Refactoring Issues and Bugs

1. Refactored Function for Store Comparison

The function responsible for comparing price totals and item availability across various stores has been refactored into a separate, dedicated function stored in an independent file. This function takes the selected items, total prices at different stores, available item counts, and missing item counts as inputs. After calculating the best store, it returns a Map containing the relevant data, including the keys **“bestStoreId”** and **“bestStoreName”**.

2. Navigation Bar Transparency Issue (Bug Fix)

A bug caused text to appear as though it was scrolling over the navigation bar when navigating pages with more vertical content than the screen size. Investigation revealed that the navigation bar’s transparency was the root cause, creating an illusion of text scrolling over it. The issue was resolved by making the navigation bar completely opaque, ensuring that the text is properly displayed behind it.

3. Search Bar Implementation in the Items Section

Initially, the Stores section lacked a search bar, making it difficult for users to find products efficiently. The solution was to implement and optimize the search bar in the Items section, where it is more relevant and useful.

4. Restriction on Negative Price Inputs

A bug allowed users to input negative values for item prices, which were then stored in the database as valid entries. To ensure data accuracy, the input validation was refactored to only accept positive values.

5. Removal of Unnecessary Raw Data from the Store Section

When navigating to the Stores section, the interface initially displayed raw database data instead of properly formatted product names and prices. The UI was updated to remove unnecessary data and display only a structured list of available items, with prices appearing upon clicking an item name.

6. Improved Error Handling for Invalid Inputs

Previously, incorrect inputs such as single spaces in all fields triggered full error messages, which were overwhelming for users. The error handling mechanism was refined to provide user-friendly messages such as **“Invalid name”**, **“Invalid price”**, and **“Please enter a valid store name”** instead of full technical error outputs.

7. Fix for Navigation Buttons Submitting Unexpectedly

A bug caused the **Next** and **Previous** buttons to unintentionally submit forms. This issue was resolved by modifying the button behaviors to function strictly as navigation controls without triggering unintended submissions.

8. Home Button Disappearance Issue (Bug Fix/Refactoring)

A bug caused the **Logo button**, which also serves as a home button, to become inactive when scrolling down. The issue stemmed from redundant and unnecessary code within the class responsible for rendering the logo. After refactoring and cleaning up the code, the home button now remains functional and visible regardless of scrolling.

9. Restriction on Extremely Large Price Inputs

A bug allowed users to enter excessively large numerical values for item prices, causing inaccurate price calculations and improper rounding. The issue was resolved by imposing input limits that align with realistic item price formats.

10. Prevention of Blank Item Names

Users could previously create unclickable items by entering only a single space as an item name, making the item inaccessible from menus. The issue was resolved by implementing an input validation check that requires a properly entered name before allowing the item to be added. If left blank, an error message is displayed, prompting the user to enter a valid name.