

Programming with JavaScript – QAP2

Course Name: Programming with JavaScript

Current Week: (2025/06/02)

Introduction:

The purpose of this assessment is to help us understand how the class is doing in terms of the review material that we have covered during the previous couple of weeks. The only purpose of this assessment is for us to improve our approach to review and ensure that what we're currently doing is an effective strategy. Completion of this assessment is mandatory - if you don't submit a solution, it will be marked as incomplete. If you do submit a solution, it will be marked as complete, as you will receive full marks.

Again, the goal here is to help you all in the best way that we can, so please do be honest when answering the questions related to how long it took, which resources you used, etc. And please ensure that you do your own work – don't just copy off a friend to get it done, earnestly do your best with it. If you can't get it completely working, give us what you have. While it will be graded, the grade will not count against you, it's just a way for us to see where everybody is, and to know which concepts, if any, we, as a class, may be struggling with.

Deadline: You will have until the end of the day on **Saturday June 14, 2025 (4:00pm)** to submit your assessment solutions. Please ensure you answer all the questions outlined in the instructions portion of this document as well in your submission.

Instructions: Your name: _____

You are allowed to complete the assessment problems below in whatever way you can but please answer the following questions/points as part of your submission:

1. How many hours did it take you to complete this assessment? (Please keep try to keep track of how many hours you have spent working on each individual part of this assessment as best you can - an estimation is fine; we just want a rough idea.)

Answer: [your answers should be formatted like this]...

2. What online resources you have used? (My lectures, YouTube, Stack overflow etc.)
3. Did you need to ask any of your friends in solving the problems. (If yes, please mention name of the friend. They must be amongst your class fellows.)
4. Did you need to ask questions to any of your instructors? If so, how many questions did you ask (or how many help sessions did you require)?
5. Rate (subjectively) the difficulty of Making this all! from your own perspective, and whether you feel confident that you can solve a similar but different problem requiring some of the same techniques in the future now that you've completed this one.

Objective:

Strings are a fundamental element of any programming language, and JavaScript is no different. To become proficient in any programming language you encounter in the future, it's crucial to be comfortable with strings and regular expressions. This QAP is designed to provide you with practical experience and mastery in working with strings and regular expressions. Utilize regular expressions effectively to tackle various questions and challenges.

We have covered numerous string functions in our sessions, and there is a video recording of my lecture on regular expressions available - <https://youtu.be/zSDQ0HAiLq4> - (please watch it if you haven't already 😊).

Additionally, Vanessa conducts TA sessions twice a week, providing great support. You can join these sessions on Mondays and Wednesdays to ask any questions you may have. Moreover, while working on this QAP, it is recommended that after submission, you contact Vanessa to discuss the problems you found most challenging and get some hands-on assistance with the given questions. If you still feel the need, Dr. Qurat ul ain or myself can offer a session to go over the solutions to the problems.

This assignment also focuses on JavaScript's ability to manipulate the Document Object Model (DOM). You will modify elements dynamically, handle user input, and apply event-driven programming to solve various problems.

You must complete each problem using DOM interactions instead of only returning values from functions. Your submission should include an HTML file, an external JavaScript file, and a CSS file for styling.

Deadline: Submit by Saturday, June 14, 2025 (4:00 PM).

Instructions:

Complete the following tasks using JavaScript and DOM manipulation (**Please refer to the `qap2.js` attached script file for details of the questions – you would want to link that file with the html file and solve your questions there**)

Task 1: Convert String to Lower Snake Case

Create a webpage with an input field where the user enters a string. When they click a button, convert the string into **lower snake case** by replacing spaces, dots, or tabs with underscores and making all letters lowercase. Display the transformed string below the input field.

Task 2: Create a Video Element

Design a webpage where users enter a video URL in a text field and specify the width. When they click a button, a `<video>` element should be generated dynamically below with the provided width. Include a checkbox for users to enable playback controls.

Task 3: Parse a Date String

Provide an input field where users enter a date in the format YYYY-MM-DD. On clicking a button, validate the input, and if correct, convert it into a Date object and display it in a formatted manner (e.g., "January 1, 2025"). If the format is incorrect, show an error message.

Task 4: Convert a Date Object to a String

Add a date picker input element where users select a date. When they click a button, convert the date to the YYYY-MM-DD format and display it below.

Task 5: Parse Geographic Coordinates

Create an input field where users enter coordinates in various formats, such as "42.9755,-77.4369" or "[-77.4369, 42.9755]". On clicking a button, validate and standardize the format to "(lat, lng)" and display it.

Task 6: Format and Display Multiple Coordinates

Allow users to enter multiple geographic coordinates separated by commas. On clicking a button, validate and format them into a structured list (e.g., ((42.9755, -77.4369), (42.9755, -62.1234))).

Task 7: Determine MIME Type from Filename

Create an input field where users enter a filename (e.g., file.jpg, document.pdf). When they click a button, determine and display the MIME type based on the extension.

Task 8: Generate a License Link

Allow users to enter a license code (e.g., CC-BY-NC). On clicking a button, generate and display an anchor (<a>) tag linking to the correct Creative Commons license page.

Task 9: Convert Values to Boolean

Provide an input field where users enter different boolean-like values (Yes, No, 1, 0, true, false). On clicking a button, convert the value into a pure Boolean (true or false) and display the result.

Task 10: Build a URL

Create a form with input fields for a search query, sort order, result count, and license type. When users submit the form, generate a properly formatted API URL and display it.

Submission Instructions:

- Submit all files in a folder named firstName_qap2, zip it, and upload it OR share a GitHub repository link.
- Ensure all features work properly and provide a readme file with a brief explanation.

Enjoy coding!