|  |  |
| --- | --- |
|  | **CODE:**  **TYPE: Labs**  **LOC:**  **DURATION:** |

**FRONT-END - BÀI TẬP THỰC HÀNH**

**Day 5 - JAVASCRIPT**

**5.1.** Viết đoạn chương trình khai báo một mảng các số nguyên (ít nhất có 08 giá trị).

- Viết hàm hiển thị danh sách các giá trị chẵn trong mảng và tổng các giá trị chẵn trong mảng

- Viết hàm hiển thị danh sách các giá trị lẽ trong mảng và tổng các giá trị lẽ trong mảng

- Viết hàm hiển thị danh sách các giá trị trong mảng ban đầu có giá trị nằm trong khoảng [a, b]. Trong đó, a và b nhập vào từ bàn phím và a ≥ b.

- Viết hàm sắp xếp các giá trị trong mảng chứa các giá trị chẵn, mảng chứa các giá trị lẽ và mảng ban đầu theo thứ tự tăng dần. (Viết hàm sắp xếp theo các thuật toán (ít nhất 2 thuật toán): Selection sort, Insertion sort, Merge sort, Quick sort, Stooge Sort)

**5.2.** Viết đoạn chương trình nhập vào thông tin của một người gồm: họ và tên, ngày sinh, giới tính. Hãy hiển thị thông tin và tuổi của người được nhập;

**5.3.** Khai báo Array chứa Data như sau:

*var users = [*

*{ id: 1, first\_name: "Eamon", last\_name: "Harhoff", email: "eharhoff0@imageshack.us", gender: "Male", age: 76, salary: 18888 },*

*{ id: 2, first\_name: "Laney", last\_name: "Whittam", email: "lwhittam1@issuu.com", gender: "Female", age: 42, salary: 15018 },*

*{ id: 3, first\_name: "Lynett", last\_name: "Twinberrow", email: "ltwinberrow2@gov.uk", gender: "Female", age: 99, salary: 13343 }*

*];*

**Yêu cầu:**

* Sử dụng hàm **filter** để lọc và hiển thị các user có *gender* là male và age dưới 40;
* Sử dụng hàm **reduce** để tính trung bình chung độ tuổi của các user

**Day 6 - JAVASCRIPT**

**6.1. Viết hàm validate sử dụng Regular Expression:**

- Kiểm tra số credit card – chuỗi chỉ chứa giá trị số có độ dài 16 số;

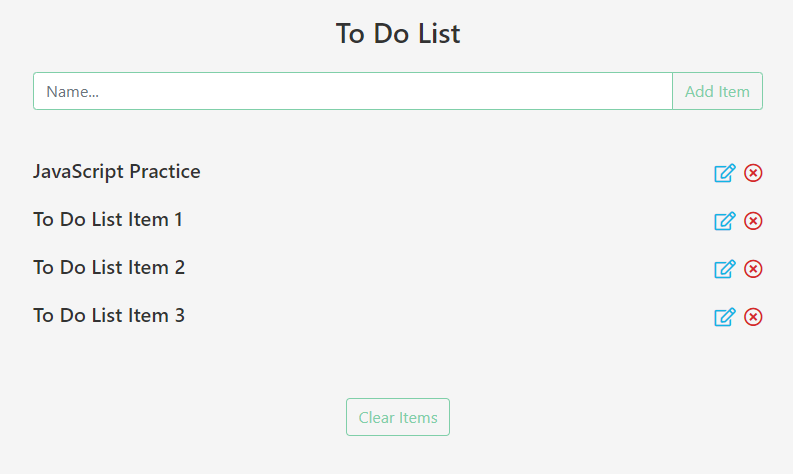
- Kiểm tra một giá trị có phải là số hay không?

- Kiểm tra một chuỗi có thỏa mãn định dạng email hay không?

- Kiểm tra một chuỗi có phải là chuỗi theo cú pháp url web hay không? Ví dụ: <http://www.w3schools.com/>, <https://www.w3schools.com/>, [www.w3schools.com](http://www.w3schools.com) là hợp lệ.

- Kiểm tra một chuỗi thỏa mãn: Chỉ chứa ký tự alphabet và số, không chứa các ký tự đặc biệt.

**6.2. Xây dựng Form và xử lý sự kiện cho bài toán Todo List sau:**



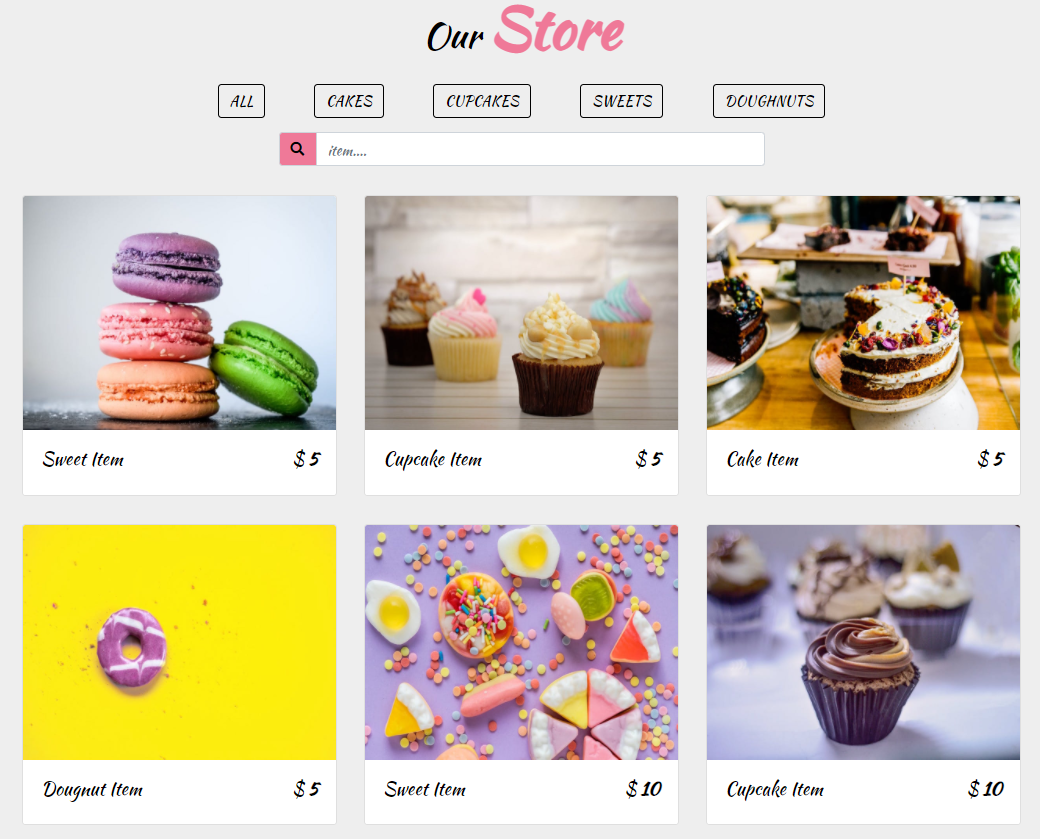
Sử dụng các giải pháp sau đây trong **JavaScript** để giải quyết:

* DOM Manipulation
* Event Listeners
* Nested Functions
* Local Storage API
* Arrays và các hàm xử lý trong Array: ***filter(), splice(), indexOf(),*** …
* Các Function trong JS (gợi ý): các hàm của ***classList, insertAdjacentHTML() hoặc appendChild(), removeChild(), …***

**Yêu cầu:**

* Lập trình form như trên
* Validate: dữ liệu nhập vào ô Textbox là không được để trống. Hiển thị lỗi validate không hợp lệ ngay phía dưới ô Textbox
* Xử lý sự kiện:
  + Khi nhập dữ liệu vào ô Textbox và nhấn nút **[Add Item]** thì chèn một row mới xuống vùng danh sách các Item to do list đã Add;
  + Khi click vào **icon edit** ở row tương ứng của item to do list thì thực hiện
    - Lấy Text của to do list tương ứng hiển thị lên ô Textbox
    - Xóa item to Do list này ra khỏi danh sách các Item to do list
  + Khi click vào **icon delete** ở row tương ứng của item to do list thì thực hiện xóa item to do list này ra khỏi danh sách các item to do list;
  + Khi click vào button **[Clear Items]** thì xóa tất cả các item to do list đang có;

**6.3. Lập trình Page để Filter các Item có cấu trúc như sau:**



Sử dụng các giải pháp sau đây trong **JavaScript** để giải quyết:

* DOM Manipulation
* Control Structures
* Array.forEach()
* JavaScript CSS Manipulation
* EventListeners
* Immediately Invoked Function Expressions
* CSS Font chữ: @import url("https://fonts.googleapis.com/css?family=Kaushan+Script");
* Hình ảnh: tự tìm ảnh phù hợp

**Yêu cầu:**

* Lập trình page chứa các loại bánh như hình
* Xử lý sự kiện khi lick vào các Button:
  + **[ALL]** Hiển thị tất cả các loại bánh
  + **[CAKES]** Lọc và hiển thị những item bánh có tên là Cake
  + **[CUPCAKES]** Lọc và hiển thị những item bánh có tên là Cupcake
  + **[SWEETS]** Lọc và hiển thị những item bánh có tên là Sweet
  + **[DOUGNUTS]** Lọc và hiển thị những item bánh có tên là Dougnut
* Xử lý sự kiện khi người dùng nhập keyword vào ô input (sự kiện **keyup**): Lọc và hiển thị các loại bánh có tên phù hợp với keyword được nhập