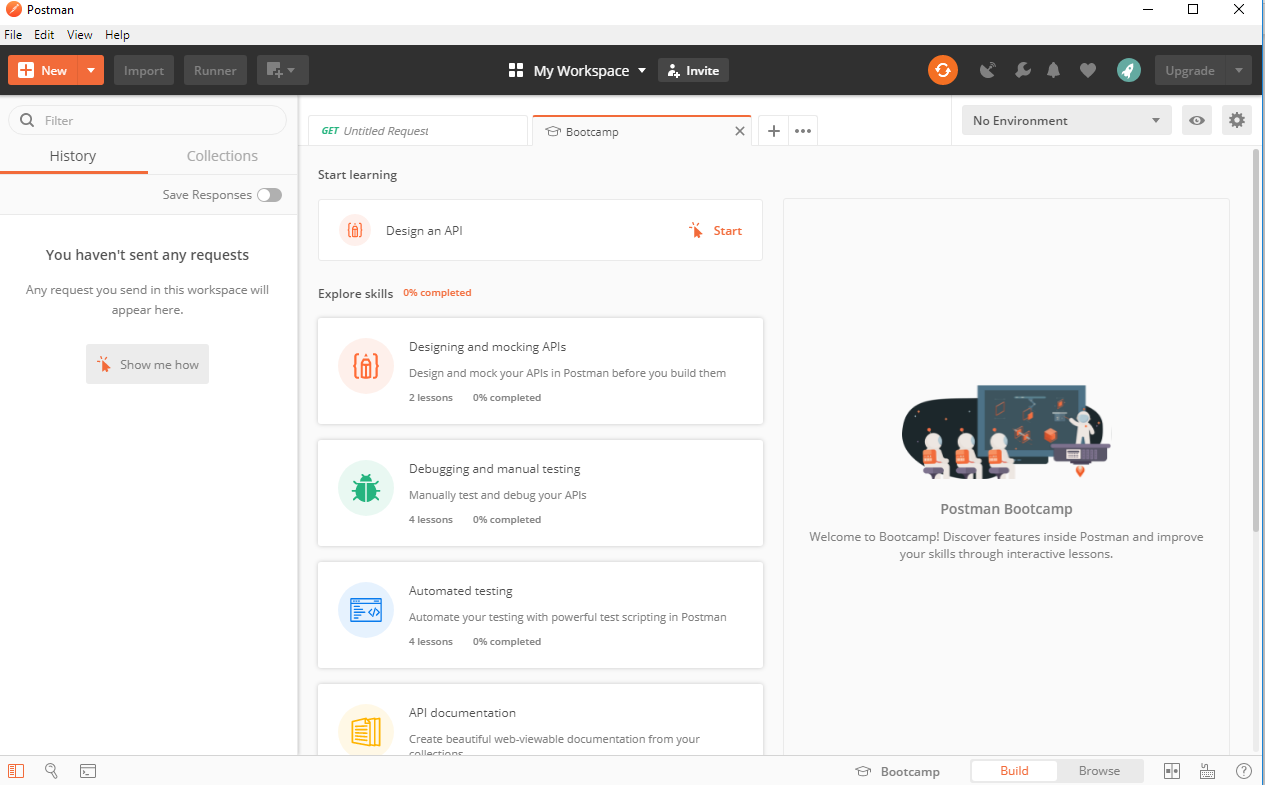
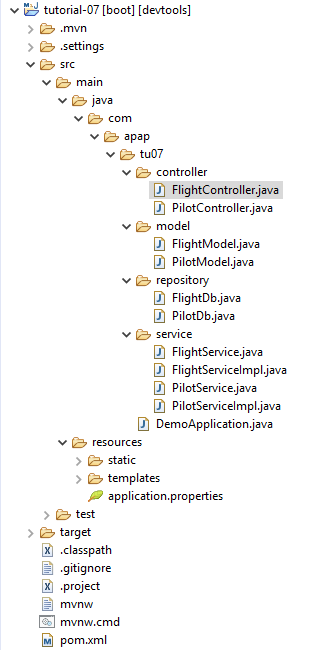
1. **Pendahululan**

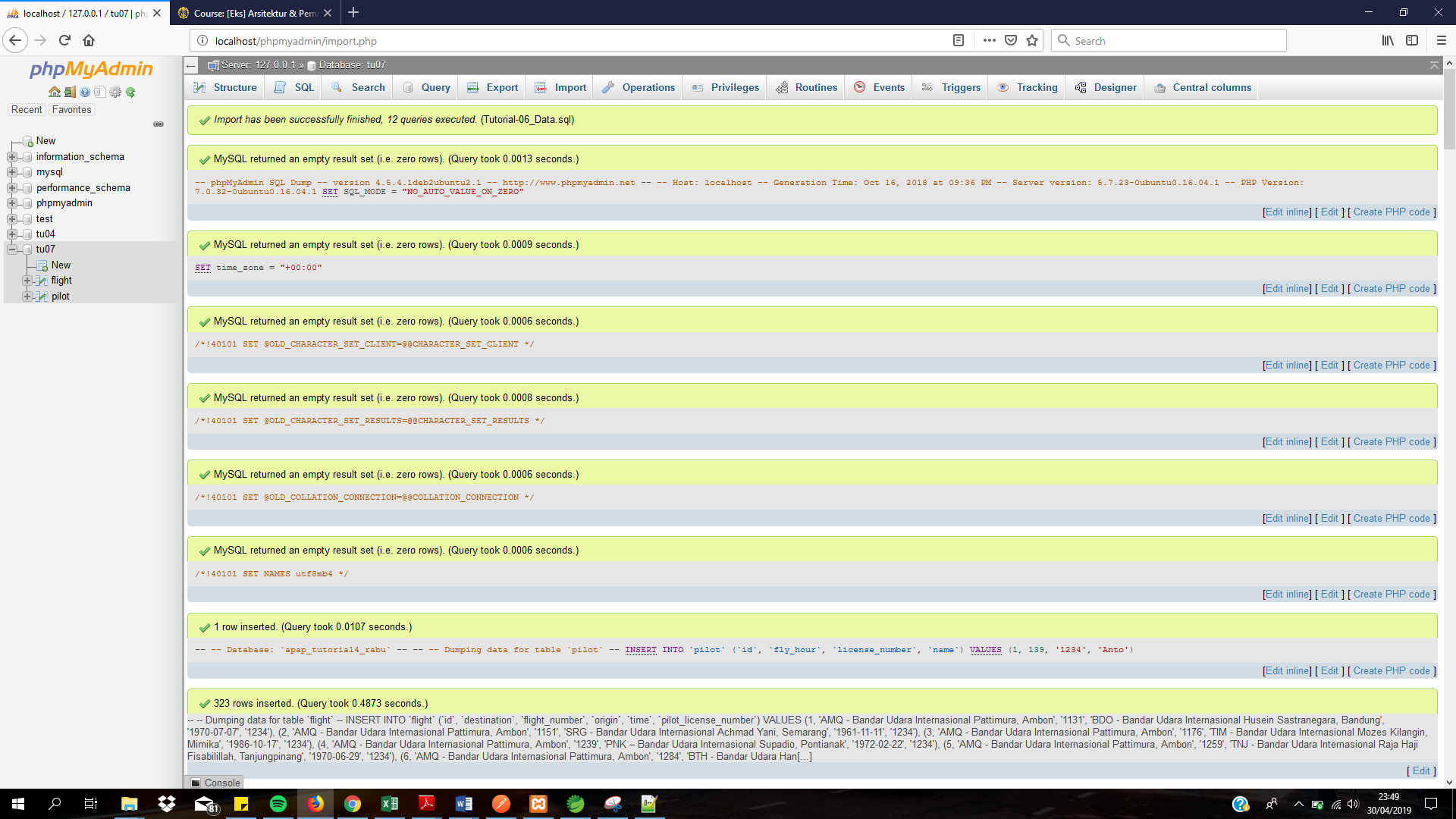
* Install Postman, jalankan aplikasinya.



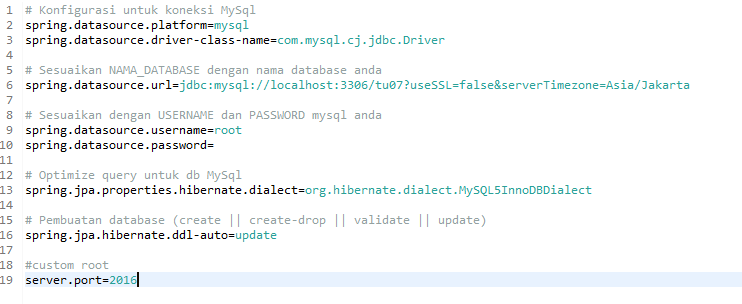
* Import project Tutorial-06 dan ubah menjadi Tutorial-07. Ubah Package dari \*.tu06 menjadi \*.tu07



* Buat DB dengan nama tu07, lalu import SQL yang ada di scele.

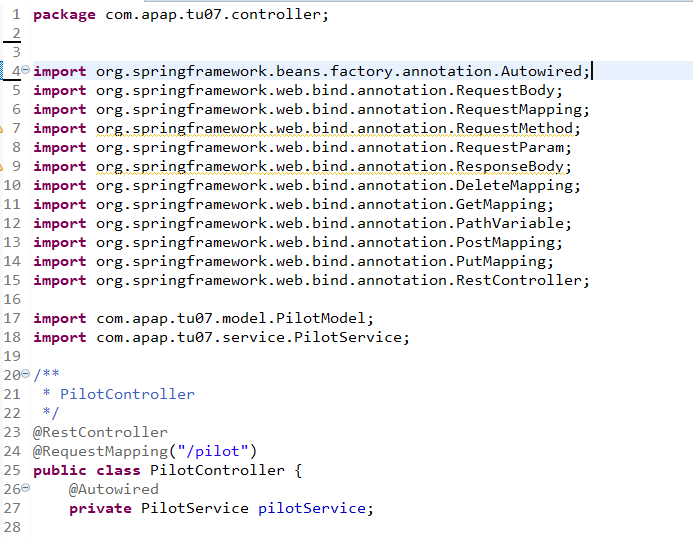


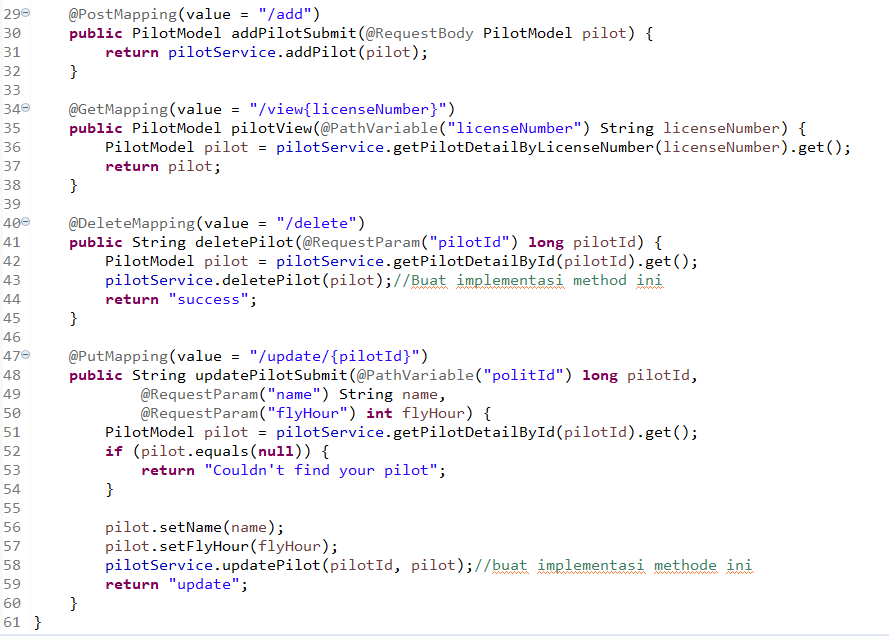
* Sesuaikan application.properties dengan envi. Tambahkan custom port.



1. **Membuat RestController**

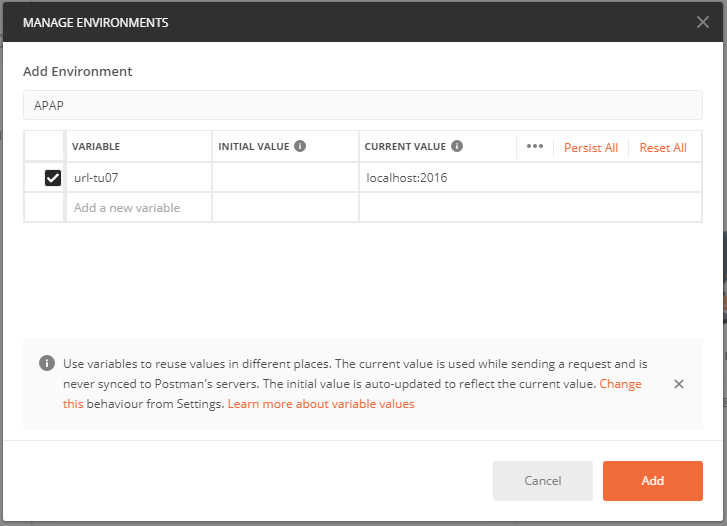
* Ubah PilotControl seperti berikut:



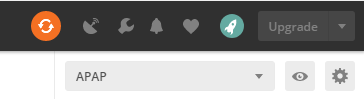


1. **Mengakses methode GET**

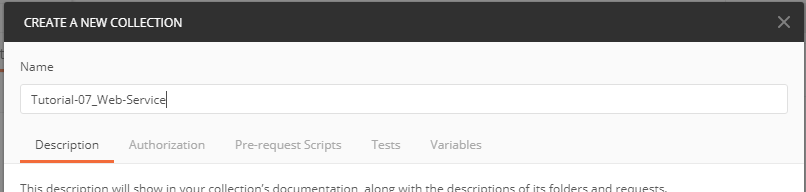
* Buat Environment baru bernama “APAP” dan buat variable dengan spedifikasi sbb:



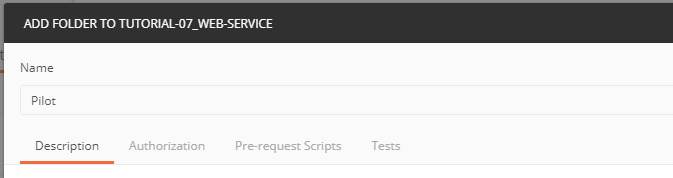
* Pilih environment APAP



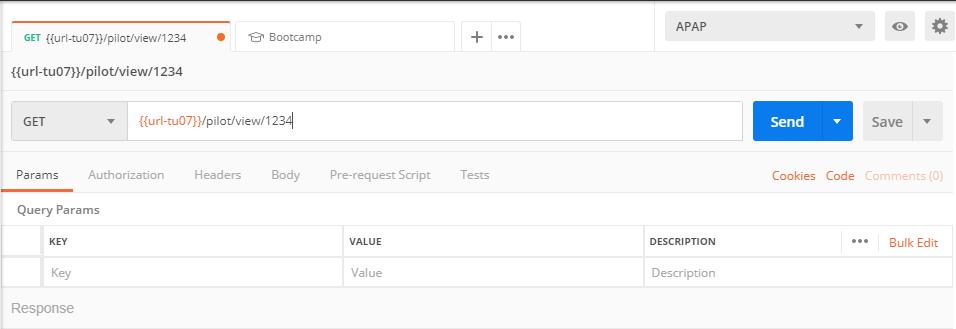
* Buat Collection Baru



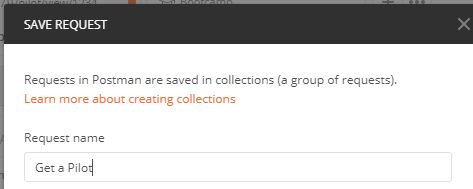
* Buat Folder Baru dengan nama Pilot



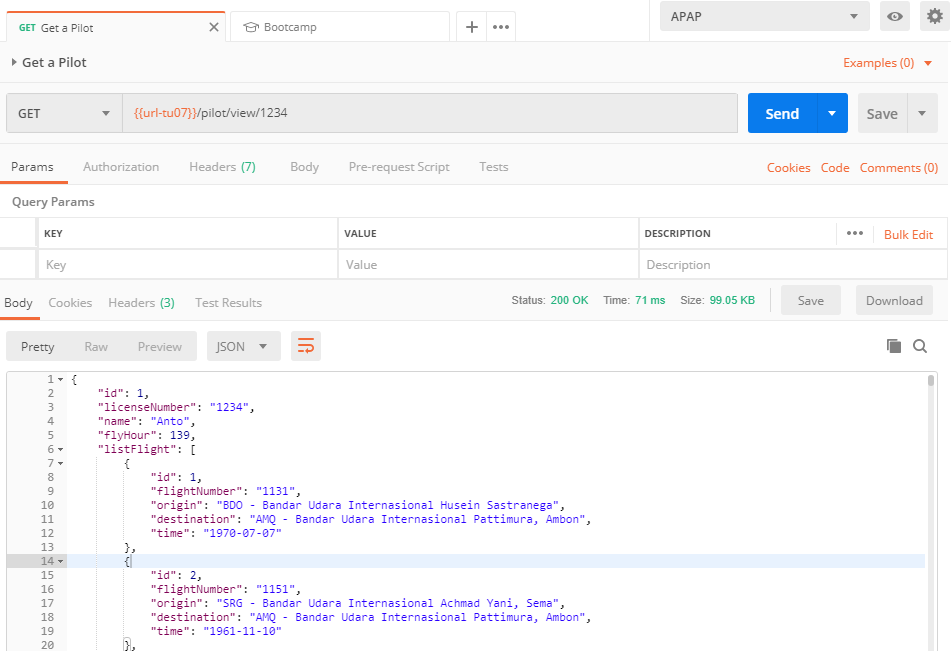
* Buat Request baru dengan method GET dengan URL “{{url-tu07}}/pilot/view/1234 “



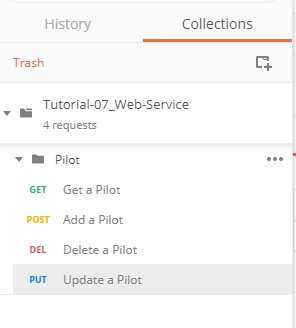
* Lalu save dan beri nama “Get a Pilot” pada folder yang sudah dibuat



* Kirim request dengan mengkilk tombol “Send”, dan akan keluar body sbb:



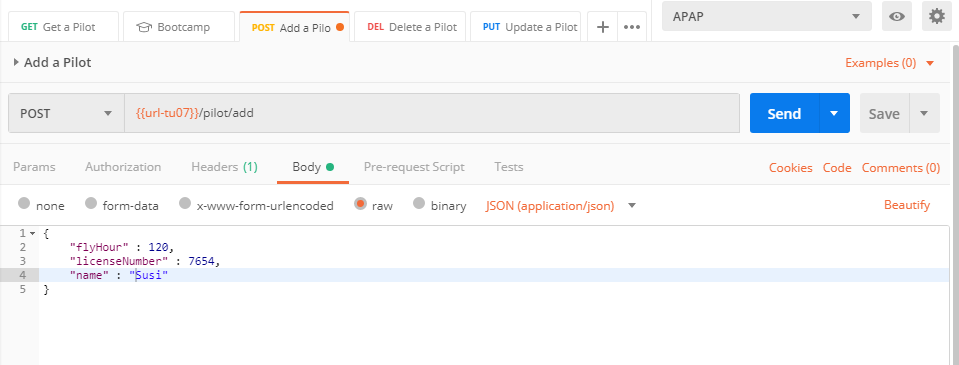
* Buat Request yang lainnya



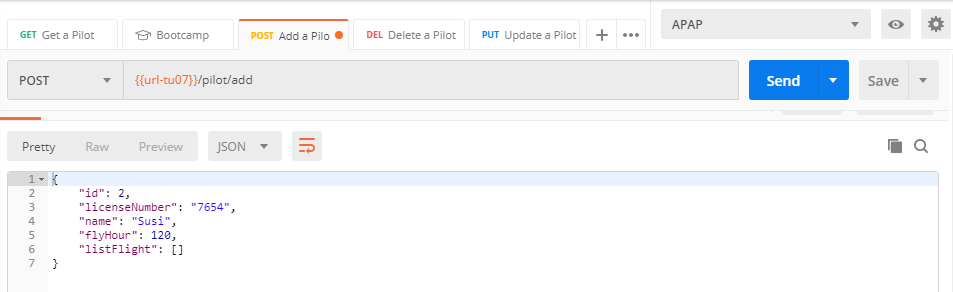
1. **Mengakses method POST**

Jika menggunakan field URL pada browser kita sebenarnya menggunakan method GET, sementara method lain seperti PUT, DELETE, POST, dll. tidak bisa menggunakan URL browser. Postman digunakan agar bias mengakses tanpa membuat terlebih dahulu sistem yang menerima API dari kita. Terdapat aplikasi lainnya selain Postman.

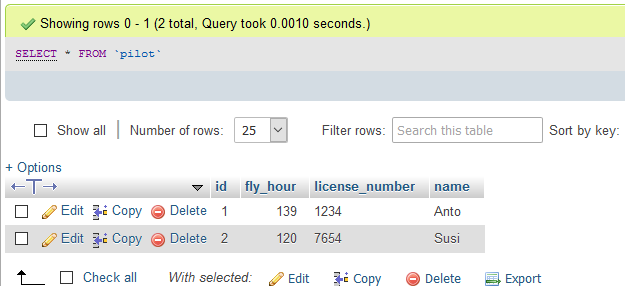
* Akses request post yang telah dibuat untuk “Add a Pilot.” Lalu tambahkan request body sbb:



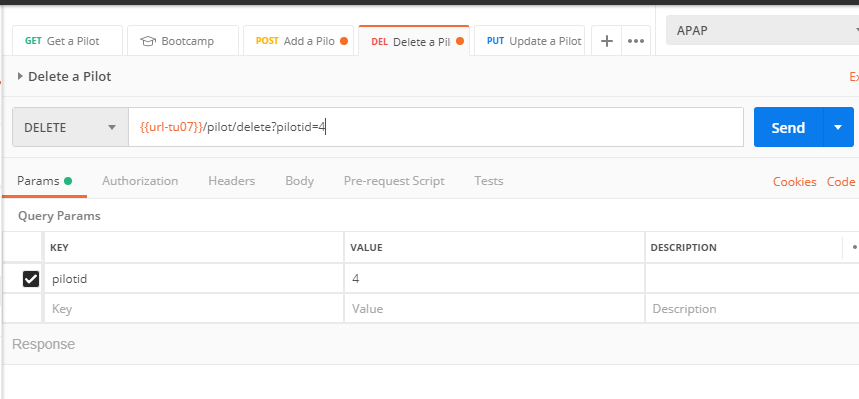
* Kirim *request* dengan menekan tombol “Send”. Anda akan menerima pesan pada *response body* di bawah, sbb.:



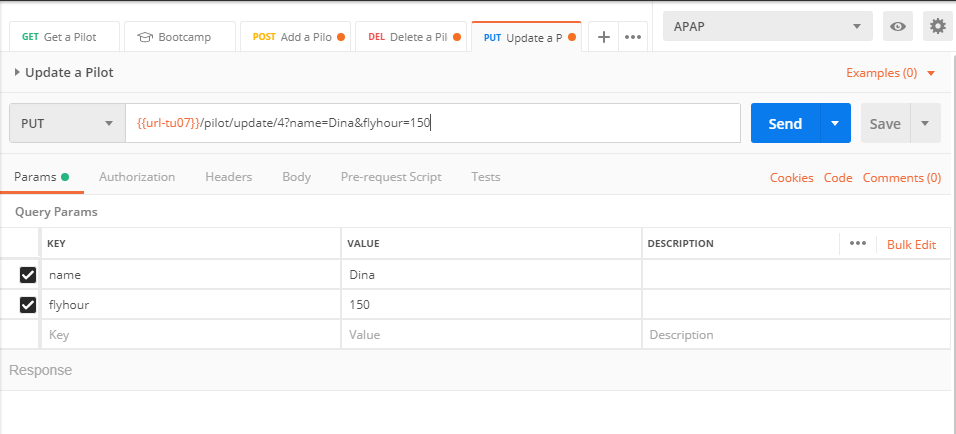
* Cek pada DB apakah data tersebut sesuai.



* Karena pada *delete* dan *update* pilot menggunakan *parameter*, maka jangan lupa untuk menambahkan *paramater* pada Postman.
* Method delete:



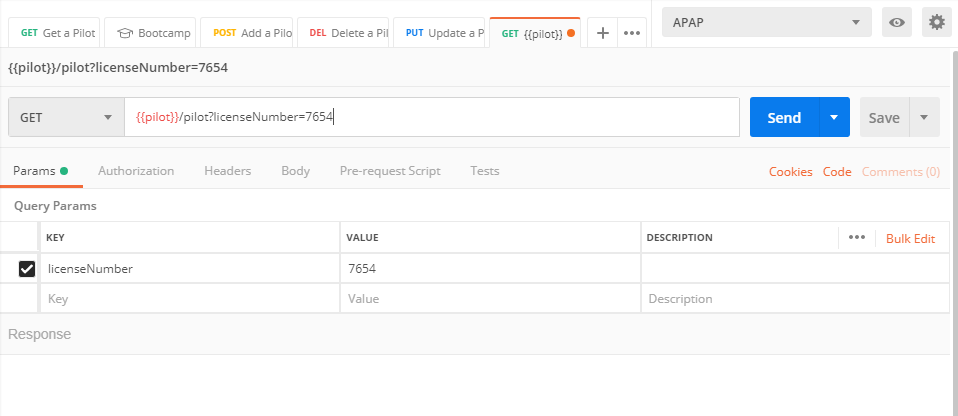
* Method update:



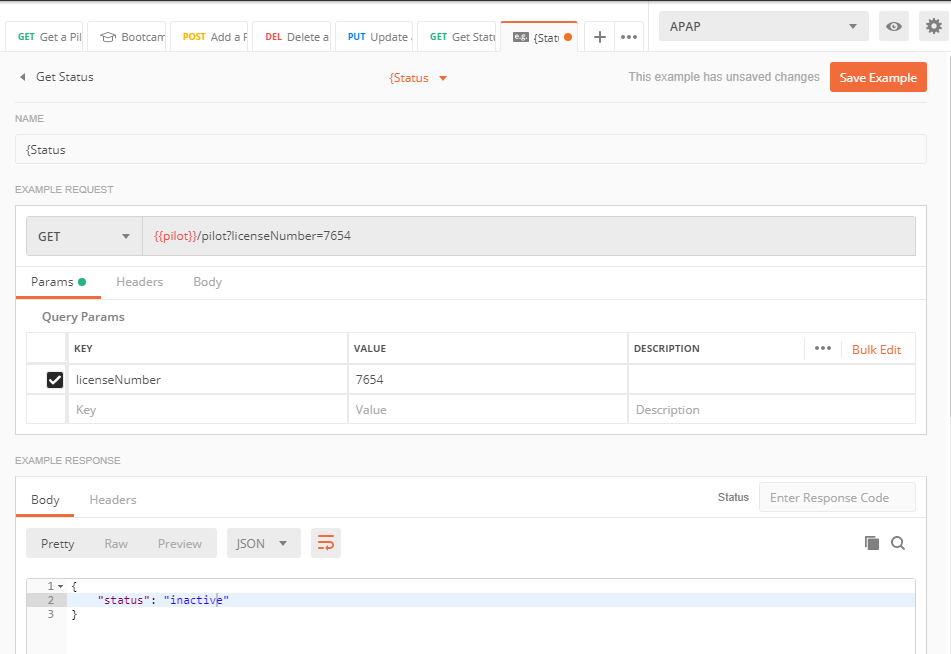
1. **Mockserver**

Mockserver adalah server bayangan API yang bisa diakses dengan spesifikasi *custom*. Mockserver berguna ketika spesifikasi mengharuskan beberapa sistem untuk berkomunikasi satu sama lainnya sedangkan hanya tersedia API *documentation* dan belum ada implementasinya. Dengan kata lain Mockserver mempermudah percepatan *development* sistem yang sedang dikembangkan secara paralel. Mockserver merupakan *service* yang ada pada Postman.

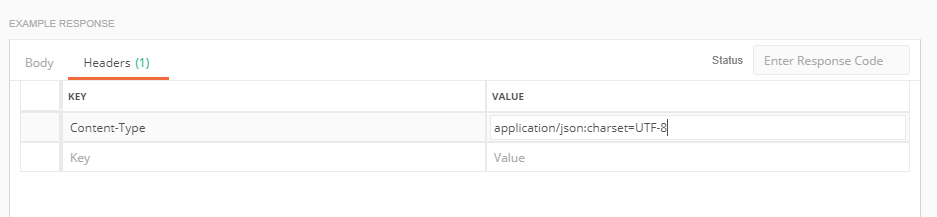
* Buat Collection baru dengan nama “Tutorial-07\_Mock-Server”
* Buatlah request baru pada collection tersebut dengan nama “Get Status”



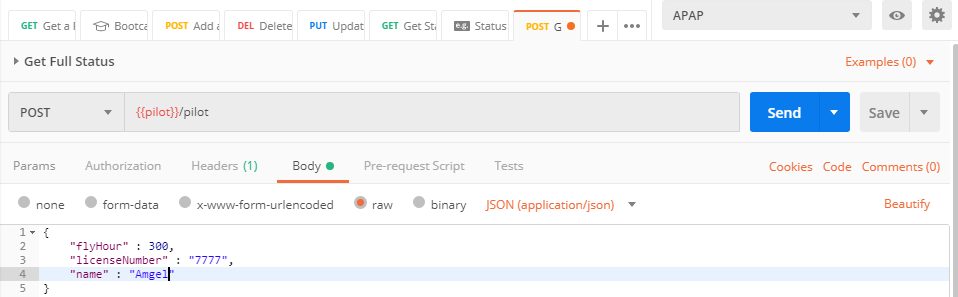
* Klik “Examples (0)” >> “Add Example”
* Buat *example* baru yang bernama “Status” sbb:



* Untuk *response header* spesifikasinya sbb:



* Buatlah *request* lain dengan nama “Get Full Status” lalu Save.



* Buat *example* baru yang bernama “Full Status” sbb.:

