

Wine Catalog

Your goal is to implement a web user interface that will allow browsing of a **Wine Catalog**.

The recommended completion time of this test is: **2 - 3 days**.

ASSETS

As part of this package, you are provided with the following:

mockups/

- `default.png` - A screenshot of what your application should look like.
- `info.png` - A screenshot of what extended wine details should look like.
- `sort.png` - A screenshot of what the expanded sort dropdown look like.
- `filter_by_category.png` - A screenshot of what applied category filters should look like.
- `interacting_filters.png` - A screenshot of how all filters interacting together should look like.
- `shopping_cart.png` - A screenshot of how an expanded shopping cart should look like.
- `wine_details.gif` - An animation of how a catalog item should appear when interacted with.
- `shopping_cart.gif` - An animation of how the shopping cart should appear when interacted with.

lib/

- `jquery-3.2.1.min.js` - The imported third-party library for DOM traversal and manipulation.
- `underscore.js` - The imported third-party library to be used for Javascript utilities functions.

css/

- `reset.css` - A stylesheet used to reset default browser styling.
- `left.css` - A stylesheet used to style elements in the left panel of the application.
- `center.css` - A stylesheet used to style elements in the center panel of the application.
- `right.css` - A stylesheet used to style elements in the right panel of the application.
- `main.css` - A stylesheet that should contain any other generic elements of the application.

js/

- `common.js` - A script that will contain any global helpers or variables.
- `cart.js` - A script that will contain your implementation for how your shopping cart will behave.
- `catalog.js` - A script that defines all the data for rendering your wine catalog.
- `catalogItem.js` - A script that defines the structure of a wine item in your catalog.
- `category.js` - A script that provides logic and static definitions of wine categories.
- `main.js` - The script that you will be implementing the core of your Javascript implementation.

images/

- `catalog/*.png` - These are the static assets corresponding to each catalog item.
- `wine.png` - This is the static image that should be used to create the application logo.

`index.html` - This is where the main markup for your application should be implemented.

GUIDELINES / RESTRICTIONS

You will be evaluated based on:

- The level of completion of your implementation
- The accuracy of your implementation in accordance with the provided mockups
- The level of knowledge and organization of your approach to the solution
- The cleanliness of your final, submitted solution

You are encouraged to:

- Use **all** of the provided files and assets
- Annotate your solution as much as possible to communicate your intentions in the code
- Modularize your implementation as much as possible
- Create additional files to your solution to assist with organizing your solution
- Use an object-oriented approach to help with readability and organization
- Pay **very close attention** to all styles and animations as detailed in the provided mockups
- Double-check your solution before submitting

You are **HEAVILY** discouraged from:

- Using any additional third-party libraries
- Using any additional images or static resources
- Introducing any external frameworks
- Sourcing any external snippets directly (e.g. stackoverflow)

CLARIFICATIONS

- Screenshots were captured on a very high-resolution monitor so font looks smaller than it is in actuality. Base font-size is as provided: 12px.
- There are three filters available in this application:
 - **SEARCH**: This will match wines that contain the search term in their name (case insensitive).
 - **SORT BY**: This needs to contain all sort criteria detailed in mockups/sort.png. (HINT: The dropdown is a <select> element and the asc/desc character can be achieved with plain text)
 - **CATEGORY**: This will only match wines that satisfy the requested category.
- Interactions on the wine item:
 - Clicking the plus icon on the wine item will increment the # of selected units in the cart by 1
 - Hovering on the wine item will activate a hover state
 - Clicking on the **up** caret on the wine item will activate a detail state
 - Clicking on the **down** caret on the wine item in a detail state will revert to the default state
- Interactions on the shopping cart
 - Clicking on the shopping cart toggle will slide left/right the shopping cart panel accordingly
 - Each row of the shopping cart is a distinct wine added from the center panel
 - After adding a unit to the shopping cart, the total amount must be recalculated
 - Clicking the **X** icon at the start of the shopping cart row will remove **ALL** units for that wine from the cart, followed by a recalculation