Assignment 2: Utilizing External API Services

OBJECTIVE

Students will use their JavaScript and/or jQuery knowledge to create a dynamic web page to display data obtained from two different API services. Students must develop a website that utilizes on-demand API calls to a currency exchange rate API service and a weather API service.

TASKS

1. Display the current weather information (20 marks)

You are required to create a web page and retrieve the current local weather using a free online service. There are several free weather service providers available; it's up to you to find one, study its API, figure out what endpoint to use, and write the JavaScript code to retrieve the current weather for Calgary. You should retrieve and display at least the temperature and a description of the weather (e.g. partially cloudy, sunny, snowing, etc) and a timestamp. Remember: do not push your API key to a public repository! (add .env to your .gitignore file if pushing to GitHub).

- The temperature is correctly displayed for a selected location. (5 marks)
- The weather description and timestamp are correctly displayed. (5 marks)
- CSS style is properly applied to your weather information. (5 marks)
- The API key is protected by storing it in a .env file. (2.5 marks)
- The code is well-documented, commented, best practices are applied, and the code is organized. (2.5 marks)

2. Create a currency exchange service (20 marks)

Using any free currency exchange API service (example: https://www.exchangerate-api.com/), create a website where you can enter a determined amount of currency and obtain the converted value in another currency. Remember: do not push your API key to a public repository! (add .env to your .gitignore file if pushing to GitHub).

- At least 10 different currencies are available to be selected from. (5 marks)
- Users can select “from” currency and “to” currency. (5 marks)
- The exchange rate for the selected operation is visually displayed on screen. Ex.: 1 USD = 1.44 CAD (5 marks)
- The API key is protected by storing it in a .env file. (2.5 marks)

- The code is well-documented, commented, best practices are applied, and the code is organized. (2.5 marks)

3. Data persistency and rate limiting (10 marks)

- Using your knowledge of sessionStorage and localStorage, find a way to store the weather information and the currency exchange rates locally, and retrieve them when you restore the navigation, without needing to send another API request. (5 marks)
- Apply any rate-limiting technique to avoid multiple consecutive API calls. (5 marks)

Grading

If your code does not work, you won't receive marks for that section, even if most of the code is correct. **Ensure that you correctly link your scripts and test before submitting.**

If the wrong exchange rate or the wrong result is displayed: -10 marks.
If the weather for the wrong location is displayed: -10 marks.
If data cannot be updated after being stored locally: -10 marks. (You should still be able to fetch data after storing the values locally after a reasonable time has elapsed!)

Your assignment will be graded from a total of 50 marks.

Submission

Submit your assignment folder inside of a .zip file to Assignment 2's Dropbox. Make sure to include all necessary files.

Alternatively, you can upload your assignment folder to GitHub and share the repository link in D2L's Dropbox. Make sure its visibility is set to public.