In [1]:

```
# I started off by reviewing the Python API Tutorial:
# https://www.dataquest.io/blog/python-api-tutorial/
```

In [2]:

```
# This is the initial library required to work with the API requests
import requests as requests
```

In [3]:

```
# This code sets up the information that will be passed to the Google API
# For the Google API part go here: https://console.developers.google.com/apis/
# 1) Sign up 2) Create a project 2) Get a key!
# Note that in the free version you are limited to a max size of 640x640
# Help documentation here: https://developers.google.com/maps/documentation/javascript/?hl=en_US
GOOGLE_MAPS_API_URL = 'https://maps.googleapis.com/maps/api/staticmap'
params = {
    "address":"40 Heerengracht St, Cape Town City Centre, Cape Town, 8000, South Africa",
    "center":"5 Nereide St, Paarden Eiland, Cape Town, 7405",
    "zoom":"13",
    "size":"640x640",
    "maptype":"satellite",
    "key":"xxx"
}
```

In [4]:

```
# And this code makes the actual request, using the variable defined above
req = requests.get(GOOGLE_MAPS_API_URL, params=params)
```

In [5]:

```
# We can check if the response was positive
# 4* and 5* responses are negative!
# More info on errors if required: https://developers.google.com/maps/documentation/static-maps/error-messages
print(req)
```

```
<Response [200]>
```

In [6]:

```
# We can also look at the headers to see what information was returned
req.headers
```

Out[6]:

```
{'Content-Type': 'image/png', 'Date': 'Tue, 20 Mar 2018 03:24:45 GMT', 'Expires': 'Wed, 21 Mar 2018 03:24:45 GMT', 'Cache-Control': 'public, max-age=86400', 'Var
y': 'Accept-Language', 'Access-Control-Allow-Origin': '*', 'Server': 'staticmap', 'Content-Length': '267207', 'X-XSS-Protection': '1; mode=block', 'X-Frame-Option
s': 'SAMEORIGIN', 'Alt-Svc': 'hq=":443"; ma=2592000; quic=51303431; quic=51303339; quic=51303335,quic=":443"; ma=2592000; v="41,39,35"'}
```

In [7]:

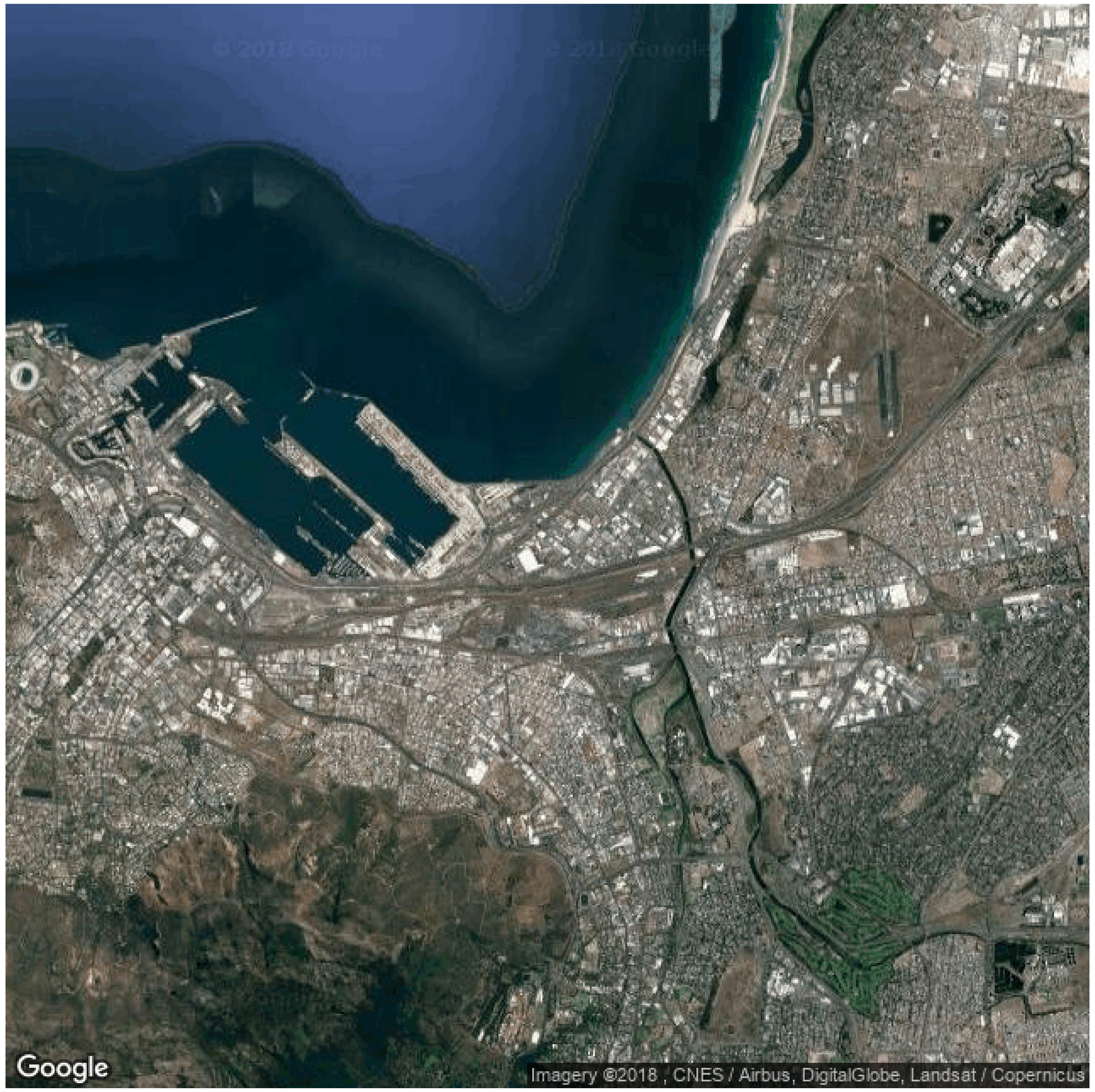
```
# The image is returned as binary response content -
# see http://docs.python-requests.org/en/master/user/quickstart/#binary-response-content
# We need some additional libraries to deal with this in order to actually GET the image to display
from PIL import Image
from io import BytesIO
```

In [8]:

```
my_map = Image.open(BytesIO(req.content))
```

```
In [9]: my_map
```

Out[9]:


© 2018 Google
© 2018 Google
Google
Imagery ©2018 , CNES / Airbus, DigitalGlobe, Landsat / Copernicus