



NASA

MIDTERM PROJECT

สมาชิก

นางสาวกาญจน์นิชา คำจริง 61102010135

นายปวริศ ศรีพิบูลย์ 61102010151

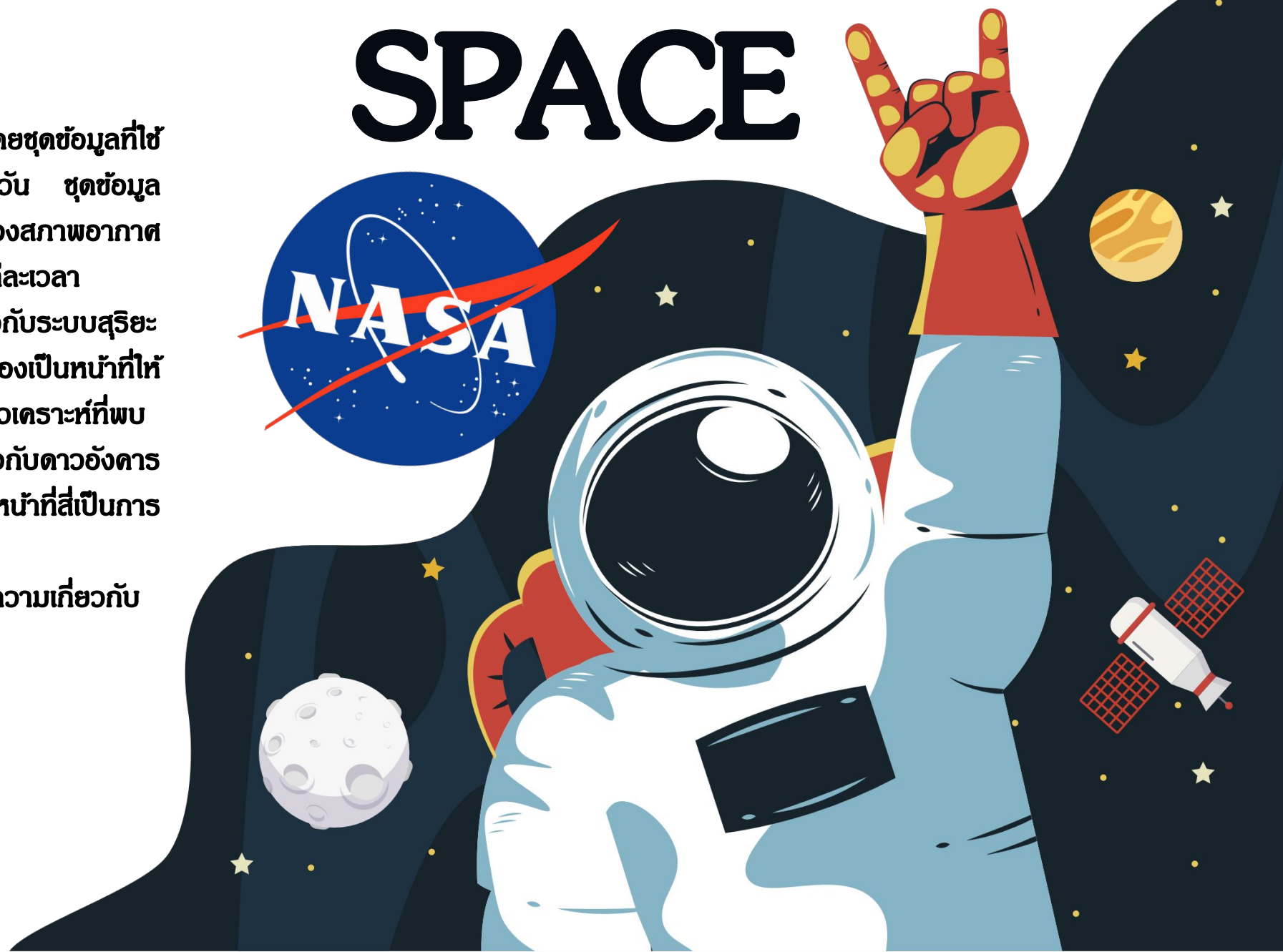
นางสาวเพชรสิริ ศิริยม 61102010154

เว็บของเราใช้ข้อมูล API ของ NASA โดยชุดข้อมูลที่ใช้จะประกอบไปด้วย ชุดข้อมูลภาพถ่ายประจำวัน ชุดข้อมูลดาวเคราะห์ที่พบเจอในแต่ละวัน ชุดข้อมูลของสภาพอากาศดาวอังคาร และภาพถ่ายของโลกในแต่ละวันแต่ละเวลา

โดยหน้าแรกใช้เป็นการแสดงข้อมูลเกี่ยวกับระบบสุริยะ และมีการแสดงภาพถ่ายประจำวัน หน้าที่สองเป็นหน้าที่ให้ความรู้เกี่ยวกับดาวเคราะห์และแสดงข้อมูลดาวเคราะห์ที่พบเจอในแต่ละวัน หน้าทีสามเป็นหน้าข้อมูลเกี่ยวกับดาวอังคาร มีการแสดงสภาพอากาศของดาวอังคาร หน้าทีสี่เป็นการแสดงภาพของโลกในแต่ละวันและ

ผู้ใช้สามารถค้นหาตามวันที่ได้และมีบทความเกี่ยวกับพระจันทร์ประกอบอยู่

SPACE



FRONT-END : HTML CSS JAVASCRIPT

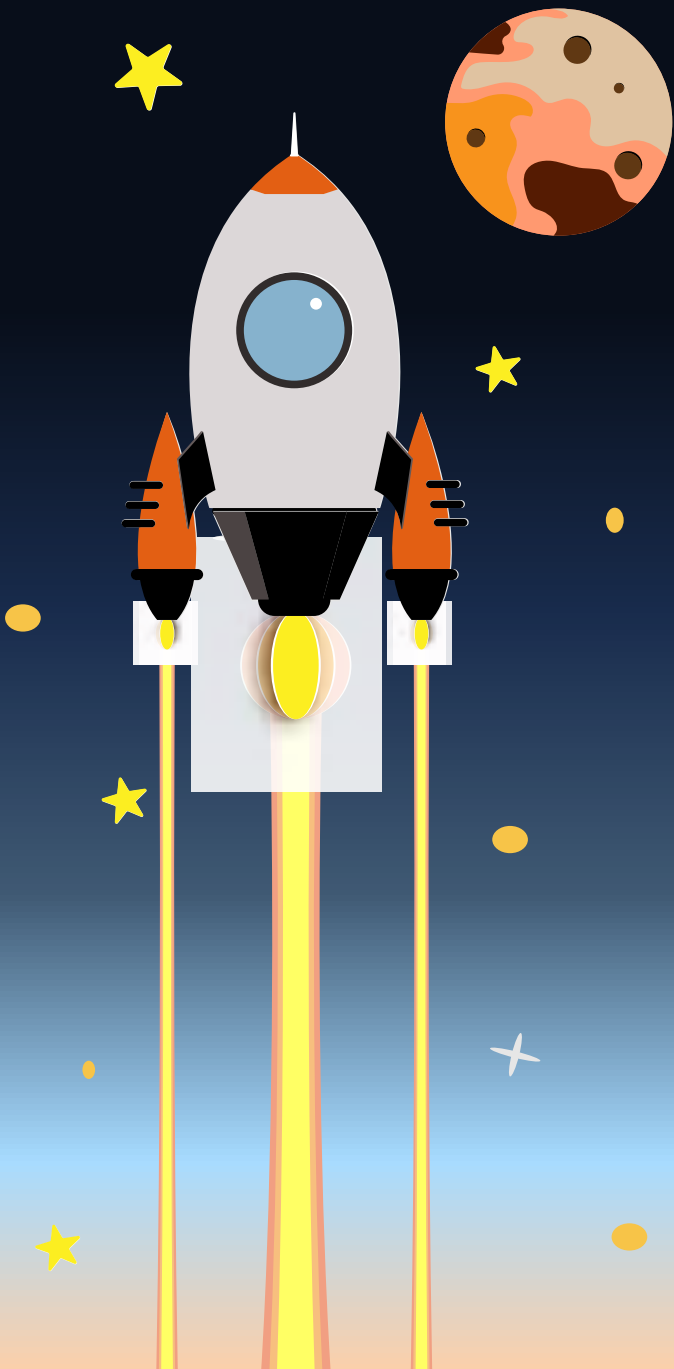
BACK-END : FLASK





API : <https://api.nasa.gov/planetary/apod>

```
{'copyright': 'Chuck Ayoub',  
'date': '2018-11-14',  
'explanation': "What's inside this cosmic cave? A stellar nursery 10 "  
               'light-years deep. The featured skyscape is dominated by '  
               'dusty Sh2-155, the Cave Nebula. In the telescopic image, data '  
               'taken through a narrowband filters tracks the nebular glow of '  
               'hydrogen, oxygen, and sulfur, colors that together form the '  
               'Hubble Palette. About 2,400 light-years away, the scene lies '  
               'along the plane of our Milky Way Galaxy toward the royal '  
               'northern constellation of Cepheus. Astronomical explorations '  
               'of the region reveal that it has formed at the boundary of '  
               'the massive Cepheus B molecular cloud and the hot, young '  
               'stars of the Cepheus OB 3 association. The bright rim of '  
               'ionized hydrogen gas is energized by radiation from the hot '  
               'stars, dominated by the bright star just to the left of the '  
               'cave entrance. Radiation driven ionization fronts are likely '  
               'triggering collapsing cores and new star formation within.',  
'hdurl': 'https://apod.nasa.gov/apod/image/1811/CaveNebula\_Ayoub\_2469.jpg',  
'media_type': 'image',  
'service_version': 'v1',  
'title': 'The Cave Nebula in Hydrogen, Oxygen, and Sulfur',  
'url': 'https://apod.nasa.gov/apod/image/1811/CaveNebula\_Ayoub\_960.jpg'}
```

```
1 def fetchAPOD():
2     URL_APOD = "https://api.nasa.gov/planetary/apod"
3     api_key = 'PIQgwKgT5WieoxPWMksJNr1GtdtIktdVc01dc6Jr'
4     now = datetime.datetime.now()
5     year = now.year
6     month = now.month
7     day = now.day - 1
8     date = str(year)+'-'+str(month)+'-'+str(day)
9     date
10    params = {
11        'api_key': api_key,
12        'date': date,
13        'hd': 'True'
14    }
15    response = requests.get(URL_APOD, params=params).json()
16    APOD = {'title': response['title'],
17           'explanation': response['explanation'],
18           'daily': response['date'],
19           'hdurl': response['hdurl']}
20
21
22    return APOD
```


2





API:<https://api.nasa.gov/neo/rest/v1/feed>

```
[{'absolute_magnitude_h': 19.7,
'close_approach_data': [{'close_approach_date': '2021-03-09',
'close_approach_date_full': '2021-Mar-09 13:49',
'epoch_date_close_approach': 1615297740000,
'miss_distance': {'astronomical': '0.1716840417',
'kilometers': '25683566.951311179',
'lunar': '66.7850922213',
'miles': '15959028.4735611102'}},
'orbiting_body': 'Earth',
'relative_velocity': {'kilometers_per_hour': '72146.4174182538',
'kilometers_per_second': '20.0406715051',
'miles_per_hour': '44828.9780896922'}},
'estimated_diameter': {'feet': {'estimated_diameter_max': 2238.8501681036,
'estimated_diameter_min': 1001.2442334633},
'kilometers': {'estimated_diameter_max': 0.6824015094,
'estimated_diameter_min': 0.3051792326},
'meters': {'estimated_diameter_max': 682.4015094011,
'estimated_diameter_min': 305.1792325939},
'miles': {'estimated_diameter_max': 0.4240245083,
'estimated_diameter_min': 0.1896295249}},
'id': '2523637',
'is_potentially_hazardous_asteroid': False,
'is_sentry_object': False,
'links': {'self': 'http://www.neowsapp.com/rest/v1/neo/2523637?api\_key=PIQgwKgT5WieoxPWMksJNr1GtdtIktdVc01dc6Jr'},
'name': '523637 (2010 LT108)',
'nasa_jpl_url': 'http://ssd.jpl.nasa.gov/sbdb.cgi?sstr=2523637',
'neo_reference_id': '2523637'}
```

```
1 def fetchAsteroidNeowsFeed():
2     URL_NeoFeed = "https://api.nasa.gov/neo/rest/v1/feed"
3     api_key = '7UbKT3dVXEXJHRbZ7VV0J2eJ6aXMK7XGAgXba2UH'
4     params = {
5         'api_key': api_key,
6         'start_date': '2021-03-09'
7     }
8     response = requests.get(URL_NeoFeed, params=params).json()
9     NEO = response['near_earth_objects']['2021-03-09']
10    ID = []
11    name = []
12    link = []
13    for i in range(1,13):
14        myNeoFeed = NEO[i]
15        ID.append(myNeoFeed['id'])
16        name.append(myNeoFeed['name'])
17        link.append(myNeoFeed['nasa_jpl_url'])
18    myFeed = zip(ID,name,link)
19    return myFeed
```

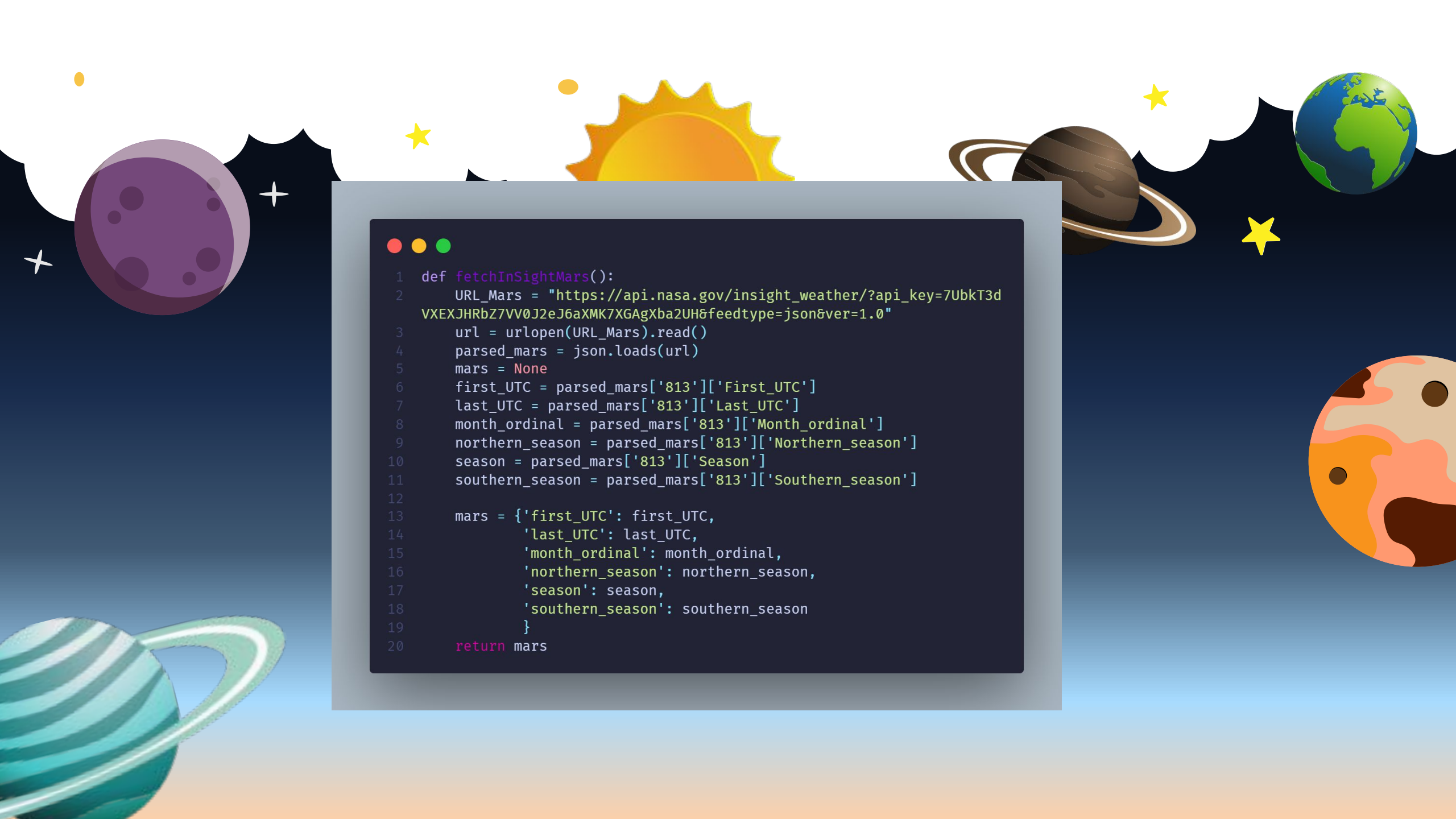
3



API : [https://api.nasa.gov/insight_weather/](https://api.nasa.gov/insight_weather/?api_key=DEMO_KEY&feedtype=json&ver=1.0)
[?api_key=DEMO_KEY&feedtype=json&ver=1.0](https://api.nasa.gov/insight_weather/?api_key=DEMO_KEY&feedtype=json&ver=1.0)

```
{
  "813": {
    "First.UTC": "2021-03-10T13:35:28Z",
    "Last.UTC": "2021-03-11T14:14:59Z",
    "Month_ordinal": 12,
    "Northern_season": "late winter",
    "PRE": {
      "av": 727.909,
      "ct": 100620,
      "mn": 706.2391,
      "mx": 743.8608
    },
    "Season": "winter",
    "Southern_season": "late summer",
    "WD": {
      "most_common": null
    }
  },
}
```





```
1 def fetchInSightMars():
2     URL_Mars = "https://api.nasa.gov/insight_weather/?api_key=7UbKT3d
3     VXEXJHRbZ7VV0J2eJ6aXMK7XGAgXba2UH&feedtype=json&ver=1.0"
4     url = urlopen(URL_Mars).read()
5     parsed_mars = json.loads(url)
6     mars = None
7     first.UTC = parsed_mars['813']['First.UTC']
8     last.UTC = parsed_mars['813']['Last.UTC']
9     month_ordinal = parsed_mars['813']['Month_ordinal']
10    northern_season = parsed_mars['813']['Northern_season']
11    season = parsed_mars['813']['Season']
12    southern_season = parsed_mars['813']['Southern_season']
13
14    mars = {'first.UTC': first.UTC,
15           'last.UTC': last.UTC,
16           'month_ordinal': month_ordinal,
17           'northern_season': northern_season,
18           'season': season,
19           'southern_season': southern_season}
20    return mars
```

4



API : [https://api.nasa.gov/EPIC/api/
natural/date/{YYYY-MM-DD}
?api_key=DEMO_KEY](https://api.nasa.gov/EPIC/api/natural/date/{YYYY-MM-DD}?api_key=DEMO_KEY)

```
[{'attitude_quaternions': {'q0': 0.495585,  
'q1': -0.356553,  
'q2': -0.694341,  
'q3': 0.380992},  
'caption': "This image was taken by NASA's EPIC camera onboard the NOAA DSCOVR spacecraft",  
'centroid_coordinates': {'lat': 24.56543, 'lon': 170.683594},  
'coords': {'attitude_quaternions': {'q0': 0.495585,  
'q1': -0.356553,  
'q2': -0.694341,  
'q3': 0.380992},  
'centroid_coordinates': {'lat': 24.56543, 'lon': 170.683594},  
'dscovr_j2000_position': {'x': 339005.145834,  
'y': 1368757.776568,  
'z': 645861.927788},  
'lunar_j2000_position': {'x': 381104.35964,  
'y': 104675.95663,  
'z': -35701.90868},  
'sun_j2000_position': {'x': 56531896.481815,  
'y': 129098775.627199,  
'z': 55963516.666649}},  
'date': '2019-05-30 01:09:10',  
'dscovr_j2000_position': {'x': 339005.145834,  
'y': 1368757.776568,  
'z': 645861.927788},  
'identifier': '20190530011359',  
'image': 'epic_1b_20190530011359',  
'lunar_j2000_position': {'x': 381104.35964,  
'y': 104675.95663,  
'z': -35701.90868},  
'sun_j2000_position': {'x': 56531896.481815,  
'y': 129098775.627199,  
'z': 55963516.666649},  
'version': '03'}
```



```
1 def get_image():
2     scr_date = request.args.get('date')
3     if not scr_date:
4         scr_date = '2019-05-30'
5     url = 'https://api.nasa.gov/EPIC/api/natural/date/{0}?api_key=PIQ
6 gwKgT5WieoxPwMksJNr1GtdtIktDvc01dc6Jr'
7     url_img = url.format(scr_date)
8     data = urlopen(url_img).read()
9     parsed = json.loads(data)
10    img = []
11    for i in range(6):
12        earth = parsed[i]
13        get_image = earth['image']
14        date = earth['date']
15        date = date[0:10]
16        date_fom = date.replace('-', '/')
17        img_link = fetchEPICImage(get_image, date_fom)
18        img.append(img_link)
19    mylist = img
20    return mylist
```



```
1 https://epic.gsfc.nasa.gov/archive/natural/2018/06/
14/png/epic_1b_20180614015122.png
```



```
1 def fetchEPICImage(get_image, date_fom):
2     URL_EPIC = "https://epic.gsfc.nasa.gov/archive/natural/"
3     URL_EPIC = URL_EPIC + date_fom
4     URL_EPIC = URL_EPIC + '/png/'
5     URL_EPIC = URL_EPIC + get_image + '.png'
6     return URL_EPIC
```



THANK

YOU

