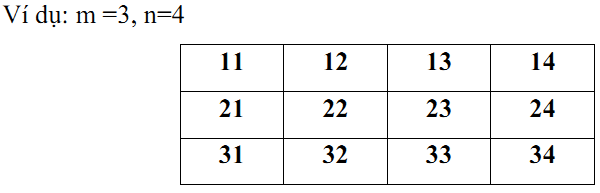
**varLab05 – TKWeb – Javascript (P1)**

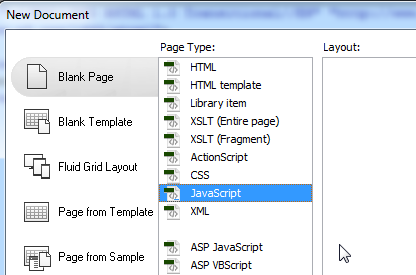
**1/ Nhập và xuất với javascript**

**Yêu cầu :** Dùng javascript viết 1 chương trình nhập số dòng và số cột sau đó xuất ra trang web dạng sau

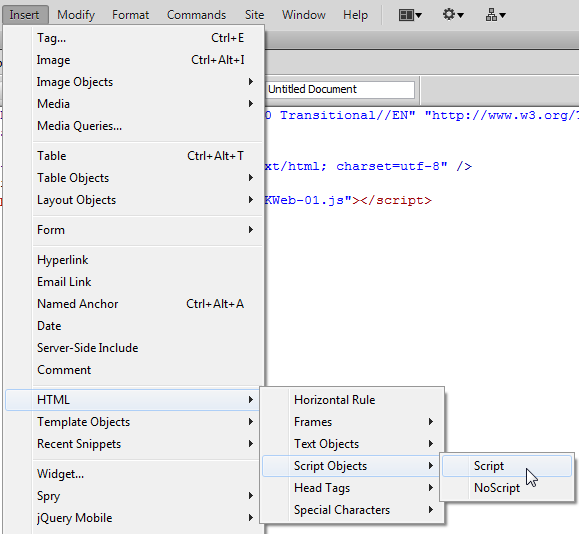


**Hướng dẫn:**

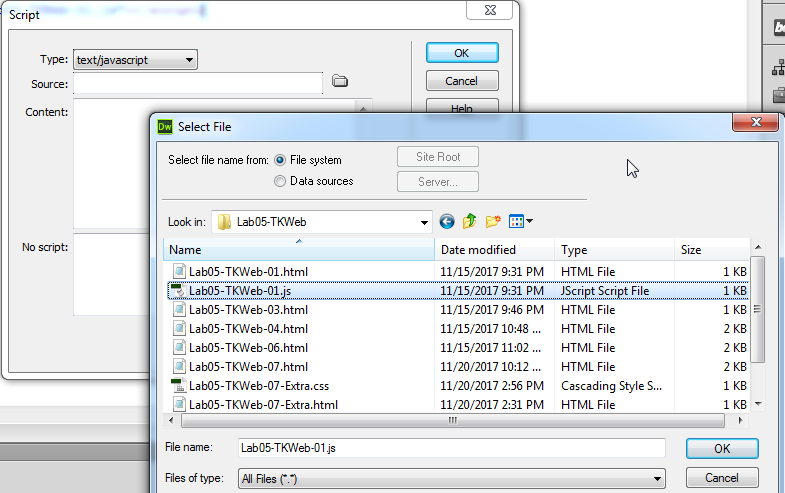
* Thực hiện tạo mới 1 trang html
* Thực hiện File > New > Javascript để tạo một file .js như sau



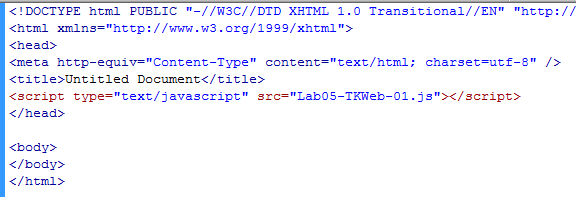
* Nhấn ctrl + s để lưu lại file js này. (Ví dụ đặt tên file là Lab05-TKWeb-01.js)
* Quay lại trang html và đặt con trỏ chuột sau tag <title> … </title> và trong tag <head> </head>
* Sau đó chọn



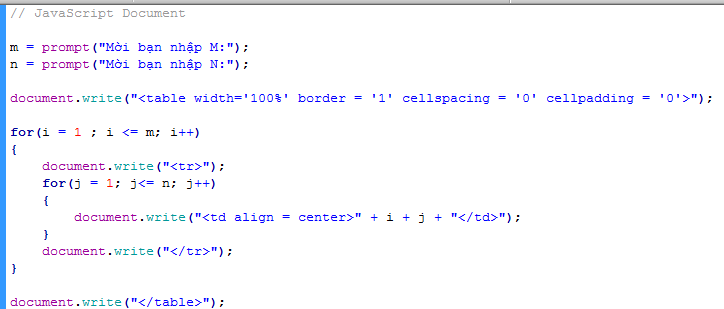
* Tại mục Source chọn file .js tương ứng (Ví dụ đặt tên file là Lab05-TKWeb-01.js)



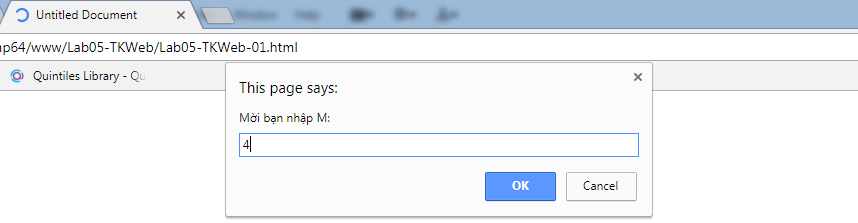
* Lúc này ta thấy xuất hiện một liên kết giữa html và js như sau là thành công

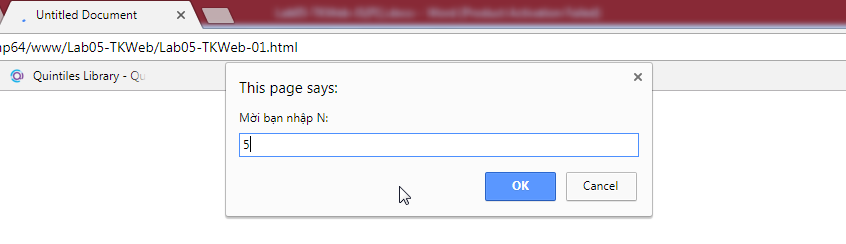


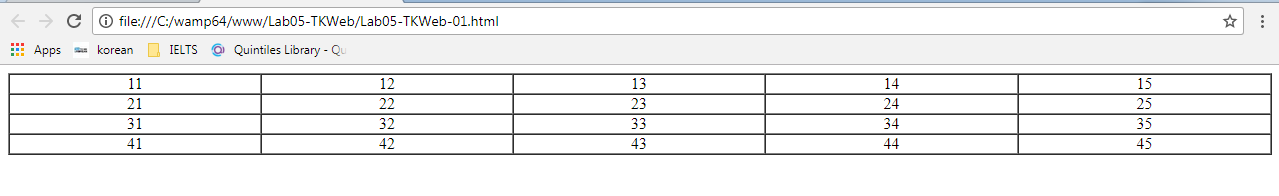
* Trong file javascript ta sẽ có các bước sau :
  + Dùng lệnh prompt để nhập thông tin từ hộp thoại của trình duyệt
  + Dùng document.write để viết nội dung html thông qua javascript (Ở đây ta viết html để tạo một table sau khi biết thông tin số dòng m và số cột n)



* Quay lại trang html và chạy thử trên trình duyệt ta sẽ có

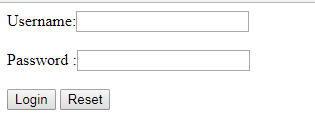






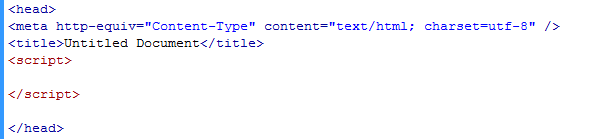
**2/ Gọi hàm thông qua javascript**

**Yêu cầu :** Viết chương trình tao form để nhập và kiểm tra tính hợp lệ username và password. Nếu username=admin và password=123 thì hiển thị thông báo “ Welcome to world of JavaScript”. Ngược lại thì hiển thị thông báo “ Invalid Username or password”.

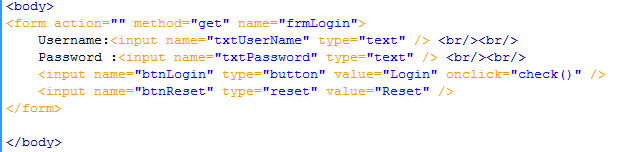


**Hướng dẫn :**

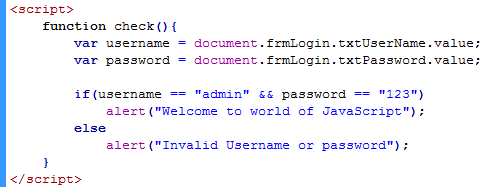
* Ngoài cách tạo riêng file javascript ta có thể insert trực tiếp một script vào trang html bằng cách. Để con chuột tại một vị trí trong <head></head>. Chọn Insert một Javascript như phần 1 nhưng không cần chọn source. Ta sẽ có kết quả như sau :



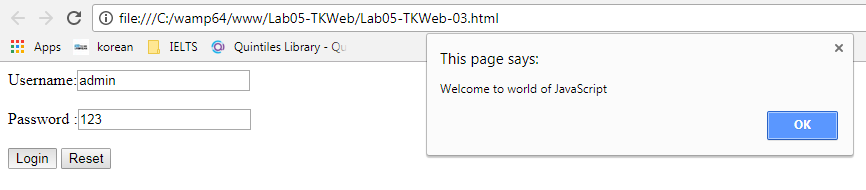
* Phần html ta thực hiện thêm form với 2 khung nhập và 2 nút
  + Ở đây lưu ý ta có 1 hàm tên check() được gọi khi nhấn chuột vào nút Login thông qua sự kiện onclick.
  + Hàm check() này sẽ viết trong javascript

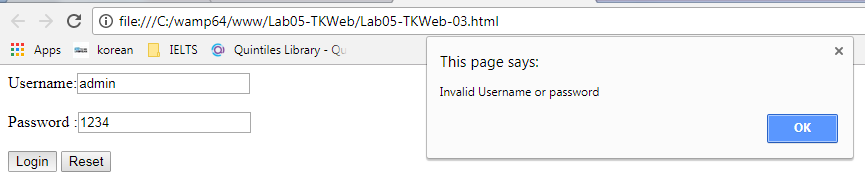


* Hàm check ta sẽ dùng thuộc tính name của form và name của các control trong form để truy xuất thông tin người dùng nhập
  + Dùng hàm alert để xuất một hộp thoại thông báo của trình duyệt



* Chạy chương trình ta sẽ có kết quả





**3/ Dùng regular expression để ràng buộc thông tin nhập bằng javascript**

**Yêu cầu** : Viết chương trình nhập vào một địa chỉ email. Kiểm tra xem email đó có hợp lệ không?

Email hợp lệ là:

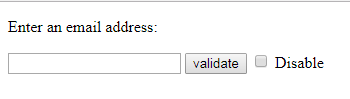
- @ không ở vị trí đầu tiên của chuỗi

- Phải có ít nhất 1 ký tự ở giữa @ và .

- Phải có ít nhất 1 ký tự sau dấu .

- Không được có khoảng trắng.

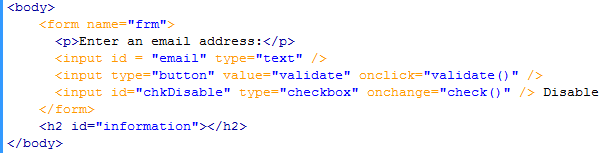
- Ngoài ra có 1 checkbox disable. Nếu check vào thì không thể nhập email được



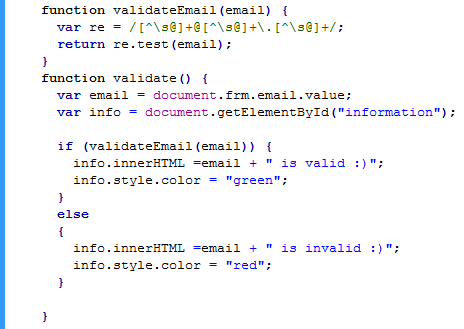
**Link tham khảo** : https://www.w3schools.com/js/js\_regexp.asp

**Hướng dẫn** :

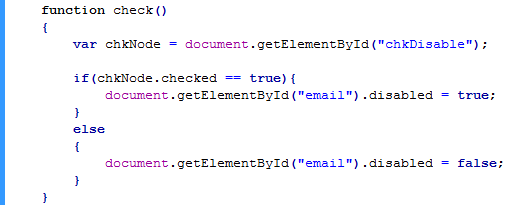
* Xây dựng html như sau
  + Hàm validate thực hiện khi nhấn nút validate
  + Hàm check thực hiện khi nhấn checkbox



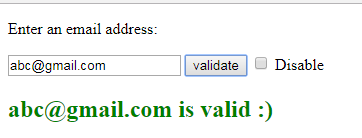
* Hàm validate
  + Sẽ gọi hàm validateEmail
  + Dùng hàm getElementById để lấy đối tượng thẻ có Id là information trong cấu trúc html
  + innerHTML để thay đổi nội dung của một đối tượng thẻ
  + style.color để thay đổi thuộc tính của một đối tượng thẻ
* Regular Expression : /[^\s@]+@[^\s@]+\.[^\s@]+/
  + Rule1 : [^\s@] nghĩa là ký tự bất kỳ miễn không là khoảng trắng “\s” hoặc ký tự “@”
  + Rule2 : + nghĩa là được lặp lại nhiều lần
  + Rule3 : @ buộc phải có 1 ký tự @
  + Rule4 : \. buộc phải có 1 ký tự “.”
  + Rule5 : Dạng chung sẽ là chuỗi@chuỗi.chuỗi
  + Ví dụ :
    - Vậy sẽ là chuỗi có dạng “[abc123@gmail.com](mailto:abc123@gmail.com)” thỏa
    - Chuỗi dạng “abc [123@gmail.com](mailto:123@gmail.com)” phạm quy tắc 1
    - Chuỗi dạng “[abc@abc@gmail.com](about:blank)” phạm quy tắc 5
    - Chuỗi dạng “abc@gmail” phạm quy tắc 5



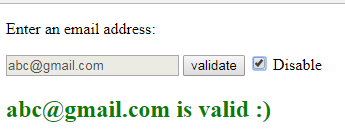
* Hàm check
  + Thuộc tính checked có hai giá trị true và false để kiểm tra nó đang được check hay không
  + Thuộc tính disabled để vô hiệu hóa một control



* Khi chạy chương trình ta sẽ có kết quả







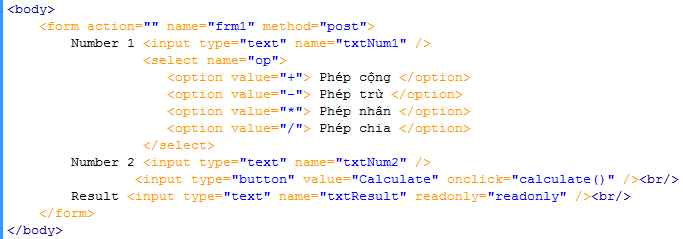
**4/ Tính toán dùng javascript**

**Yêu cầu** : Viết trang web mô phỏng máy tính cá nhân thực hiện được các phép toán cơ bản như +, l, \*, /

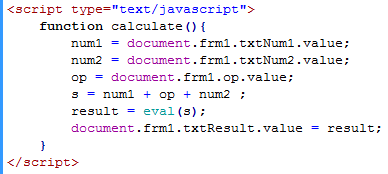


**Hướng dẫn :**

* Tạo trang html như sau



* Hàm calculate trong javascript
  + Dùng hàm eval để tính toán một biểu thức chuỗi



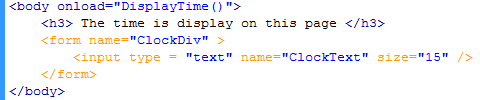
**5/ Xuất đồng hồ dạng digital**

**Yêu cầu :** Viết chương trình hiển thị đồng hồ điện tử trên trang web dưới dạng digital.

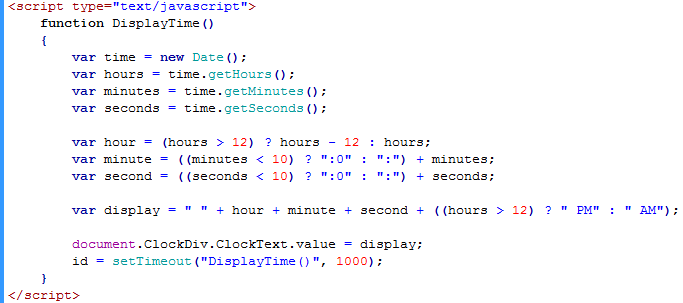


**Hướng dẫn :**

* Xây dựng nội dung html như sau
  + Chú ý ta có sự kiện onload của thẻ body nghĩa là hàm này thực hiện tự động khi trang web được load

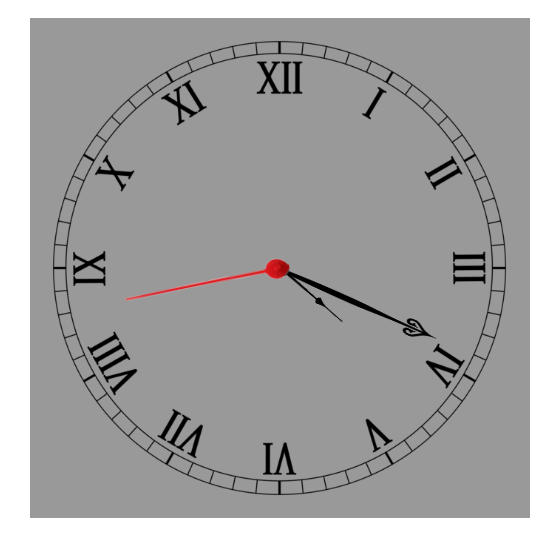


* Hàm DisplayTime
  + New Date sẽ có ta một đối tượng ngày giờ hiện hành
  + getHours, getMinutes, getSeconds cho thông tin cụ thể giờ phút giây
  + setTimeout là cơ chế hẹn giờ để gọi hàm. Ta hẹn cứ 1000 milisecond thì hàm DisplayTime được gọi lại



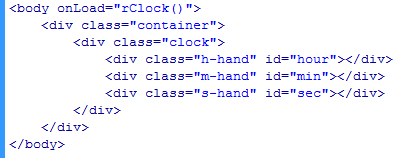
**5/ Xuất đồng hồ dạng analog**

**Yêu cầu :** Viết chương trình hiển thị đồng hồ điện tử trên trang web dưới dạng analog.

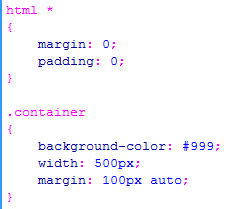


**Hướng dẫn** :

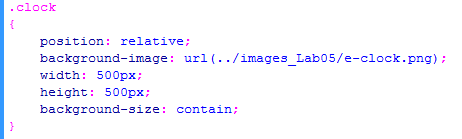
* Tạo trang html có nội dung sau
  + Thẻ div clock dùng thể làm background với ảnh cái đồng hồ
  + Thẻ div h-hand là ảnh kim giờ
  + Thẻ div m-hand là ảnh kim phút
  + Thẻ div s-hand là ảnh kim giây
  + Các ảnh trong các thẻ này sẽ xoay tùy vào giờ hiện tại



* Link html có với một file css như sau
  + Xóa các canh lề
  + Cho khung container rộng 500px, canh trên dưới 100px và tự đông canh trái phải

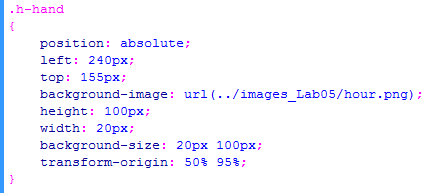


* + Khung clock đặt position là relative để các khung nó chứa (m-hand, h-hand, s-hand) có thể dễ dàng canh vị trí theo khung này mà không cần quan tâm đến layout trang web
  + Ảnh nền khung clock là ảnh e-clock.png
  + Định khung cho đồng hồ là 500 x 500
  + Background-size giúp toàn bộ ảnh phủ toàn bộ khung này (không bị cắt xén)

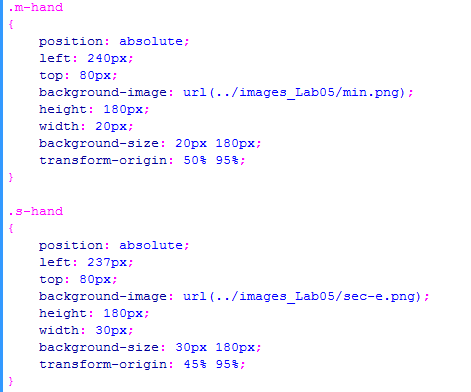


* + Khung chứa kim giờ sẽ canh chỉnh theo khung cha là clock nhờ position là absolute
  + Vị trí canh là left 240px và top 155px theo khung clock
  + Dùng ảnh hour.png
  + Định chiều cao và rộng cho ảnh
  + Transform-origin giúp định nghĩ điểm gốc mà khung này sẽ xoay quanh.(Nhìn ảnh ta sẽ thấy tọa độ x là 50% và y là 95% chính là vị trí cái lỗ của cây kim này. Đây chính là gốc chọn để khung này xoay quanh gốc này.)

ảnh hour.png 



* + Tương tự cho m-hand và s-hand (kim phút và kim giây)



* Giờ ta tạo một file js và kết nối với html đang có
  + Hàm rClock sẽ gọi khi trang web được load thông qua sự kiện onload của thẻ body
  + Hàm clock gọi trong hàm rClock và cứ sau mỗi 1000 milisecond
  + clockDivs là một danh sách các đối tượng thẻ div lần lượt là h-hand, m-hand, và s-hand
  + Thêm thuộc tính transform là rotate ta sẽ có các thẻ này xoay quanh một gốc (đã định nghĩa trong css bằng transform-origin)
  + Deg là quay theo độ

