

Weather API Task

The NOAA National Weather Service provides a convenient web API for retrieving weather forecast data for locations in the USA. In this task, you will write python code to access the API to download a forecast for a given site and plot it on a graph. Additionally, you are asked to wrap your code in a script that can take input from the command line and run for any valid location in the USA.

The task

Use the NOAA National Weather Service in the US as your data source. The service has a web API that allows you to request forecast data for a given grid point in the USA. Details of the API are documented at:

<https://www.weather.gov/documentation/services-web-api>

Use the endpoint <https://api.weather.gov/> as the base URL.

The stages of your task are:

1. For a US location of your choice, use the API to identify an appropriate grid box. This will provide you with a response that includes values for the following properties: "gridId", "gridX", and "gridY".
2. Use the values obtained in (1) to make a request to the weather API for the most recent weather forecast. **Tip:** you can get the required URL from the "forecastGridData" property of the response.
3. Extract the "minTemperature" and "maxTemperature" components from the API response (which is provided in JSON format).
4. Calculate the mean temperature time series from the minimum and maximum temperatures.
5. Plot and annotate a line graph of the minimum, mean and maximum temperatures against the date/time on the X-axis.
6. Having completed the task, refactor your code so that it can be called using a command-line script that takes the longitude and latitude values as inputs. The script should carry out steps (1) to (5) for the chosen location.

You should write your solution in **python**.

Presenting your solution

Your interview will begin with a 15-minute session in which you will be asked to share your screen and present your solution to the weather API task. You will be asked to show the code, talk through the details, and demonstrate that it works correctly. Please be prepared to show your code in an appropriate editor.

Please pay attention to the following factors that we are looking for in your solution:

- write clean, modular, well-described code
- appropriate use of variables, functions and/or classes
- use of imported libraries to do complex tasks (thereby minimising the amount of code required)
- error-handling to deal with invalid inputs
- anything else you think would be useful or helpful