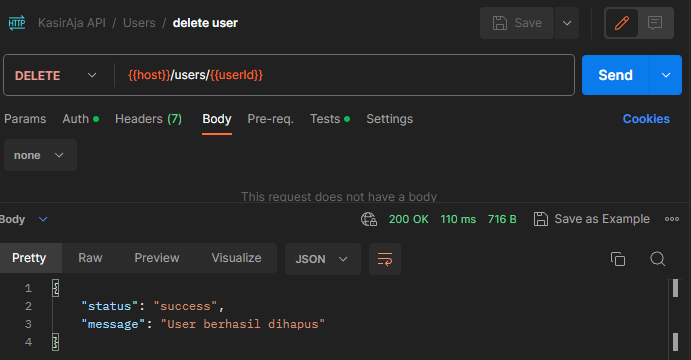
**PRAKTIK**

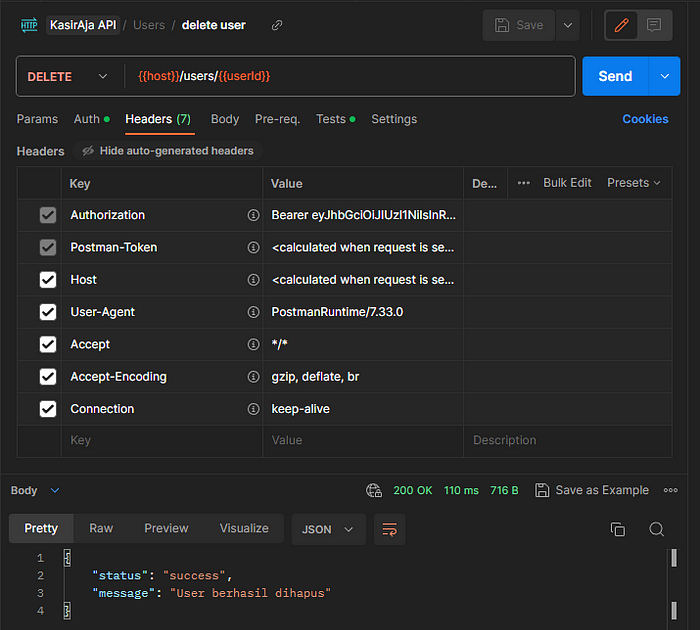
| **Input**  describe('Delete User', () => {  it('Successfully deleted user', () => {  cy.request('DELETE', 'https://reqres.in/api/users/2').then((response) => {  expect(response.status).equal(204)  })  });  }); |
| --- |
| **Output** |

**TUGAS**

**Implement DELETE method**

Oke selanjutnya kita *move* ke implementasi dari method DELETE. Disini kita akan menggunakan request API berikut.





Dari gambar diatas kita dapat mengambil informasi bahwa request API tersebut memiliki method DELETE, kemudian kita juga membutuhkan parameter userId yang diletakkan di path setelah /users/. Selain itu, request tersebut juga harus memiliki headers yang didalamnya terdapat token Authorization yang bisa kita dapatkan setelah menjalankan API Login. Mirip-mirip ya kayak yang PUT tadi, bedanya sekarang kita gaperlu kirim payload nih, harusnya makin gampang dong ya.. Oke gas kita ke file “user.api.js” dulu buat bikin function asynchronous deleteUser(). Kurang lebih code untuk function ini seperti ini ya..

| async deleteUser(token, userId) {  const response = await chai  .request(this.host)  .delete("/users/" + userId)  .set("Authorization", "Bearer " + token);  return response;  } |
| --- |

Berikut full code dari file “user.api.js”



Selanjutnya kita tinggal panggil aja deh function ini ke file “user.spec.js” didalam TC kita. Nah hasilnya gini nih.

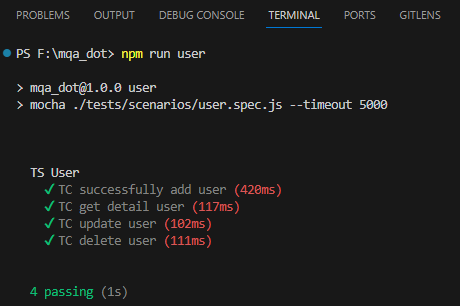
| it("TC delete user", async () => {  const response = await new user().deleteUser(token, userId);  expect(response.statusCode).to.be.equal(200);  expect(response.body.status).to.be.equal(message.success);  expect(response.body.message).to.be.equal(message.successDeleteUser);  }); |
| --- |

karena kita gaperlu kirim payload, jadi tambah pendek kan code nya hehe. Oiya jangan lupa buat tambahin assertion juga ya. Disini assertion nya mengecek statusCode, status, dan message.

Nah berikut full code dari Test Suite User yaa, file “user.spec.js”



*Last but not least,* kita jalankan dulu TS User ini. Karena kita sudah ada script di file “package.json” untuk menjalankan file “user.spec.js” maka kita tinggal jalankan script tersebut.



Yak sudah berhasil ya. *BTW* code ini bisa juga diimplementasikan di module collection lain di API Collection KasirAja ini. Kan masih banyak nih, jadi semangat!!

