Assignment 2: Server-Side Scripting using Python, JSON and Tomorrow.io API

Grading Guidelines

Graders will test your homework on Firefox and Chrome.

Grading Guidelines:

Search Form (2.5 points in total)

- The initial web page with proper fields in the search form. (**1 point**)
  - "Weather Search" header with background, "Fill out the form to get weather info" sub header.
  - State should be a drop-down list consisting of all state names.
  - Check here checkbox and two buttons i.e., Submit and Clear.
- Validation for the search inputs and fields disabled when **Check here** checkbox is checked, the error message(s) should indicate if fields are missing. **(0.5 point**)
  - If there is no error message for invalid input, 0.5 point will be deducted.
- The "Clear" button resets the search form to the initial state and clears the results area. (**1 point)**
  - For each incorrectly reset or cleared field, 0.25 point will be deducted, for a maximum deduction of 0.5 point.

Search Result Page (4 points)

Show weather information retrieved from API service in a card view format as described in the homework description. **(2 points)**

- For each missing data in the card, 0.25 points will be deducted with a maximum of 1 point.
- For incorrect mismatch of data, 0.25 point will be deducted
- If On hovering has no shadow effect, 0.25 points will be deducted.

Show the forecast weather information retrieved from the API service in a tabular format as described in the homework description. (**2 points**)

- For each missing column in the table, 0.25 points will be deducted with a maximum of 0.75 points.
- For each incorrect column value or column order mismatch, 0.25 points will be deducted with a maximum of 0.5 point.

- For each incorrect format of date & status in the row, 0.25 points will be deducted.
- On clicking the record, it should direct to that detailed page with the right date else 0.25 points will be deducted.

Detailed Summary of Weather Page (3.5 points)

Show the information retrieved from the API service in a card format as described in the homework description and in referral video. (**1 point**)

- For each missing data in the card, 0.25 points will be deducted with a maximum of 0.5 point.
- Missing the unit for the corresponding value, 0.25 points will be deducted.
- The icon should be present in the card, 0.25 points will be deducted.
- The styling of the card should be similar, 0.25 points will be deducted.

Show the arrow toggle button for the expansion and to hide the Weather Charts (vice versa). **(0.5 points)**

- On clicking the arrow, the chart should be displayed, and the arrow icon should toggle and vice versa, 0.25 point will be deducted.
- For any missing content on expansion or hide, 0.25 will be deducted.

Temperature Range (Min, Max) Weather Chart should be as described in the description and in the referral video. **(1 point)**

- The chart should display the weather on a day basis, 0.5 points will be deducted.
- The chart should have the corresponding label on the axis, 0.25 points will be deducted.

Hourly Weather (For Next 5 days) Weather Chart should be as described in the description and in the referral video. **(1 point)**

- The chart should display the weather on an hourly basis, 0.5 points will be deducted.
- The chart should have the corresponding label on the axis, 0.25 points will be deducted.

IMPORTANT

- You may call Google Geocoding API and HighCharts API from JavaScript
- You should not call Tomorrow.io API directly from JavaScript. Doing this will result in a **4 points** penalty.
- You may call HighCharts API from JavaScript

- If either React, Angular, or Bootstrap are used, **4 points** will be deducted.
- If Python templates are used, **4 points** will be deducted.
- JQuery may be used, with no penalty.
- You should implement the Assignment using Python.
- You should deploy Assignment 2 on AWS/GCP/Azure cloud server as described in the **Cloud Set up (Python)** documentation. Otherwise, **3 points** will be deducted. This will be verified with an additional link in the Table of Assignments, showing a sample API call to your "cloud" service and verifying the corresponding JSON result.
- Use of code downloaded from GitHub for any component of the app front-end will result in a **5-point** penalty.
- Use of Python "templates" will result in a 5-point penalty.

- **Source code submission**

  Students should submit their source code electronically on D2L Brightspace as a single ZIP file containing all front-end and back-end code and any additional files. In addition, on your course homework page, you **must** update the Assignment 2 link to refer to your new home page in the cloud.