

1. Học viên lấy template trên link Github: <https://github.com/vtthien/JAVASCRIPT>
 - Vào mục “**DeThi**”
 - Có 2 folder, ứng với 2 bài làm:
 - ✓ DeThi
 - JSExam
 - Index.html
 - JSExamWithJQuery
 - Index.html
2. Học viên tiến hành hiện thực chức năng của 1 máy tính bỏ túi đơn giản (Calculator), học viên **PHẢI** hiện thực thuần thông qua javascripts trên file “**JSExam\index.html**”, có thể hiện thực bằng kỹ thuật của JQuery thông qua file “**JSExamWithJQuery\index.html**” sau khi hoàn thành bằng Javascript hoặc chỉ có thể hiện thực bằng JQuery.
3. Học viên hiện thực toàn bộ code Javascript trong file Index.html, hoặc có thể tạo ra 1 file **script.js** để hiện thực code javascript. Khi hoàn thành
 - ✓ DeThi
 - JSExam
 - Index.html (có thể hiện thực toàn bộ code js vào file này)
 - Script.js (nếu hiện thực javascript vào file này) (optional)
 - JSExamWithJQuery
 - a. Index.html (có thể hiện thực toàn bộ code js vào file này)
 - b. Script.js (nếu hiện thực javascript vào file này)(optional)
4. Sau khi hoàn thành bài thi, học viên nén toàn bộ source code, nén folder “DeThi”, đặt tên lại file nén có đính kèm tên của mình, sau đó, gửi file nén bài thi về email thienvt151@gmail.com với tiêu đề “[TTĐT][JAVASCRIPT] – Bài thi javascript – [Tên học viên]”
 - Ví dụ: Tôi tên Vũ Trọng Thiên, file nén “DeThi.zip”, sẽ đổi tên thành “**DeThi_VuTrongThien.zip**”.
 - Gửi file “**DeThi_VuTrongThien.zip**” về email thienvt151@gmail.com với tiêu đề “[TTĐT][JAVASCRIPT] – Bài thi javascript – Vũ Trọng Thiên”

Lưu ý: BÀI THI HỢP LỆ là bài thi gửi về trước 21h15 ngày 26/7/2018.

5. Hướng dẫn hiện thực:
Học viên hiện thực cơ bản các chức năng theo gợi ý sau:
 - **BƯỚC 1:** Hiện thực quá trình khởi tạo trang web (hàm **onLoadPage()**)
Khi mới mở trang web, hoặc refresh lại page, máy tính phải ở trạng thái là chưa có nhập số nào, chính vì vậy, các chức năng tính toán phải ở trạng thái disable.
Disable các đối tượng sau (tìm theo classname hoặc theo id): '**box-calculation**', '**box-result**', '**times**', '**divide**', '**equal**'
Tìm các đối tượng theo class, và thiết lập lại thuộc tính '**disabled**' về '**true**'

LƯU Ý: Hàm `getElementsByClassName` sẽ trả về 1 mảng các đối tượng, nên phải duyệt qua các phần tử để thiết lập `disabled` về `true` (viết 1 hàm `function disableAllItems(arr)`, truyền vào 1 mảng, sau đó chỉnh thuộc tính `disabled` trên từng phần tử mảng)

- **BƯỚC 2:** Hiện thực nút nhấn **“CE”**, refresh lại ô tính toán (box-calculation) và ô kết quả (box-result) bằng cách thiết lập lại value của các ô về rỗng thông qua hàm **“processCEButton_Click”**
- **BƯỚC 3:** Xử lý nút nhấn cho các số Number, học viên có thể tự hiện thực từng hàm cho từng đối tượng nút nhấn number, hoặc có thể viết tổng quát hóa bằng cách dùng 1 hàm **“function btnNumber_Click(objButton)”** duy nhất, bằng cách lấy dữ liệu của nút đang nhấn.
 - ✓ Lấy được dữ liệu nút đang nhấn
 - ✓ Nối dữ liệu được nhấn (số nhấn) vào trong dữ liệu hiện có trên màn hình (ô box-calculation). (Ví dụ: box-calculation đang có dữ liệu là “8+” khi nhấn thêm nút số 7 thì box-calculation sẽ có dữ liệu là “8+7”)
- **BƯỚC 4:** Hiện thực nút nhấn dấu chấm thông qua hàm **“function btnDot_Click()”**, tương tự như cách xử lý ở bước 3 cho các nút nhấn số.
- **BƯỚC 5:** Hiện thực nút Del để xóa giá trị nhập gần nhất, ví dụ trên box-calculation đang chứa “8+7” khi nhấn nút Del thì box-calculation sẽ còn “8+”
- **BƯỚC 6:** Xử lý cho các phép toán khi được nhấn **“function btnOperation_Click(objButton)”** để thêm các phép toán +, -, *, / vào biểu thức.
- **BƯỚC 7:** Hiện thực tính toán bằng việc hiện thực chức năng cho nút nhấn **“=”**, sử dụng hàm **“eval”** để tính toán biểu thức text để trả về dữ liệu tính toán.
Hiện thực hàm **“function btnEqual_Click()”**

===== HẾT =====