**LearnAPI Documentation**

This REST API created only for exhibiting FastAPI ability of doing a CRUD operation combined with mysql-connector-python package. FastAPI is an open-source Framework for creating REST API using python scripting language. It is claimed to be the fastest and easiest framework. This documentation is created as a guidance for the user who wanted to use the REST API know what kind of request that need to be sent, what response the API will return, what the API does to the System, and what causing API to return an Error code.

This API only consisted by only one file ‘main.py’ that is used by uvicorn command to run it as development server and is only available locally with this URL ‘<http://localhost:8000/>’. To use it globally in development scenario, it need to be forwarded by ngrok packages. The URL ngrok use is quite complicated and not suited for release scenario.

Database is consisted of four column id(inaccessible | PK), name, username, password.

**1. Insert to database operation**

To do insert operation, send the request to ‘<http://localhost:8000/insert/>’ with POST methods with json as body parameters. Example:

Result:

import requests

response = request.post( # on the left before parenthesis is the method that is used by requests

‘http://localhost:8000/insert/’,

json={ # this is where the body parameters that will be sent goes

“name”: “string name”,

“username”: “string username”,

“password”:”12345”

})

print(response.text) # print the response returned by REST API

print(response.status\_code)

If the value requested is invalid it will return:

{'message': 'added user row with string name|string username', 'status':'oke'}

200

{ "detail": [{"loc": ["string",0],"msg": "string","type": "string"}]}

422

**2. Update to database operation**

To update specified column with specified id from database. The column consisted of three available options: name, username, and password. Send request to ‘<http://localhost:8000/update/>{column}/{id}’ with PUT method and specify the new value with body parameter. Example:

URL:

url = ‘[**http://localhost:8000/update/name/1/**](http://localhost:8000/update/name/1/)’

Body Param:

send the request:

body\_param = {

“value”: “new name”

}

Result:

import requests

response = requests.put(url, json=body\_param)

print(response.status\_code, response.text, sep=” | ”)

200 | {"message":"successfully update name of table with ID: 15 to test"}

**3. Delete from database operation**

To delete a row with the selected id from database. Send request to ‘<http://localhost:8000/delete/>{id}/’ with DELETE methods, where the id is the targeted row id. Example:

send request:

import requests

response = requests.delete(‘http://localhost:8000/delete/1/’)

print(response.text, response.status\_code, sep=” | ”)

result:

200 | table with ID: 1 have been deleted

**4. Select all row from database**

To select all the row in the database omitting the password column. Send request to ‘<http://localhost:8000/select/>’ with GET method. Example:

result:

import requests

response = requests.get(‘http://localhost:8000/select/’)

print(response.text, response.status\_code, sep=” | ”)

200 | [[id, name, username], ...]