UNITTEST

unittest yang dilakukan ada 7 jenis,

test untuk setiap methodnya. Kegagalan terhadap 3 method dibawah ini disebabkan saya belum mampu untuk mengambil id dari order yang baru saja di tambahkan dengan 1x eksekusi.

Jadi proses matching data menjadi gagal.

```
===== short test summary info =====
FAILED tests/unittest.py::TestOrder::test_get_by_id - AssertionError: 200 != 404
FAILED tests/unittest.py::TestOrder::test_update_item - AssertionError: 200 !=...
FAILED tests/unittest.py::TestOrder::test_update_order - AssertionError: 200 ...
===== 3 failed, 4 passed in 3.51s =====
```

PUT DAN PATCH

put dan path pada dasarnya protokol http yang memiliki fungsi sama yaitu untuk update field atau resources. Yang berbeda adalah input awal yang diberikan. Put request, entitas entitas yang kita berikan dianggap sebagai modifikasi sumber awal sehingga dilakukan update atau pembaharuan terhadap data yang tersimpan sesuai entitas yang diberikan.

Pada patch, hanya entitas yang terlampir lah yang akan mengalami pembaharuan. Sehingga request client terhadap server dilakukan persis dengan data request tersebut.

Yang menjadi pembeda unik adalah. Ketika kita memberi request PUT maka entitas yang kita sediakan harus sama lengkapnya dengan field-field entitas yang tersimpan. Sedangkan patch hanya perlu entitas yang ingin diubah.

PUT

GEThttp://127.0.0.1:5000/orders/5f861d48bad93599bac66235

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTests

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

1

{

2

"transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",

3

"customer_name": "PT. PRIMATIGA",

4

"customer_code": "1678593",

5

"transaction_amount": "70700",

6

"transaction_discount": "0",

7

"transaction_payment_type": "29",

8

"transaction_additional_field": "",

9

"transaction_state": "UNPAID",

10

"transaction_code": "CGKFT20200715121",

11

"transaction_order": 121

}

BodyCookiesHeaders (4)Test Results

PrettyRawPreviewVisualizeJSON

1

{

2

"_id": {

3

"\$oid": "5f861d48bad93599bac66235"

4

},

5

"transaction_id": "d0090c40-539f-479a-8274-899b9970bddc",

6

"customer_name": "PT. PRIMATIGA",

7

"customer_code": "1678593",

8

"transaction_amount": "70700",

9

"transaction_discount": "0",

10

"transaction_payment_type": "29",

11

"transaction_additional_field": "",

12

"transaction_state": "UNPAID",

13

"transaction_code": "CGKFT20200715121",

14

"transaction_order": 121

}

PATCH

GEThttp://127.0.0.1:5000/orders/5f860ff7c0fd033e686ed228

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTests

none

form-data

x-www-form-urlencoded

raw

binary

1

{

2

"transaction_state": "UNPAID"

3

}

BodyCookiesHeaders (4)Test Results

PrettyRawPreviewVisualizeJSON

12

"transaction_state": "UNPAID",