**LAB 4: LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG VÀ MÔ HÌNH BOM**

Sử dụng oop ; Sử dụng các hàm trong window, document, gọi hàm trong sự kiện…

### Bài 1: Giải phương trình bậc 2 theo cách oop

Viết mã javascript tạo một đối tượng chứa các hệ số a, b, c của 1 phương trình bậc 2. Sau đó gọi phương thức giai() để giải và xuất nghiệm ra màn hình.

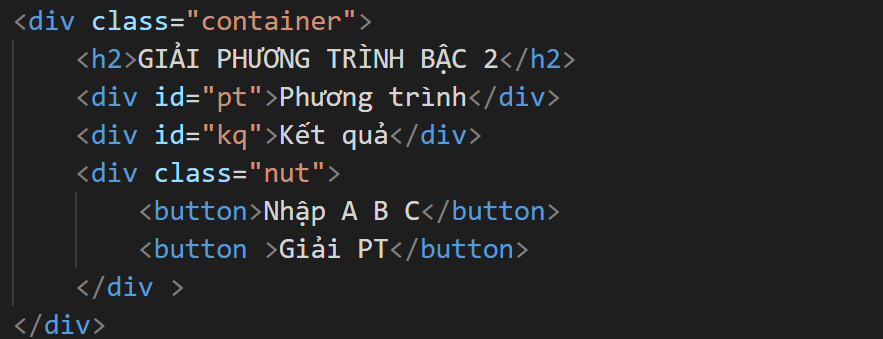


Hướng dẫn thực hiện:

1. Tạo file **bai1-ptb2.html**
2. Code html trong body:

Tạo 1 div có id là pt để hiện phương trình, 1 div có id là kq để hiện kết quả

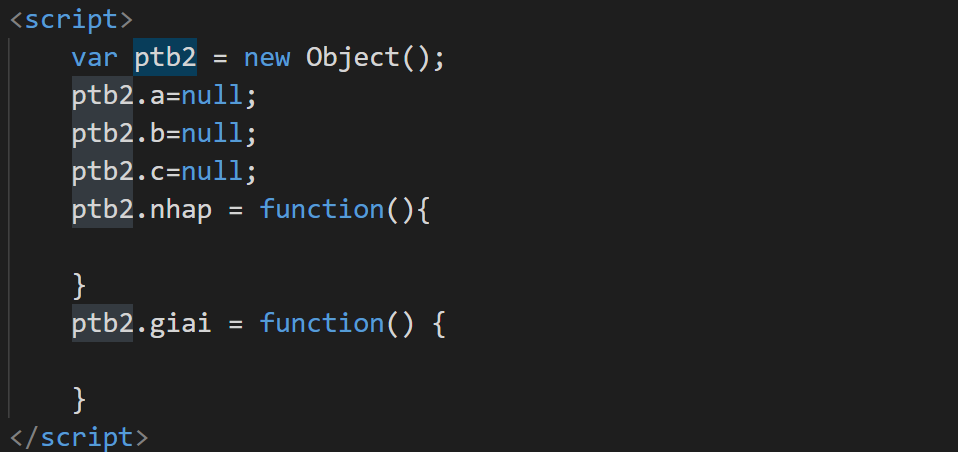
Một button để nhập 3 hệ số a b c. Một button để giải phương trình.



1. Định dạng cho đẹp:



1. Trong tag head, định nghĩa đối tượng ptb2 + khai báo 3 hệ số a b c + khai báo 2 hàm nhap và giai



1. Gọi hàm

* Trong sự kiện click của button **Nhập A B C** : gọi hàm **ptt2.nhap()**

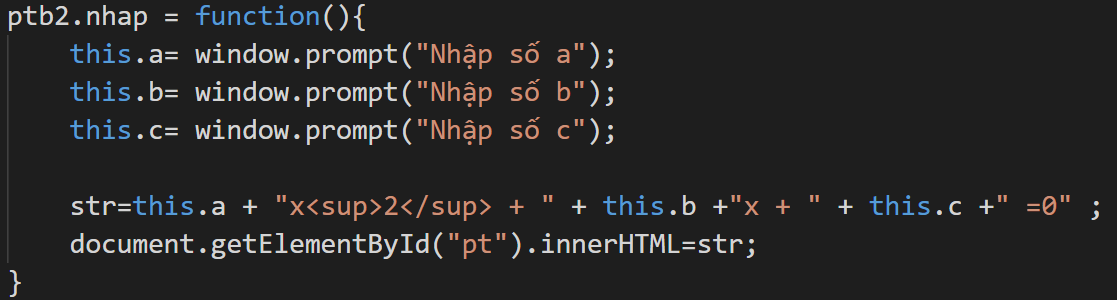


* Trong sự kiện click của button **Giải PT**: gọi hàm **ptt2.giai()**

1. Viết hàm nhập :

Trong hàm nhap của ptb2, code để nhập 3 giá trị lưu vào a , b, c của đối tượng ptb2.

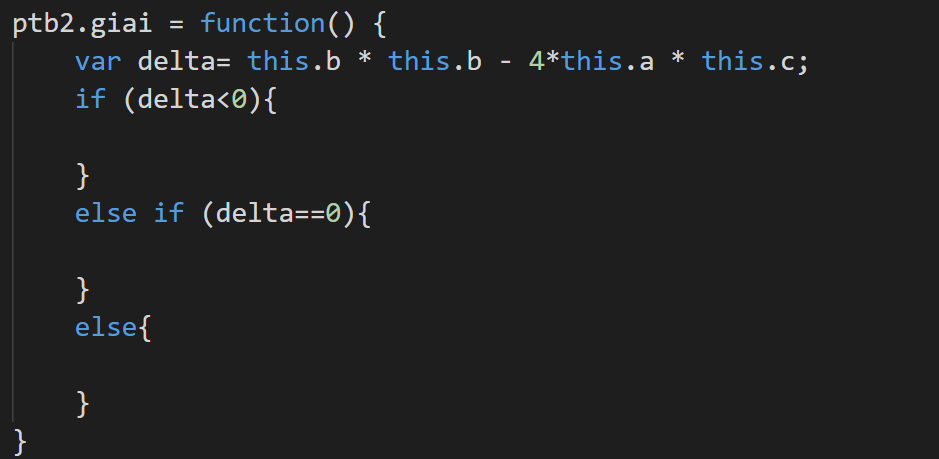
Khi nhập xong thì xuất phương trình ra trong div có id là pt



Xem thử trang web, nhập giá trị và phải thấy phương trình hiện ra:



1. Viết hàm giải:



- Viết code biện luận 3 trường hợp của delta và hiện kết quả trong div có id là kq

1. Cải thiện nhập abc:

Khi nhập a,b,c: bắt buộc nhập cho đến khi user nhập số (không cho nhập chữ)

Gợi ý: dùng vòng lặp do while bao quanh hàm prompt, cho đến khi isNAN(this.a)==true

1. Nhập họ tên sinh viên trong tag title

### Bài 2 : Đồng hồ

Viết mã javascript hiển thị thời gian hiện tại của máy tính, cập nhật lại mỗi giây.

1. Tạo file **bai3-dongho.html v**à code html

<style>

        .container { width: 1140px; margin:auto; border:1px solid darkgrey}

        header { background: darkolivegreen; height: 120px; position: relative;}

        main{ background: darkseagreen; min-height: 400px;}

        footer { background: darkslategray; height: 100px;}

</style>

<body>

    <div class="container">

        <header>

            <span id="dongho" >Đồng hồ</span>

        </header>

        <main></main>

        <footer></footer>

    </div>

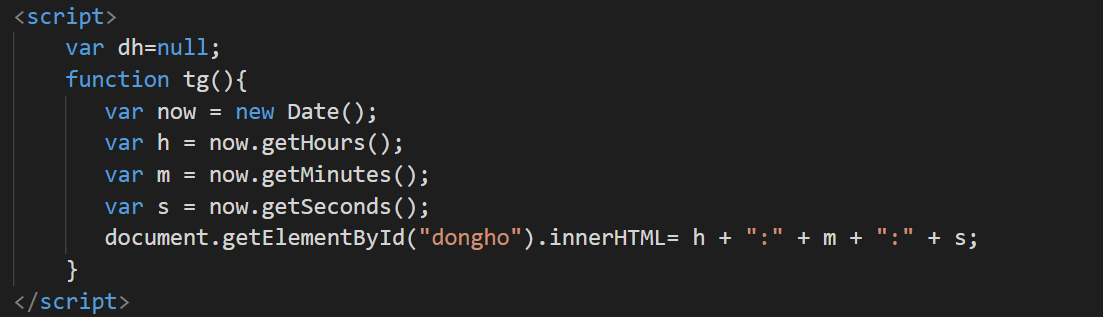
</body>

Xem thử trang web:

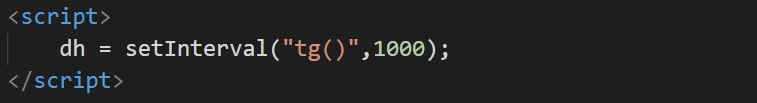


1. Code ở cuối trang :

Khai báo biến **dh** và định nghĩa hàm **tg** lấy giờ phút giây hiện tại đưa vào tag có id là dongho



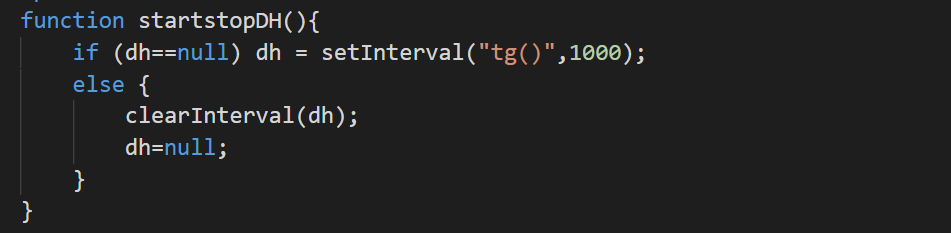
1. Gọi hàm tg mỗi giây



1. Xem thử và định dạng



1. Định nghĩa hàm StartStopDH



1. Yêu cầu: trong trang web, khi đồng hồ đang chạy, nhắp vào thì đồng hồ dừng lại, nhắp lại sẽ chạy tiếp.

Thực hiện: Gọi hàm StatStopDH vừa mới định nghĩa ở trên trong sự kiện click của tag span dongho

1. Lời chào:

* Viết lệnh để 10 giây sau khi nạp trang hiện ra lời chào: **Tôi là <Họ tên>. Chào Bạn! Chúc an lành**

Gợi ý: Dùng hàm setTimeout (“lời chào”, 10 giây);

1. Thêm nút Nạp lại

* Trong tag main, thêm code tạo 1 button:



* Viết lệnh nạp lại trang trong sự kiện click của button **Nạp lại** vừa thêm

### Bài 3: Điều khiển window

1. Tạo trang **thongtin.html** và viết code để hiện thông tin của mình:

<h3>Họ tên: Nguyễn Văn Tèo </h3>

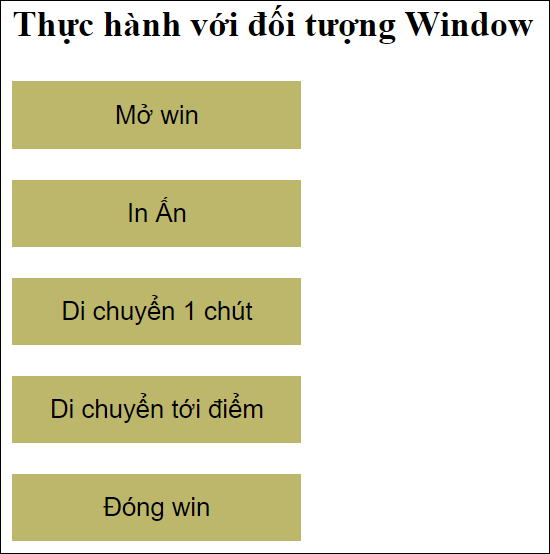
<h3>Email: meomeo@gmail.com </h3>

<h3>Mã SV: PS123456 </h3>

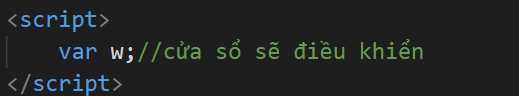
Định dạng màu chữ, màu nền, border, canh giữa font-size cho đẹp

1. Tạo file **bai4-window.html**

* Code html tạo chữ và các nút như hình dưới (dùng tag h1 chứa chữ và tag button tạo các nút)

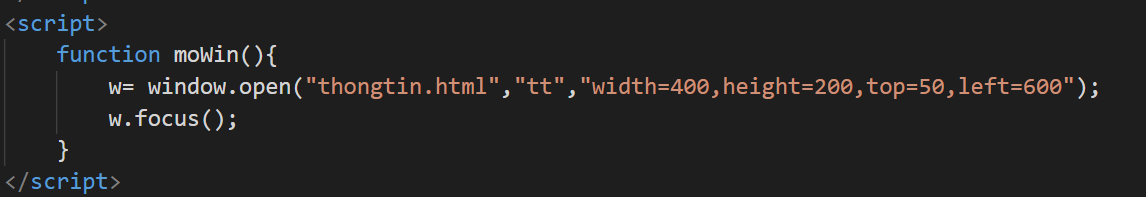


1. Khai báo biến w chứa thông tin cửa sổ



1. Mở Win

* Dùng lệnh window.open để mở trang thongtin.html, tên cửa sổ là tt, cửa sổ có width là 400, height là 200, top là 50, left là 600. Mở xong gán vào biến w và gọi hàm focus() để cửa sổ ở trên top.



* Gọi hàm moWin() trong sự kiện click của button Mở Win
* Test: Nhắp nút Mở Win phải thấy cửa sổ mới



1. **Đóng Win**

* Viết lệnh **w.close()** trong sự kiện click của nút Đóng win
* Test: Khi mở win lên, nhắp nút **Đóng win** thì cửa sổ phải mất.

1. **Di chuyển 1 chút: di chuyển cửa sổ tt qua phải 5px qua xuống dưới 10px**

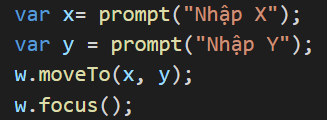
* Trong sự kiện click của nút Di chuyển 1 chút, gọi hàm dichuyen1chut()
* Định nghĩa hàm dichuyen1chut và viết lệnh:



* Test: Khi mở win lên, nhắp nút **Di chuyển 1 chút** thì cửa sổ phải di chuyển.

1. **Di chuyển tới điểm: cho user nhập tọa độ x, y và di chuyển cửa sổ đến đó.**

* Trong sự kiện click của nút Di chuyển tới điểm, gọi hàm dichuyenDen()
* Định nghĩa hàm dichuyenDen và viết lệnh:



* Test: Khi mở win lên, nhắp nút **Di chuyển tới điểm** + nhập x, y thì cửa sổ phải di chuyển đến đó.

1. **In Ấn**

Trong sự kiện click của button In Ấn, gọi lệnh **w.print()** để hiện hộp thoại in ấn

1. Định dạng các button và trang web cho đẹp.

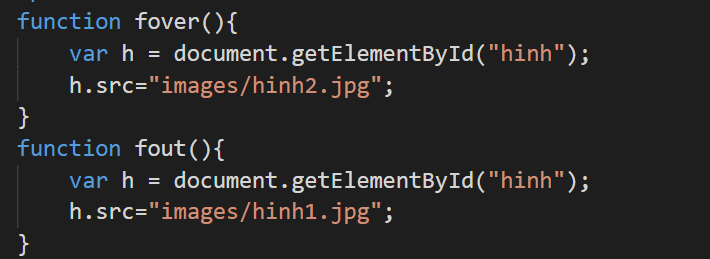
### Bài 4 : Gọi hàm trong sự kiện html

Viết mã javascript để điều khiển hành vi người dùng khi đưa chuột vào hoặc ra khỏi hình.

1. Chụp 2 hình selfie chính mình và lưu vào folder images của website với tên **hinh1.jpg** và **hinh2.jpg**
2. Tạo file **bai5.html** và code html để chèn hình và đặt tên cho tag img là **hinh:**



1. Định nghĩa 2 hàm chuyển hình: chọn đối tượng hình và gán 2 địa chỉ 2 file hình khác nhau



1. Gọi hàm trong sự kiện mouseover và mousemove của **hinh**

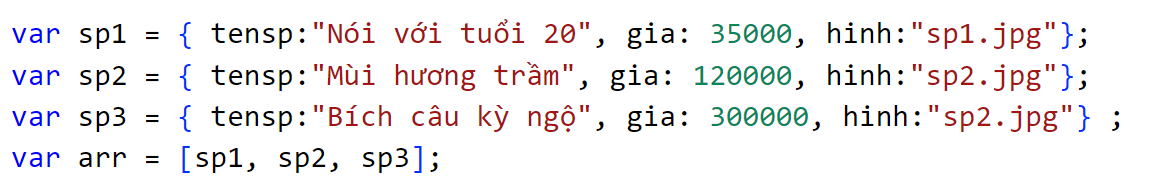


1. Định dạng hình cho đẹp: độ rộng, độ cao, border, margin…

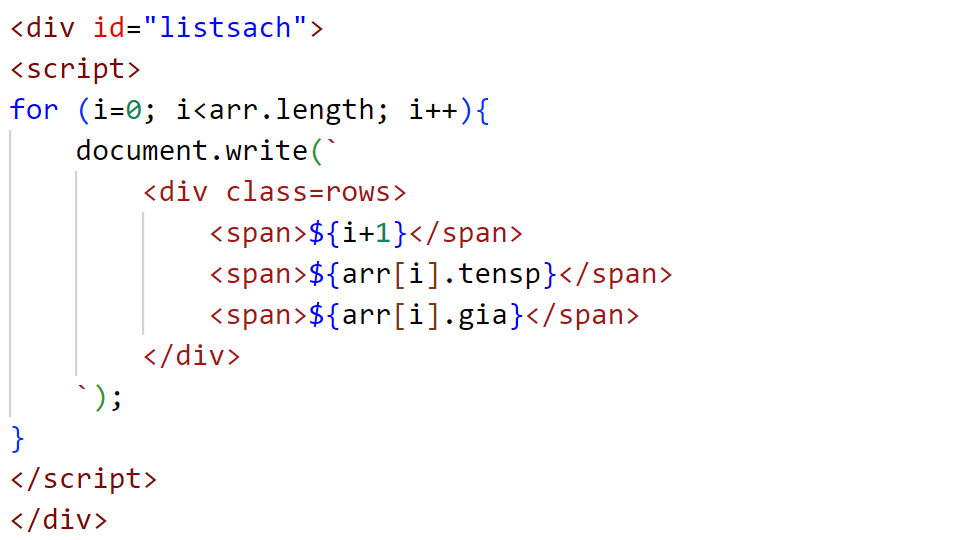
### Bài 5: Ứng dụng OOP và mảng

Tạo 3 đối tượng, mỗi đối tượng gồm các thuộc tính tensp, gia, hinh. Sau đó đưa 3 đối tượng vào mảng. Cuối cùng xuất mảng ra trang web.

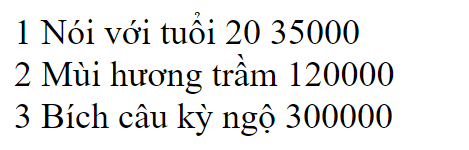
1. Tạo trang bai2.html
2. Tạo 3 đối tượng và mảng chứa 2 đối tượng



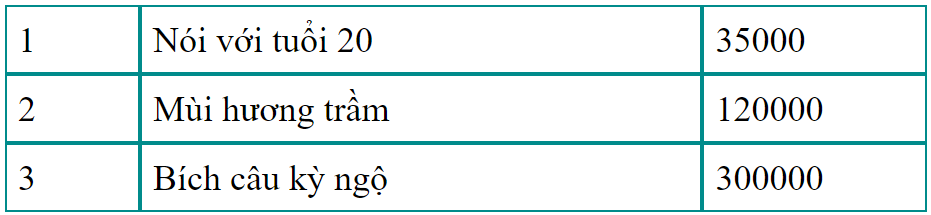
1. Lặp qua mảng để hiện



Xem kết quả phải được như sau:



1. Định dạng để được như hình



Có thể làm thêm (nếu muốn): canh phải cột giá, hiện hình sản phẩm, thêm vài quyển sách