THE UNIVERSITY OF SYDNEY

School of Information Technologies

Web Application Development (COMP5347)
Semester 1, 2018
Assignment 1: Bookstore Web Page

Individual Work: 10%

Submit your work on Canvas by11:59 pm 9th April, 2018

Introduction

In this assignment, you will demonstrate that you have the basic understanding of the core client-side technologies (HTML, CSS and JavaScript) and are able to create a HTML pages with desirable style and certain interactive features. You will be given a small start file set, which contains a complete CSS file and a skeleton HTML file. Your task is to produce an online bookstore Web page that implements the below requirements without using any third-party CSS frameworks or JavaScript libraries.

Main Requirements

The following functionalities must be implemented in your Web page:

- The main page should display the main information of each book including a thumbnail picture, title, author, year, price, publisher and category. (See Figure 1 as an example).
- End users can search books with their titles. The rows that match the search term will be highlighted with a colored background. (see Figure 2 as an example)
- End users should be able to filter books through their categories. The list of category works like a dropdown and should cover an extra category that is currently not in the bookstore for boundary test use. Meanwhile, end users should always have the option to return to the default status (display all books no matter which category they belong to).(see Figure 3 as an example)
- The functions of the search and filter should be able to work together and combine the result of each other. (See Figure 4 as an example).

Figure 1. Main page requirements

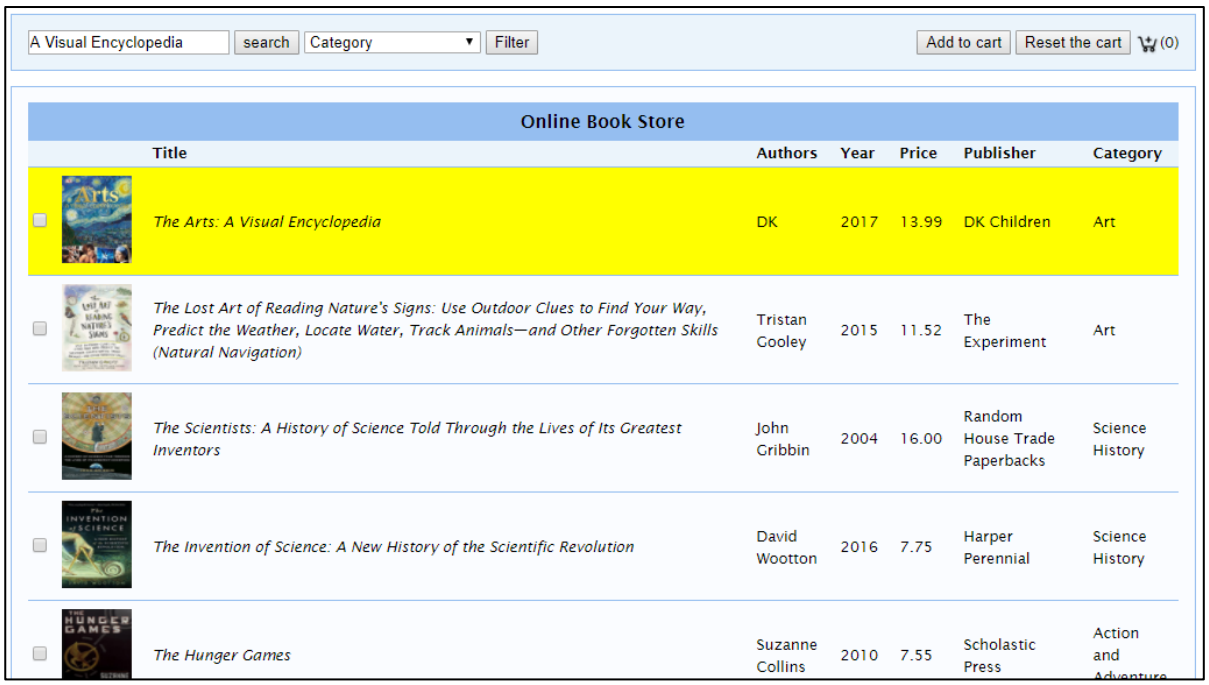

Figure 2. Highlight the searched row

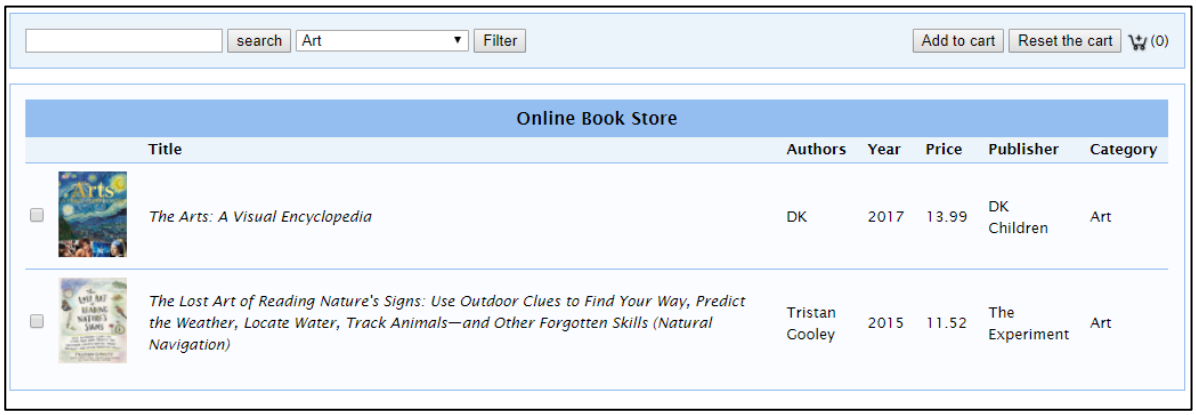

Figure 3. Filter the books

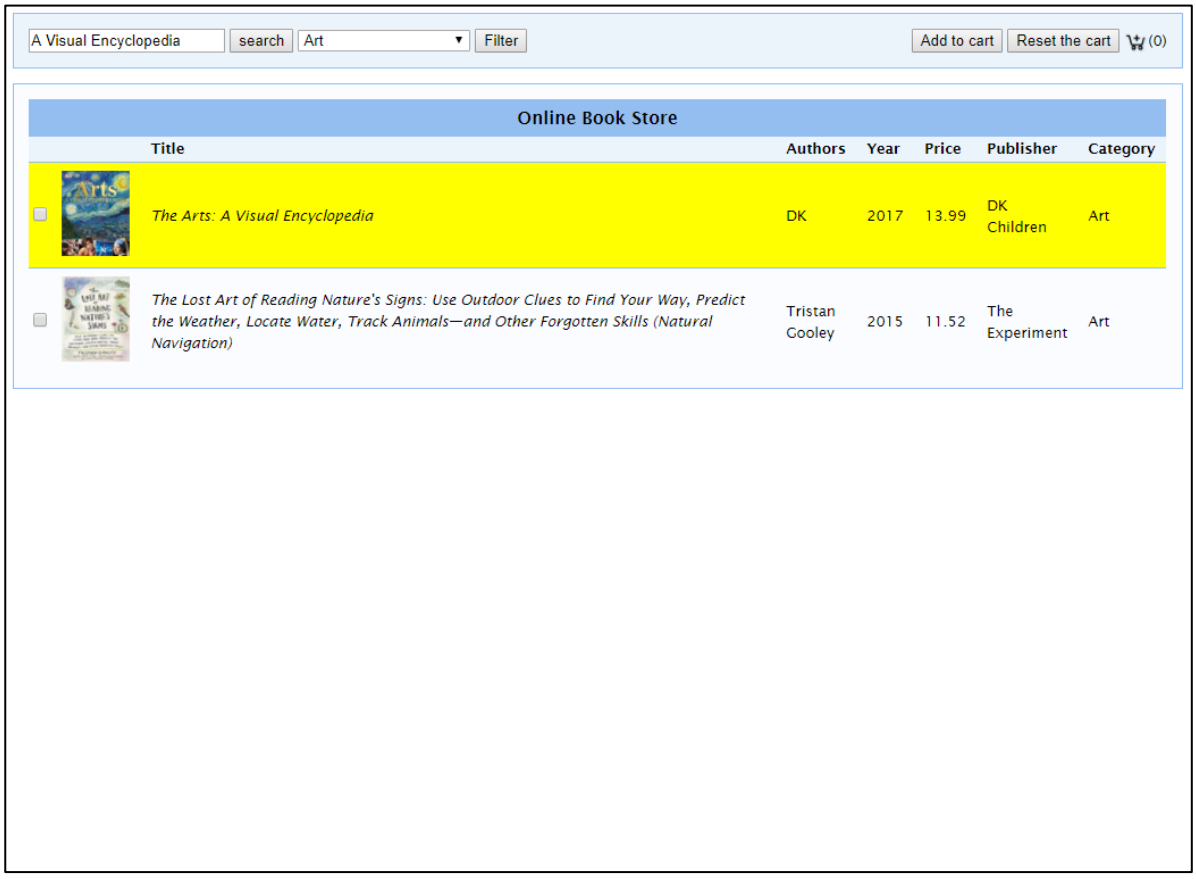

Figure 4. Combined result of search and filter functions

- End users should be able to select books and add them to the "Shopping Cart" through the checkbox ahead of them with the button "Add to cart".
- End users can clear the shopping cart through the "Reset the Cart" button. Users

should be prompted with a message box to confirm their desired action ("Reset the cart" or "Cancel").

- The "Shopping Cart" should always show the correct number of books in the cart based on the end user actions (see Figure 5 as an example).

Figure 5. Add to cart and reset the cart requirement

Design and Implementation Requirements

The online book store data and required thumb images are prepared in the zip file. The layout does not have to be the same as the example in Figure 1. For instance, you may modify the given CSS file to have slightly different color scheme, border and text styles.

You should consider the behavior of boundary cases and design suitable UI changes to notify the end users (e.g. an alarm). Sample boundary cases include, but are not restricted to the followings:

- Search term does not appear in any title
- Users select the category that does not contain any books

Each function mentioned above should work on the output of other functions.
Make sure that your HTML, CSS rules and JavaScript code are kept in separate files. Inline style or JavaScript code is not accepted.
Make sure that your HTML and CSS files are "clean" enough. They should not contain any unnecessary automatically generated content such as scattered element with pieces of styles.
Use of jQuery or other JavaScript library is not allowed in this assignment.
Use of Bootstrap or other CSS framework is not allowed either.

Mark Distribution

- Static features (4 marks)
  - 2 marks for having all HTML elements and structure as per the main requirements
  - 2 marks for applying similar CSS styles as per the main requirements

- Interactive features (6 marks)
  - 1 mark each for normal search function
  - 1 mark for normal filter function
  - 1.5 mark for combining the search and filter functions

- 1.5 marks for the cart functionality
- 1 mark for boundary cases

Deliverable and Submission Guideline

- **Source code submission**
  Submit a zip file containing the HTML, CSS and JavaScript files on Canvas site, by 11:59 pm 9th April, 2018. Please do not include the image files in the submission.

  Submit a hard copy assignment cover sheet with your signature to your tutor in Week 5 demo. You can download the cover sheet from: http://sydney.edu.au/engineering/it/current_students/undergrad/policies/assignment_sheet_individual.pdf

  The zip file should include your First and Last name and unikey in the file name. For example, if your is John Smith and UniKey is abcd1234, you would label your zip file as John_Smith_abcd1234_Assignment1.zip

Important Note: A software system will be used to detect software code similarity and allegations will be reported to the University's central unit.

- **System Demo**
  Each student will demo to their tutor in week 5 lab with the submitted version.