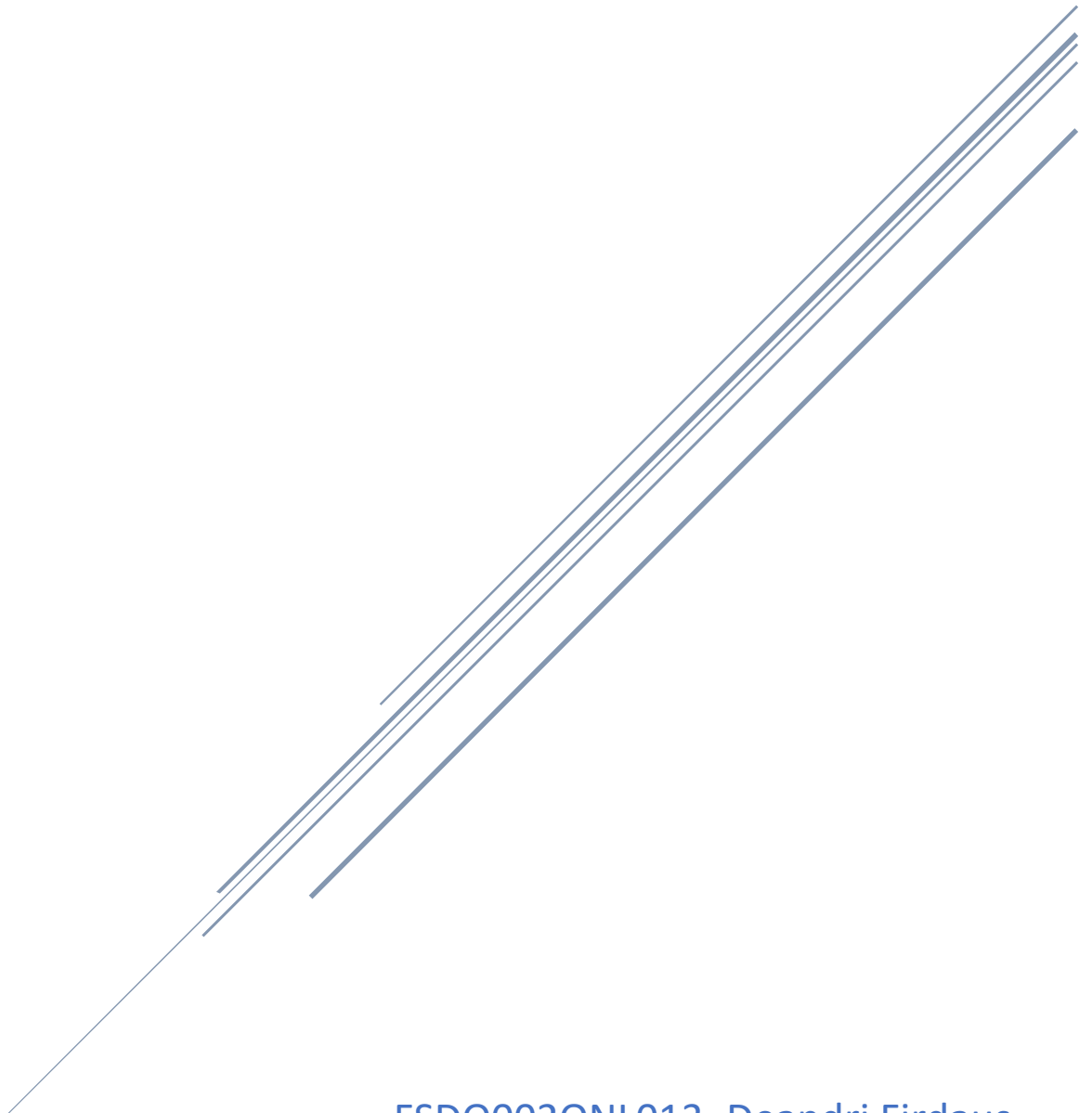


# PROJECT 1

Payment Application using Heroku as a Web Hosting



FSDO002ONL013\_Deandri Firdaus  
Hactiv8 Bootcamp OCBC NISP

## 1. Link dan Keterangan Project

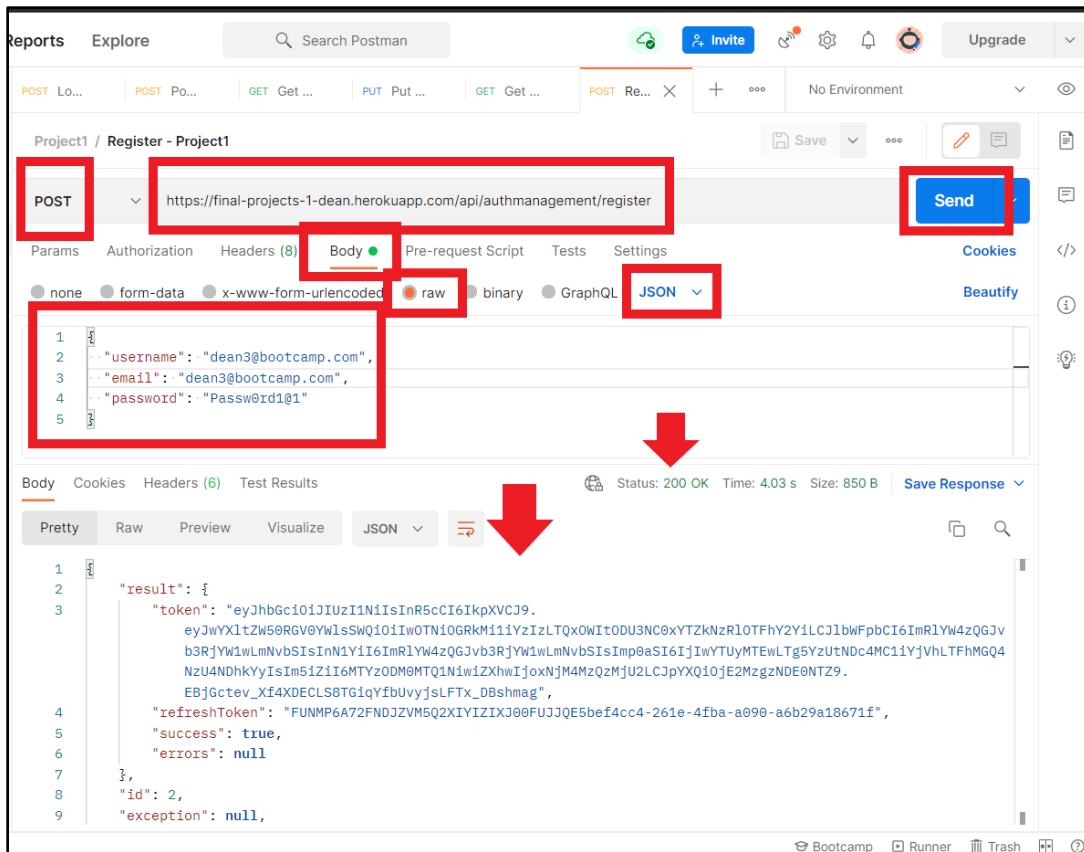
Beberapa link terkait Project1 adalah sebagai berikut:

- Project ini telah diupload seluruhnya ke Heroku dan GitHub.
- Link GitHub = <https://github.com/henchhh/final-project-1-dean>
- Link Heroku = <https://final-projects-1-dean.herokuapp.com/>
- Link Postman 1 = <https://go.postman.co/workspace/Sesi10~78116a65-10de-432a-a936-5cf3d43f1e8b/collection/18450595-ce0fa4f2-d776-479c-9655-8096b1f0ac67>
- Link Postman 2 = Postman link2 : [https://github.com/henchhh/final-project-1-dean/blob/main/Project1.postman\\_collection.json](https://github.com/henchhh/final-project-1-dean/blob/main/Project1.postman_collection.json)
- Saya pribadi dalam pengerjaan proyek ini memiliki beberapa kendala di namespace, docker tidak dapat membaca Sqlite hanya MySQL, source code untuk UseMySQL() di Startup.cs, mengganti di database online diatur menjadi latin-general-ci, dan beberapa source code lainnya.

## 2. Register

Melakukan register dengan cara berikut:

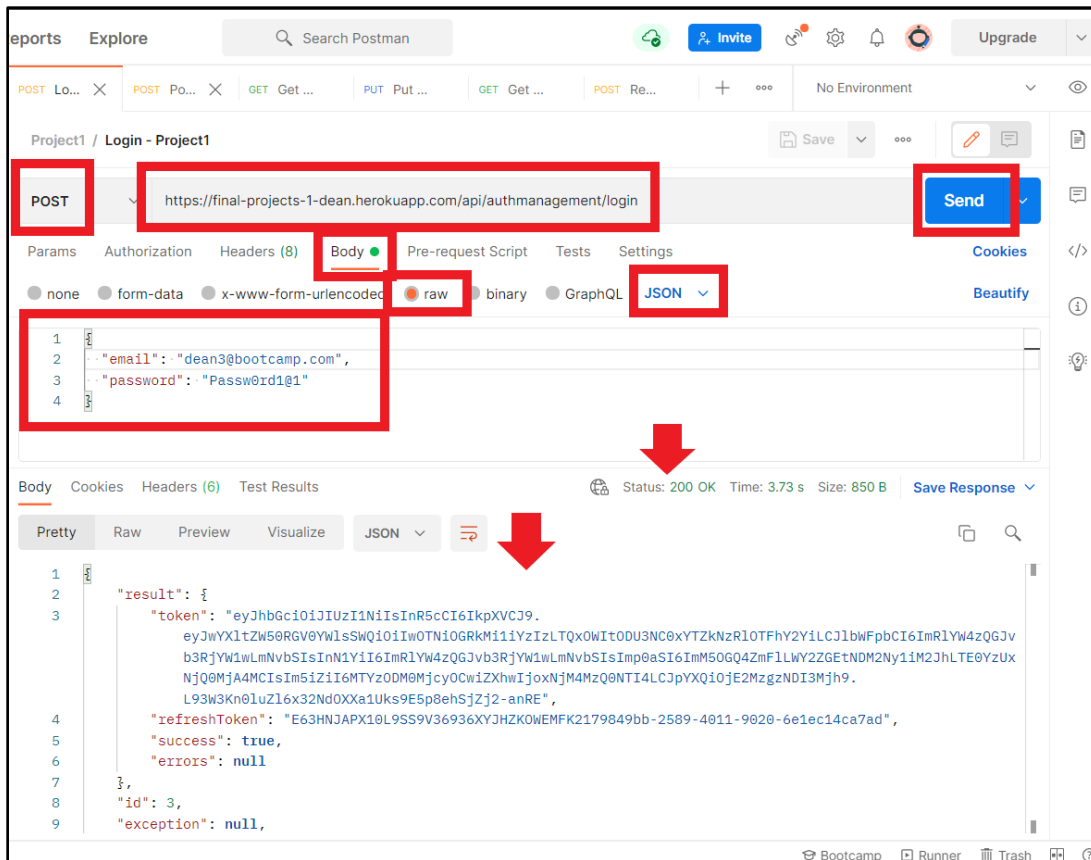
- Memasukkan link berikut ke postman => <https://final-projects-1-dean.herokuapp.com/api/authmanagement/register>.
- Mengganti method menjadi POST.
- Mengubah dari Params menjadi Body.
- Mengubah dari none menjadi raw.
- Mengubah dari Text menjadi JSON.
- Memasukkan username, email, dan password yang ingin didaftarkan. Hal ini perlu digunakan mengingat proyek ini menggunakan Jwt dimana database tidak dapat diakses sebelum mendapatkan token.
- Klik Send.
- Tampil status bernilai Ok dan akan didapatkan token yang nantinya digunakan untuk mengakses database.
- Untuk ke depannya, tidak perlu melakukan register lagi, hanya perlu melakukan login untuk mendapatkan token.



### 3. Login

Melakukan login dengan cara berikut:

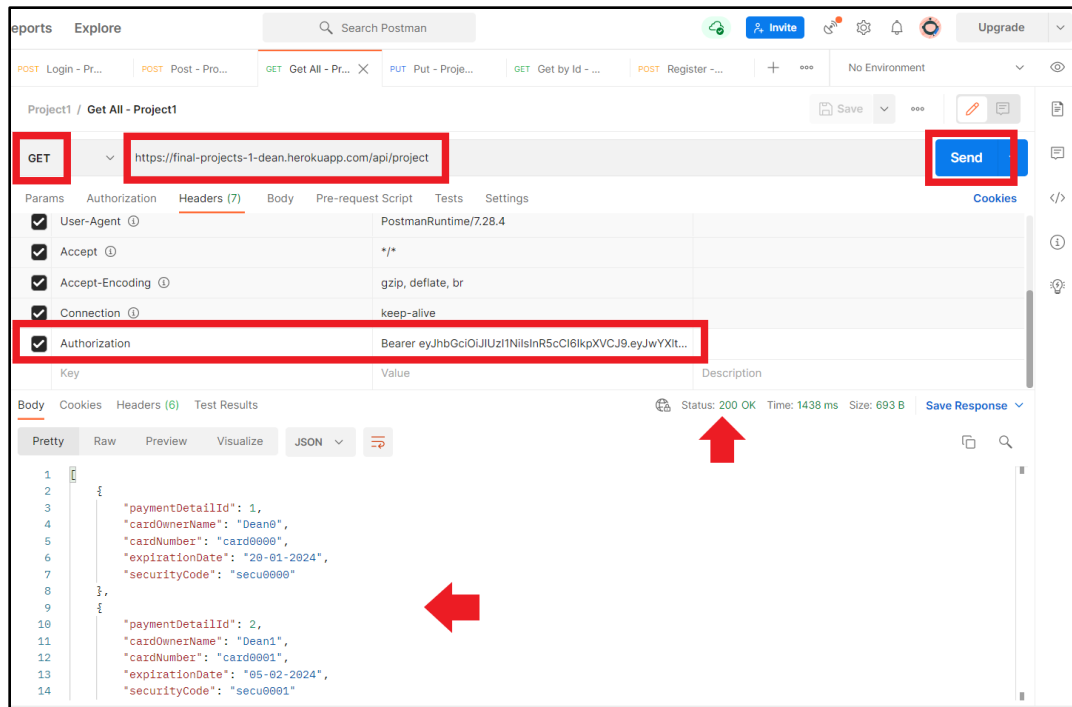
- Memasukkan link berikut ke postman => <https://final-projects-1-dean.herokuapp.com/api/authmanagement/login>.
- Mengganti method menjadi POST.
- Mengubah dari Params menjadi Body.
- Mengubah dari none menjadi raw.
- Mengubah dari Text menjadi JSON.
- Memasukkan email dan password yang telah didaftarkan sebelumnya. Hal ini perlu digunakan untuk mendapatkan token yang nantinya berfungsi untuk mengakses database.
- Klik Send.
- Tampil status bernilai Ok dan akan didapatkan token yang nantinya digunakan untuk mengakses database.



## 4. Get All Payment Data

Mendapatkan seluruh data payment dengan cara berikut:

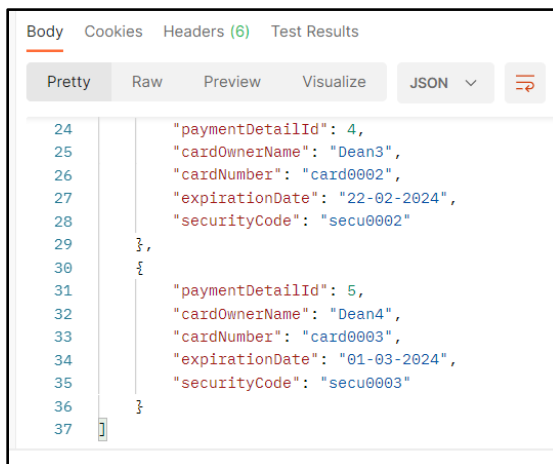
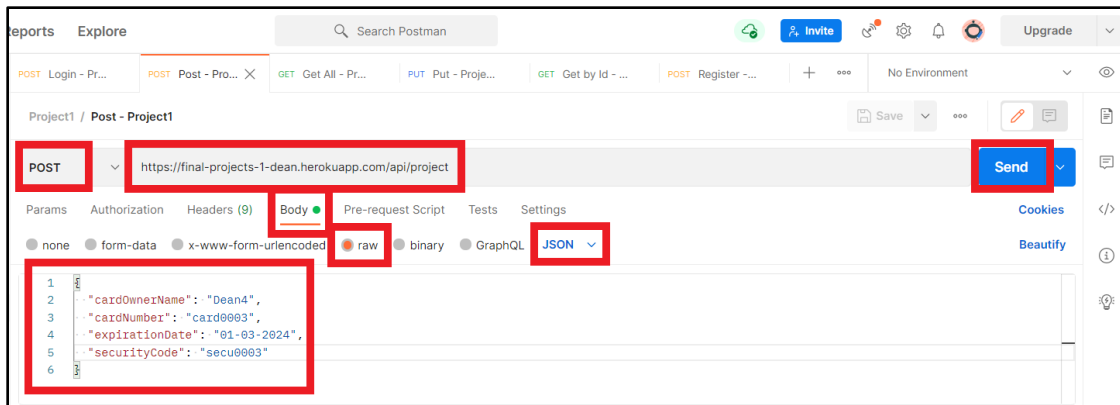
- Memasukkan link berikut ke postman => <https://final-projects-1-dean.herokuapp.com/api/project>.
- Menggunakan method GET.
- Klik Header, scroll sampai bawah, lalu tambahkan key baru, yaitu Authorization.
- Value-nya => Bearer {token}. Token didapatkan setelah register ataupun login.
- Klik Send.
- Tampil seluruh data payment yang telah dimasukkan.



## 5. Post New Payment Data

Memasukkan data payment baru dengan cara berikut:

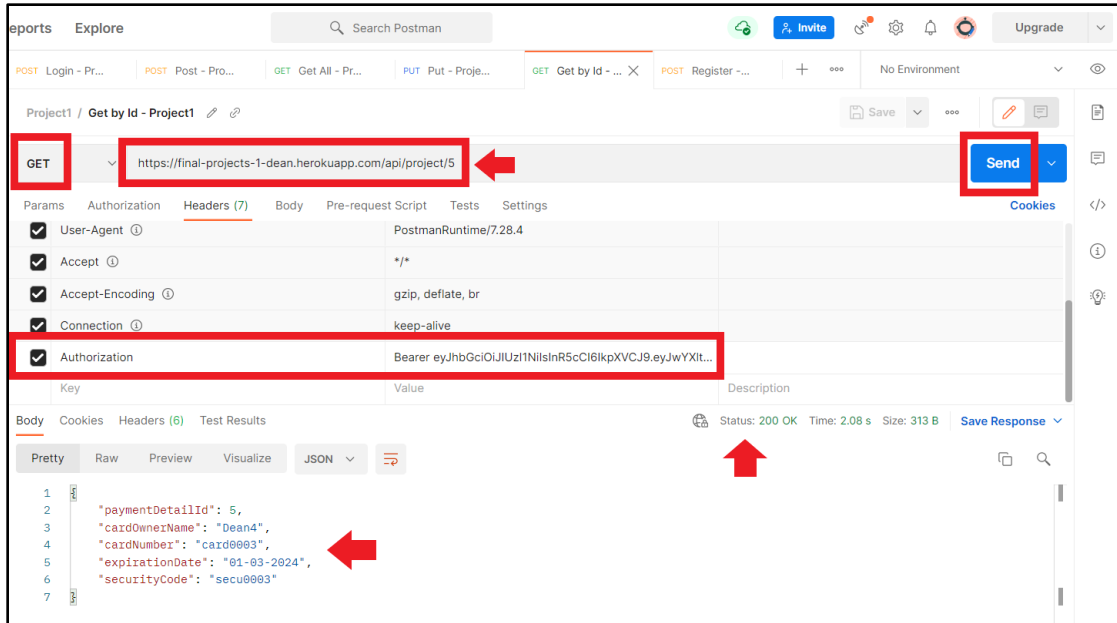
- Memasukkan link berikut ke postman => <https://final-projects-1-dean.herokuapp.com/api/project>.
- Mengganti method menjadi POST.
- Klik Header, scroll sampai bawah, lalu tambahkan key baru, yaitu Authorization.
- Value-nya => Bearer {token}. Token didapatkan setelah register ataupun login.
- Mengubah dari Params menjadi Body.
- Mengubah dari none menjadi raw.
- Mengubah dari Text menjadi JSON.
- Memasukkan data baru yang berisikan cardOwnerName, cardNumber, expirationDate, dan securityCode. Tidak perlu memasukkan paymentDetailId karena akan bernilai otomatis dan auto increment.
- Klik Send.
- Coba cek data tersebut menggunakan proses nomor 3.



## 6. Get Data by ID

Mendapatkan data berdasarkan paymentDetailId dengan cara berikut:

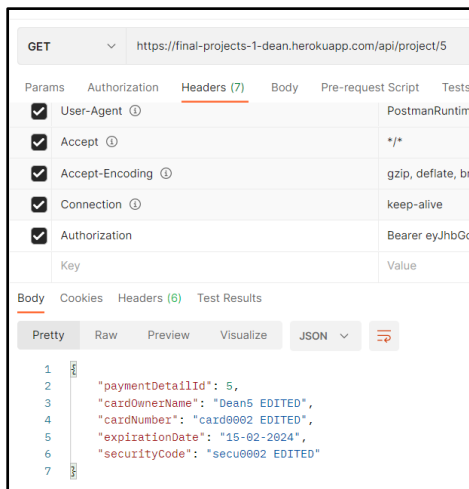
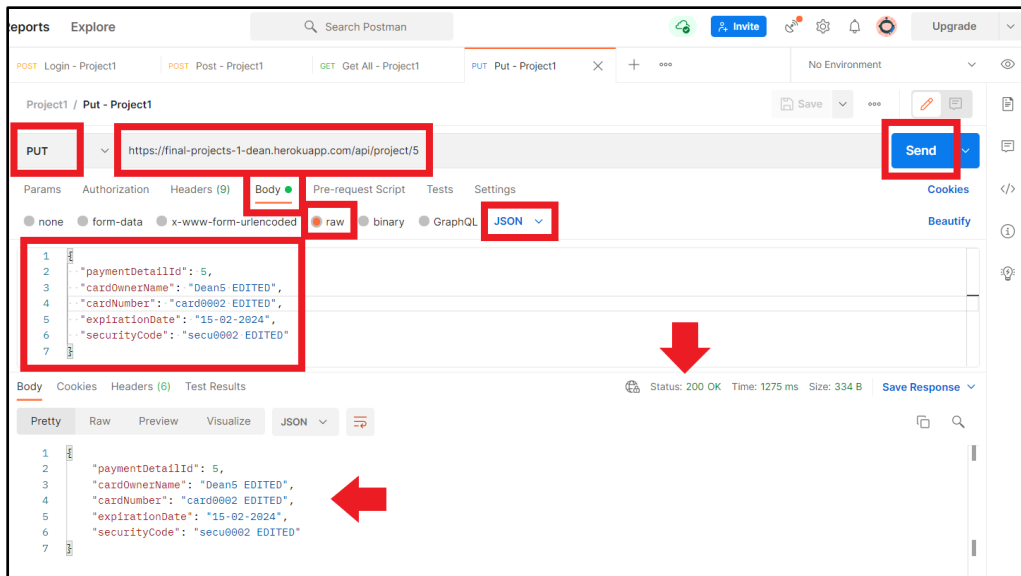
- Memasukkan link berikut ke postman => [https://final-projects-1-dean.herokuapp.com/api/project/masukkan\\_paymentDetailID](https://final-projects-1-dean.herokuapp.com/api/project/masukkan_paymentDetailID).
- Akan dicoba untuk memanggil data yang telah diinput sebelumnya dengan paymentDetailId bernilai 5.
- Menggunakan method GET.
- Klik Header, scroll sampai bawah, lalu tambahkan key baru, yaitu Authorization.
- Value-nya => Bearer {token}. Token didapatkan setelah register ataupun login.
- Klik Send.
- Tampil detail data payment berdasarkan paymentDetailId yang telah dimasukkan.



## 7. Put Data by ID

Mengedit salah satu data berdasarkan paymentDetailId dengan cara berikut:

- Memasukkan link berikut ke postman => [https://final-projects-1-dean.herokuapp.com/api/project/masukkan\\_paymentDetailId](https://final-projects-1-dean.herokuapp.com/api/project/masukkan_paymentDetailId).
- Akan dicoba untuk mengubah isi dari data yang telah diinput sebelumnya dengan paymentDetailId bernilai 5.
- Menggunakan method PUT.
- Klik Header, scroll sampai bawah, lalu tambahkan key baru, yaitu Authorization.
- Value-nya => Bearer {token}. Token didapatkan setelah register ataupun login.
- Klik Send.
- Menampilkan data tersebut menggunakan proses 5.



## 8. Delete Data by ID

Menghapus salah satu data berdasarkan paymentDetailId dengan cara berikut:

- Memasukkan link berikut ke postman => [https://final-projects-1-dean.herokuapp.com/api/project/masukkan\\_paymentDetailID](https://final-projects-1-dean.herokuapp.com/api/project/masukkan_paymentDetailID).
- Akan dicoba untuk menghapus data yang telah diinput sebelumnya dengan paymentDetailId bernilai 5.
- Menggunakan method DELETE.
- Klik Header, scroll sampai bawah, lalu tambahkan key baru, yaitu Authorization.
- Value-nya => Bearer {token}. Token didapatkan setelah register ataupun login.
- Klik Send.
- Menampilkan semua data menggunakan proses 3.



Reports Explore

Search Postman

POST Login - Pr... POST Post - Pro... GET Get All - Pr... PUT Put - Proje... DELETE Delete - Pr... GET Get by Id - ... + No Environment

Project1 / Delete - Project1

DELETE https://final-projects-1-dean.herokuapp.com/api/project/5 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

☒ User-Agent ① PostmanRuntime/7.28.4

☒ Accept ① \*/\*

☒ Accept-Encoding ① gzip, deflate, br

☒ Connection ① keep-alive

☒ Authorization Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJwYXIt...

Body Cookies Headers (6) Test Results Status: 200 OK Time: 1460 ms Size: 334 B Save Response

Reports Explore

Search Postman

POST Login - Pr... POST Post - Pro... GET Get All - Pr... PUT Put - Proje... DELETE Delete - Pr... GET Get by Id - ... + No Environment

Project1 / Get All - Project1

GET https://final-projects-1-dean.herokuapp.com/api/project Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

☒ User-Agent ① PostmanRuntime/7.28.4

☒ Accept ① \*/\*

☒ Accept-Encoding ① gzip, deflate, br

☒ Connection ① keep-alive

☒ Authorization Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJwYXIt...

Body Cookies Headers (6) Test Results Status: 200 OK Time: 1198 ms Size: 693 B Save Response

Pretty Raw Preview Visualize JSON

```
17 {
18   "paymentDetailId": 3,
19   "cardOwnerName": "Dean2",
20   "cardNumber": "card9992",
21   "expirationDate": "15-02-2024",
22   "securityCode": "secu9992"
23 },
24 {
25   "paymentDetailId": 4,
26   "cardOwnerName": "Dean3",
27   "cardNumber": "card9992",
28   "expirationDate": "22-02-2024",
29   "securityCode": "secu9992"
30 }
```