

A4: Individual Assignment 4

Summary

Assignment goals

This assignment tests your knowledge in the following aspects:

- Using JavaScript to read and display content, and dynamically update HTML content on the page with user interaction.
- Implement a responsive layout based on provided design using CSS Grids.
- Ensuring separation of concerns between content, presentation and behaviour layers in web pages.

Assignment overview

A4 requires you to use the starting code provided to implement a flight selectin website.

Note:

The focus of this assignment is JavaScript. Please plan your work and manage your time well in the next weeks to work on this assignment.

Do not modify the HTML that is given to you, if you do you will get a 0 (zero). This is a CSS and JS assignment.

Instructions

Step 1: Download the starter code

Download the starter code using the link below. This gives you all the files and folders, organized as required into sub-folders, etc., that you are required to use to develop a solution for this assignment.

Step 2: Using JS, read content from the JS object and display in main container

Your first main task is to read the flight content provided in the file named *flight-data.js* available in the *js* folder and display it as a collection of elements inside the main container.

Hint: Think about this like you are reading each flight information into a box. Choose an appropriate semantic box, and then use headings, paragraphs, and buttons appropriately to display the content as shown in the reference.

NOTES:

1. The flight content must not be hard coded into the HTML.
2. Do not modify any HTML.
3. Do not modify flight-data.js
4. Update only the CSS file named main.css and JS file named script.js.

Step 3: Using JS, change web page behaviour when the user clicks on the button.

1. Using JS, add an event listener to the button ("select") so that when a user clicks on the button to choose a flight,
 - a. the flight code and destination are retrieved from the correct content information box, and,
 - b. the flight code and destination are written in the aside named "flight selection bag" as a paragraph.
 - c. Basically, you are implementing something like a "shopping cart" for flights in this assignment.
2. Things to keep in mind:
 - a. Remember that you are reading content for each flight using JS.
 - b. Think about how you can dynamically set event listeners to buttons created in this way.
 - c. Use console.log() to verify whether you are reading the correct content at each step of your implementation.

Step 4: Update website appearance using CSS

1. Update the CSS code in main.css
2. Using CSS Grids and Flexbox, make this website responsive and make needed changes in the appearance to make it look the way you see in the pdf.
3. Some style classes are given to you in the starting code that are empty.
 - a. Use them to style the content that is appearing dynamically.
 - b. You might need to add additional styles classes.

- c. You will need to use CSS along with JavaScript.
- 4. Responsive behaviour:
 - a. You will only have 2 views, mobile and desktop. The breakpoint for the mobile view is 840px.
 - b. Make sure that the desktop view appears beyond 840px, and you change some appearance based on the reference.
 - i. The header and footer change a bit in the mobile view.
 - ii. The main content also change.

Step 5 Organize your code correctly and use citations

1. Make sure that you organize your code well.
 - a. Your code must be readable and properly indented.
2. Use comments appropriately.
 - a. Do not use comments for each line of the code.
 - b. Use comments to describe parts of your web page and what each part does.
3. **Citations:**
 - a. If you have learned any concept from content other than those provided in the course, include a citation in both the comments in the code and in the README.md file that is in your submission.

Step 6: Validate your code

1. Validate your CSS file - **fix ALL errors and warnings:**
 - a. Validate `main.css`.
 - b. Validate it via: https://jigsaw.w3.org/css-validator/#validate_by_input
2. Save these validation files as PDF files in this folder named files.
 - a. Name these PDF files with the name of the CSS file name was `main.css`, save the PDF as **W3C-main-css.pdf**

Step 7: Organize your code correctly and use citations

1. If you have created folders, etc., as specified above, your folder structure must look as shown below.
2. Folder structure for this website:

A4

```
|-- www_root
|   |-- README.md
|   |-- index.html
|   |-- img
|   |   |---- <any images that you use, with citations>
|   |-- js
|   |   |---- flight-data.js
|   |   |---- script.js
|   |-- css
```

```
|      |      |---- main.css
|-- files
|      |---- W3C-main-css.pdf
```

Step 8: Save the folder in .ZIP format

1. Compress the folder named **A4** into ZIP format, i.e., to get **A4.zip**
2. Submit **A4.zip** on Brightspace, in the assignment submission drobox.

Marking Rubric

JS behaviour	Meets expectations 10 points	Somewhat meets expectations 5 points	Does not meet expectations yet 0 points	Criterion Score
Read data from flight-data.js using JS and display in HTML dynamically	<p>Content is read from flight-data.js as specified in Step 2.</p> <p>Data is not hardcoded into HTML, it is read using JS and new HTML elements are created using JS to display content to the web page (inside the main container).</p> <p>flight-data.js is not modified either.</p>	<p>Content reading through JS is somewhat implemented.</p> <p>What is implemented does not work reliably.</p> <p>(See feedback for more info)</p>	<p>Work does not meet expectations yet.</p> <p>(See feedback)</p>	/10
Add JS event listener and handler to process attempts to "select" to flights (Part 1 - retrieval of flight info)	Using JS, add an event listener to the button so that when a user clicks on the button to select a flight, the flight code and destination are retrieved from the correct content information box.	<p>JS event management to retrieve content is somewhat implemented.</p> <p>Event may be generated but is not properly handled.</p> <p>(See feedback)</p>	<p>Work does not meet expectations yet.</p> <p>(See feedback)</p>	/10
Add JS event listener and handler to process attempts to "select" to flights (Part 2 - posting flight info to the "flight selection bag")	In the same event handler as used in Part 1 above, the flight code and destination are written in the aside named "flight selection bag" as a paragraph.	<p>JS event management to post content to "flight selection bag" is somewhat implemented.</p> <p>Event may be generated but is not properly handled.</p> <p>(See feedback)</p>	<p>Work does not meet expectations yet.</p> <p>(See feedback)</p>	/10

Content presentation and website appearance	Meets expectations 5 points	Moderately meets expectations 2.5 points	Does not meet expectations yet 0 points	Criterion Score
Overall, organize the web page content	Demonstrates excellent understanding of web page layout styling	Demonstrates good understanding of web page layout styling	Work does not meet expectations yet	/5

(retrieved from JS) in layout as specified				
Implement layout styling for content retrieved using JS using CSS Grid	CSS Grids properties used appropriately to create layout appearance as specified, for content retrieved using JS.	Organizes most content aspects as specified, other layout styling may be used extensively (e.g., grids or other styles like Floats) in addition to Grids.	Work does not meet expectations yet	/5
Use spacing and borders appropriately, as per reference video	Spacing and borders used appropriately to display content	Spacing and borders used mostly appropriately. (See feedback)	Work does not meet expectations yet	/5

Responsive design	Meets expectations 10 points	Somewhat meets expectations 5 points	Does not meet expectations yet 0 points	Criterion Score
Web pages are responsive (implemented using Flexbox and Grids)	<p>The web page is responsive.</p> <p>The website must be responsive and include desktop, and mobile views. Use the media query breakpoints as specified in Step 4 (item 4).</p> <p>The header and footer must change when you switch views as specified in the reference video.</p>	<p>Web page is somewhat responsive, but not according to assignment requirements.</p> <p>(See feedback)</p>	<p>Work does not meet expectations yet.</p> <p>(See feedback)</p>	/10

Code styles and validity	Meets expectations 5 points	Does not meet expectations yet 0 points	Criterion Score
Only one CSS style file used to style pages. No use of style attribute or style element.	Included in the submission as expected	Multiple style files used (other than main.css), or style attributes/style elements used.	/5
Use valid CSS	CSS is valid	CSS is not valid	/5
Make code readable & include comments appropriately	Code is readable, indented and comments are used as specified	<p>Work does not meet expectations yet.</p> <p>(See feedback)</p>	/5

Attributions and code organization	Meets expectations 5 points	Does not meet expectations yet 0 points	Criterion Score
Citations and notes	Includes citations for any external resources used. If no material used, then includes a note in the README file saying as such. If external resources are not used, includes a note in the README file that says that the assignment was developed based on content presented in the class and available in textbook.	Citations or note not included.	/5
Use valid CSS	CSS is valid	CSS is not valid	/5
Code organized as specified	Code is organized as specified in the assignment requirements. File names are used as specified. Folders and files are organized as specified.	Work does not meet expectations yet. (See feedback)	/5

Accessibility	Meets expectations 5 points	Does not meet expectations yet 0 points	Criterion Score
Use semantic elements and containers only for content sections; no generic container (i.e., div) elements	Included in the submission as expected. Some divisions may be used for layout-related aspects only, not as replacements of elements like sections, articles, nav, aside, etc.	Many generic containers used, including for all content related aspects	/5
Make content containers accessible	Semantic elements/containers are accessible.	Semantic containers are not accessible.	/5

Total / 100