

Cafe & Wifi

This is an API consisting of a few cafes and some information about them. You can look up any cafe and decide whether it is a good place to work based on the information provided.

GET Search Cafes By Location

http://127.0.0.1:5000/search?loc=Peckham

The /search route will search the cafe database for a cafe that matches the location queried. Use the loc parameter to pass a location name.

PARAMS

loc	Peckham
-----	---------

Example Request

Search Cafes By Location

```
import requests

url = "http://127.0.0.1:5000/search?loc=Peckham"

payload={}
headers = {}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.text)
```

View More

GET Get a Random Cafe

http://127.0.0.1:5000/random

The /random route will search the cafe daatabase for a random selected cafe.

Example Request

Get a Random Cafe

```
import requests

url = "http://127.0.0.1:5000/random"

payload={}
headers = {}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.text)
```

View More

GET Get All Cafes

http://127.0.0.1:5000/all

The /all route will search the cafe database and return all cafes that exist in the database.

Example Request

Get All Cafes

```
import requests

url = "http://127.0.0.1:5000/all"

payload={}
headers = {}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.text)
```

View More

POST Add a New Cafe

http://127.0.0.1:5000/add

The /add route is used to add a new cafe to the database.

BODY urlencoded

name	The V&A Cafe
map_url	https://goo.gl/maps/QABQjZsexWAd98HN9
img_url	https://www.google.pl/maps/place/The+V%26A+Cafe/@51.4969754,-0.1727111,3a,75y,90t/data=!3m8!1e2!3m6!1sAF1QipPs-cZ-t_VpHLqZePFtpg4km5qFg3YGdBO_XBmR!2e10!3e12!6shhttps:%2F%2FIh5.googleusercontent.com%2Fp%2FAF1QipPs-cZ-t_VpHLqZePFtpg4km5qFg3YGdBO_XBmR%3Dw203-h151-k-no!7i9280!8i6944!4m9!1m2!2m1!1slondon+cafe!3m5!1s0x4876055e12958739:0xc2fa7d3d21a20b9c!8m2!3d51.4973733!4d-0.1724352!15sCgtsb25kb24gY2FmZVoNIgtsb25kb24gY2FmZZIBBGNhZmXgAQA#
location	Kensington
seats	30+
has_toilet	True
has_wifi	Fals
has_sockets	True
can_take_calls	False
coffee_price	\$3.40

Example Request

Add a New Cafe

```
import requests

url = "http://127.0.0.1:5000/add"

payload='name=The%20V%26A%20Cafe&map_url=https%3A%2F%2Fgoo.gl%2Fmaps%2FQABQjZsexWAd98HN9&i
headers = {}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

View More

PATCH Update Coffee Price For a Cafe

http://127.0.0.1:5000/update/25?new_coffee_price=\$3.75

This uses the /update/ route, to search for the given cefe in the databse and updates its coffee price with the data new_coffee_price send with the request.

PARAMS

new_coffee_price	\$3.75
------------------	--------

Example Request

Update Coffee Price For a Cafe

```
import requests

url = "http://127.0.0.1:5000/update/25?new_coffee_price=$3.75"

payload={}
headers = {}

response = requests.request("PATCH", url, headers=headers, data=payload)

print(response.text)
```

View More

DEL Delete a Cafe by Id

http://127.0.0.1:5000/report-closed/23?api-key=TopSecretAPIKey

This uses /report-closed/ route to find in the database the requested id = cafe_id and deletes this entry in the database if a correct api-key is sent with the request

PARAMS

api-key	TopSecretAPIKey
---------	-----------------

Example Request

Delete a Cafe by Id

```
import requests

url = "http://127.0.0.1:5000/report-closed/23?api-key=TopSecretAPIKey"

payload={}
headers = {}

response = requests.request("DELETE", url, headers=headers, data=payload)

print(response.text)
```

View More