

## Python 深度学习基础第一次作业

### ● 功能

实现了从 DarkskyAPI 提供的天气预测 API 获取合肥（任意经纬度地区）接下来一周的天气情况，制成天气预测图，包括最高温曲线，最低温曲线，和湿度曲线，并展示在窗口中（已上传至 <https://github.com/milier00/Python>）

### ● 说明

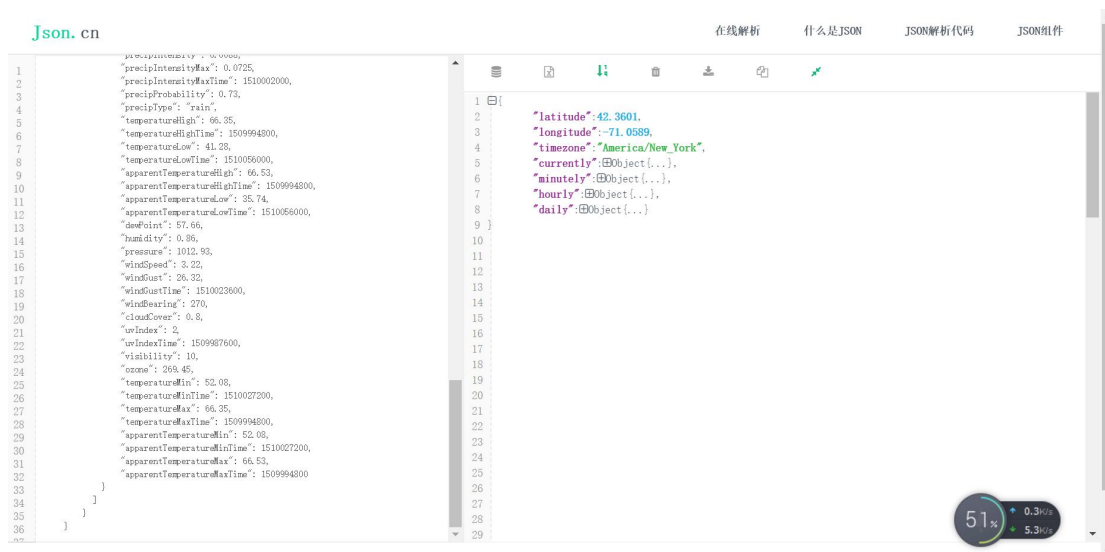
一共两个 py 文件，weather.py 中包含了实现从 API 获取数据、绘图、保存图片功能的函数；而 mian.py 实现了 GUI 交互，将得到的图显示在对话框中。mian.py 中设计了窗口的类 class win(QDialog)，并且 from weather import next\_week\_weather。

### ● 使用

运行程序即可。

### ● 实现

➤ **数据获取：**我们设计了两种获取数据的方式，一种是本地，判断条件为 remote=False，另一种是从 API 下载，判断条件为 remote=True。采取第二种。查阅合肥经纬度，从 DarkskyAPI 获取该经纬度的下周天气预测数据并下载。发现这是一个 json，为清楚地观察其结构，用 online editor 解构 json，发现它有现在的、每分钟的、每小时的、每天的天气数据，



点开我们需要的 daily，发现更多数据种类，

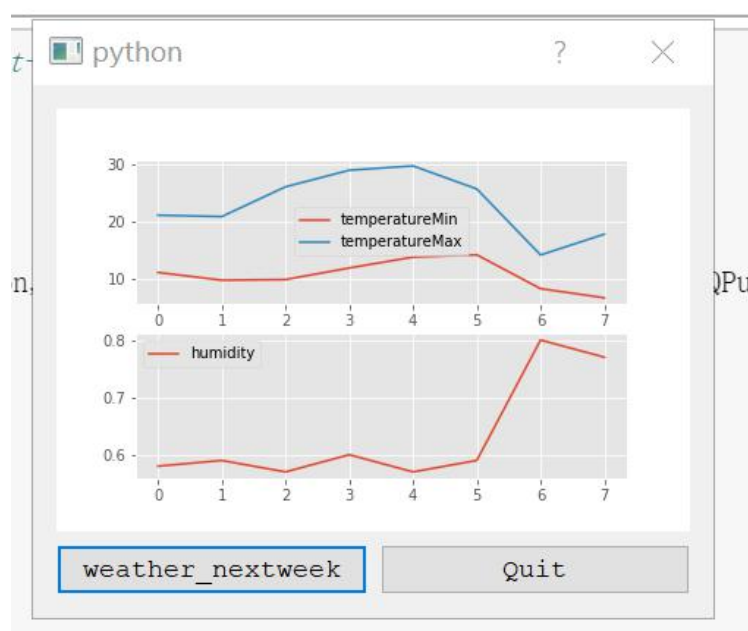
```

10  "icon": "rain",
11  "data": {
12    "time": 1509944000,
13    "summary": "Rain starting in the afternoon, continuing until evening.",
14    "icon": "rain",
15    "sunriseTime": 1509967519,
16    "sunsetTime": 1510003992,
17    "moonPhase": 0.59,
18    "precipIntensity": 0.0088,
19    "precipIntensityMax": 0.0725,
20    "precipIntensityMaxTime": 1510002000,
21    "precipProbability": 0.73,
22    "precipType": "rain",
23    "temperatureHigh": 66.35,
24    "temperatureHighTime": 1509994800,
25    "temperatureLow": 41.28,
26    "temperatureLowTime": 1510056000,
27    "apparentTemperatureHigh": 66.53,
28    "apparentTemperatureHighTime": 1509994800,
29    "apparentTemperatureLow": 35.74,
30    "apparentTemperatureLowTime": 1510056000,
31    "dewPoint": 57.66,
32    "humidity": 0.86,
33    "pressure": 1012.93,
34    "windSpeed": 3.22,
35    "windGust": 26.32,
36    "windGustTime": 1510023600,
37    "windBearing": 270,
38    "cloudCover": 0.8,
39    "uvIndex": 2,
40    "uvIndexTime": 1509987600,
41    "visibility": 10,
42    "ozone": 269.45,
43    "temperatureMin": 52.08,
44    "temperatureMinTime": 1510027200,
45    "temperatureMax": 66.35,
46    "temperatureMaxTime": 1509994800,
47    "apparentTemperatureMin": 52.08,
48    "apparentTemperatureMinTime": 1510027200,
49    "apparentTemperatureMax": 66.53,
50    "apparentTemperatureMaxTime": 1509994800
51  }
52 }
53 }
54 }

```

所以必须在选取数据时提取特定的 **attributes**。首先要得到以天为单位的，设计了函数 **daily\_data\_of\_attributes**；再得到我们需要的每天的气温和湿度数据，设计了 **get\_daily\_data** 函数来实现。

- **绘图：**我们将从 API 获取的下周七天每天的最高温、最低温和湿度数据用 **matplotlib.pyplot** 绘成折线图，并下载到本地保存 **plt.savefig('weather\_nextweek.png')**。将上述功能封装为函数 **next\_week\_weather**，在主函数中只需 **import next\_week\_weather**。
- **交互：**设计类 **class win(QDialog)** 实现窗口设计，包括两个按钮 **weather\_nextweek** 和 **Quit**，分别用来显示天气预测图和退出。



## ● 未实现的功能

- 可在下拉框中选择城市或输入经纬度，得到该地点下周的天气预报；
- 可自主选择天气预报包括的内容，除了温度湿度外，还有日出日落时间、月相、风速等，并用可视化图文给出结果；
- 最终生成个性化的某地下周天气预报图，包含用户选择的天气选项和用户输入的文本信息。该图可以用来发送给亲友，提醒 ta 下周所在地点的天气情况。

## ● 困难

- 第一次使用 API，如何从 json 中提取需要的数据是难题，不熟悉语法消耗时间；
- 在获取数据时没有完整看完 API 文档，结果获得的数据是默认华氏温度为单位的，从意识到这个问题到试图 DIY 解决它，再到文档中介绍了在 url 后加单位选项 `units=si` 就可以获得摄氏温度，明白了 API 文档要仔细看的道理；

```
url = "https://api.darksky.net/forecast/%s/%s,%s?units=si" % (api_key, lat, long)
```

- 试图实现上述“未实现的功能”的过程很曲折，中途想用 Qt Designer，尝试失败，最后不断删减到现在只有两个按钮的情况。。。