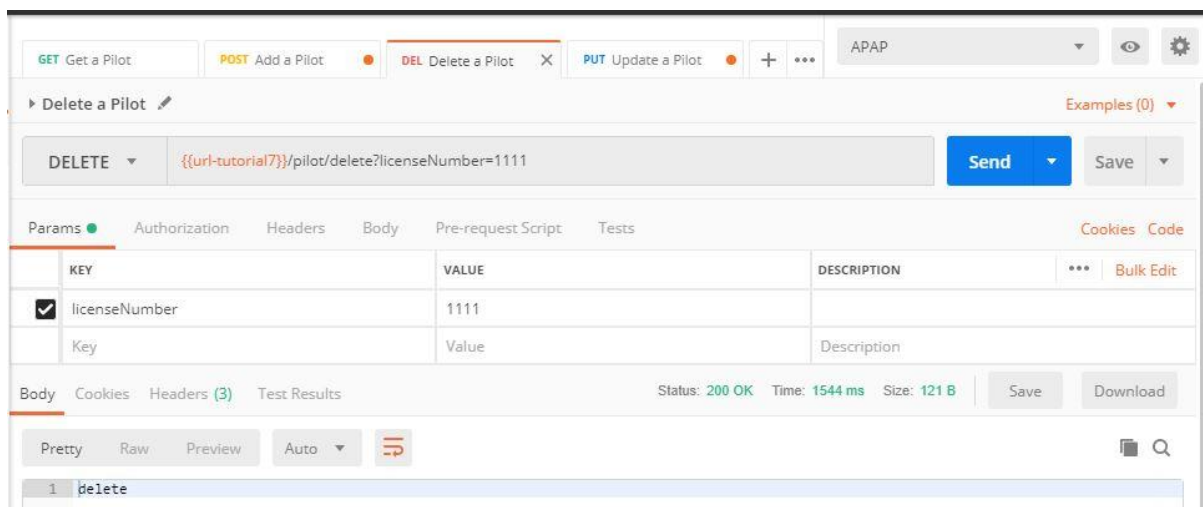
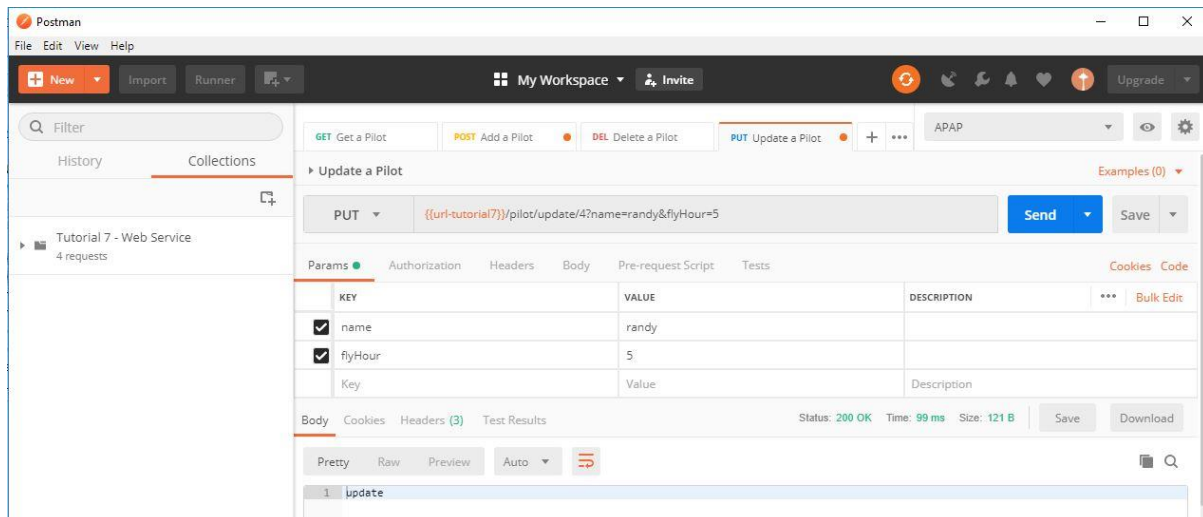
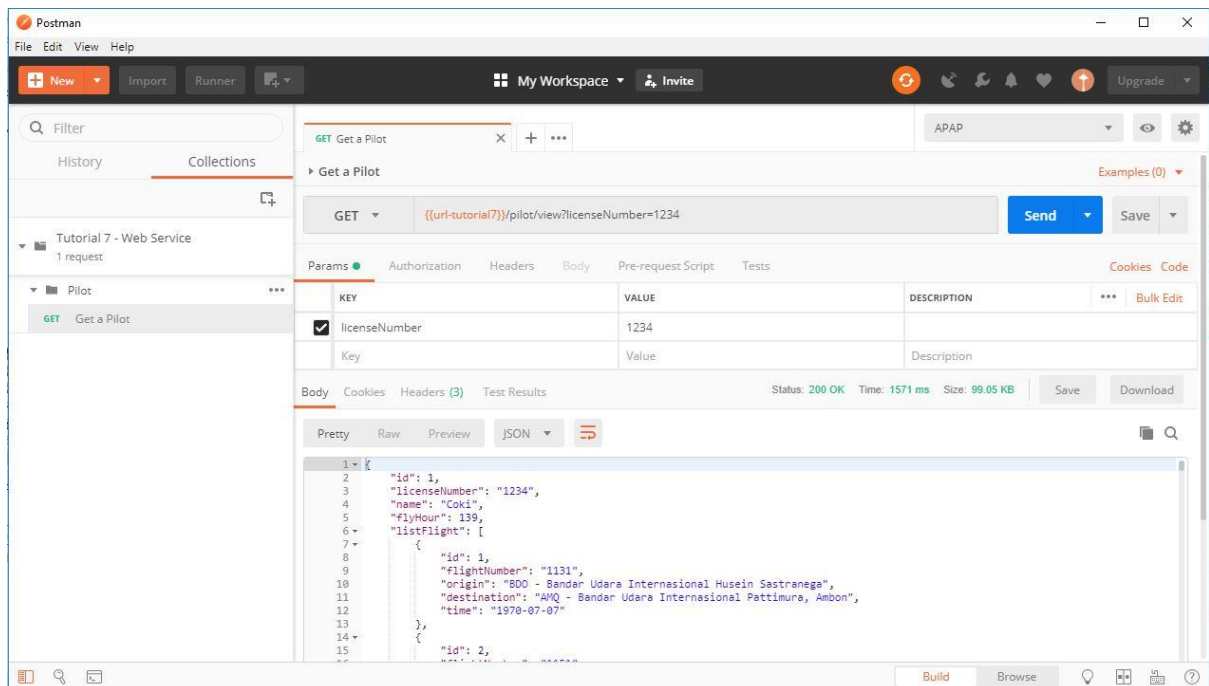
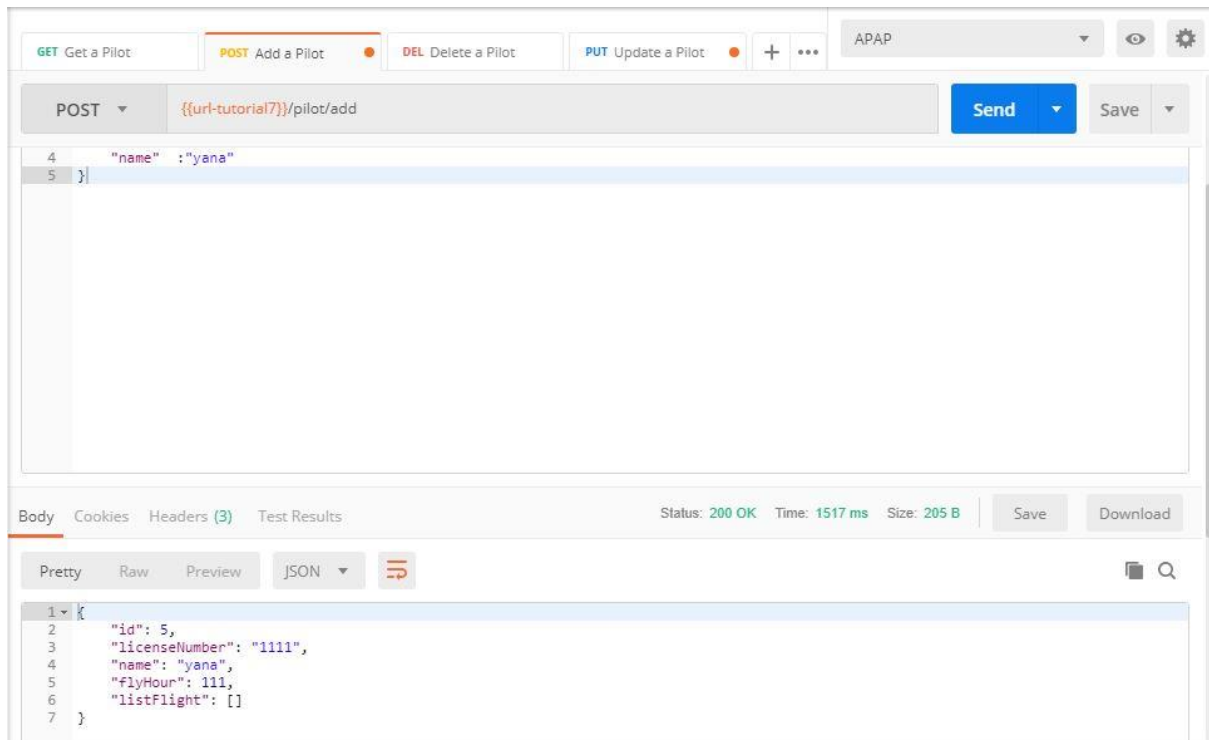
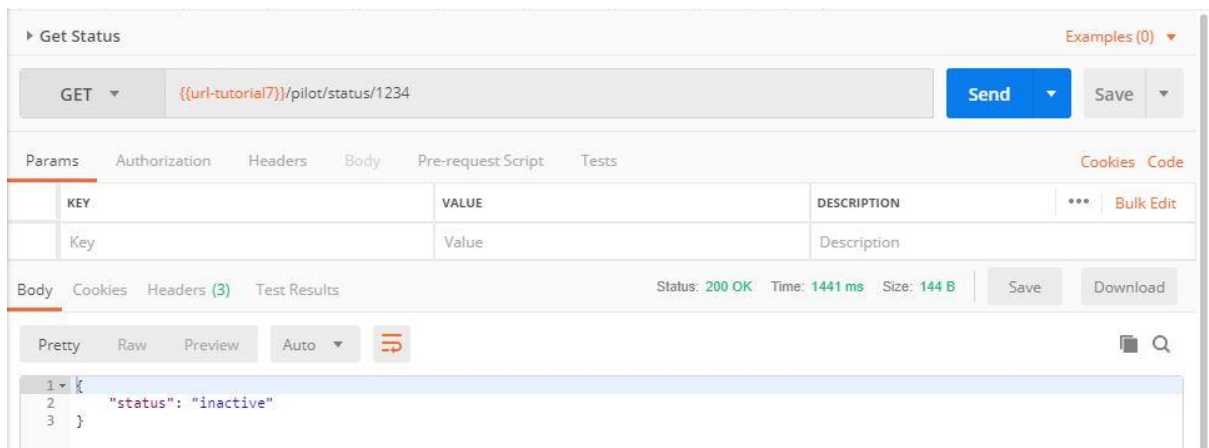
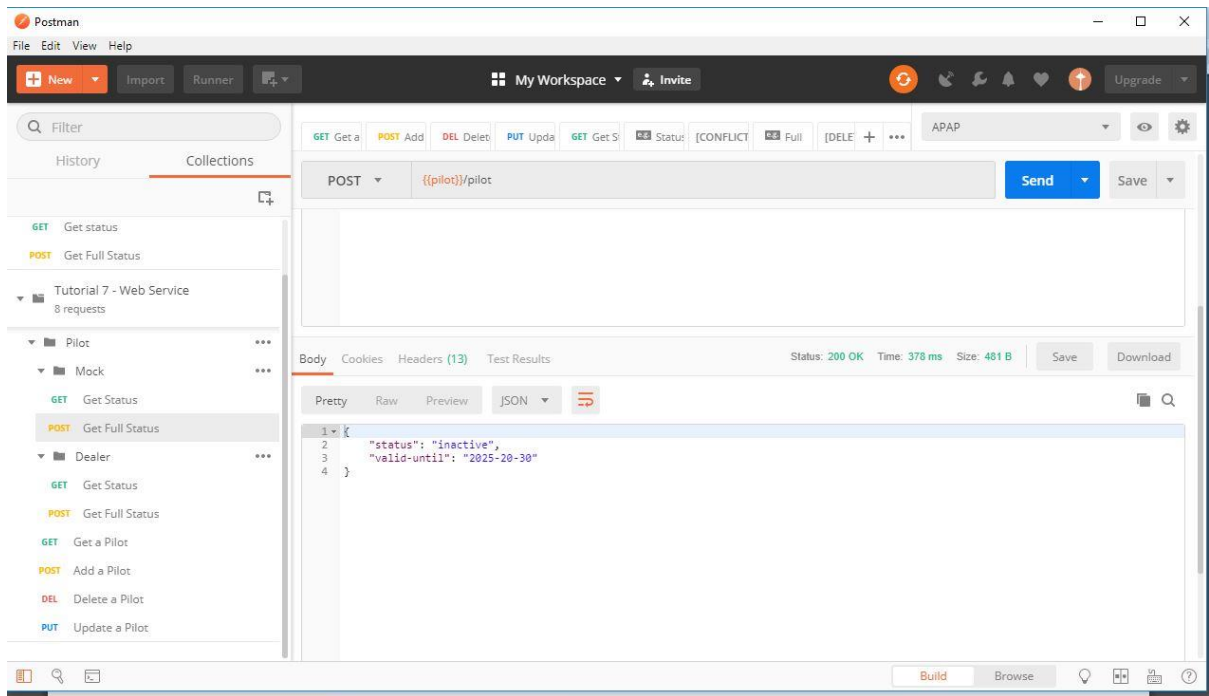


Dari tutorial 7 , saya belajar mengenai cara membuat dan menggunakan Rest Controller , yang meliputi metode POST , PUT , DELETE , GET . Saya juga belajar cara membuat Mockserver dan membuat service customer .

Dokumentasi pengerjaan tutorial :







GET Get a POST Add DEL Delete PUT Update GET Get S Status: POST Get Full M + ... APAP

Get Full Status Examples (0)

GET `{{url-tutorial7}}/pilot/full/1234` Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (3) Test Results Status: 200 OK Time: 4353 ms Size: 196 B Save Download

Pretty Raw Preview JSON

```

1 {
2   "status": "inactive",
3   "valid-until": "2026-08-30T00:00:00.000+0000"
4 }

```

LATIHAN

1.

Fitur menambah penerbangan

GET Get POST Add DEL Delete PUT Update GET Get S Status: POST Get Full M + ... [DELETED] + ... APAP

POST `{{url-tutorial7}}/flight/add`

```

0 pilot {
7 {
8   "licenseNumber": "1111",
9   "id": 2
10 }
11 }
12 }

```

Body Cookies Headers (3) Test Results Status: 200 OK Time: 379 ms Size: 288 B

Pretty Raw Preview JSON

```

1 {
2   "id": 19,
3   "flightNumber": "7779",
4   "origin": "singapore",
5   "destination": "tokyo",
6   "time": "2017-10-10",
7   "pilot": {
8     "id": 2,
9     "licenseNumber": "1111",
10    "name": null,
11    "flyHour": 0

```

Fitur melihat penerbangan

The screenshot shows a REST client interface with a GET request titled "Get a Flight". The URL is `{{url-tutorial7}}/flight/view/7778`. The response status is 200 OK, with a time of 39 ms and a size of 291 B. The response body is displayed in JSON format:

```
1 {
2   "id": 18,
3   "flightNumber": "7778",
4   "origin": "bandung",
5   "destination": "london",
6   "time": "2017-10-10",
7   "pilot": {
8     "id": 2,
9     "licenseNumber": "1111",
10    "name": "yana",
11    "flyHour": 111
12  }
13 }
```

Fitur menghapus penerbangan

The screenshot shows a REST client interface with a DELETE request titled "Delete a Flight". The URL is `{{url-tutorial7}}/flight/delete/19`. The response status is 200 OK, with a time of 446 ms and a size of 121 B. The response body is displayed in JSON format:

```
1 {
2   "id": 19,
3   "flightNumber": "7778",
4   "origin": "bandung",
5   "destination": "london",
6   "time": "2017-10-10",
7   "pilot": {
8     "id": 2,
9     "licenseNumber": "1111",
10    "name": "yana",
11    "flyHour": 111
12  }
13 }
```

Fitur mengubah penerbangan

Update a Flight Examples (0)

PUT `{{url-tutorial7}}/flight/update/18?destination=Jakarta&origin=Yogyakarta&time=2016-11-11` Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> destination	Jakarta			
<input checked="" type="checkbox"/> origin	Yogyakarta			
<input checked="" type="checkbox"/> time	2016-11-11			
Key	Value	Description		

Body Cookies Headers (3) Test Results Status: 200 OK Time: 456 ms Size: 121 B Save Download

Pretty Raw Preview Auto ≡

1 update

Fitur untuk melihat semua penerbangan

GET GET POST DELETE PUT GET Status POST Get Full M [DELETED] + ... APAP APAP 👁 ⚙

Get all flights Examples (0)

GET `{{url-tutorial7}}/flight/all` Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (3) Test Results Status: 200 OK Time: 394 ms Size: 813 B Save Download

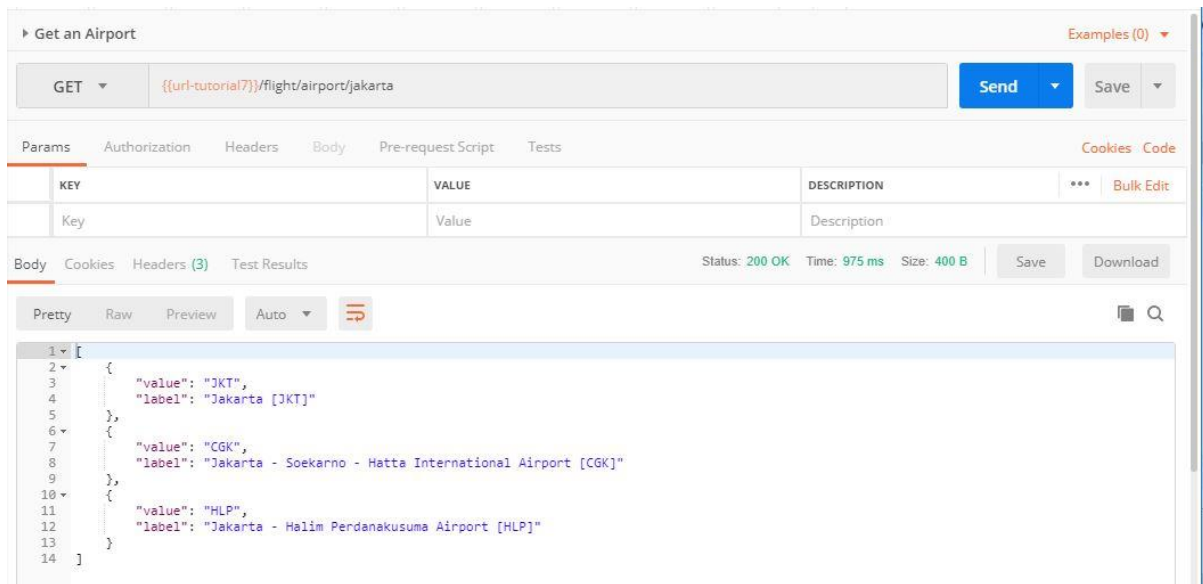
Pretty Raw Preview JSON ≡

```

1 {
2   {
3     "id": 6,
4     "flightNumber": "Narita",
5     "origin": "Malaysia",
6     "destination": "Jogjakarta",
7     "time": "2018-10-16",
8     "pilot": {
9       "id": 1,
10      "licenseNumber": "81234QQ",
11      "name": "Johnny",
12      "flyHour": 17
13    }
14  },
15  {
16    "id": 13,
17    "flightNumber": "Singapore Airlines",

```

2.



Dalam pengerjaan nomor dua , pertama diambil URL untuk akses API (yang berisikan API key , kode negara Indonesia, dan term untuk pencarian) dari Travel Innovation Sandbox. Term (kata kunci untuk pencarian) dikosongkan karena merupakan variabel input saat request di Postman .

Setelah itu , dibuat metode dalam FlightController untuk mendapatkan data (dengan menggunakan API) dan saya juga membuat request dalam Postman seperti gambar di atas .

Response pun ditampilkan dalam hasil di atas yang berisikan bandara-bandara di kota Jakarta.