

```
# Import necessary packages
import sys
import requests
import re
import pandas as pd

get_ipython().run_line_magic('matplotlib', 'inline')
import matplotlib.pyplot as plt
import numpy as np

import urllib
import json
import urllib.request, urllib.parse, urllib.error
import ssl

# Get data from website
url = "https://skyscanner-skyscanner-flight-search-v1.p.rapidapi.com/apiservices/brows

headers = {
    'x-rapidapi-key': "f61128c512msh40ead389b17dbdfp16f199jsn29c5ca7ec2a5",
    'x-rapidapi-host': "skyscanner-skyscanner-flight-search-v1.p.rapidapi.com"
}

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

# make sure we got it

# print(data.text)

# Something smart u did that I didn't look into

ctx = ssl.create_default_context()
ctx.check_hostname = False
ctx.verify_mode = ssl.CERT_NONE

# Load the JSON
try:
    js = json.loads(data.text)
except Exception as err:
    print(f"error: {err}")
    sys.exit(-1)

# Here is how you can look at your keys
print(js.keys())

#Here is how you retrieve data from a spacific key
print(js['Quotes'])
```

```
27, 'DestinationId': 48018, 'DepartureDate': '2021-01-13T00:00:00'}, 'QuoteDateTi
```

```
quotes = js['Quotes']
```

```
flight_info = {0:[0,0,0],1:[0,0,0],2:[0,0,0],3:[0,0,0],4:[0,0,0],5:[0,0,0],6:[0,0,0]}
# {day:[dir,indir,daycount],...}
# 1= friday
# 2= sat
# 3= sun
# 4= mon
# 5= tues
# 6= wed
# 0= thurs
```

```
total_flights = 0
```

```
for q in quotes:
```

```
    print(q['MinPrice'])
```

```
    total_flights +=1
```

```
    # update total flight count for that day
```

```
    dt = q['QuoteDateTime']
```

```
    d = dt.split(':')
```

```
    day = (int(d[-2]))%7
```

```
    flight_info[day][2] +=1
```

```
    # Add count to correct day for dir or undir
```

```
    if q['Direct'] == 'true':
```

```
        flight_info[day][0] = flight_info[day][0] + (q['MinPrice']-flight_info[day][0])/1
```

```
    else:
```

```
        flight_info[day][1] = flight_info[day][1] + (q['MinPrice']-flight_info[day][1])/1
```

```
    # print(q.keys())
```

```
print('total flights: ', total_flights)
```

```
print(flight_info)
```

```
591
```

```
593
```

```
593
```

```
595
```

```
596
```

```
596
```

```
597
```

```
606
```

```
608
```

```

610
total flights: 10
{0: [0, 0, 0], 1: [0, 594.5, 2], 2: [0, 599.5, 2], 3: [0, 595.0, 1], 4: [0, 608.0, 1], 5: [0, 608.0, 1], 6: [0, 608.0, 1], 7: [0, 608.0, 1], 8: [0, 608.0, 1], 9: [0, 608.0, 1]}

# Plot averages
import matplotlib.patches as mpatches

ys = np.arange(7)
x_dir = []
x_undir = []

for y in ys:
    x_dir.append(flight_info[y][0])
    x_undir.append(flight_info[y][1])

width = .35

ax = plt.subplot(111)
ax.bar(ys - width/2, x_dir, width, color='g', align='center')
ax.bar(ys + width/2, x_undir, width, color='b', align='center')

green = mpatches.Patch(color='g', label='direct flights')
plt.legend(handles=[green])

blue = mpatches.Patch(color='b', label='indirect flights')
plt.legend(handles=[blue])

plt.xticks(ticks = np.arange(7), labels = ['Thurs', 'Fri', 'Sat', 'Sun', 'Mon', 'Tues', 'Wed'])
plt.xlabel("Day of the week")
plt.ylabel("Avg Price (USD)")
plt.ylim(0,700)
plt.show()

```



