JavaScript 1

Course Assignment

Brief

Use the attached resources (at the bottom) to complete the following assignment on the various pages.

Index Page

1. On the index page the API is provided please connect to the URL provided.
   https://api.magicthegathering.io/v1/cards
2. Convert the results to JSON format
3. Create a function, that takes the JSON Object as an argument.
4. Call the function from the method that returns the JSON data and pass in the JSON data.
5. Inside the function that you just created, append all the data to the div with the “cards” ID attached to it
6. Styles will be provided for you; the result will mimic the following design.

7. The HTML for the card above is in the index.html file, use this as a base to display all the cards. They should all follow the same design pattern. When done delete the hard coded card.
8. You will need the following keys to achieve this: name, imageUrl and id.
9. If you look at the “View More” button the id is used and passed in as a query string in the URL.
10. Don’t worry about the query string for the next page as there will be a function that will strip the URL and return the ID to you.
11. Above all the cards is a search box.

12. The search box should work in the following way, the search button will trigger a click event. This event will delete all the HTML inside the div that has the ID “cards” attached to it.
13. It should then get the value from the search text box, make a call to the api get all the results back. Thereafter it should filter through all the results by the value that was in the textbox.
14. If it finds any results it should be added to a new array and the new array should be display as HTML like you did with all the cards but it should only return what the user searched for. If the application doesn’t find any results a suitable message should be displayed.
15. Some images returned from the API will be broken.
16. Please test to see if the value is undefined, if the value is undefined, please use the image below as a placeholder
17. https://via.placeholder.com/223x310
18. Please write all js in the script.js file.

Card Specific Page

1. There is a page called card-specific.html if you click on the view more button on the index page you will notice “?id=xxxxxxxxxxxxxxxx” at the end of the URL, it will link to this page.
2. Inside the JavaScript file specific.js from lines 1 – 4 there is prewritten JavaScript that returns the value of the id in the Query String.
3. Take that value and make an API call, but append the ID to end of the API Call. The URL should look as follows:https://api.magicthegathering.io/v1/cards/3b072bac-4508-5bfa-adc3-13a9163284d3
4. If you open up the URL above it will only return details to you for that specific ID.
5. Append the image to the div called “cardImage”, set the width of the image to 100%.
6. Please get the following keys: imageUrl, name, text, rarity and colors. Display their values.

   The specific card should look as follows:

7. The HTML for the card template will be inside the card-specific.html file please delete it once you are pulling data from the API.
8. Test to see if there is no query string if there isnt display a suitable message.
9. Please write all js in the specific .js file.

About Page

1. On the about page. Replace the word “Magic” with the “Something” in all the text inside the div with “aboutText” ID attached to it.
2. Also on the about us page there is a div with and ID “moreInfoTrigger” it should look like the image below.
3. If you click on this heading it should toggle display: block and display none on the div with the “moreInfoContent” ID attached to it.
4. If style is display block it should look like the image below:
5. Please write all JavaScript in the about.js file.

Contact Form

1. Validate the contact form on the contact.html page.
2. The name and last name textboxes shouldn’t be empty.
3. The email textbox should have a valid email address inside it.
4. The number format should be as follows:
   xxx xxx xxxx
   xxx-xxx-xxxx
   xxx.xxx.xxxx
5. There are error messages inside the HTML. Display these messages if there is an issue with the values for each textbox respectively.
6. If the value of the textbox is valid, hide the error message.
7. All this should happen after a user clicks the submit button.
8. Please write all JavaScript in the contact.js file

Marking Criteria

- Can apply knowledge of JavaScript to develop simple dynamic and interactive solutions
- Can apply knowledge of Document Object Model (DOM) to influence structure, content and design in HTML documents
- Masters industry-relevant software for programming and web development with JavaScript
- Masters industry-relevant tools and techniques for testing and troubleshooting source code / Can study the development process of simple, interactive and dynamic solutions with JavaScript
- Can find information and material that is relevant to the development of dynamic and interactive web solutions and services built with JavaScript
- Can develop simple, relevant interactive solutions with JavaScript Events

Submission

- Please submit your assignment as a zip file.
- Please include a PDF report on the process you went through to complete this CA and the final product. Your report should include reflection on your submission, and must have at least the following main sections: introduction, body, conclusion, and references (if applicable). Your report must be 800-2000 words.

Resources

- JavaScript Course Assignment files