JavaScript 1

Module 1 ▼

Module 2 ▼

Module 3 ▼

Module 4 ▼

Course Assignment ▲

Course Assignment (../javascript-1/ca.html)

Marking Criteria (../javascript-1/marking-criteria.html)

Course Assignment

Table of Contents

Course Assignment

Brief

In this assignment, you will be required to modify your existing Cross-Course Project to fetch products from an external API. Your project should already have a product list page and a product detail page with hardcoded data, but you will now need to modify them to fetch data dynamically from the API.

The product list page should display a list of all products fetched from the API, and the user should be able to click on a product to view its details on a separate product detail page. You should ensure that the product data is not hardcoded into the HTML, but instead fetched dynamically from the API to display the latest data. Your API calls should include a loading indicator to show the user that data is being fetched. Additionally, you should implement error handling to catch any errors that may occur when fetching data and display an appropriate message on the page. Make sure that your project is tested thoroughly to ensure that it works as expected.

Here are the API documents for each of the APIs.

- Rainy Days API docs
- GameHub API docs
- Square Eyes API docs

The base URL for the API is `https://api.noroff.dev/api/v1/` .

Level 1 is required.

Level 2 is optional.

Choosing appropriate variable and function names will form part of your assessment, as will proper and consistent formatting of your code.

Level 1 Process

1. Look through your website and assess where API calls need to be made to dynamically add content. You should have a products page and a product details page, but there might be other pages like the home page where products are loaded.
2. Remove the hardcoded products from the HTML.
3. Fetch the products from the API and dyncamically add them to the page with links to the product details page. Use the ID as a query string parameter so that on the product details page you know which item has been clicked on.
4. On the product details page make a request to the API for the specific data for the product that the user has clicked on.
5. Add loading indicators when making API calls.
6. Ensure proper error handling.
7. Test the site.
8. Remove console.logs you’ve been using for testing.
9. Deliver the link to your site and your repository.

Level 2 Process

Make a working shopping cart so that the user can add items to a shopping cart and the correct products display on the checkout page.

Information

Rules

- You may only use plain JavaScript for this assignment, no libraries or frameworks. You will be given a mark of ‘not passed’ if you use a library or framework for your JavaScript code.

Submission

- A link to your repository with the latest code committed, and a link to the site deployed on Netlify.