!python3 -m pip install requests

Requirement already satisfied: requests in /Users/in7erval/opt/anaconda3/lib/python3.8/site-packages (2.24.0)

Requirement already satisfied: certifi>=2017.4.17 in /Users/in7erval/opt/anaconda3/lib/python3.8/site-packages (from requests) (2020.6.20)

Requirement already satisfied: idna<3,>=2.5 in /Users/in7erval/opt/anaconda3/lib/python3.8/site-packages (from requests) (2.10)

Requirement already satisfied: chardet<4,>=3.0.2 in /Users/in7erval/opt/anaconda3/lib/python3.8/site-packages (from requests) (3.0.4)

Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /Users/in7erval/opt/anaconda3/lib/python3.8/site-packages (from requests) (1.25.9)

In [2]:

import requests

In [3]:

requests.get("https://randomuser.me/api/")

Out[3]:

<Response [200]>

In [4]:

response = requests.get("https://randomuser.me/api/")

response.text

Out[4]:

'{"results":[{"gender":"male","name":{"title":"Mr","first":"Lucca","last":"Steunebrink"},"location":{"street":{"number":1077,"name":"Kruswei"},"city":"Camperduin","state":"Flevoland","country":"Netherlands","postcode":22166,"coordinates":{"latitude":"46.8205","longitude":"-148.5897"},"timezone":{"offset":"+3:30","description":"Tehran"}},"email":"lucca.steunebrink@example.com","login":{"uuid":"8a91f222-c74e-45b3-8aad-d86e93abedfc","username":"purplesnake867","password":"nellie","salt":"WCr4uRpq","md5":"8804faf534ac472c02f5ebd59fbd5e1b","sha1":"f259f265ef8ea856223d02f7b34cd74d42ddde6d","sha256":"41cc8065b15f22676b94dee5b467d73c9863887b6bb7b919fdb3cc573568cf0d"},"dob":{"date":"1971-11-01T04:09:45.580Z","age":51},"registered":{"date":"2009-07-14T20:13:57.146Z","age":13},"phone":"(526)-798-9754","cell":"(396)-008-4102","id":{"name":"BSN","value":"04744916"},"picture":{"large":"https://randomuser.me/api/portraits/men/89.jpg","medium":"https://randomuser.me/api/portraits/med/men/89.jpg","thumbnail":"https://randomuser.me/api/portraits/thumb/men/89.jpg"},"nat":"NL"}],"info":{"seed":"781485a7762b5c61","results":1,"page":1,"version":"1.3"}}'

In [8]:

response = requests.get("https://api.thedogapi.com/")

response.text

Out[8]:

'{"message":"The Dog API"}'

In [9]:

response = requests.get("https://api.thedogapi.com/v1/breeds")

response.text[:100] # чтобы не засорять вывод

Out[9]:

'[{"weight":{"imperial":"6 - 13","metric":"3 - 6"},'

In [12]:

response = requests.get("https://api.thecatapi.com/v1/breeds")

response.text[:100] # чтобы не засорять вывод

Out[12]:

'[{"weight":{"imperial":"7 - 10","metric":"3 - 5"},"id":"abys","name":"Abyssinian","cfa\_url":"http:'

In [13]:

response = requests.get("https://api.thedogapi.com/v1/breeds")

response

Out[13]:

<Response [200]>

In [14]:

request = response.request

request

Out[14]:

<PreparedRequest [GET]>

In [15]:

request.url

Out[15]:

'https://api.thedogapi.com/v1/breeds'

In [16]:

request.path\_url

Out[16]:

'/v1/breeds'

In [17]:

request.method

Out[17]:

'GET'

In [18]:

request.headers

Out[18]:

{'User-Agent': 'python-requests/2.24.0', 'Accept-Encoding': 'gzip, deflate', 'Accept': '\*/\*', 'Connection': 'keep-alive'}

In [19]:

response

Out[19]:

<Response [200]>

In [20]:

response.text[:100] # чтобы не засорять вывод

Out[20]:

'[{"weight":{"imperial":"6 - 13","metric":"3 - 6"},"height":{"imperial":"9 - 11.5","metric":"23 - 29"'

In [22]:

response.status\_code

Out[22]:

200

In [23]:

response.headers

Out[23]:

{'x-dns-prefetch-control': 'off', 'x-frame-options': 'SAMEORIGIN', 'strict-transport-security': 'max-age=15552000; includeSubDomains', 'x-download-options': 'noopen', 'x-content-type-options': 'nosniff', 'x-xss-protection': '1; mode=block', 'vary': 'Origin', 'pagination-count': '172', 'pagination-page': '0', 'pagination-limit': '1000', 'access-control-expose-headers': 'Pagination-Count, Pagination-Page, Pagination-Limit', 'content-type': 'application/json; charset=utf-8', 'x-response-time': '7ms', 'X-Cloud-Trace-Context': '411e43544cef3356a069c9a354967541', 'Date': 'Wed, 20 Apr 2022 18:32:06 GMT', 'Server': 'Google Frontend', 'Content-Length': '76246'}

In [24]:

response =

Владислав Фролов, сегодня в 23:29

requests.get("https://api.thedogapi.com/v1/breeds")

response

Out[24]:

<Response [200]>

In [25]:

response.status\_code

Out[25]:

200

In [26]:

response.reason

Out[26]:

'OK'

In [27]:

response = requests.get("https://api.thedogapi.com/v1/breedz")

response

Out[27]:

<Response [404]>

In [28]:

response.status\_code

Out[28]:

404

In [29]:

response.reason

Out[29]:

'Not Found'

In [30]:

response = requests.get("https://api.thedogapi.com/v1/breeds/1")

response.headers

Out[30]:

{'x-dns-prefetch-control': 'off', 'x-frame-options': 'SAMEORIGIN', 'strict-transport-security': 'max-age=15552000; includeSubDomains', 'x-download-options': 'noopen', 'x-content-type-options': 'nosniff', 'x-xss-protection': '1; mode=block', 'vary': 'Origin', 'content-type': 'application/json; charset=utf-8', 'x-response-time': '2ms', 'X-Cloud-Trace-Context': '51ec7d3331702d09c8af76e71d6928e4;o=1', 'Date': 'Wed, 20 Apr 2022 18:35:24 GMT', 'Server': 'Google Frontend', 'Content-Length': '357'}

In [32]:

headers = {"X-Request-Id": "<my-request-id>"}

response = requests.get("https://example.org", headers=headers)

response.request.headers

Out[32]:

{'User-Agent': 'python-requests/2.24.0', 'Accept-Encoding': 'gzip, deflate', 'Accept': '\*/\*', 'Connection': 'keep-alive', 'X-Request-Id': '<my-request-id>'}

In [33]:

response = requests.get("https://api.thedogapi.com/v1/breeds/1")

response.headers.get("Content-Type")

Out[33]:

'application/json; charset=utf-8'

In [34]:

response = requests.get("https://api.thedogapi.com/v1/breeds/1")

response.headers.get("Content-Type")

Out[34]:

'application/json; charset=utf-8'

In [35]:

response.json()

Out[35]:

{'weight': {'imperial': '6 - 13', 'metric': '3 - 6'},

'height': {'imperial': '9 - 11.5', 'metric': '23 - 29'},

'id': 1,

'name': 'Affenpinscher',

'bred\_for': 'Small rodent hunting, lapdog',

'breed\_group': 'Toy',

'life\_span': '10 - 12 years',

'temperament': 'Stubborn, Curious, Playful, Adventurous, Active, Fun-loving',

'origin': 'Germany, France',

'reference\_image\_id': 'BJa4kxc4X'}

In [36]:

response.json()["name"]

Out[36]:

'Affenpinscher'

In [37]:

requests.post("https://api.thedogapi.com/v1/breeds/1")

Out[37]:

<Response [405]>

In [38]:

requests.get("https://api.thedogapi.com/v1/breeds/1")

Out[38]:

<Response [200]>

In [39]:

requests.put("https://api.thedogapi.com/v1/breeds/1")

Out[39]:

<Response [405]>

In [40]:

requests.delete("https://api.thedogapi.com/v1/breeds/1")

Out[40]:

<Response [405]>

In [41]:

requests.get("https://randomuser.me/api/").json()

Out[41]:

{'results': [{'gender': 'female',

'name': {'title': 'Mrs', 'first': 'Lily', 'last': 'Wang'},

'location': {'street': {'number': 6076, 'name': 'Highgate'},

'city': 'Nelson',

'state': 'Taranaki',

'country': 'New Zealand',

'postcode': 85864,

'coordinates': {'latitude': '59.5393', 'longitude': '90.0372'},

'timezone': {'offset': '-8:00',

'description': 'Pacific Time (US & Canada)'}},

'email': 'lily.wang@example.com',

'login': {'uuid': '9d92968b-df5e-4c0d-a513-14ee051768ab',

'username': 'bluebutterfly341',

'password': 'barron',

'salt': '8t4XcZkv',

'md5': '99ec15fc75db854d3a3fbbaded5486a0',

'sha1': 'eff1d3723d72387a9f0ccbe79b6c4286750b5892',

'sha256': '1cb5bcceb1bbea333d7f0e6ec497a8d6890ff54ced8d0fcf4a642d41d4c63001'},

'dob': {'date': '1993-01-15T12:02:56.720Z', 'age': 29},

'registered': {'date': '2018-01-19T20:45:52.604Z', 'age': 4},

'phone': '(314)-456-6368',

'cell': '(854)-685-3922',

'id': {'name': '', 'value': None},

'picture': {'large': 'https://randomuser.me/api/portraits/women/2.jpg',

'medium': 'https://randomuser.me/api/portraits/med/women/2.jpg',

'thumbnail': 'https://randomuser.me/api/portraits/thumb/women/2.jpg'},

'nat': 'NZ'}],

'info': {'seed': '9539170ec23d6cb7',

'results': 1,

'page': 1,

'version': '1.3'}}

In [42]:

requests.get("https://randomuser.me/api/?gender=female&nat=de").json()

Out[42]:

{'results': [{'gender': 'female',

'name': {'title': 'Mrs', 'first': 'Emilie', 'last': 'Flach'},

'location': {'street': {'number': 7401, 'name': 'Ahornweg'},

'city': 'Siegen',

'state': 'Baden-Württemberg',

'country': 'Germany',

'postcode': 37162,

'coordinates': {'latitude': '-68.6495', 'longitude': '-81.2650'},

'timezone': {'offset': '-12:00', 'description': 'Eniwetok, Kwajalein'}},

'email': 'emilie.flach@example.com',

'login': {'uuid': '390b0c04-ecfe-4c73-8861-47a9c917531f',

'username': 'happybutterfly357',

'password': 'loser',

'salt': '4U85WElb',

'md5': '1d214d4315c03601dab0ee1bdb00e9ac',

'sha1': '2d11706c324e1f30a4b04dc730c3a15ab4a21ffd',

'sha256': 'c8cdbde97cb96219cdbd5dfdd718d2bbe2e45b0ec7e235a9d553f49eab0e31e4'},

'dob': {'date': '1971-03-03T14:33:22.345Z', 'age': 51},

'registered': {'date': '2013-09-06T09:26:46.815Z', 'age': 9},

'phone': '0850-4321974',

'cell': '0170-6856846',

'id': {'name': '', 'value': None},

'picture': {'large': 'https://randomuser.me/api/portraits/women/57.jpg',

'medium': 'https://randomuser.me/api/portraits/med/women/57.jpg',

'thumbnail': 'https://randomuser.me/api/portraits/thumb/women/57.jpg'},

'nat': 'DE'}],

'info': {'seed': '6175040b2ea5ba56',

'results': 1,

'page': 1,

'version': '1.3'}}

In [43]:

query\_params = {"gender": "female", "nat": "de"}

requests.get("https://randomuser.me/api/", params=query\_params).json()

Out[43]:

{'results': [{'gender': 'female',

'name': {'title': 'Miss', 'first': 'Rosita', 'last': 'Eismann'},

'location': {'street': {'number': 1547, 'name': 'Jahnstraße'},

'city': 'Oederan',

'state': 'Sachsen',

'country': 'Germany',

'postcode': 13763,

'coordinates': {'latitude': '-35.1729', 'longitude': '119.2738'},

'timezone': {'offset': '+6:00', 'description': 'Almaty, Dhaka, Colombo'}},

'email': 'rosita.eismann@example.com',

'login': {'uuid': '20c27ac0-cf4d-4927-86bc-495f7b6324c7',

'username': 'sadsnake333',

'password': 'gizmo1',

'salt': 'Y0r2LjDY',

'md5': '9e57fba5a0eeb59089706a3e9c8e52c2',

'sha1': '55d31e5abc9fa3ef54a168f5f64009b3451260b8',

'sha256': 'a31bd7a6497deae4995d6e04fd7fb9be191d9c8e9363084d4dcac7aa86251709'},

'dob': {'date': '1977-09-06T14:18:56.531Z', 'age': 45},

'registered': {'date': '2008-09-02T23:38:23.845Z', 'age': 14},

'phone': '0473-2127469',

'cell': '0179-7928973',

'id': {'name': '', 'value': None},

'picture': {'large': 'https://randomuser.me/api/portraits/women/94.jpg',

'medium': 'https://randomuser.me/api/portraits/med/women/94.jpg',

'thumbnail': 'https://randomuser.me/api/portraits/thumb/women/94.jpg'},

'nat': 'DE'}],

'info': {'seed': '81c27a811fe08d35',

'results': 1,

'page': 1,

'version': '1.3'}}

In [44]:

query\_params = {"q": "labradoodle"}

endpoint = "https://api.thedogapi.com/v1/breeds/search"

requests.get(endpoint, params=query\_params).json()

Out[44]:

[{'weight': {'imperial': '45 - 100', 'metric': '20 - 45'},

'height': {'imperial': '14 - 24', 'metric': '36 - 61'},

'id': 148,

'name': 'Labradoodle',

'breed\_group': 'Mixed',

'life\_span': '10 - 15 years'}]

In [46]:

endpoint ="https://api.nasa.gov/mars-photos/api/v1/rovers/curiosity/photos"

api\_key = "DEMOOOOOO\_KEYYY" # use https://api.nasa.gov/

query\_params = {"api\_key": api\_key, "earth\_date": '2020-07-01'}

response = requests.get(endpoint, params=query\_params)

response

Out[46]:

<Response [200]>

In [51]:

response.json()

Out[51]:

{'photos': [{'id': 754118,

'sol': 2809,

'camera': {'id': 20,

'name': 'FHAZ',

'rover\_id': 5,

'full\_name': 'Front Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/fcam/FLB\_646868981EDR\_F0810628FHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754119,

'sol': 2809,

'camera': {'id': 20,

'name': 'FHAZ',

'rover\_id': 5,

'full\_name': 'Front Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/fcam/FRB\_646868981EDR\_F0810628FHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754120,

'sol': 2809,

'camera': {'id': 20,

'name': 'FHAZ',

'rover\_id': 5,

'full\_name': 'Front

Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/fcam/FRB\_646860144EDR\_F0810628FHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754121,

'sol': 2809,

'camera': {'id': 20,

'name': 'FHAZ',

'rover\_id': 5,

'full\_name': 'Front Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/fcam/FLB\_646860144EDR\_F0810628FHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754122,

'sol': 2809,

'camera': {'id': 21,

'name': 'RHAZ',

'rover\_id': 5,

'full\_name': 'Rear Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/rcam/RRB\_646869036EDR\_F0810628RHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754123,

'sol': 2809,

'camera': {'id': 21,

'name': 'RHAZ',

'rover\_id': 5,

'full\_name': 'Rear Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/rcam/RLB\_646869036EDR\_F0810628RHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754124,

'sol': 2809,

'camera': {'id': 21,

'name': 'RHAZ',

'rover\_id': 5,

'full\_name': 'Rear Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/rcam/RRB\_646860185EDR\_F0810628RHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754125,

'sol': 2809,

'camera': {'id': 21,

'name': 'RHAZ',

'rover\_id': 5,

'full\_name': 'Rear Hazard Avoidance Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/rcam/RLB\_646860185EDR\_F0810628RHAZ00337M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754126,

'sol': 2809,

'camera': {'id': 26,

'name': 'NAVCAM',

'rover\_id': 5,

'full\_name': 'Navigation Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/ncam/NLB\_646869070EDR\_F0810628NCAM00229M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754127,

'sol': 2809,

'camera': {'id': 26,

'name': 'NAVCAM',

'rover\_id': 5,

'full\_name': 'Navigation Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/ncam/NLB\_646860211EDR\_F0810628NCAM00229M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754128,

'sol': 2809,

'camera': {'id': 26,

'name': 'NAVCAM',

'rover\_id': 5,

'full\_name': 'Navigation Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/ncam/NRB\_646869070EDR\_F0810628NCAM00229M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name': 'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}},

{'id': 754129,

'sol': 2809,

'camera': {'id': 26,

'name': 'NAVCAM',

'rover\_id': 5,

'full\_name': 'Navigation Camera'},

'img\_src': 'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/ncam/NRB\_646860211EDR\_F0810628NCAM00229M\_.JPG',

'earth\_date': '2020-07-01',

'rover': {'id': 5,

'name':

'Curiosity',

'landing\_date': '2012-08-06',

'launch\_date': '2011-11-26',

'status': 'active'}}]}

In [48]:

photos = response.json()["photos"]

print(f'Найдено {len(photos)} фотографий.')

photos[7]["img\_src"]

Out[48]:

'https://mars.nasa.gov/msl-raw-images/proj/msl/redops/ods/surface/sol/02809/opgs/edr/rcam/RLB\_646860185EDR\_F0810628RHAZ00337M\_.JPG'

In [56]:

CLIENT\_ID = "CLIENT\_SECRET"

CLIENT\_SECRET = "CLIENT\_SECRET"

REDIRECT\_URI = "https://httpbin.org/anything"

In [57]:

def create\_oauth\_link():

params = {

"client\_id": CLIENT\_ID,

"redirect\_uri": REDIRECT\_URI,

"scope": "user",

"response\_type": "code",

}

endpoint = "https://github.com/login/oauth/authorize"

response = requests.get(endpoint, params=params)

url = response.url

return url

In [58]:

def exchange\_code\_for\_access\_token(code=None):

params = {

"client\_id": CLIENT\_ID,

"client\_secret": CLIENT\_SECRET,

"redirect\_uri": REDIRECT\_URI,

"code": code,

}

headers = {"Accept": "application/json"}

endpoint = "https://github.com/login/oauth/access\_token"

response = requests.post(endpoint, params=params, headers=headers).json()

return response["access\_token"]

In [ ]:

link = create\_oauth\_link()

print(f"Перейдите по ссылке, чтобы запустить аутентификацию с помощью GitHub: {link}")

In [ ]:

code = input("GitHub code: ")

access\_token = exchange\_code\_for\_access\_token(code)

print(f"Exchanged code {code} и access\_token: {access\_token}")

In [63]:

def print\_user\_info(access\_token=None):

headers = {"Authorization": f"token {access\_token}"}

endpoint = "https://api.github.com/user"

response = requests.get(endpoint, headers=headers).json()

name = response["name"]

username = response["login"]

private\_repos\_count = response["total\_private\_repos"]

print(f"{name} ({username}) | private repositories: {private\_repos\_count}")

print\_user\_info(access\_token)

Dmitry Yudakov (in7erval) | private repositories: 4

In [64]:

response = requests.get("https://api.github.com/events?per\_page=1&page=0")

response.json()[0]["id"]

Out[64]:

'21382832829'

In [65]:

response = requests.get("https://api.github.com/events?per\_page=1&page=1")

response.json()[0]["id"]

Out[65]:

'21382838218'

In [66]:

response = requests.get("https://api.github.com/events?per\_page=1&page=2")

response.json()[0]["id"]

Out[66]:

'21382842558'

In [68]:

API\_KEY = "GIPHY\_API\_KEY"

endpoint = "https://api.giphy.com/v1/gifs/trending"

params = {"api\_key": API\_KEY, "limit": 3, "rating": "g"}

response = requests.get(endpoint, params=params).json()

for gif in response["data"]:

title = gif["title"]

trending\_date = gif["trending\_datetime"]

url = gif["url"]

print(f"{title} | {trending\_date} | {url}")

Champions League Football GIF by UEFA | 2022-04-20 16:00:08 | https://giphy.com/gifs/uefa-goal-kick-bobby-firmino-E3gXCU0TPvmjEjLMsa

Fashion Vintage GIF | 2022-04-20 14:37:03 | https://giphy.com/gifs/50s-british-path-f2amVVgFXoUAo

Snoop Dogg Sunset GIF | 2022-04-20 15:41:08 | https://giphy.com/gifs/l39JrKv9ZZX2g

In [71]:

import requests

from datetime import date, timedelta

In [76]:

today = date.today()

yesterday = today - timedelta(days=1)

country = "Russia"

endpoint = f"https://api.covid19api.com/country/{country}/status/confirmed"

params = {"from": str(yesterday), "to": str(today)}

response = requests.get(endpoint, params=params).json()

total\_confirmed = 0

for day in response:

cases = day.get("Cases", 0)

total\_confirmed += cases

print(f"Total Confirmed Covid-19 cased in {country}: {total\_confirmed}")

Total Confirmed Covid-19 cased in Russia: 17829009

In [77]:

endpoint = "https://www.googleapis.com/books/v1/volumes"

query = "Моби Дик"

params = {"q": query, "maxResults": 3}

response = requests.get(endpoint, params=params).json()

for book in response["items"]:

volume = book["volumeInfo"]

title = volume["title"]

published = volume.get("publishedDate", "год издания неизвестен")

description = volume.get("description", "описание отсутствует")

print(f"{title} ({published} | {description})")

Моби Дик, или Белый кит (2014-10-17 | описание отсутствует)

Моби Дик, или, Белый кит (2020 | описание отсутствует)

Моби Дик,

или Белый Кит (2012 | В центре этого уникального произведения - погоня за Белым Китом. Захватывающий сюжет, эпические морские картины, описания ярких человеческих характеров в гармоничном сочетании с философскими обобщениями делают книгу шедевром мировой литературы)

In [ ]: