A picture containing dark, night sky

Description automatically generated

API Testing

Hasan Mahamud Rana

8732852

PROG8440 – Software Quality

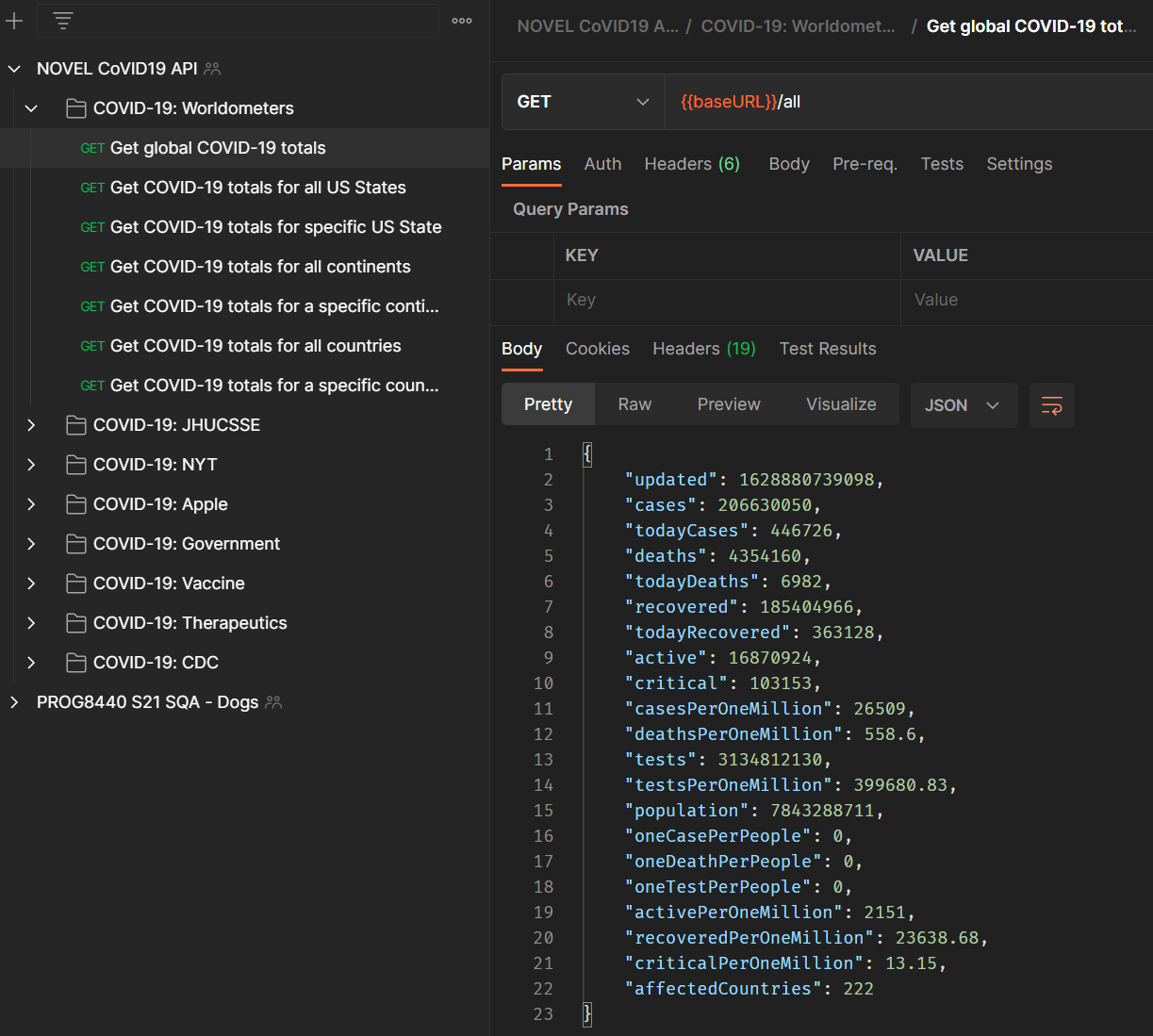
Rob Bowyer

Due Date: Aug 13, 2021 05:00 PM

**DISEASE.SH DOCS - AN OPEN API FOR DISEASE-RELATED STATISTICS**

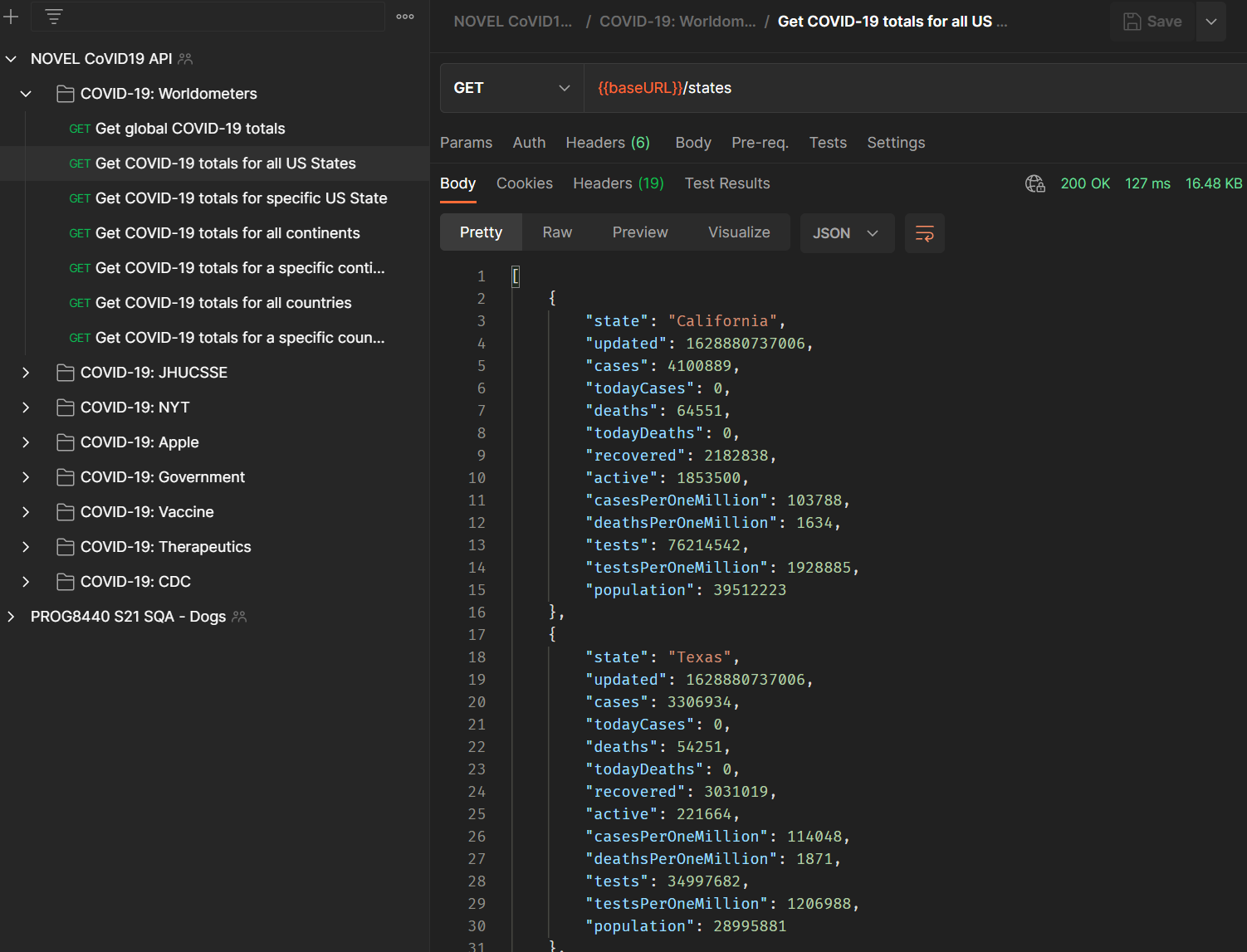
1. **Ten (10) unique API calls with different endpoints using Postman.**
   1. COVID-19: Worldometers / Get global COVID-19 totals

url = <https://disease.sh/v3/covid-19/all>



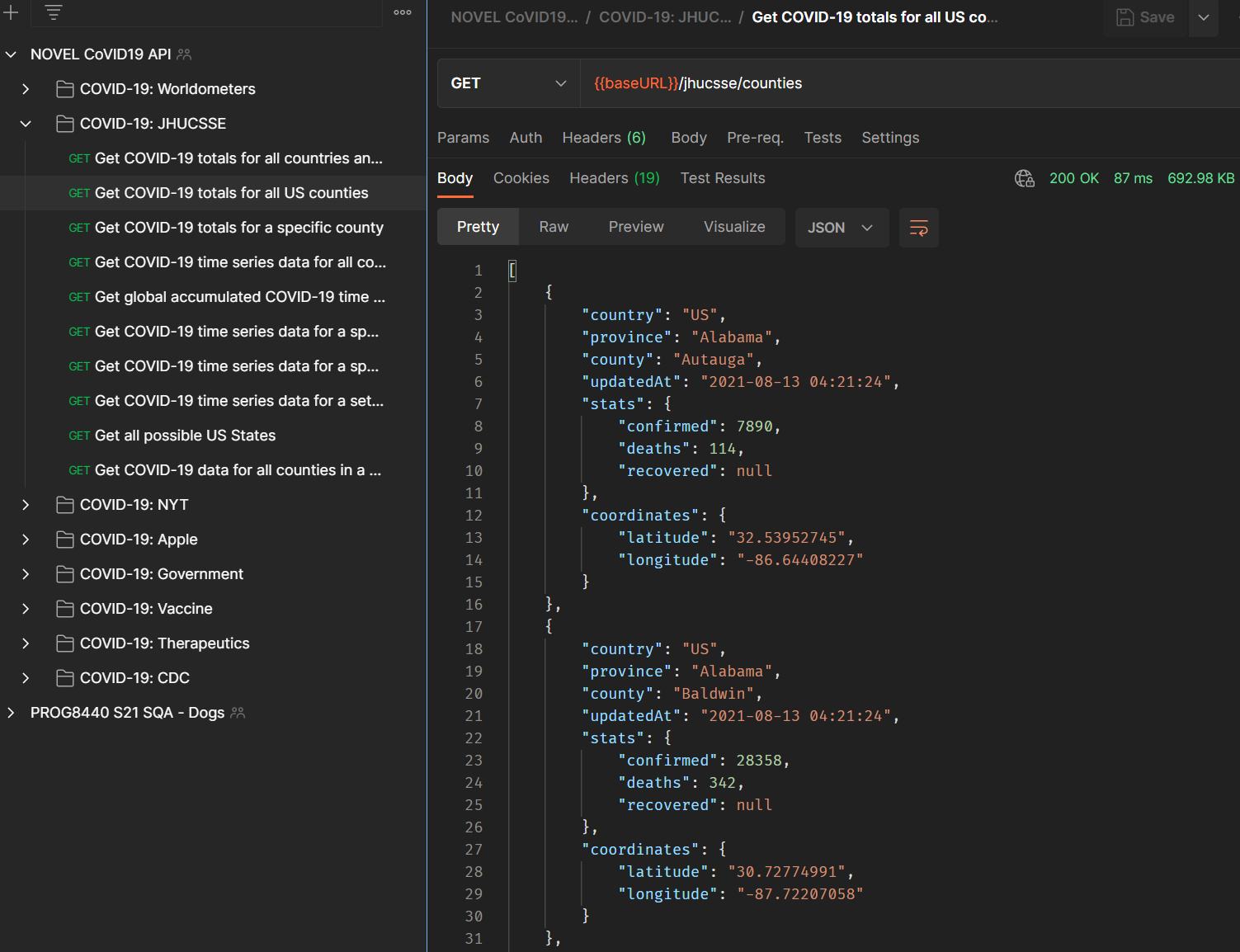
* 1. COVID-19: Worldometers / Get COVID-19 totals for all US States

url = <https://disease.sh/v3/covid-19/states>



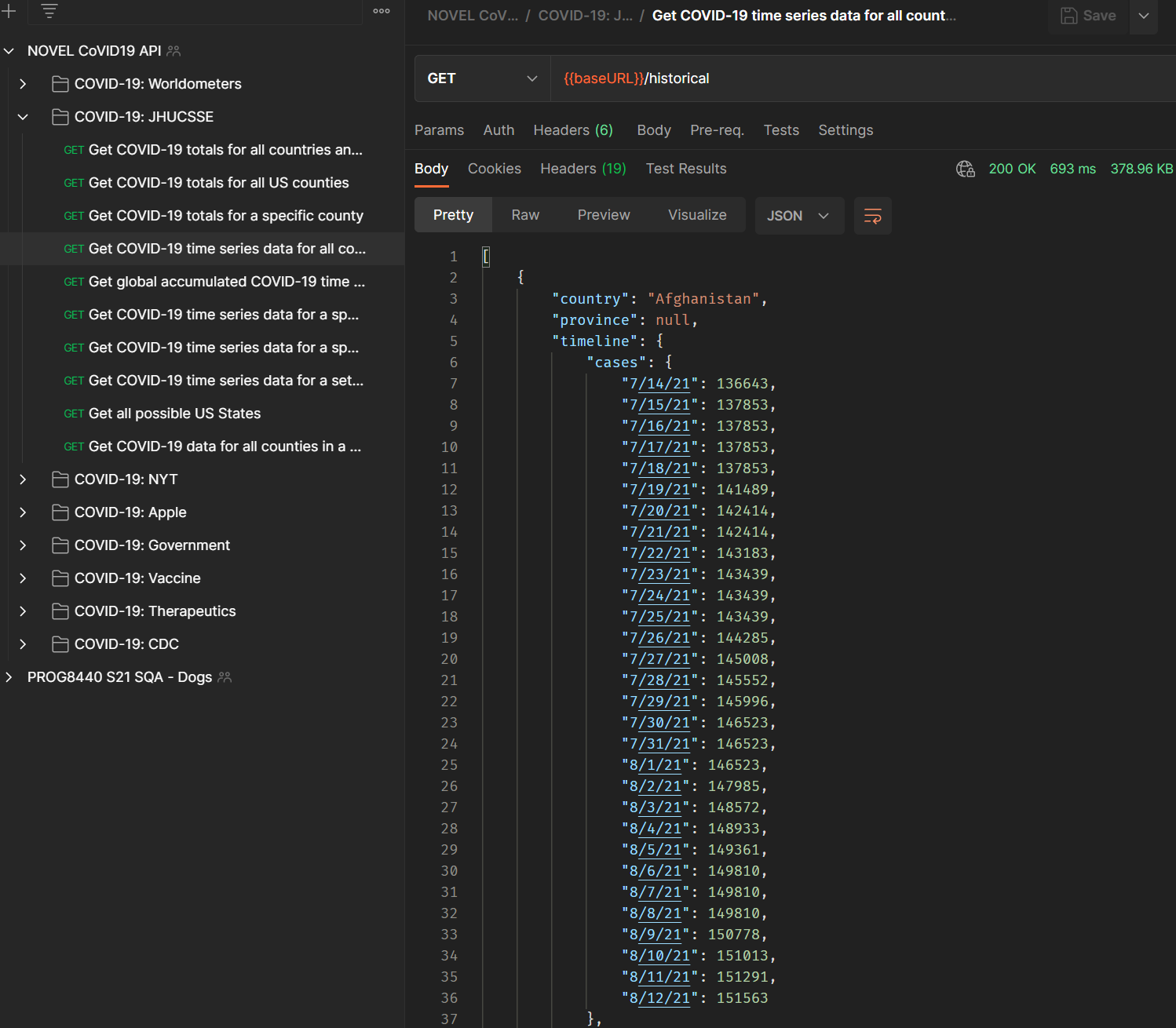
* 1. COVID-19: JHUCSSE / Get COVID-19 totals for all US counties

url = <https://disease.sh/v3/covid-19/jhucsse/counties>



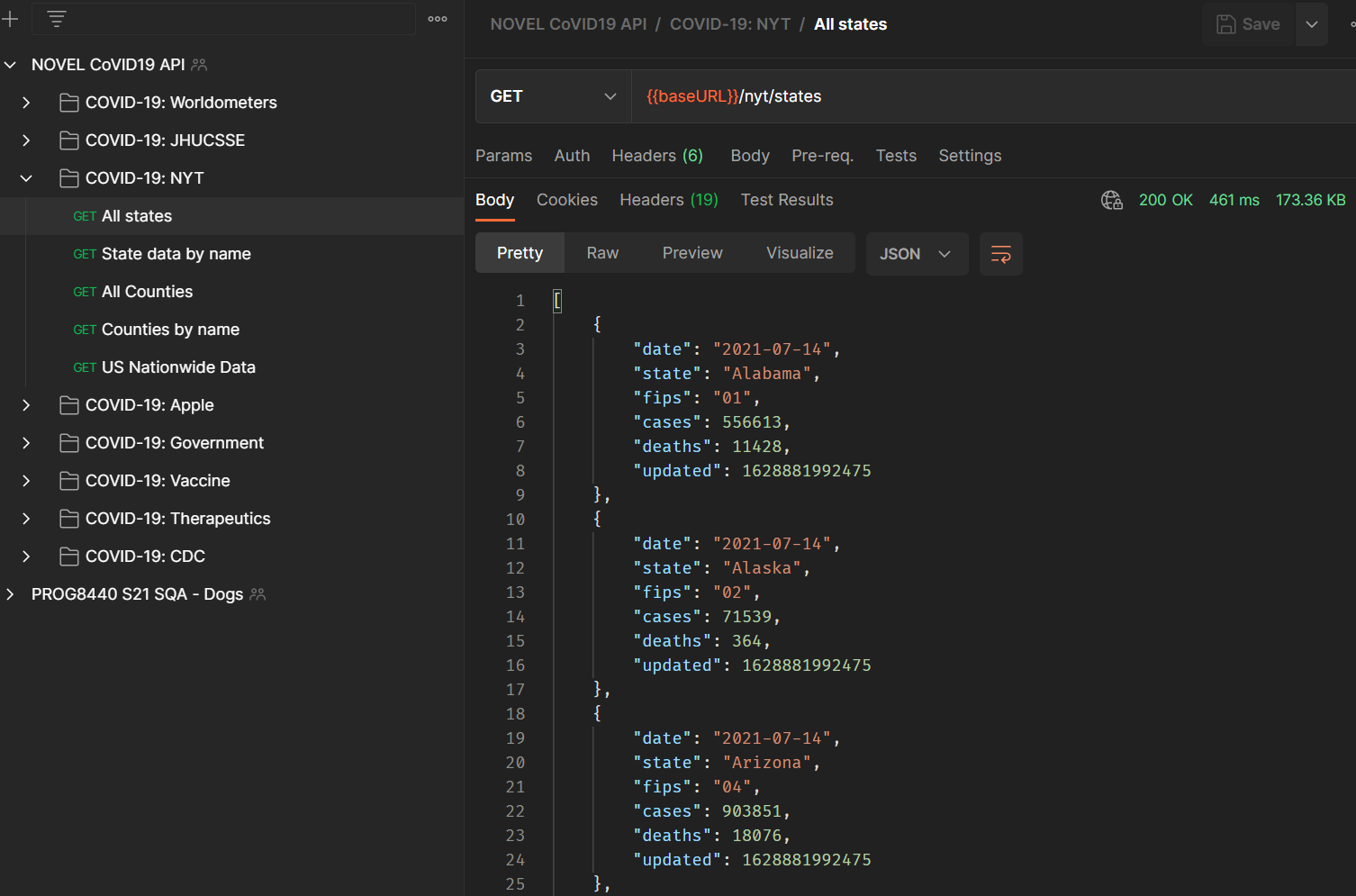
* 1. COVID-19: JHUCSSE / Get COVID-19 time series data for all countries and their provinces

url = <https://disease.sh/v3/covid-19/historical>



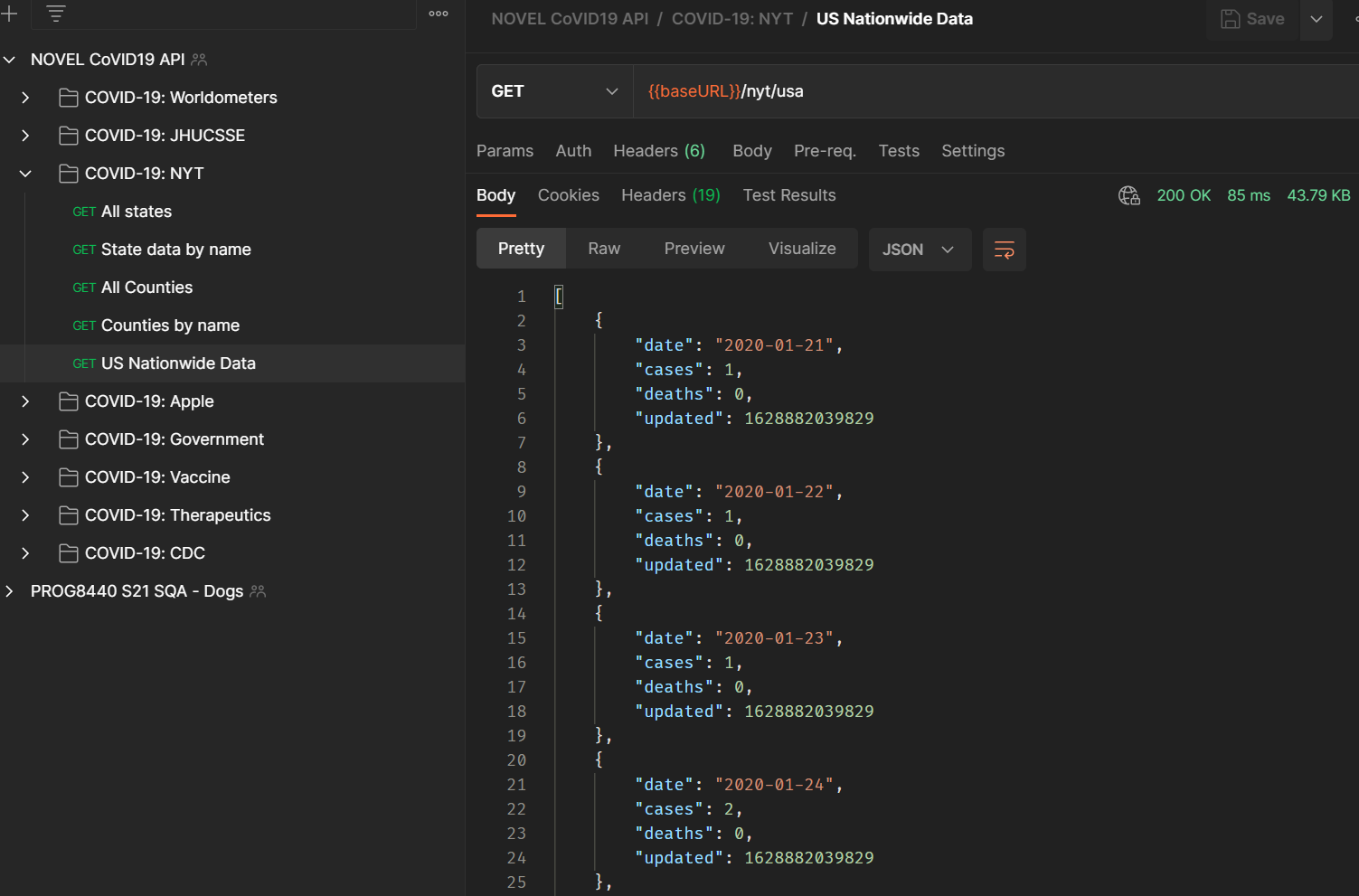
* 1. COVID-19: NYT / All states

url = <https://disease.sh/v3/covid-19/nyt/states>



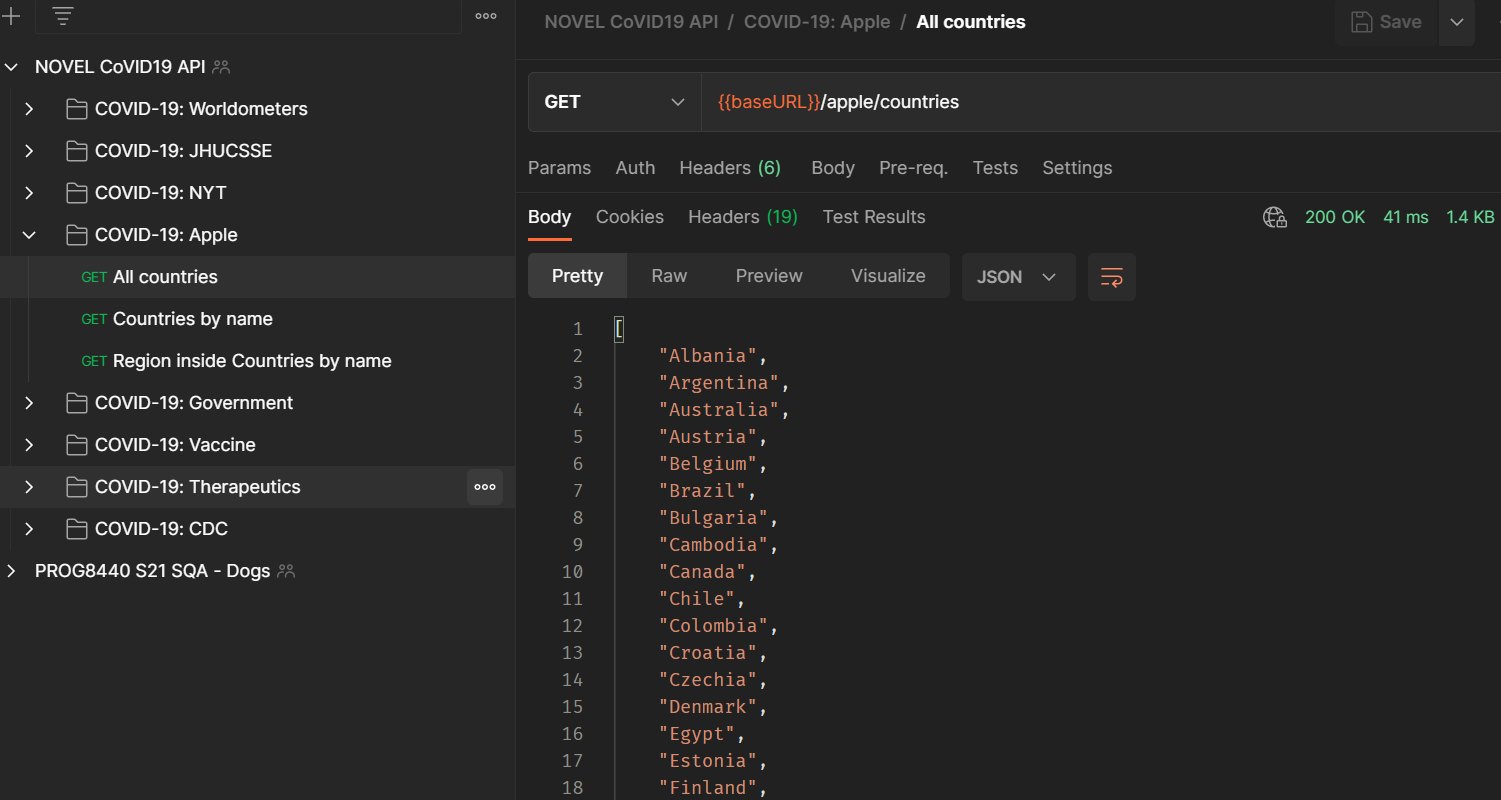
* 1. COVID-19: NYT / US Nationwide Data

url = <https://disease.sh/v3/covid-19/nyt/usa>



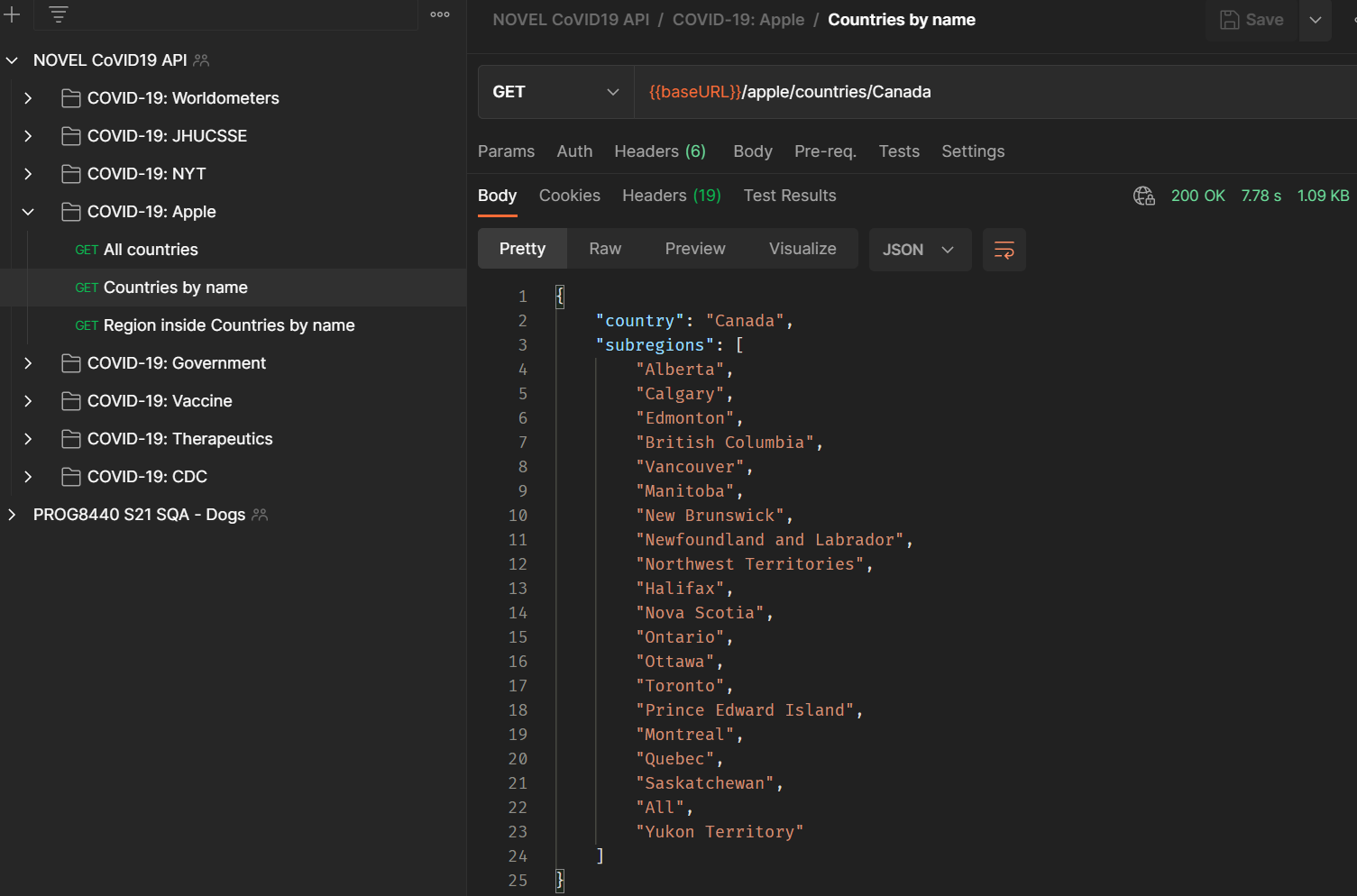
* 1. COVID-19: Apple / All countries

url = <https://disease.sh/v3/covid-19/apple/countries>



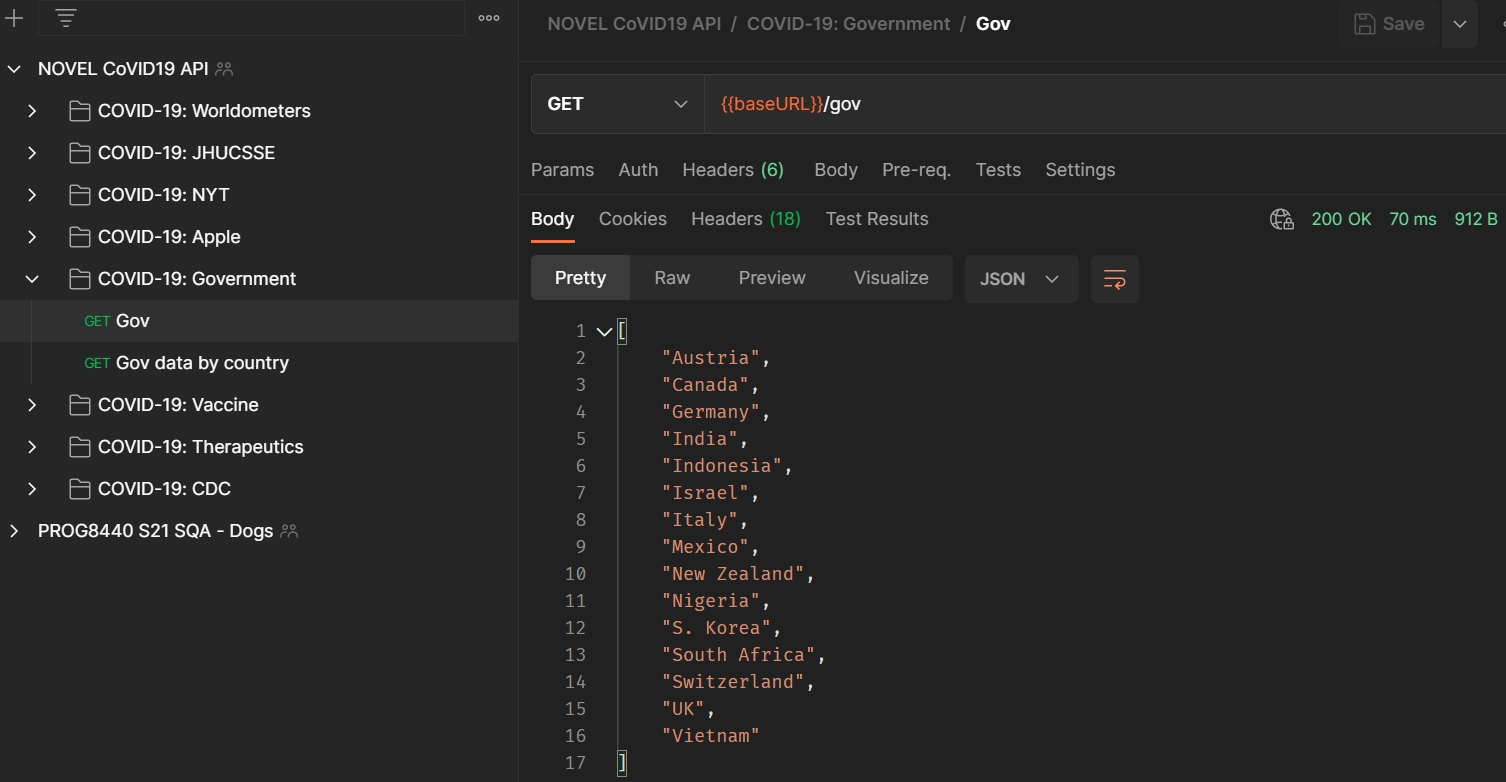
* 1. COVID-19: Apple / Single country by name

url = <https://disease.sh/v3/covid-19/apple/countries/Canada>



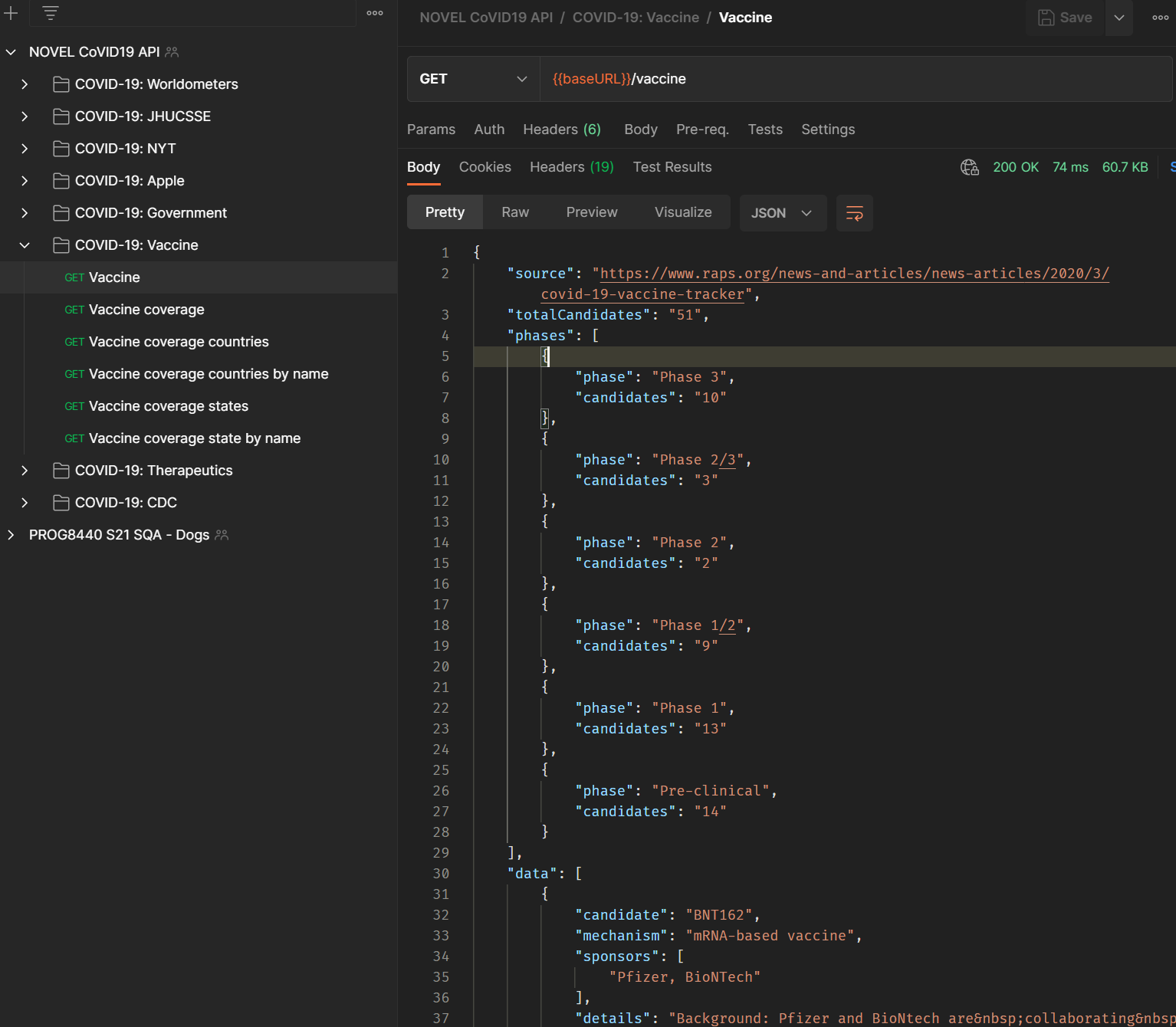
* 1. COVID-19: Government

url = <https://disease.sh/v3/covid-19/gov>



* 1. COVID-19: Vaccine

url = <https://disease.sh/v3/covid-19/vaccine>



1. **Python scripts of the above**

"""

Sample code from Postman

import requests

url = "https://disease.sh/v3/covid-19/all"

payload={}

headers = {}

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

print(response.text)

"""

# Basic setup

import requests

payload = {}

headers = {}

baseURL = 'https://disease.sh/v3/covid-19/'

def endpoints(param):

return baseURL + param

def response(url):

return requests.get(url, headers=headers, data=payload)

def responseText(param):

url = endpoints(param)

res = response(url)

return res.text

# 1. COVID-19: Worldometers / Get global COVID-19 totals

GetGlobalCOVID19Totals = responseText('all')

print(GetGlobalCOVID19Totals)

# 2. COVID-19: Worldometers / Get COVID-19 totals for all US States

GetCOVID19TotalsForAllUSStates = responseText('states')

print(GetCOVID19TotalsForAllUSStates)

# 3. COVID-19: JHUCSSE / Get COVID-19 totals for all US counties

GetCOVID19TotalsForAllUSCounties = responseText('jhucsse/counties')

print(GetCOVID19TotalsForAllUSCounties)

# 4. COVID-19: JHUCSSE / Get COVID-19 time series data for all countries and their provinces

GetCOVID19TimeSeriesDataForAllCountries = responseText('historical')

print(GetCOVID19TimeSeriesDataForAllCountries)

# 5. COVID-19: NYT / All states

GetCOVID19NYTAllStates = responseText('nyt/states')

print(GetCOVID19NYTAllStates)

# 6. COVID-19: NYT / US Nationwide Data

GetCOVID19USNationWideData = responseText('nyt/usa')

print(GetCOVID19USNationWideData)

# 7. COVID-19: Apple / All countries

GetCOVID19AppleAllCountries = responseText('apple/countries')

print(GetCOVID19AppleAllCountries)

# 8. COVID-19: Apple / Single country by name

GetCOVID19AppleAllCountriesCanada = responseText('apple/countries/Canada')

print(GetCOVID19AppleAllCountriesCanada)

# 9. COVID-19: Government

GetGovernment = responseText('gov')

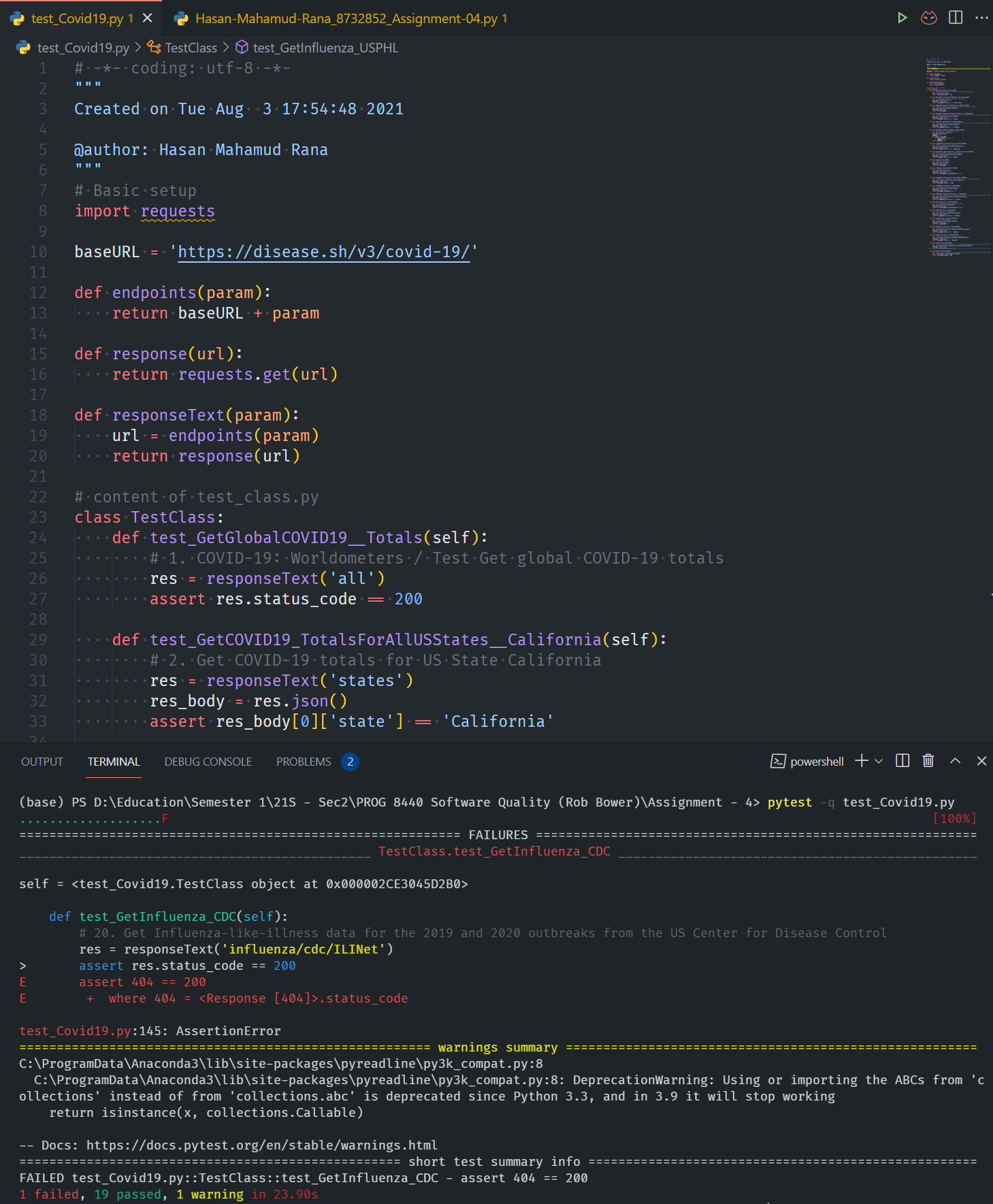
print(GetGovernment)

# 10. COVID-19: Vaccine

GetVaccine = responseText('vaccine')

print(GetVaccine)

1. **PyTest to create 20 tests (1 test will fails)**

****

# -\*- coding: utf-8 -\*-

"""

Created on Tue Aug 3 17:54:48 2021

@author: Hasan Mahamud Rana

"""

# Basic setup

import requests

baseURL = 'https://disease.sh/v3/covid-19/'

def endpoints(param):

return baseURL + param

def response(url):

return requests.get(url)

def responseText(param):

url = endpoints(param)

return response(url)

# content of test\_Covid19.py

class TestClass:

def test\_GetGlobalCOVID19\_\_Totals(self):

# 1. COVID-19: Worldometers / Test Get global COVID-19 totals

res = responseText('all')

assert res.status\_code == 200

def test\_GetCOVID19\_TotalsForAllUSStates\_\_California(self):

# 2. Get COVID-19 totals for US State California

res = responseText('states')

res\_body = res.json()

assert res\_body[0]['state'] == 'California'

def test\_GetCOVID19\_TotalsForAllUSCounties\_\_HasValue(self):

# 3. COVID-19: JHUCSSE / Get COVID-19 totals for all US counties

res = responseText('jhucsse/counties')

res\_body = res.json()

assert len(res\_body) > 1

def test\_GetCOVID19\_TimeSeriesDataForAllCountries\_\_Canada(self):

# 4. COVID-19: JHUCSSE / Get COVID-19 time series data for all countries and their provinces

res = responseText('historical/Canada')

res\_body = res.json()

assert res\_body['country'] == 'Canada'

def test\_GetCOVID19\_NYTAllStates\_\_Alabama(self):

# 5. COVID-19: JHUCSSE / Get COVID-19 time series data for all countries and their provinces

res = responseText('nyt/states/Alabama')

res\_body = res.json()

assert res\_body[0]['state'] == 'Alabama'

def test\_GetCOVID19\_USNationWideData\_\_NoDeath(self):

# 6. COVID-19: NYT / US Nationwide Data

res = responseText('nyt/usa')

res\_body = res.json()

NoDeath = 0

for item in res\_body:

if item['deaths'] == 0:

NoDeath += 1

assert NoDeath > 0

def test\_GetCOVID19\_AppleAllCountries\_\_Argentina(self):

# 7. COVID-19: Apple / All countries

res = responseText('apple/countries/Argentina')

res\_body = res.json()

assert res\_body['country'] == 'Argentina'

def test\_GetCOVID19\_AppleAllCountries\_\_RegionInsideCountry(self):

# 8. COVID-19: Apple / Region inside country

res = responseText('apple/countries/Canada')

res\_body = res.json()

assert res\_body['country'] == 'Canada'

def test\_GetGovernment(self):

# 9. COVID-19: Government

res = responseText('gov')

res\_body = res.json()

assert len(res\_body) > 0

def test\_GetVaccine\_TotalCandidates(self):

# 10. COVID-19: Vaccine

res = responseText('vaccine')

res\_body = res.json()

assert int(res\_body['totalCandidates']) > 0

def test\_GetCOVID19\_TotalsCasesFor\_\_NorthAmerica(self):

# 11. COVID-19: Worldometers / Get COVID-19 totals for North America

res = responseText('continents/North America')

res\_body = res.json()

assert res\_body['cases'] > 1000

def test\_GetCOVID19\_TodayDeaths\_Canada(self):

# 12. Get COVID-19 todays death for Canada

res = responseText('countries/Canada')

res\_body = res.json()

assert res\_body['todayDeaths'] >= 0

def test\_GetCOVID19\_TotalsForUSCounties\_\_alaska(self):

# 13. COVID-19: JHUCSSE / Get COVID-19 data for all counties in a specified US state

res = responseText('historical/usacounties/alaska')

res\_body = res.json()

assert res\_body[0]['province'] == 'alaska'

def test\_GetTherapeutics\_TrialData(self):

# 14. Get therapeutics trial data from RAPS

res = responseText('therapeutics')

res\_body = res.json()

assert int(res\_body['totalCandidates']) > 0

def test\_GetNYTCounties\_\_Autauga(self):

# 15. Get NYT data for Autauga

res = responseText('nyt/counties/Autauga')

res\_body = res.json()

assert res\_body[0]['county'] == 'Autauga'

def test\_GetGovernment\_\_NewZealand(self):

# 16. COVID-19: Government

res = responseText('gov/New Zealand')

res\_body = res.json()

assert len(res\_body) > 0

def test\_GetVaccine\_Countries\_\_Albania(self):

# 17. COVID-19: Vaccine for coutries country

res = responseText('vaccine/coverage/countries/Albania')

res\_body = res.json()

assert res\_body['country'] == 'Albania'

def test\_GetVaccine\_States\_\_Delaware(self):

# 18. COVID-19: Vaccine for coutries country

res = responseText('vaccine/coverage/states/Delaware')

res\_body = res.json()

assert res\_body['state'] == 'Delaware'

def test\_GetInfluenza\_USPHL(self):

# 19. Get Influenza report data for the 2019 and 2020 outbreaks from the US Center for Disease Control, reported by US public health labs

res = response('https://disease.sh/v3/influenza/cdc/USPHL')

res\_body = res.json()

assert len(res\_body['data']) > 0

def test\_GetInfluenza\_CDC(self):

# 20. Get Influenza-like-illness data for the 2019 and 2020 outbreaks from the US Center for Disease Control

res = responseText('influenza/cdc/ILINet')

assert res.status\_code == 200

**References**

Novel CoVID19 API (July, 2020) *disease.sh Docs - An open API for disease-related statistics*<https://disease.sh/docs/>

Bowyer, R. (August, 2021) *PROG8440\_lesson10\_APITesting* [PowerPoint slides]. eConestoga.