



## **International Space Station: Astronauts Info and Live location updates**

**GDG 2 Credit Course**

**SUBMITTED BY:**

RISHABH 17BCI0137

## **ABSTRACT**

The project is about finding and showing the location of ISS and displaying the people living on it based on the data provided in the API's.

## **INTRODUCTION OF THE PROJECT**

In this project, I will show the live location updates of International Space Station on the world map visually by using a station gif file and printing the longitudes and latitudes in the terminal. I will also list out the people living on International Space Station; all of this will be implemented using Python.

## **METHODOLOGY**

### BASIC CONCEPT

The goal of this project is to show the location of ISS overhead and astronauts information currently residing in ISS. The same will be done using the JSON API's provided, namely:

- API for ISS overhead information: <http://api.open-notify.org/iss-now.json>
- API for astronaut information: <http://api.open-notify.org/astros.json>

Both of these can be easily be used in python. Turtle, Python Library, helps in mapping the location of ISS overhead on the world map.

## CODE

```
3 import json
4 import urllib.request
5 import turtle
6 import time
7
8 # http://open-notify.org/Open-Notify-API/
9 url = 'http://api.open-notify.org/astros.json'
10 response = urllib.request.urlopen(url)
11 result = json.loads(response.read())
12 print('People in Space: ', result['number'])
13 people = result['people']
14 for i in people:
15     print(i['name'], ' in ', i['craft'])
16
17 screen = turtle.Screen()
18 screen.setup(720, 360)
19 screen.setworldcoordinates(-180, -90, 180, 90)
20 screen.register_shape('station.gif')
21
22 iss = turtle.Turtle()
23 iss.shape('station.gif')
24 iss.setheading(90)
25 iss.penup()
26
27 while(True):
28     url = 'http://api.open-notify.org/iss-now.json'
29     response = urllib.request.urlopen(url)
30     result = json.loads(response.read())
31
32     location = result['ISS_location']
33     lat = location['latitude']
34     lon = location['longitude']
35     print('Latitude: ', lat)
36     print('Longitude: ', lon)
37
38     # map.jpg: http://visibleearth.nasa.gov/view.php?id=57752 Credit: NASA
39     screen.bgpic('worldmap.gif')
40
41
42     iss.goto(float(lon), float(lat))
43     iss.pendown()
44
45     time.sleep(5)
46 turtle.done()
```

## IMPLEMENTATION AND STEP-BY-STEP EXECUTION

```
3 import json
4 import urllib.request
5 import turtle
6 import time
```

The required libraries are imported.

```
8 # http://open-notify.org/Open-Notify-API/
9 url = 'http://api.open-notify.org/astros.json'
10 response = urllib.request.urlopen(url)
11 result = json.loads(response.read())
12 print('People in Space: ', result['number'])
13 people = result['people']
14 for i in people:
15     print(i['name'], ' in ', i['craft'])
```

Line 9, is the URL link for the astronauts information. In line 10, valid JSON object is stored in response variable. In line 10, the object is converted into python dictionary and stored in result. From line 12 to line 15, we print the astronaut's information obtained.

```
17 screen = turtle.Screen()
18 screen.setup(720, 360)
19 screen.setworldcoordinates(-180, -90, 180, 90)
20 screen.register_shape('station.gif')
21
22 iss = turtle.Turtle()
23 iss.shape('station.gif')
24 iss.setheading(90)
25 iss.penup()
```

In this part of the code, we set up turtle to map the ISS location and put up station gif to mark it visually on the world map.

```

27 while(True):
28     url = 'http://api.open-notify.org/iss-now.json'
29     response = urllib.request.urlopen(url)
30     result = json.loads(response.read())
31
32     location = result['ISS_location']
33     lat = location['latitude']
34     lon = location['longitude']
35     print('Latitude: ', lat)
36     print('Longitude: ', lon)
37
38     # map.jpg: http://visibleearth.nasa.gov/view.php?id=57752 Credit: NASA
39     screen.bgpic('worldmap.gif')
40
41
42     iss.goto(float(lon), float(lat))
43     iss.pendown()
44
45     time.sleep(5)
46 turtle.done()

```

Line 28, is the URL link for the ISS location. In line 29, valid JSON object is stored in response variable. In line 30, the object is converted into python dictionary and stored in result. Lat and Lon are used to store the co-ordinates, printed in line 35 and line 36. World map background picture is added in line 39. From line 42 to 45, live location of overhead is shown in the world map. Line 46, is self-explanatory.

### API's

API for ISS overhead information: <http://api.open-notify.org/iss-now.json>

```

{"timestamp": 1552989407, "iss_position":
{"latitude": "-26.6080", "longitude": "98.2906"},
"message": "success"}

```

API for astronaut information: <http://api.open-notify.org/astros.json>

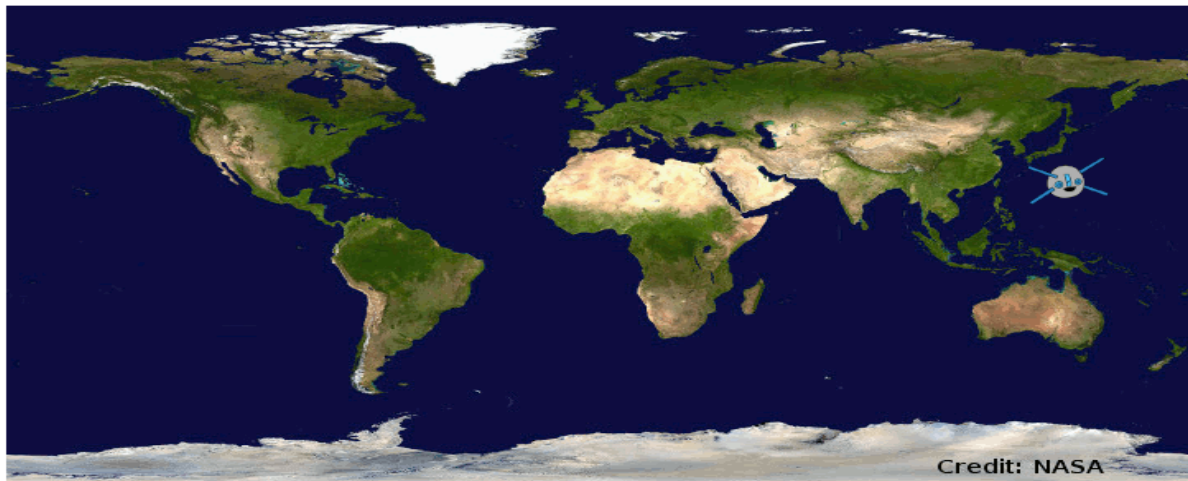
```

{"message": "success", "number": 6, "people": [{"craft": "ISS",
"name": "Oleg Kononenko"}, {"craft": "ISS", "name": "David Saint-
Jacques"}, {"craft": "ISS", "name": "Anne McClain"}, {"craft":
"ISS", "name": "Alexey Ovchinin"}, {"craft": "ISS", "name": "Nick
Hague"}, {"craft": "ISS", "name": "Christina Koch"}]}

```

## RESULT

- I am putting only one image because there's no significant change in the image for live update, as the shift is in decimal points. (Figure 1)



**Figure 1**

- This is the expected result i.e. names of the astronauts and live updates of the INTERNATIONAL SPACE STATION. (Figure 2)

```
Rishabh:iss TheBat$ python3 ./iss.py
People in Space: 6
Oleg Kononenko in ISS
David Saint-Jacques in ISS
Anne McClain in ISS
Alexey Ovchinin in ISS
Nick Hague in ISS
Christina Koch in ISS
Latitude: 12.3470
Longitude: 81.2393
Latitude: 12.6476
Longitude: 81.4676
Latitude: 12.9479
Longitude: 81.6964
Latitude: 13.2231
Longitude: 81.9068
Latitude: 13.5230
Longitude: 82.1369
Latitude: 13.7977
Longitude: 82.3484
Latitude: 14.0972
Longitude: 82.5798
```

**Figure 2**

## DIFFICULTIES FACED

- New to python so took some time to understand everything.

## CONCLUSION

The aim of the project has been successfully achieved. I am able to get the live updates and names of the astronauts living in the space station, the code has been implemented in python as per the requirements.

## REFERENCE

[1] <https://matplotlib.org/basemap/users/examples.html>

[2] <https://www.geeksforgeeks.org/turtle-programming-python/>