Final Project Data Checkpoint

Minjie Zhao

Project code:

https://github.com/Minjiezz/507final

Data sources

I am doing an script for people to searching the current weather and weather forecast.

My first data source:

My first data source is an Api that required an Api key to get access to the current weather by entering the city name.

```
city_name = input(f"Which city do you want to search for weather?")
API_key_current_weather = "d2e179d7f6dca72d1c5bcd4c1fb69d96"
search_by_city_name_url =
f"https://api.openweathermap.org/data/2.5/weather?q={city_name}&appid={API_key_current_weather}"
weather_response = requests.get(search_by_city_name_url)
current_weather = json.loads(weather_response.text)

print(current_weather)

city_weather = json.dumps(current_weather, indent=4)
city_weather_file = open("city_weather.json", "w")
city_weather_file.write(city_weather)
city_weather_file.close()
```

The Api can get the weather information of more than 20000 cities all over the world. For example, the data of Los Angeles looks like this:

```
{
  "coord": {
    "lon": -118.2437,
    "lat": 34.0522
  },
  "weather": [
    {
      "id": 800,
      "main": "Clear",
      "description": "clear sky",
      "icon": "01d"
    }
  ],
  "base": "stations",
  "main": {
    "temp": 294.25,
    "feels_like": 294.01,
    "temp_min": 289.59,
    "temp_max": 299.83,
    "pressure": 1014,
    "humidity": 61
  },
  "visibility": 10000,
  "wind": {
    "speed": 5.14,
    "deg": 230
  },
  "clouds": {
    "all": 0
  },
  "dt": 1651100524,
  "sys": {
    "type": 1,
    "id": 3694,
    "country": "US",
    "sunrise": 1651064860,
    "sunset": 1651113193
  },
  "timezone": -25200,
  "id": 5368361,
  "name": "Los Angeles",
  "cod": 200
}
```

My second data source:
My second data source is also an Api that required an Api key to get access to the weather forecast of the following 16 days.

```
cnt = input(f"How many days weather forecast do you want to search? (up to 16 days)")
API_key_weather_forecast = "d2e179d7f6dca72d1c5bcd4c1fb69d96"
search_weather_forecast_url =
f"https://api.openweathermap.org/data/2.5/forecast/daily?q={city_name}&cnt={cnt}&appid
={API_key_weather_forecast}"
forecast_response = requests.get(search_weather_forecast_url)
weather_forecast = json.loads(forecast_response.text)

print(weather_forecast)

city_weather_forecast = json.dumps(weather_forecast, indent=4)
city_weather_forecast_file = open("city_weather_forecast.json", "w")
city_weather_forecast_file.write(city_weather_forecast)
city_weather_forecast_file.close()
```

Here is an example of a part of the the weather forecast for the Los Angeles for tomorrow:

```
 {
            "dt": 1651086000,
            "sunrise": 1651064860,
            "sunset": 1651113193,
            "temp": {
                "day": 294.14,
                "min": 288.01,
                "max": 294.6,
                "night": 289.05,
                "eve": 293.88,
                "morn": 288.01
            },
            "feels_like": {
                "day": 293.55,
                "night": 288.52,
                "eve": 293.42,
                "morn": 287.56
            },
            "pressure": 1014,
            "humidity": 48,
            "weather": [
                {
                    "id": 800,
                    "main": "Clear",
                    "description": "sky is clear",
                    "icon": "01d"
                }
            ],
            "speed": 4.54,
            "deg": 176,
            "gust": 3.84,
            "clouds": 5,
            "pop": 0
        },
```

Caching:
I will modify the code form class and use it.

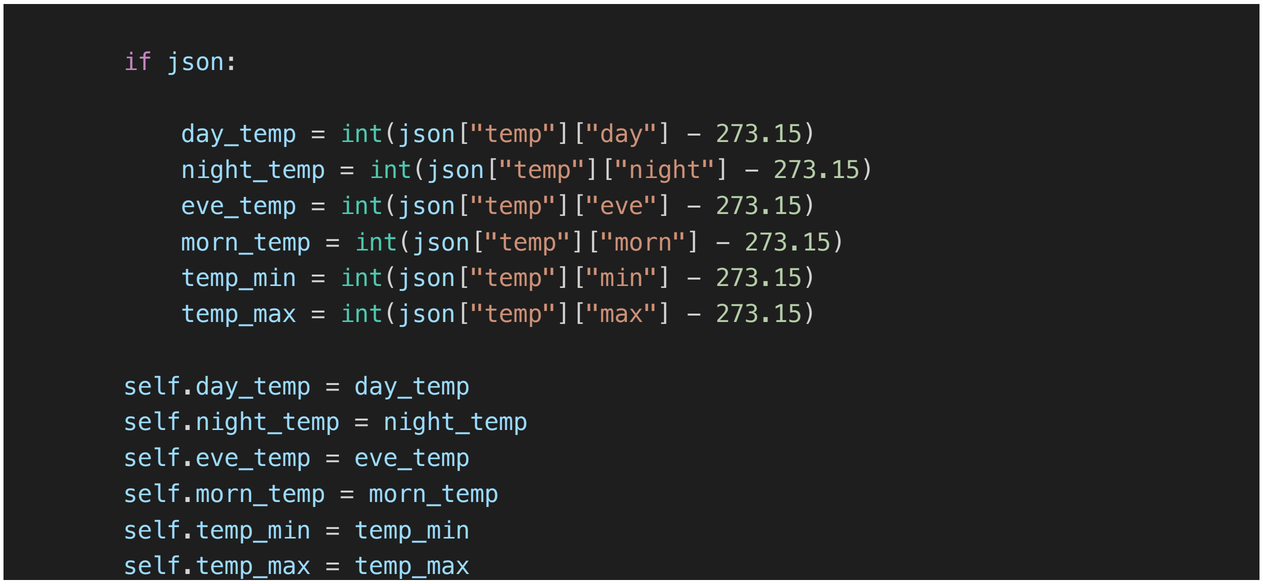
```
def make_url_request_using_cache(url, cache):
    if (url in cache.keys()): # the url is our unique key
        print("Using cache")
        return cache[url]
    else:
        print("Fetching")
        time.sleep(1)
        response = requests.get(url, headers=headers)
        cache[url] = response.text
        save_cache(cache)
        return cache[url]
```

Data Structure

I am planning to organize those two data sources by using class, these are the two classed that I am going to use:

```
class Weather:

    def __init__(self, temp="", condition="", description="", feels_like="",
temp_min='', temp_max='', pressure='', humidity='', windspeed= "", sunrise="",
sunset="", json=None):

        if json:

            condition = json["weather"][0]['main']
            description = json["weather"][0]['description']
            temp = int(json["main"]["temp"] - 273.15)
            feels_like = int(json["main"]["feels_like"] - 273.15)
            temp_min = int(json["main"]["temp_min"] - 273.15)
            temp_max = int(json["main"]["temp_max"] - 273.15)
            pressure = int(json["main"]["pressure"])
            humidity = int(json["main"]["humidity"])
            windspeed = int(json["wind"]["speed"])
            sunrise = time.strftime("%I:%M:%S",
time.gmtime(int(json["sys"]["sunrise"]) - 14400))
            sunset = time.strftime("%I:%M:%S", time.gmtime(int(json["sys"]["sunset"])
- 14400))

        self.condition = condition
        self.description = description
        self.temp = temp
        self.feels_like = feels_like
        self.temp_min = temp_min
        self.temp_max = temp_max
        self.pressure = pressure
        self.humidity = humidity
        self.windspeed = windspeed
        self.sunrise = sunrise
        self.sunset = sunset
```

```
class Forecast:

    def __init__(self, day_temp="", night_temp="", eve_temp="", morn_temp="",
temp_min='', temp_max='', json=None):
```



```
        if json:

            day_temp = int(json["temp"]["day"] - 273.15)
            night_temp = int(json["temp"]["night"] - 273.15)
            eve_temp = int(json["temp"]["eve"] - 273.15)
            morn_temp = int(json["temp"]["morn"] - 273.15)
            temp_min = int(json["temp"]["min"] - 273.15)
            temp_max = int(json["temp"]["max"] - 273.15)

        self.day_temp = day_temp
        self.night_temp = night_temp
        self.eve_temp = eve_temp
        self.morn_temp = morn_temp
        self.temp_min = temp_min
        self.temp_max = temp_max
```

I will use them to sort the information for the users(based on the input from the users).

Interaction and Presentation Plans

I plan to use commands line as my interaction technique, and the users will enter the input of the following questions to get the current weather information and the weather forecasts for next week.

These are the questions I plan to ask the users to ask:

- Which city do you want to search for weather?

- Do you want to know other information of the weather?

- Do you want to search the weather forecast for next week of the city?

- ("Do you want to show the information in detail or see the temperature change?",
  ("Show all the information for next week"),
  ("Show the day temperature changes for the week in one line"))