Documentation –

Go REST API

REST API with GIN Framework

shabrinur @ github

Simple REST API operations for:

* User registration & login
* Password reset
* CRUD with PostgreSQL
* File upload & download

# Registration Module

## Register User

Register **new user** with **valid email** as **username**.

curl -L -X POST 'http://localhost:8080/v1/registration' \

-H 'Content-Type: application/json' \

--data-raw '{

    "name": "{{display name for user}}",

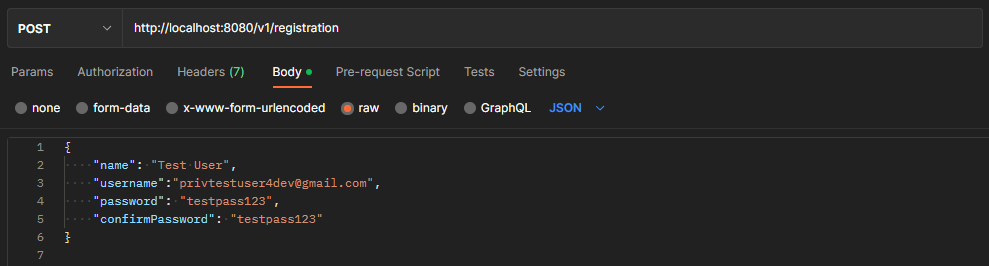
    "username":"{{valid email as username}}",

    "password": "{{password}}",

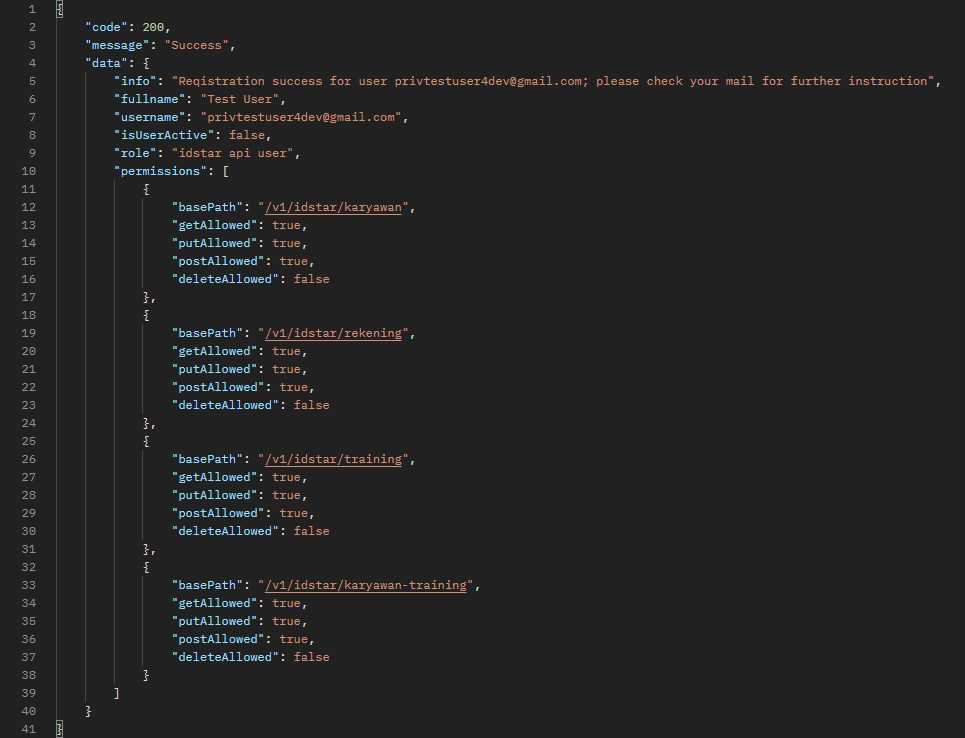
    "confirmPassword": "{{confirm password}}"

}'

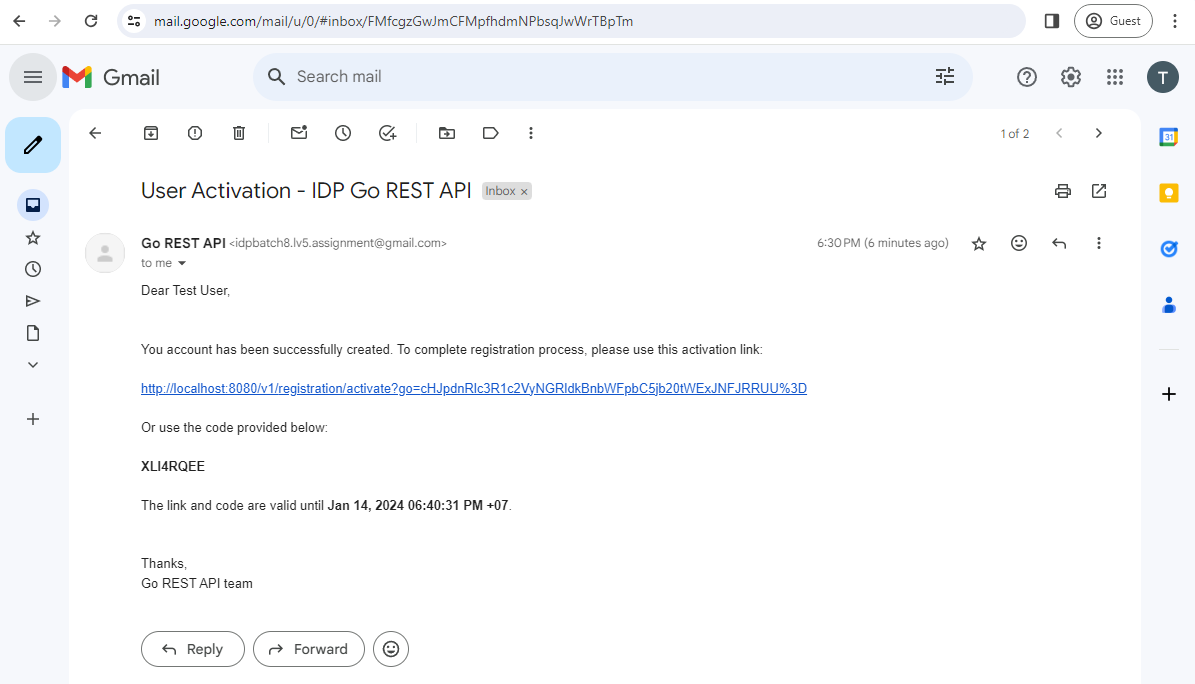
**Sample request:**



**Sample response:**



**Sample activation email:**



## Get Activation Link

Request new **activation email**.

curl -L -X POST 'http://localhost:8080/v1/registration/send-link' \

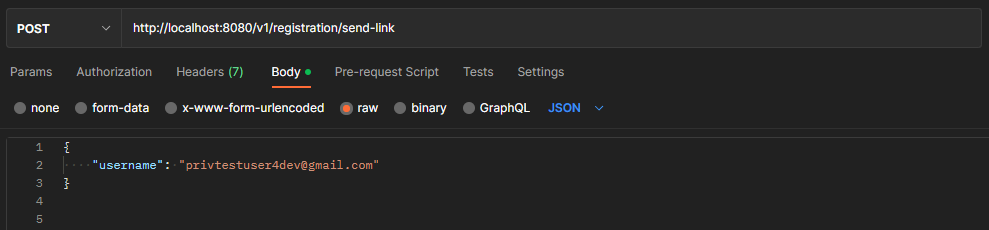
-H 'Content-Type: application/json' \

--data-raw '{

    "username": "{{registered username}}"

}'

**Sample request:**



**Sample response:**



## Activation By Code

Activate user by **code** provided in **activation email**.

curl -L -X POST 'http://localhost:8080/v1/registration/activate' \

-H 'Content-Type: application/json' \

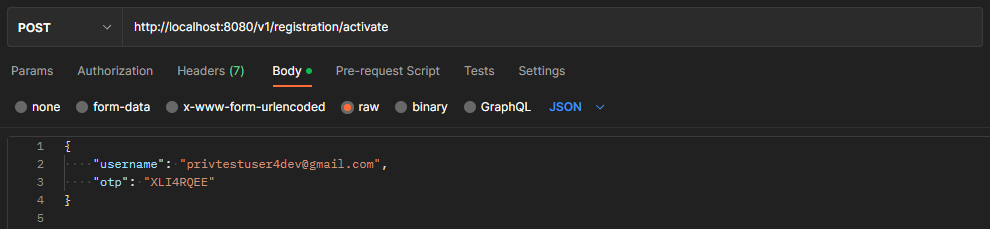
--data-raw '{

    "username": "{{registered username}}",

    "otp": "{{activation code}}"

}'

**Sample request:**



**Sample response:**



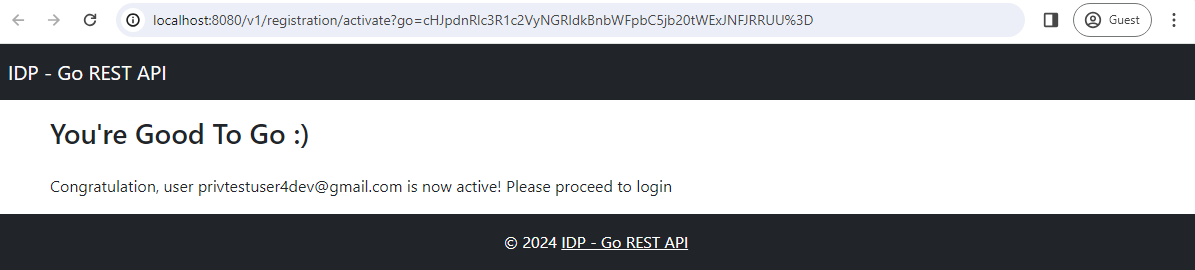
## Activation By Link

Activate user by **link** provided in **activation email**.

Open link in **web browser**.

http://localhost:8080/v1/registration/activate?go={{encoded activation code}}

When link is still **valid**, message **activation success** will be displayed.



# User Login Module

## Login Username & Password

Login using registered **username** and **password** to generate **token** for **authorization**.

curl -L -X POST 'http://localhost:8080/v1/user-login/login' \

-H 'Content-Type: application/json' \

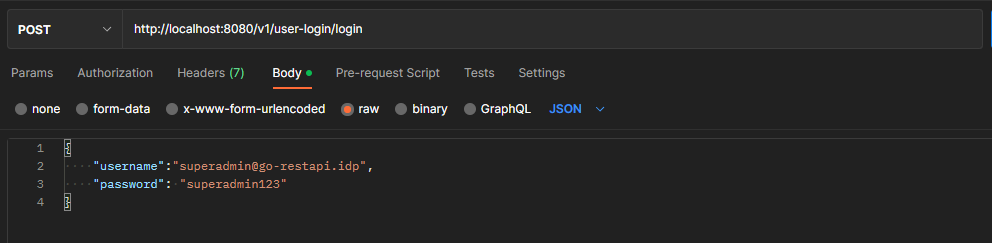
--data-raw '{

    "username":"{{username}}",

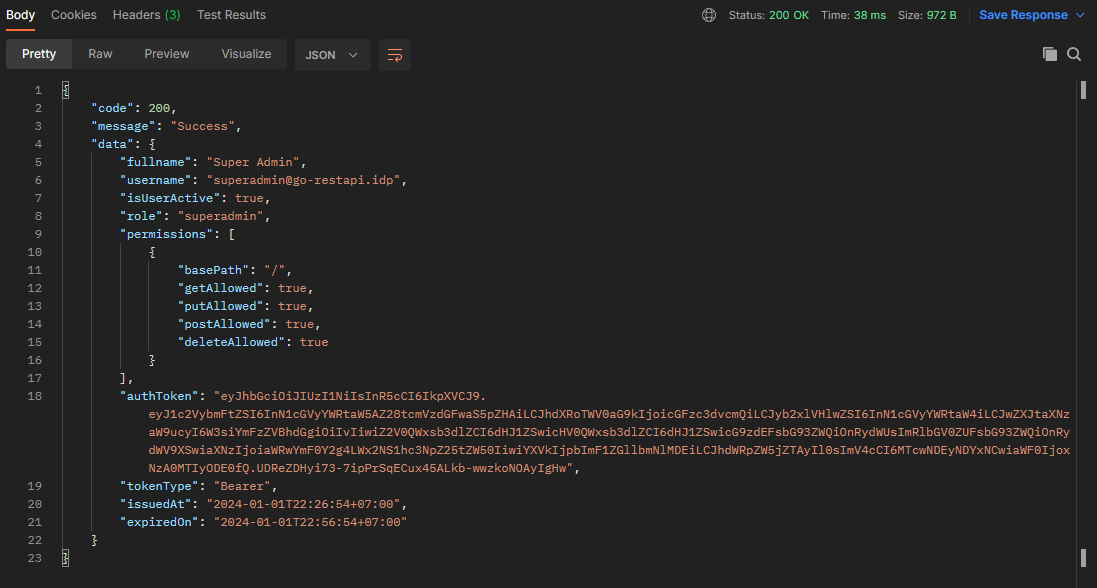
    "password": "{{password}}"

}'

**Sample request:**



**Sample response:**



## Login Google OAuth

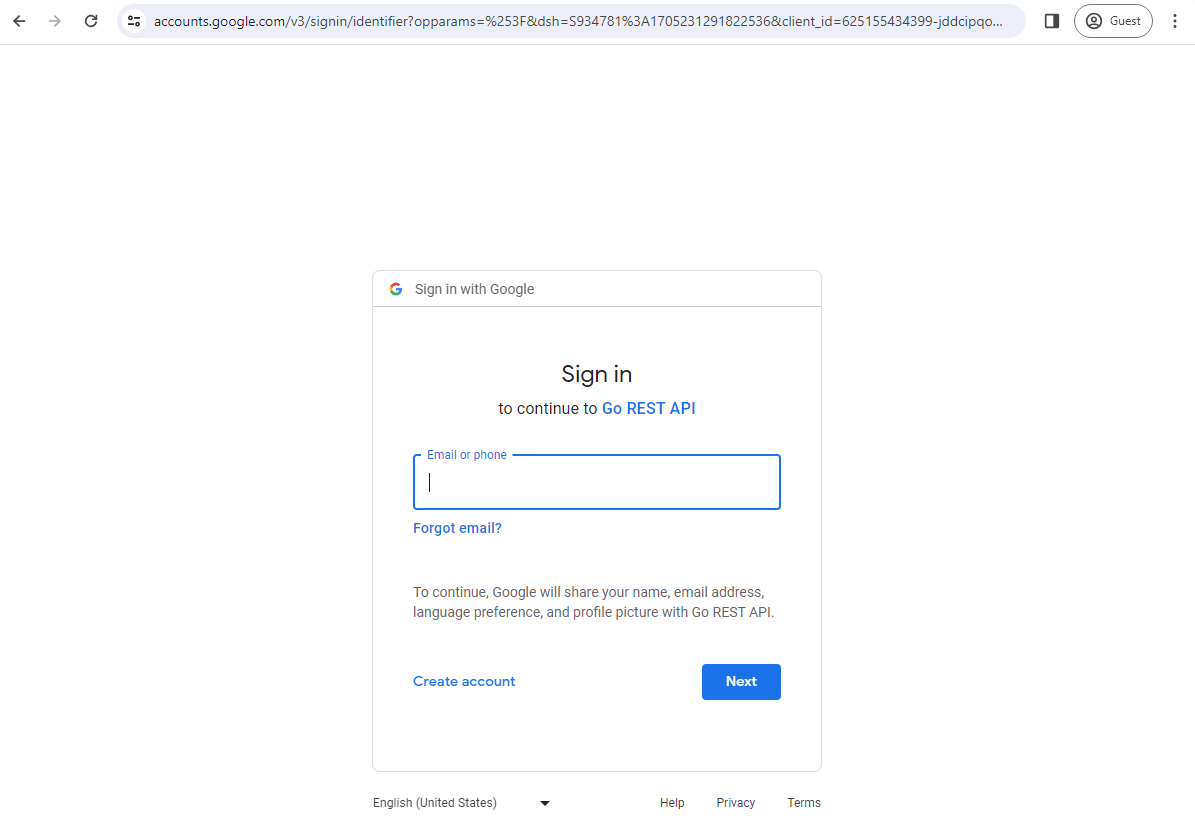
Login using **Google OAuth** to generate **token** for **authorization**.

Access address from **web browser**.

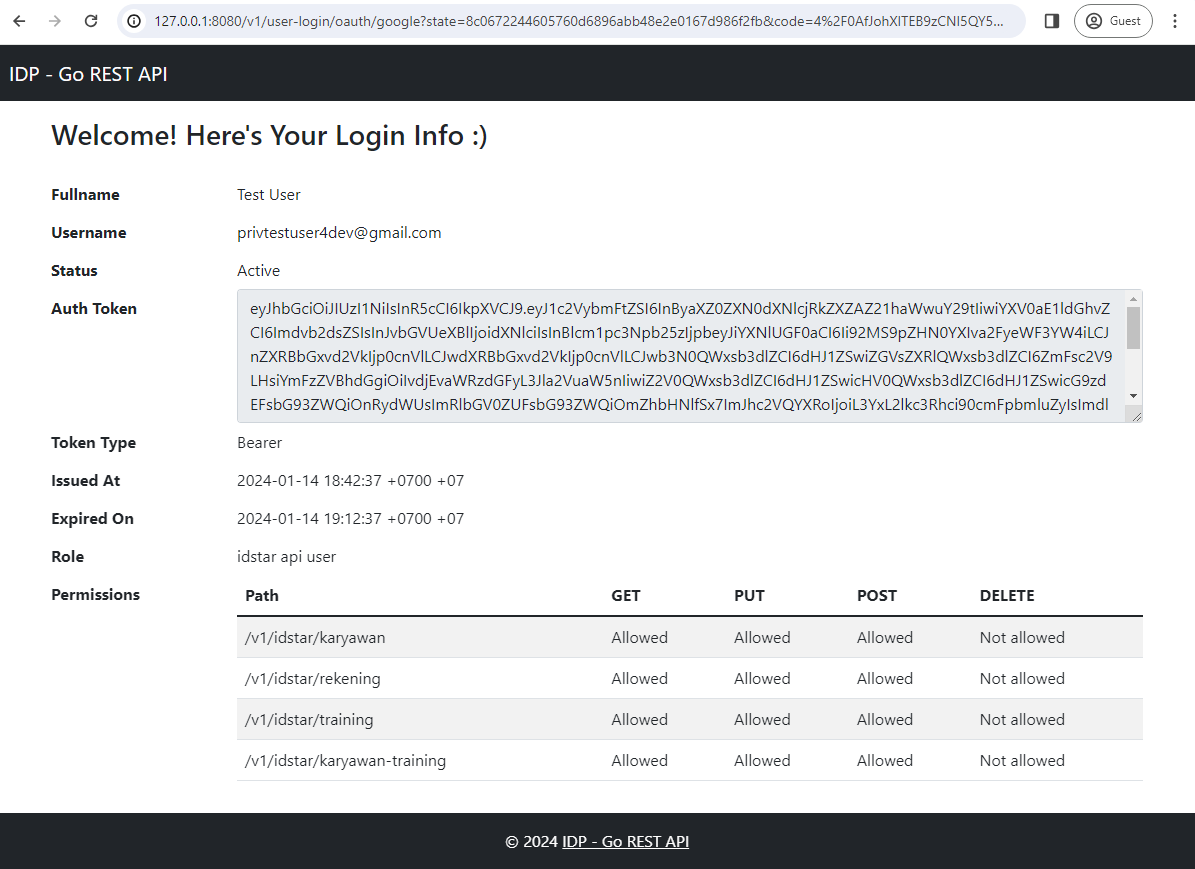
http://localhost:8080/v1/user-login/oauth/google



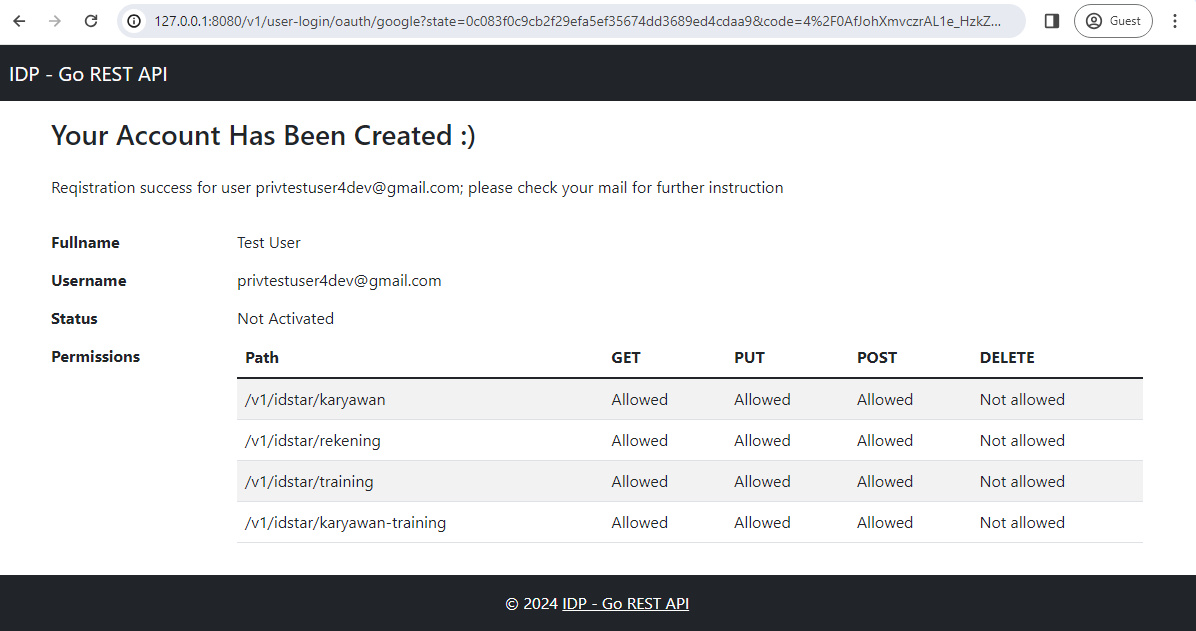
Sign in with **Google account**.

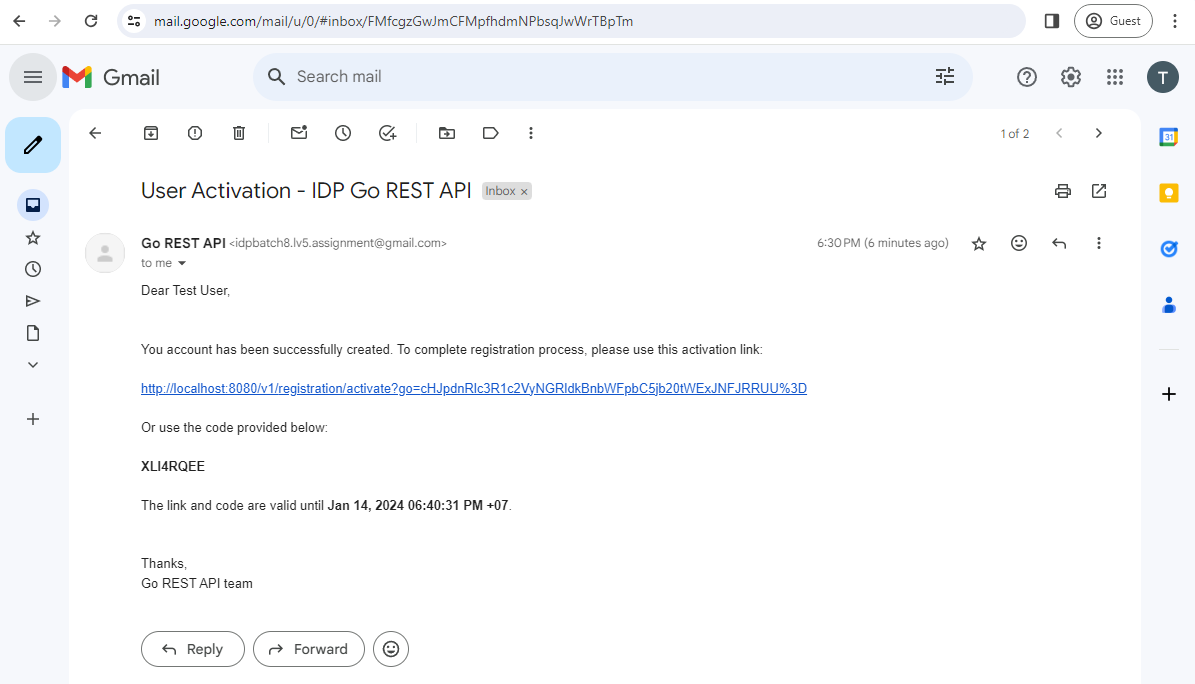


If account **already registered** in application, **login info** will be displayed.



If account **not yet registered** in application, **registration info** will be displayed and **activation email** will be sent.





# Forget Password Module

## Get OTP

Get **reset password email**.

curl -L -X POST 'http://localhost:8080/v1/forget-password/send-otp' \

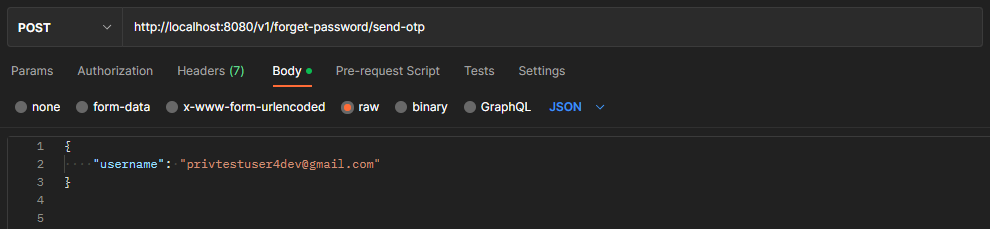
-H 'Content-Type: application/json' \

--data-raw '{

    "username": "{{registered username}}"

}'

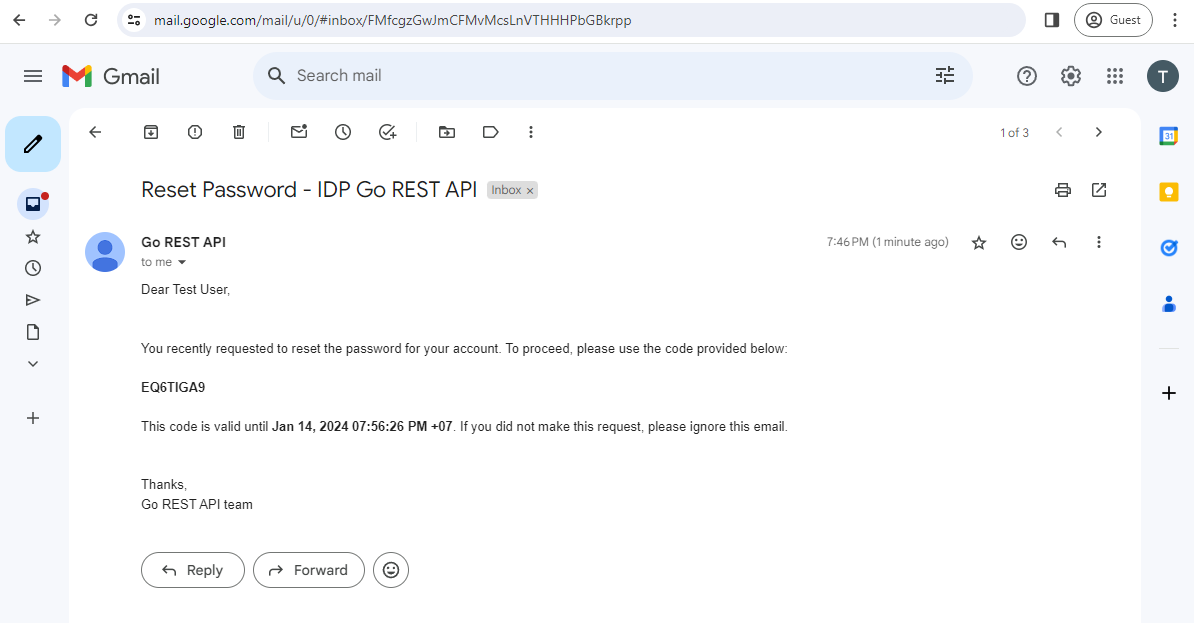
**Sample request:**



**Sample response:**



**Sample reset password email:**



## Validate OTP

Validate **code** provided in **reset password email**.

curl -L -X POST 'http://localhost:8080/v1/forget-password/validate-otp' \

-H 'Content-Type: application/json' \

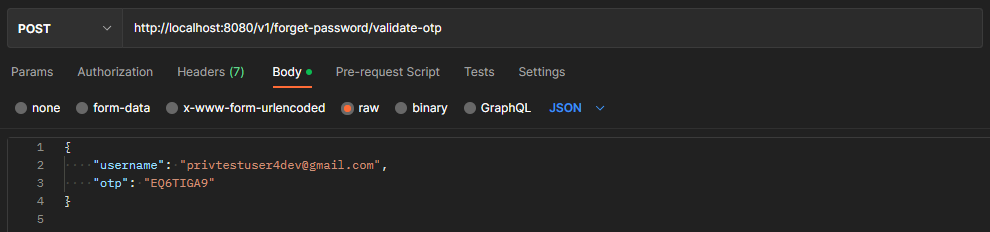
--data-raw '{

    "username": "{{registered username}}",

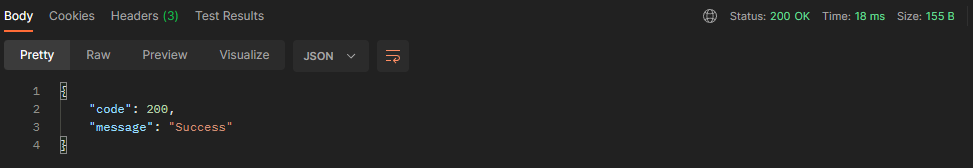
    "otp": "{{reset password code}}"

}

**Sample request:**



**Sample response:**



## Change Password

Set **new password** using **code** provided in **reset password email**.

curl -L -X POST 'http://localhost:8080/v1/forget-password/change-password' \

-H 'Content-Type: application/json' \

--data-raw '{

    "username": "{{registered username}}",

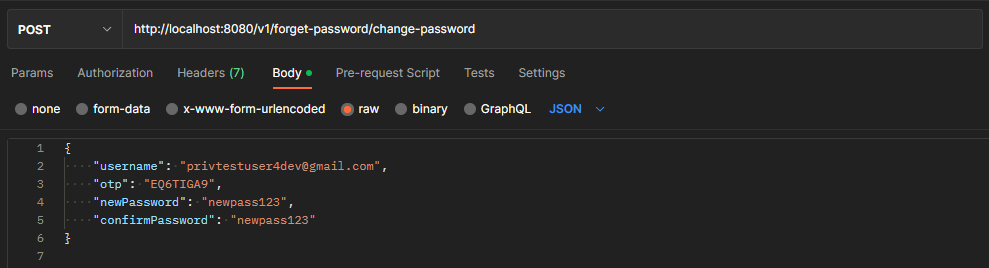
    "otp": "{{reset password code}}",

    "newPassword": "{{new password}}",

    "confirmPassword": "{{confirm new password}}"

}

**Sample request:**



**Sample response:**



# IDstar - Karyawan Module

**Note:** API under this module requires authorization type **Bearer token**. Token can be generated from **user login**.

## Create Karyawan

Create record of **karyawan** based on request body.

curl -L -X POST 'http://localhost:8080/v1/idstar/karyawan/save' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

    "dob": "{{Tanggal lahir karyawan YYYY-MM-DD}}",

    "nama": "{{Nama karyawan}}",

    "status": "{{Status active/inactive}}",

    "alamat": "{{Alamat karyawan}}",

    "detailKaryawan": {

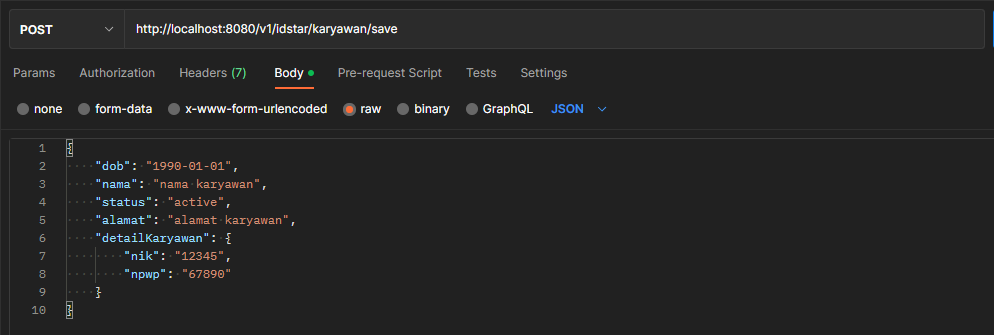
        "nik": "{{NIK karyawan}}",

        "npwp": "{{NPWP karyawan}}"

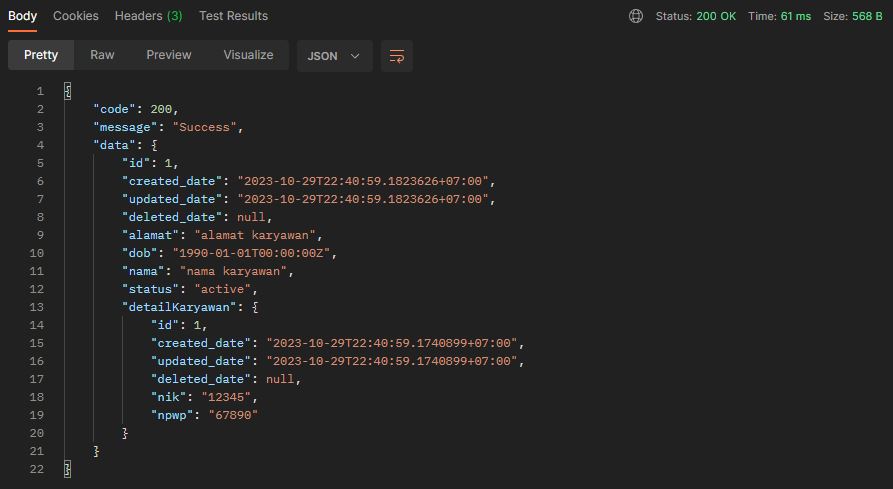
    }

}'

**Sample request:**



**Sample response:**



## Update Karyawan

Update record of **karyawan** based on request body.

curl -L -X PUT 'http://localhost:8080/v1/idstar/karyawan/update' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

    "id": {{ID karyawan numerik}},

    "dob": "{{Update tanggal lahir karyawan YYYY-MM-DD}}",

    "nama": "{{Update nama karyawan}}",

    "status": "{{Update status active/inactive}}",

    "alamat": "{{Update alamat karyawan}}",

    "detailKaryawan": {

        "id": {{ID karyawan numerik}},

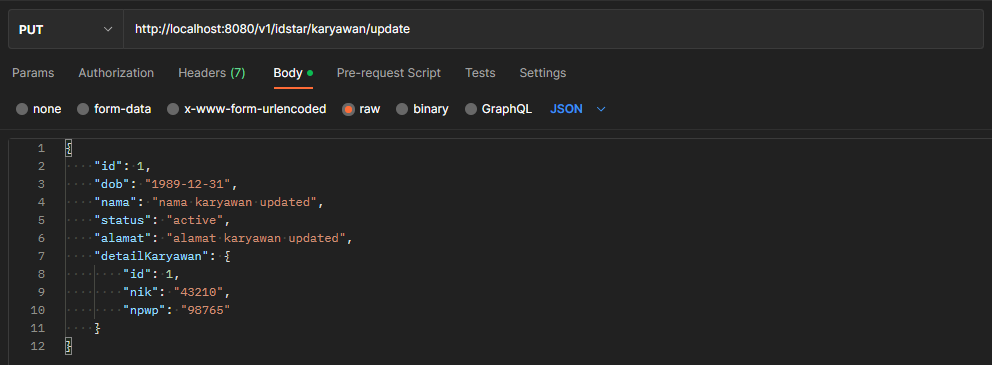
        "nik": "{{Update NIK karyawan}}",

        "npwp": "{{Update NPWP karyawan}}"

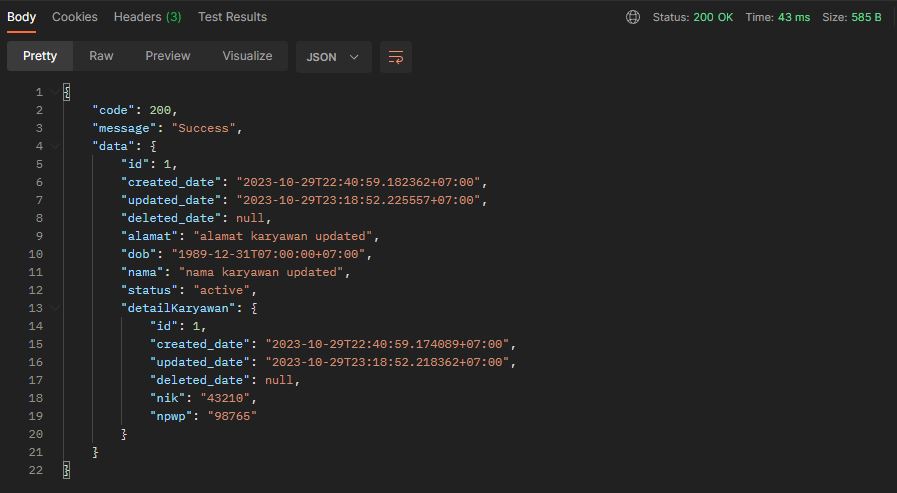
    }

}'

**Sample request:**



**Sample response:**



## Get Karyawan List

Get list of **karyawan** with paging and sorting.

Paging and sorting specification can be sent through **form-data** or **query parameter**.

curl -L -X GET 'http://localhost:8080/v1/idstar/karyawan/list' \

-H 'Authorization: Bearer {{authorization token from login}}' \

-F 'page="{{Nomor paging}}"' \

-F 'size="{{Jumlah record di 1 page}}"' \

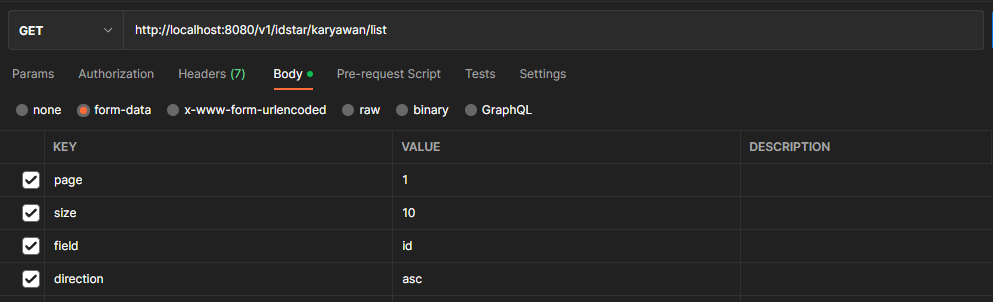
-F 'field="{{Field sorting}}"' \

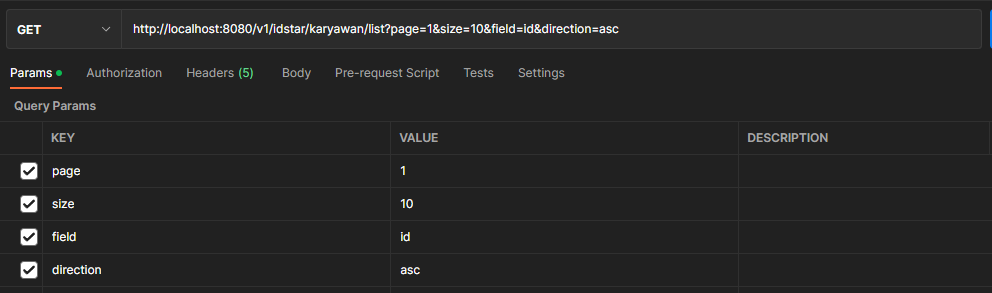
-F 'direction="{{Arah sorting asc/desc}}"'

curl -L -g -X GET 'http://localhost:8080/v1/idstar/karyawan/list?page={{Nomor paging}}&size={{Jumlah record di 1 page}}&field={{Field sorting}}&direction={{Arah sorting asc/desc}}' \

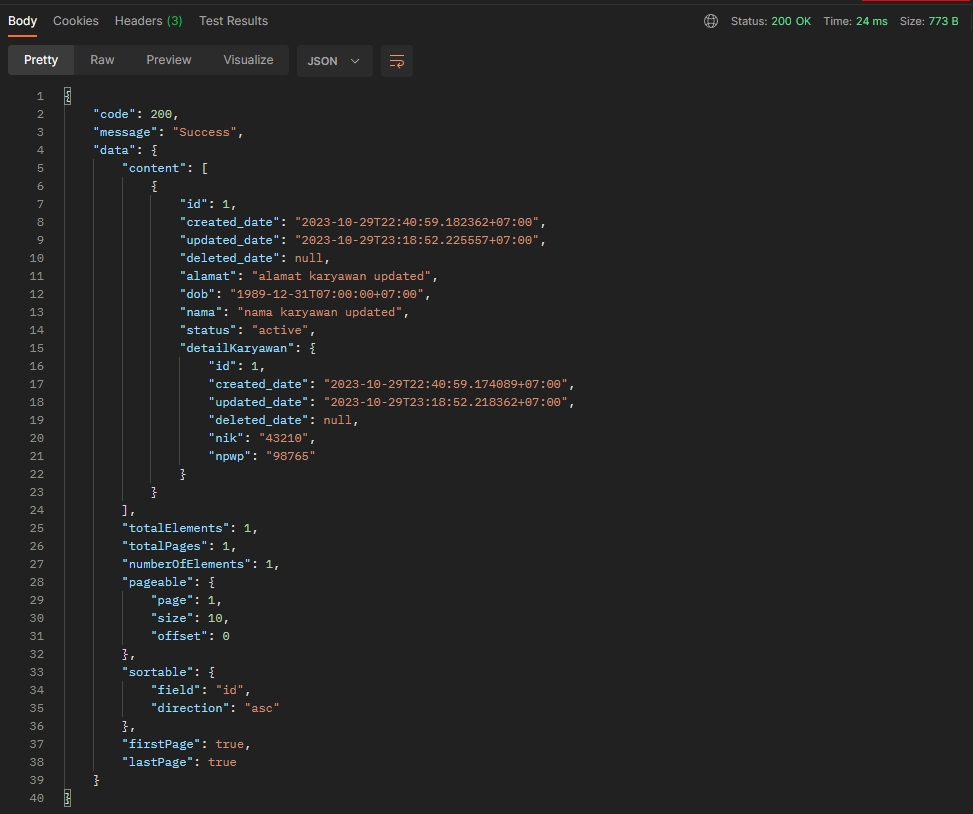
-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**





**Sample response:**



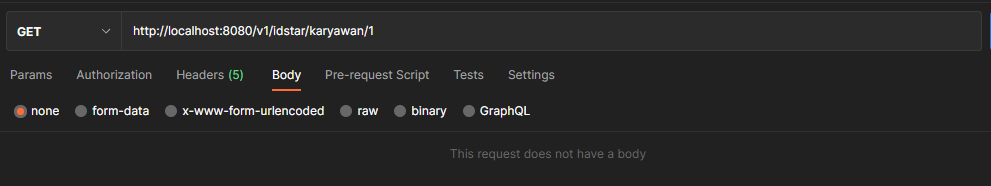
## Get Karyawan By ID

Get one **karyawan** record by **ID**.

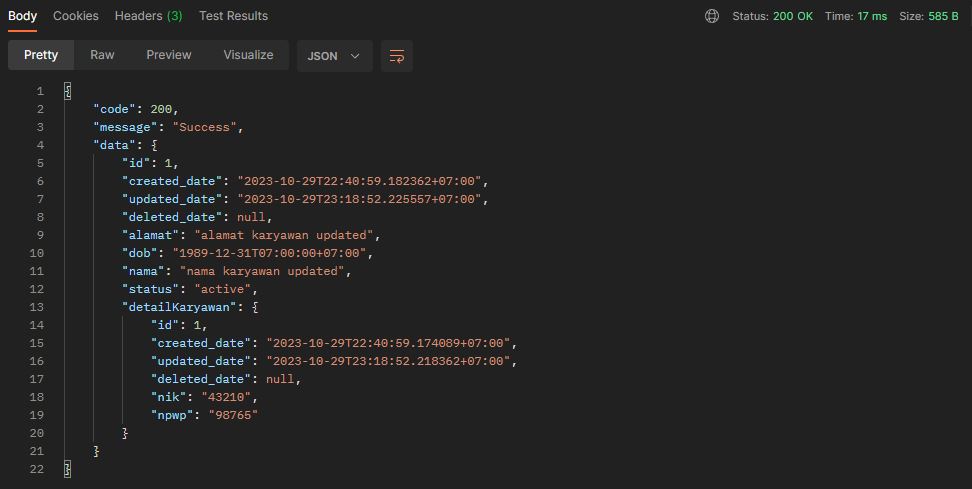
curl -L -g -X GET 'http://localhost:8080/v1/idstar/karyawan/{{ID karyawan numerik}}' \

-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**



**Sample response:**



## Delete Karyawan

Delete one **karyawan** record by **ID**.

curl -L -X DELETE 'http://localhost:8080/v1/idstar/karyawan/delete' \

-H 'Content-Type: application/json' \

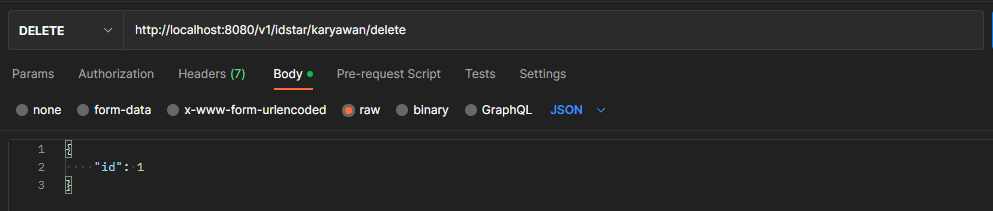
-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

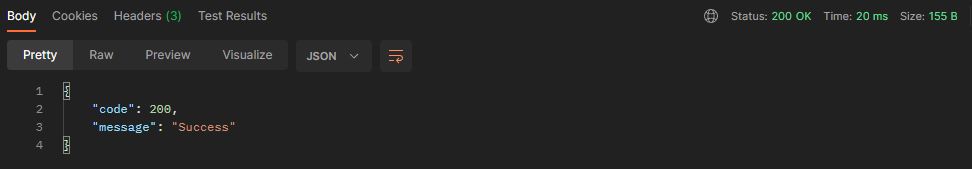
    "id": {{ID karyawan numerik}}

}'

**Sample request:**



**Sample response:**



# IDstar - Rekening Module

**Note:** API under this module requires authorization type **Bearer token**. Token can be generated from **user login**.

## Create Rekening

Create record of **rekening** based on request body.

curl -L -X POST 'http://localhost:8080/v1/idstar/rekening/save' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

    "jenis": "{{Jenis rekening}}",

    "nama": "{{Nama pemilik rekening}}",

    "rekening": "{{Nomor rekening}}",

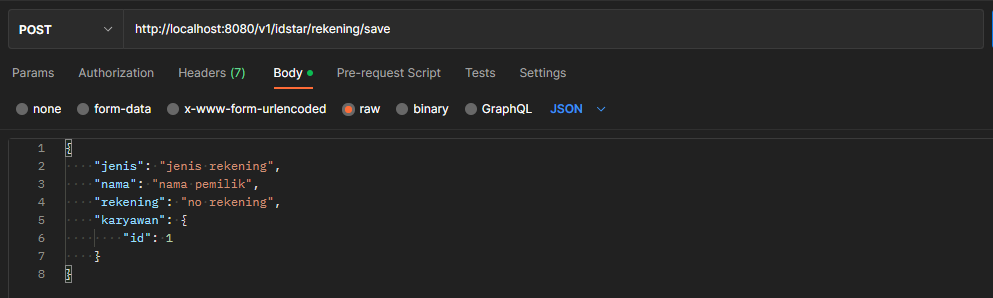
    "karyawan": {

        "id": {{ID karyawan numerik}}

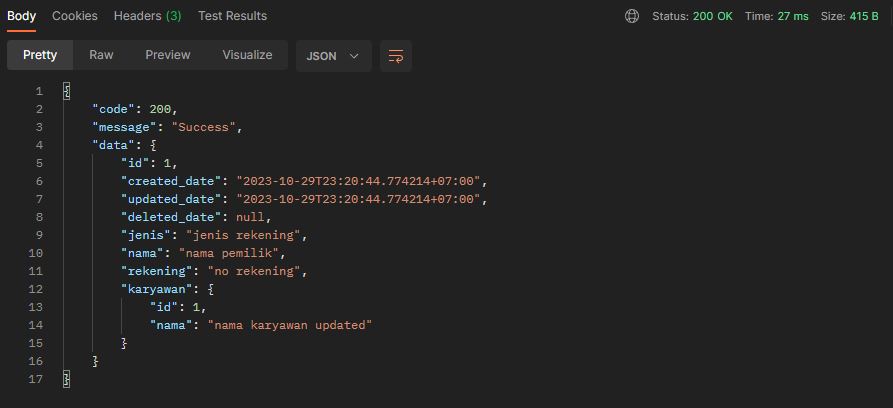
    }

}'

**Sample request:**



**Sample response:**



## Update Rekening

Update record of **rekening** based on request body.

curl -L -X PUT 'http://localhost:8080/v1/idstar/rekening/update' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

    "id": {{ID rekening numerik}},

    "jenis": "{{Update jenis rekening}}",

    "nama": "{{Update pemilik rekening}}",

    "rekening": "{{Update nomor rekening}}",

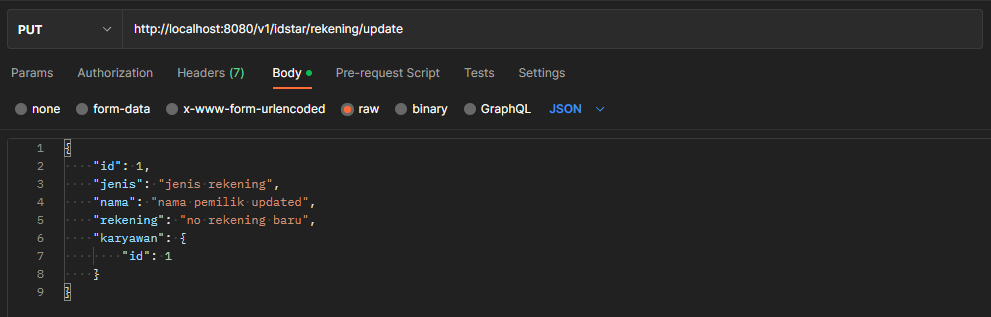
    "karyawan": {

        "id": {{ID karyawan numerik}}

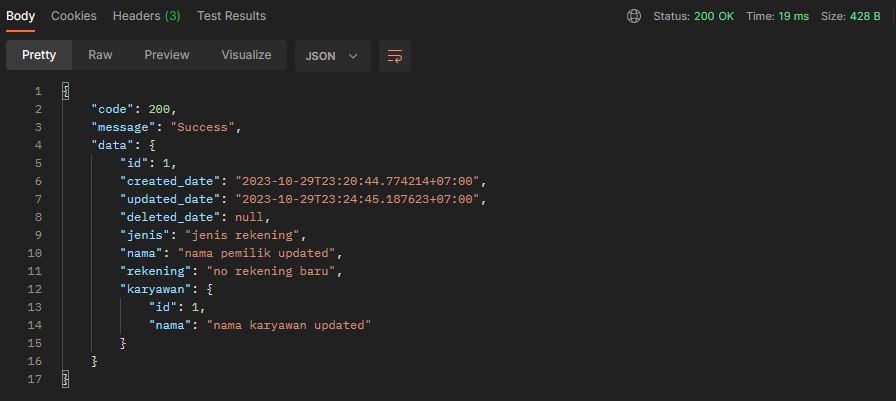
    }

}'

**Sample request:**



**Sample response:**



## Get Rekening List

Get list of **rekening** with paging and sorting.

Paging and sorting specification can be sent through **form-data** or **query parameter**.

curl -L -X GET 'http://localhost:8080/v1/idstar/rekening/list' \

-H 'Authorization: Bearer {{authorization token from login}}' \

-F 'page="{{Nomor paging}}"' \

-F 'size="{{Jumlah record di 1 page}}"' \

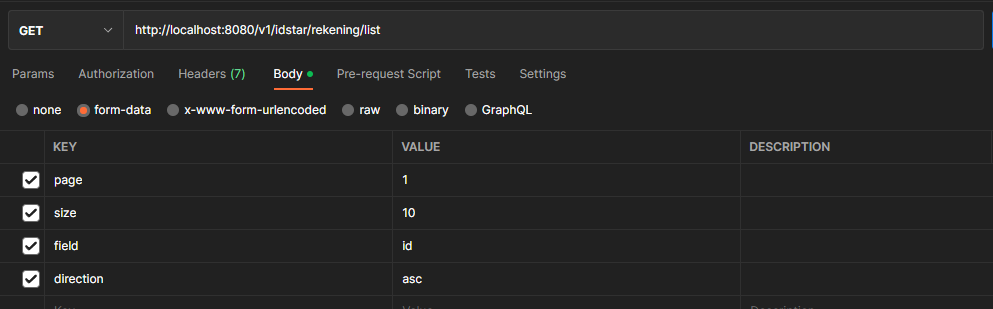
-F 'field="{{Field sorting}}"' \

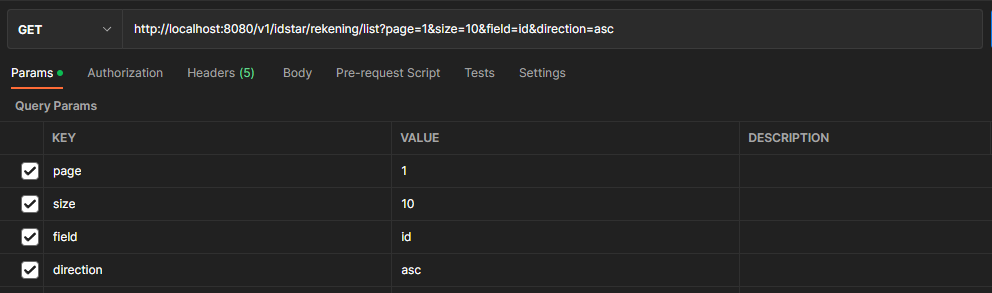
-F 'direction="{{Arah sorting asc/desc}}"'

curl -L -g -X GET 'http://localhost:8080/v1/idstar/rekening/list?page={{Nomor paging}}&size={{Jumlah record di 1 page}}&field={{Field sorting}}&direction={{Arah sorting asc/desc}}' \

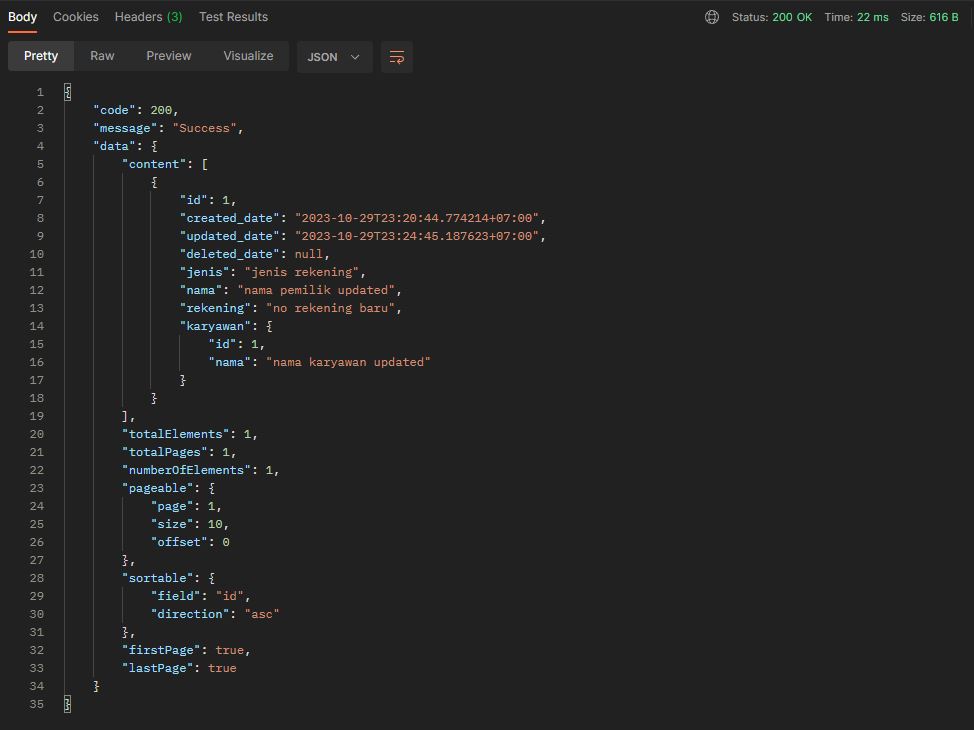
-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**





**Sample response:**



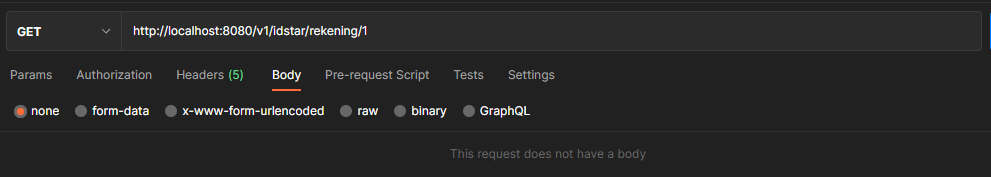
## Get Rekening By ID

Get one **rekening** record by **ID**.

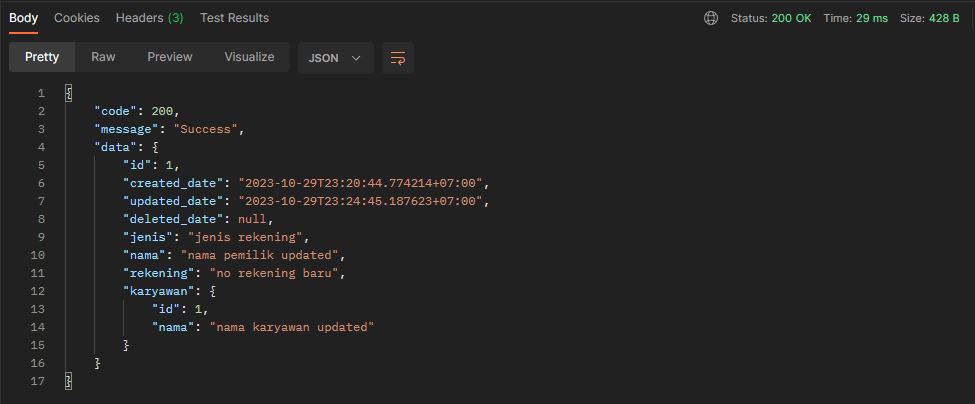
curl -L -g -X GET 'http://localhost:8080/v1/idstar/rekening/{{ID rekening numerik}}' \

-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**



**Sample response:**



## Delete Rekening

Delete one **rekening** record by **ID**.

curl -L -X DELETE 'http://localhost:8080/v1/idstar/rekening/delete' \

-H 'Content-Type: application/json' \

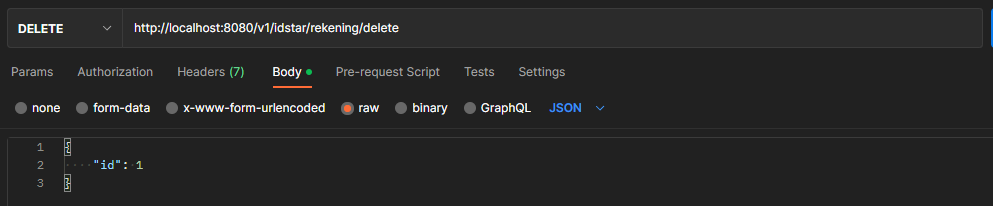
-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

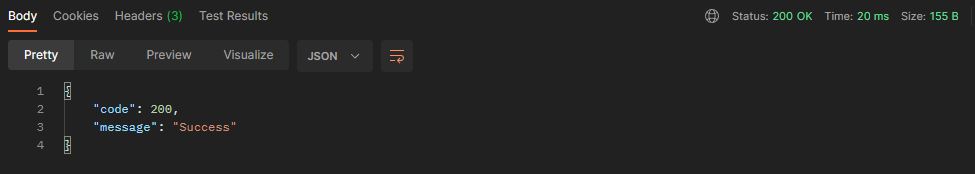
    "id": {{ID rekening numerik}}

}'

**Sample request:**



**Sample response:**



# IDstar - Training Module

**Note:** API under this module requires authorization type **Bearer token**. Token can be generated from **user login**.

## Create Training

Create record of **training** based on request body.

curl -L -X POST 'http://localhost:8080/v1/idstar/training/save' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

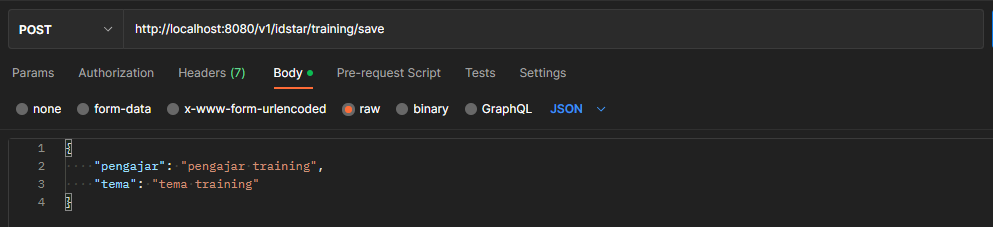
--data-raw '{

    "pengajar": "{{Pengajar training}}",

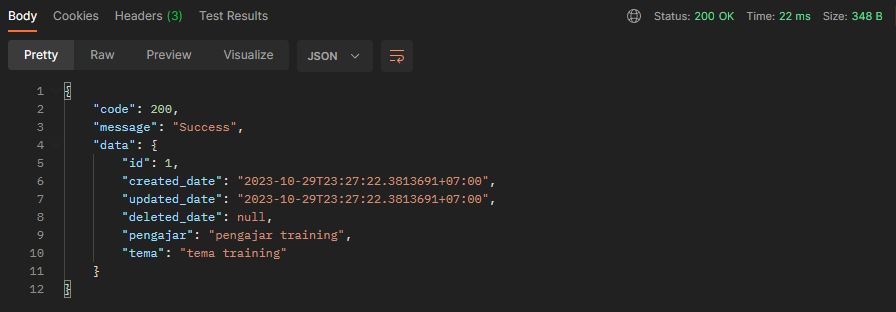
    "tema": "{{Tema training}}"

}'

**Sample request:**



**Sample response:**



## Update Training

Update record of **training** based on request body.

curl -L -X PUT 'http://localhost:8080/v1/idstar/training/update' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

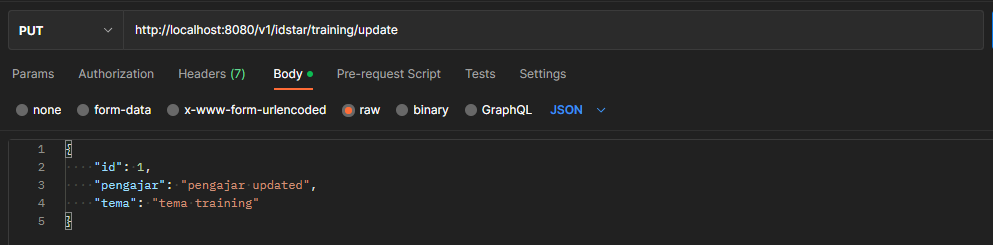
    "id": {{ID training numerik}},

    "pengajar": "{{Update pengajar training}}",

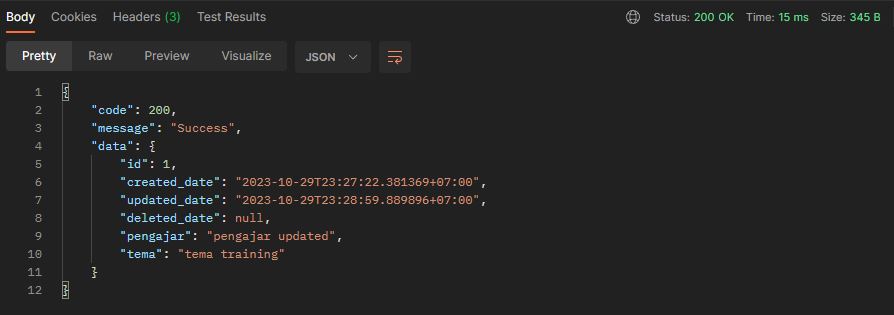
    "tema": "{{Update tema training}}"

}'

**Sample request:**



**Sample response:**



## Get Training List

Get list of **training** with paging and sorting.

Paging and sorting specification can be sent through **form-data** or **query parameter**.

curl -L -X GET 'http://localhost:8080/v1/idstar/training/list' \

-H 'Authorization: Bearer {{authorization token from login}}' \

-F 'page="{{Nomor paging}}"' \

-F 'size="{{Jumlah record di 1 page}}"' \

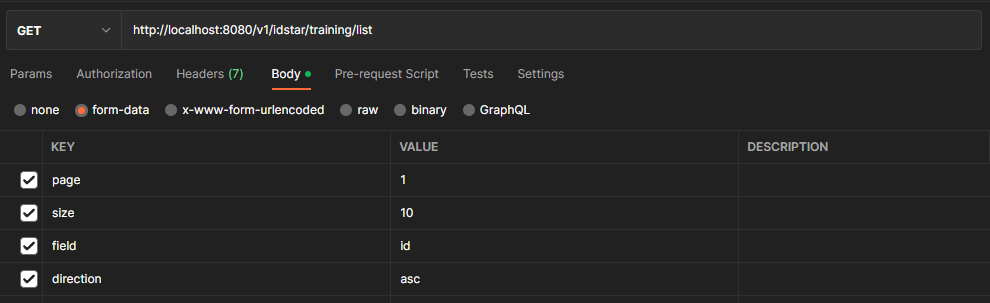
-F 'field="{{Field sorting}}"' \

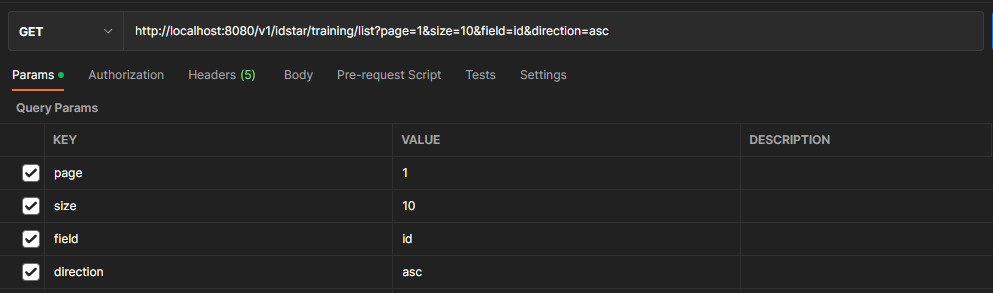
-F 'direction="{{Arah sorting asc/desc}}"'

curl -L -g -X GET 'http://localhost:8080/v1/idstar/training/list?page={{Nomor paging}}&size={{Jumlah record di 1 page}}&field={{Field sorting}}&direction={{Arah sorting asc/desc}}' \

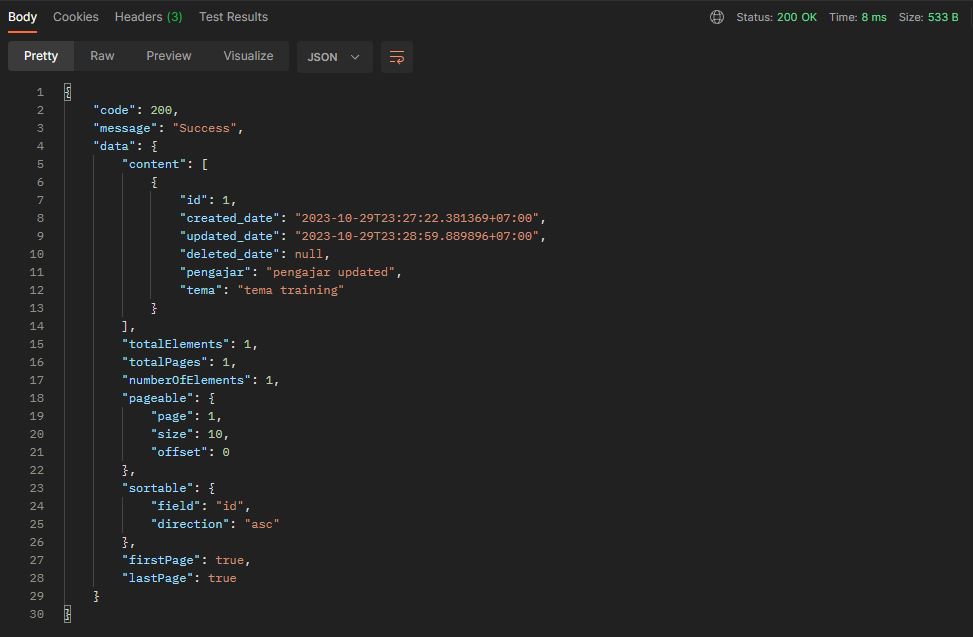
-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**





**Sample response:**



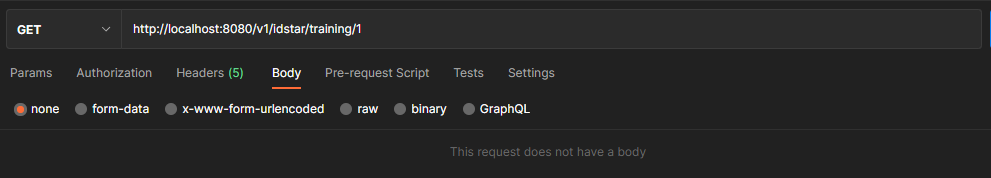
## Get Training By ID

Get one **training** record by **ID**.

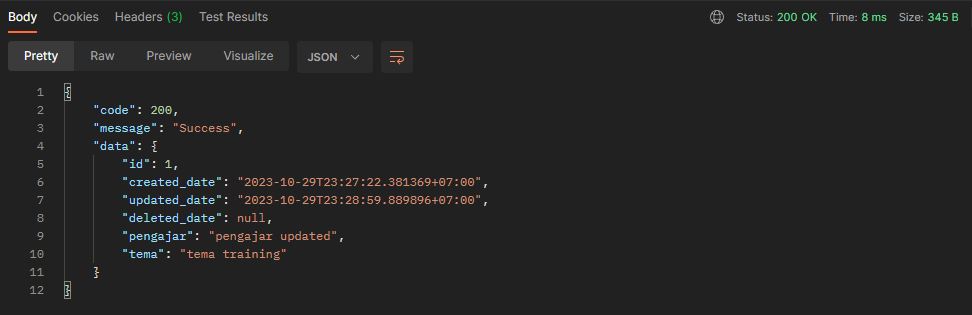
curl -L -g -X GET 'http://localhost:8080/v1/idstar/training/{{ID training numerik}}' \

-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**



**Sample response:**



## Delete Training

Delete one **training** record by **ID**.

curl -L -X DELETE 'http://localhost:8080/v1/idstar/training/delete' \

-H 'Content-Type: application/json' \

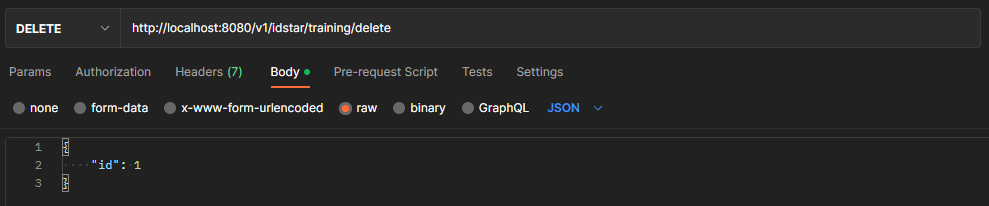
-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

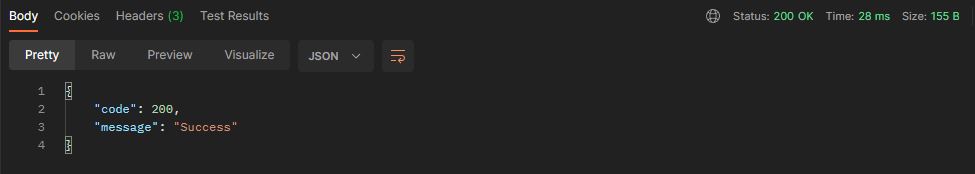
    "id": {{ID training numerik}}

}'

**Sample request:**



**Sample response:**



# IDstar - Karyawan Training Module

**Note:** API under this module requires authorization type **Bearer token**. Token can be generated from **user login**.

## Create Karyawan Training

Create record of **karyawan training** based on request body.

curl -L -X POST 'http://localhost:8080/v1/idstar/karyawan-training/save' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

    "karyawan": {

        "id": {{ID karyawan numerik}}

    },

    "training": {

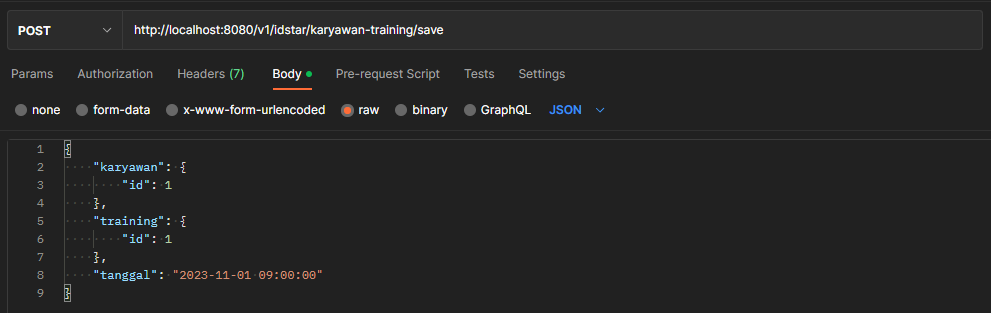
        "id": {{ID training numerik}}

    },

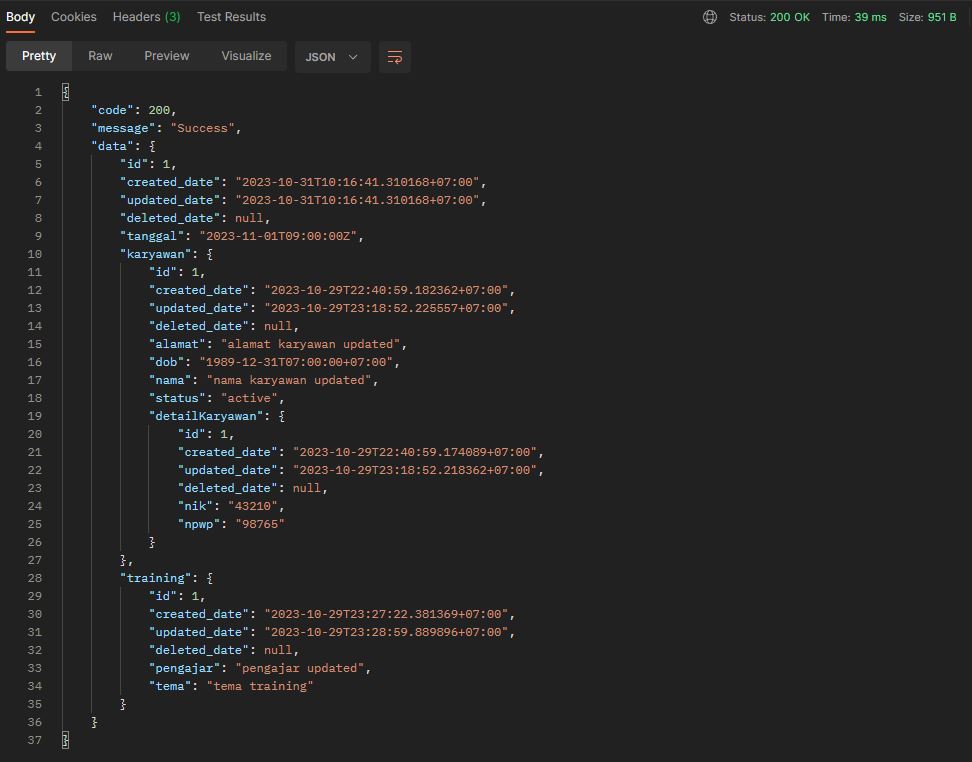
    "tanggal": "{{Tanggal training YYYY-MM-DD HH24:mm:ss}}"

}'

**Sample request:**



**Sample response:**



## Update Karyawan Training

Update record of **karyawan training** based on request body.

curl -L -X PUT 'http://localhost:8080/v1/idstar/karyawan-training/update' \

-H 'Content-Type: application/json' \

-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

    "id": {{ID karyawan-training numerik}},

    "karyawan": {

        "id": {{ID karyawan numerik}}

    },

    "training": {

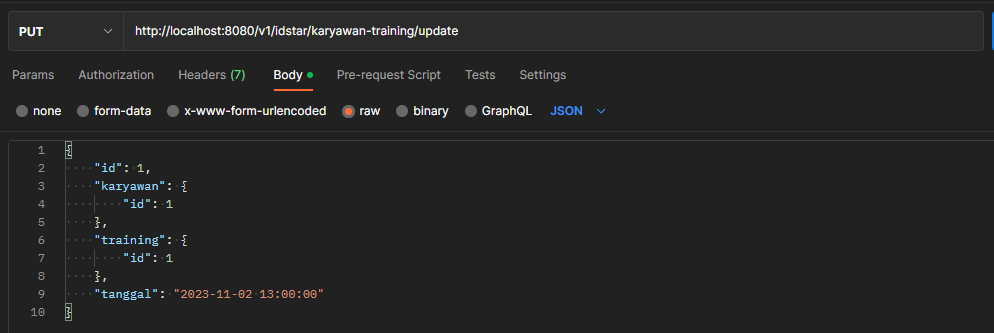
        "id": {{ID training numerik}}

    },

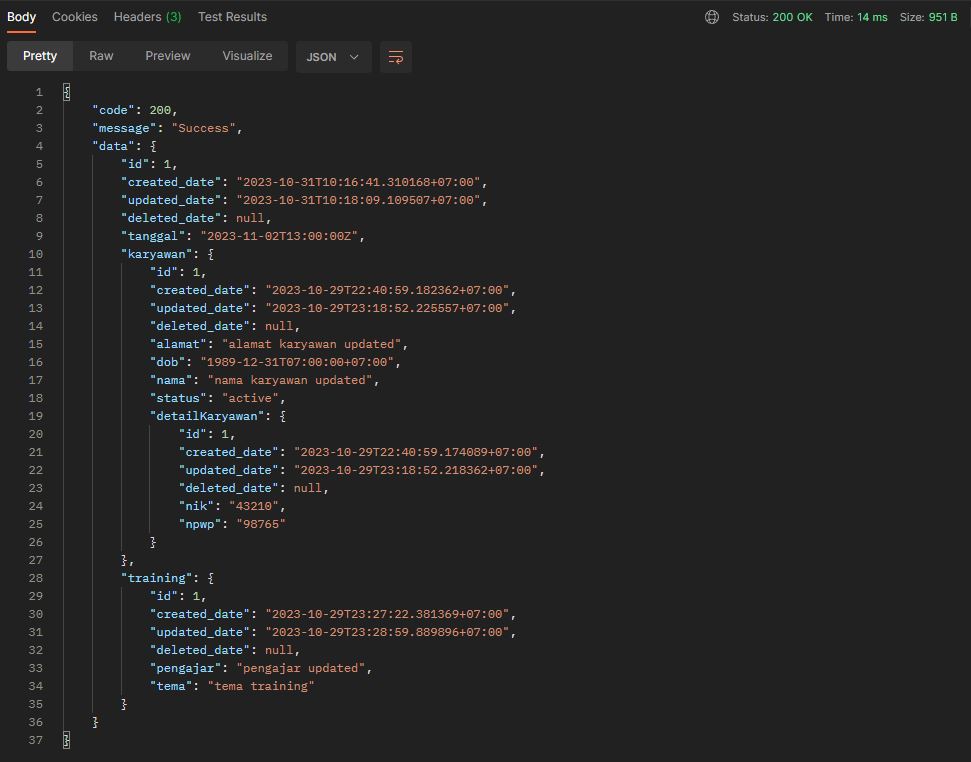
    "tanggal": "{{Update tanggal training YYYY-MM-DD HH24:mm:ss}}"

}'

**Sample request:**



**Sample response:**



## Get Karyawan Training List

Get list of **karyawan training** with paging and sorting.

Paging and sorting specification can be sent through **form-data** or **query parameter**.

curl -L -X GET 'http://localhost:8080/v1/idstar/karyawan-training/list' \

-H 'Authorization: Bearer {{authorization token from login}}' \

-F 'page="{{Nomor paging}}"' \

-F 'size="{{Jumlah record di 1 page}}"' \

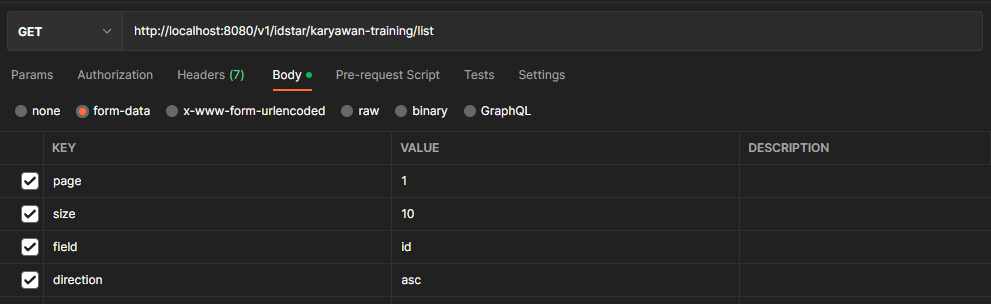
-F 'field="{{Field sorting}}"' \

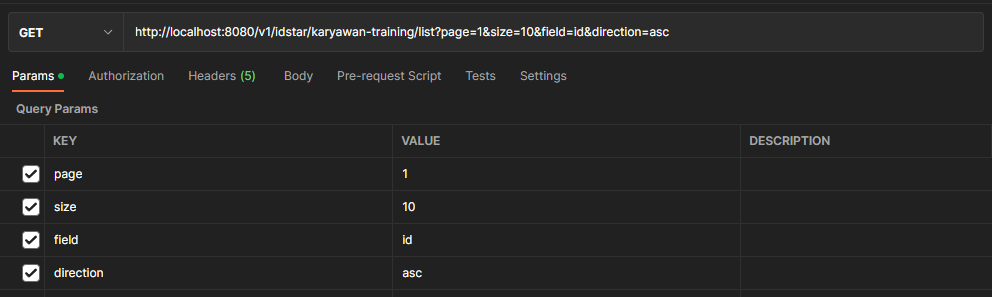
-F 'direction="{{Arah sorting asc/desc}}"'

curl -L -g -X GET 'http://localhost:8080/v1/idstar/karyawan-training/list?page={{Nomor paging}}&size={{Jumlah record di 1 page}}&field={{Field sorting}}&direction={{Arah sorting asc/desc}}' \

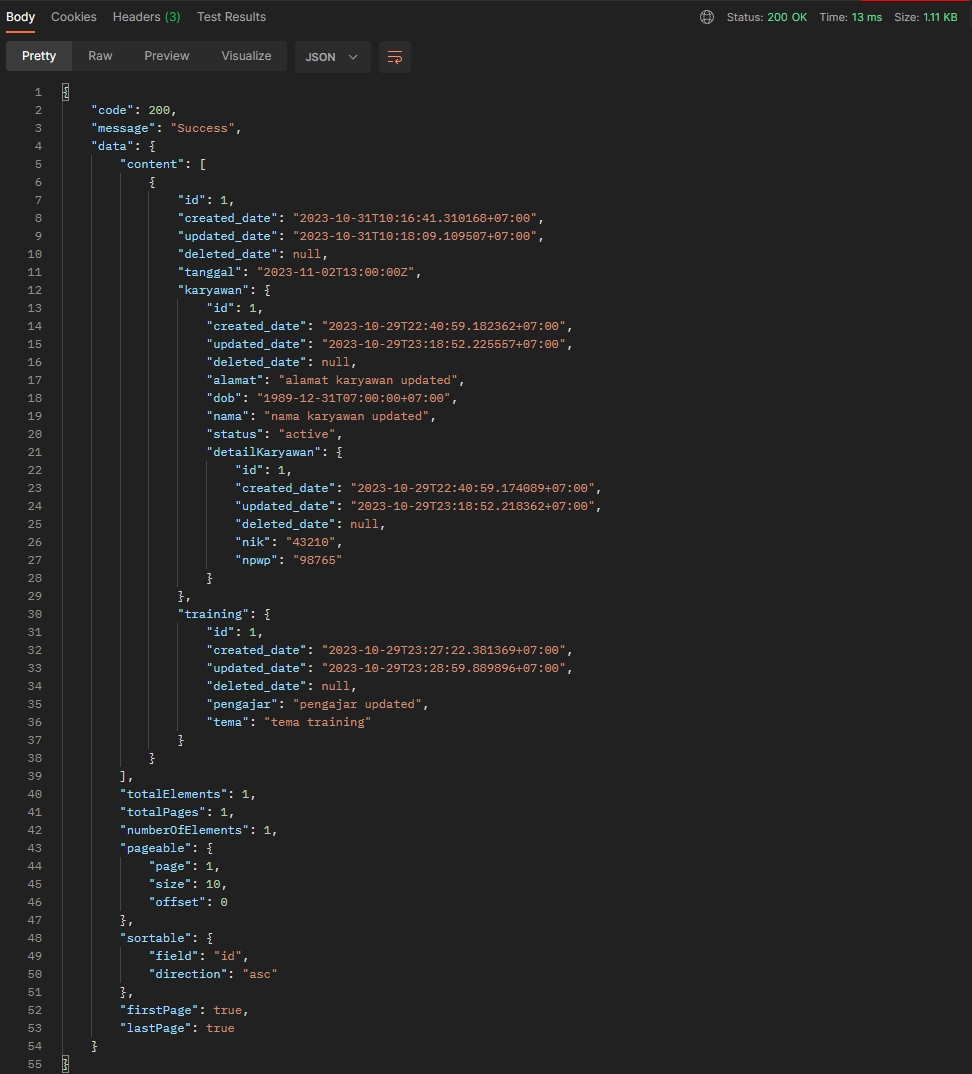
-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**





**Sample response:**



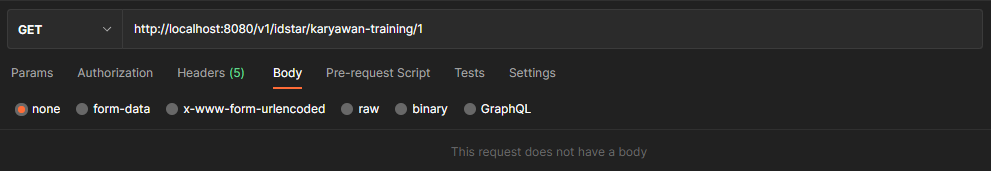
## Get Karyawan Training By ID

Get one **karyawan training** record by **ID**.

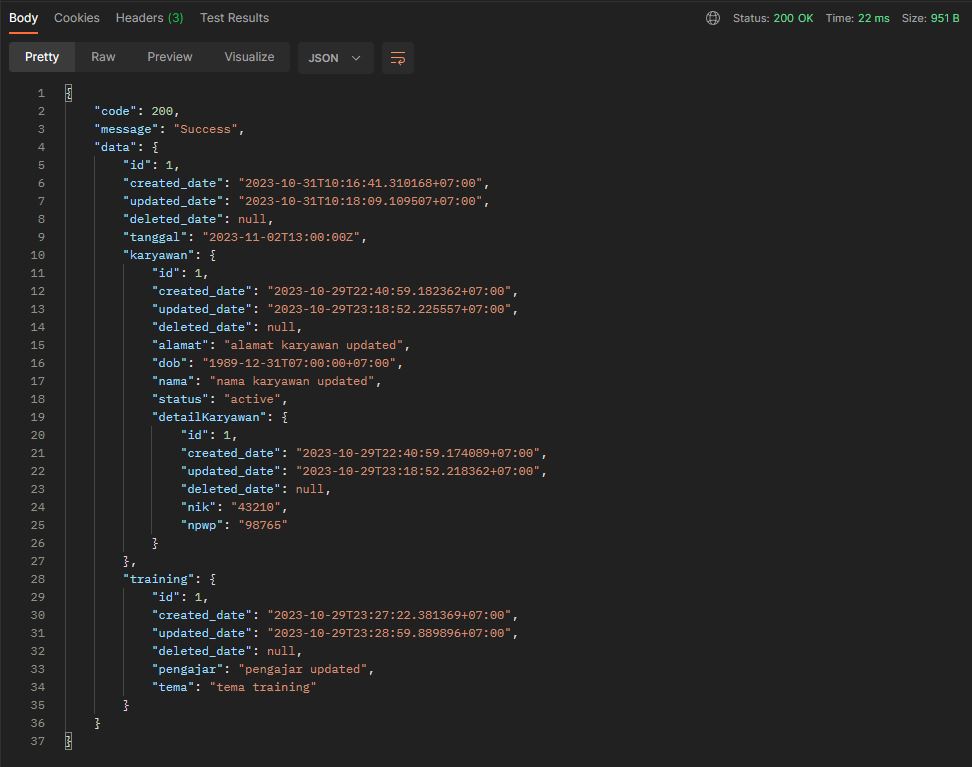
curl -L -g -X GET 'http://localhost:8080/v1/idstar/karyawan-training/{{ID karyawan-training numerik}}' \

-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**



**Sample response:**



## Delete Karyawan Training

Delete one **karyawan training** record by **ID**.

curl -L -X DELETE 'http://localhost:8080/v1/idstar/karyawan-training/delete' \

-H 'Content-Type: application/json' \

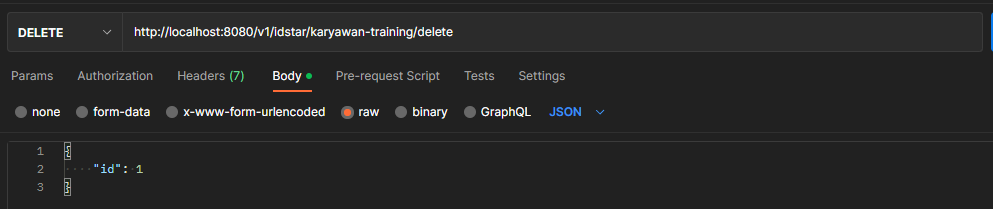
-H 'Authorization: Bearer {{authorization token from login}}' \

--data-raw '{

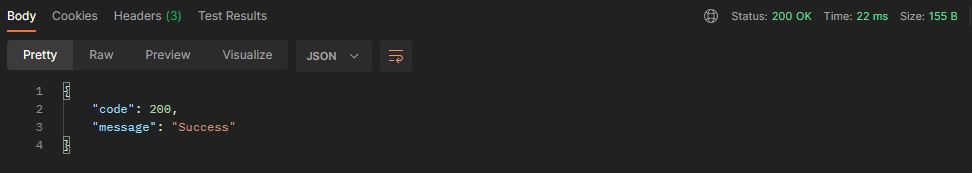
    "id": {{ID karyawan-training numerik}}

}'

**Sample request:**



**Sample response:**



# File Module

**Note:** API under this module requires authorization type **Bearer token**. Token can be generated from **user login**.

## Upload File

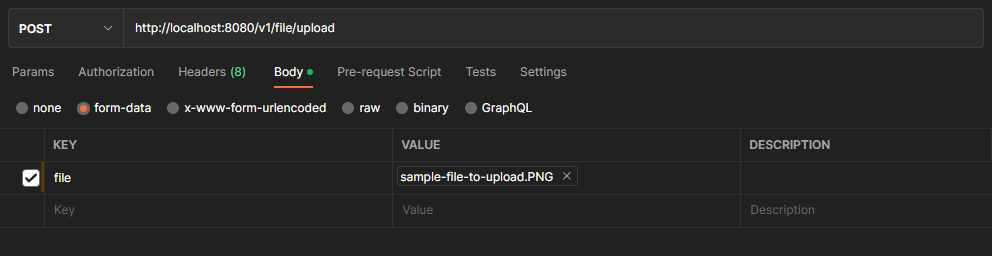
Upload a **file** from local storage to server and return the **generated name** and **download URI** for the file.

curl -L -X POST 'http://localhost:8080/v1/file/upload' \

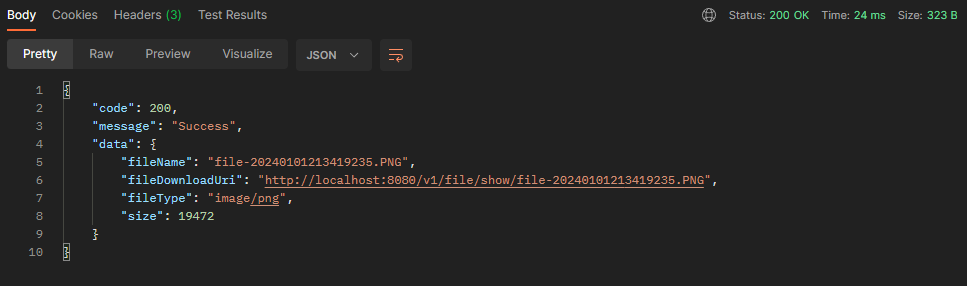
-H 'Authorization: Bearer {{authorization token from login}}' \

-F 'file=@"/path/to/file"'

**Sample request:**



**Sample response:**



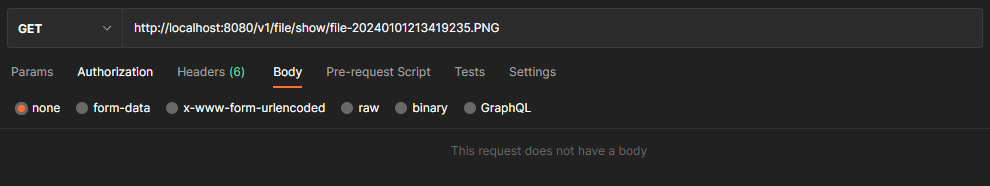
## Show File

Get file that has been uploaded by **generated name**.

curl -L -g -X GET 'http://localhost:8080/v1/file/show/{{generated filename}}' \

-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**

****

**Sample response:**



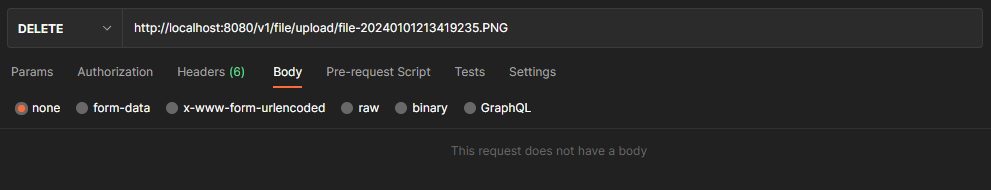
## Delete File

Delete file that has been uploaded by **generated name**.

curl -L -g -X DELETE 'http://localhost:8080/v1/file/delete/{{generated filename}}' \

-H 'Authorization: Bearer {{authorization token from login}}'

**Sample request:**



**Sample response:**

