JavaScript - Introduction

Introduction

- JavaScript là ngôn ngữ lập trình của HTML và Web. Nó là ngôn ngữ lập trình phổ biến nhất thế giới.
- Javascript có thể chạy được trên mọi trình duyệt web
- JavaScript là một trong **ba ngôn ngữ mà** tất cả các nhà phát triển web **phải** học:
  - **HTML** để xác định nội dung của các trang web
  - **CSS** để chỉ định cách bố trí các trang web
  - **JavaScript** để lập trình hành vi của các trang web

JavaScript output

- Hàm alert() : hiển thị cửa sổ in chuỗi ký tự hoặc số
  - Cách sử dụng:

```
<script type="text/javascript">
    alert("Fasttrack SE");
</script>
```

- Hàm document.write() : in chuỗi kí tự hoặc số lên màn hình
  - Cách sử dụng:

```
<script type="text/javascript">
    document.write("Fasttrack SE");
</script>
```

JavaScript output

- Hàm innerHTML : dùng để truy cập vào một phần tử HTML
  - Cách sử dụng:

```
<script type="text/javascript">
    document.getElementById("demo").innerHTML = 5 + 6;
</script>
```

- Hàm console.log () : in chuỗi kí tự hoặc số lên màn hình
  - Cách sử dụng:

```
<script type="text/javascript">
    console.log(5+6);
</script>
```

JS variables & types

- Các kiểu dữ liệu trong javascript

Kiểu dữ liệu	Mô tả
Numbers	Các số nguyên hoặc số thực. Ví dụ: 487 hoặc 25.95
Logical hoặc Boolean	True hoặc False
Strings	“Hello World”
Null	Keyword đặc biệt chỉ giá trị null.

JS variables & types

- Biến JavaScript là containers để lưu trữ giá trị dữ liệu .
- Khai báo biến
  - Cú pháp: var ten_bien = [value];
  - Ví dụ:

```
<script>

  var new_amount;
  var answer = null;
  var old_cost = 12.50;
  var number = 23;
  var fastrack = ‘Fast track se’;
  result = true;

</script>
```

JS operator

Toán tử số học:

Kí hiệu	Mô tả
+	Phép cộng
-	Phép trừ
*	Phép nhân
/	Phép chia
%	Phép chia lấy dư

JS operator

Toán tử gán:

Toán tử	Ý nghĩa
x = y	x = y
x += y	x = x + y
x -= y	x = x - y
x *= y	x = x * y
x /= y	x = x / y
x %= y	x = x % y

JS operators

Toán tử so sánh :

Toán tử	Ý nghĩa
==	Trả về TRUE nếu 2 giá trị bằng nhau và trả về FALSE nếu 2 giá trị khác nhau
!=	Trả về TRUE nếu 2 giá trị khác nhau và trả về FALSE nếu 2 giá trị bằng nhau
>	Vd : a>b . Kết quả sẽ trả về TRUE nếu a lớn hơn b và ngược lại sẽ trả về FALSE nếu a nhỏ hơn b
>=	Vd : a>=b . Kết quả sẽ trả về TRUE nếu a lớn hơn hoặc bằng b và ngược lại sẽ trả về FALSE nếu a nhỏ hơn b
<	Vd : a<b . Kết quả sẽ trả về TRUE nếu a nhỏ hơn b và ngược lại sẽ trả về FALSE nếu a lớn hơn b
<=	Vd : a<b . Kết quả sẽ trả về TRUE nếu a nhỏ hơn hoặc bằng b và ngược lại sẽ trả về FALSE nếu a lớn hơn b
===	So sánh giá trị bằng nhau và kiểu dữ liệu giống nhau

JS operators

Toán tử logic: ! && ||

- Toán tử && (and) : chỉ đúng khi cả 2 đều đúng
- Toán tử || (or) : chỉ sai khi cả hai giá trị đều sai

x	y	Toán tử	Kết quả
True	True	&&	True
True	False	&&	False
False	True	||	True
False	False	||	False
True		!	False

JavaScript If...Else Statements

- Dạng 1 : Khuyết Else

```
<script>
    if(Điều kiện){
        //Khối câu lệnh
    }
</script>
```

- Ví dụ

```
<script>
    var x = 5;
    if(x>1){
        alert('x lớn hơn 1 ');
    }
</script>
```

JavaScript If...Else Statements

- Dạng 2 (đầy đủ):

```
<script>
    if(Điều kiện){
        //Khối câu lệnh
    }else{
        //Khối câu lệnh
    }
</script>
```

- Ví dụ

```
<script>
    var a = 5,b=6;
    if(a>b){
        alert('a lớn hơn b');
    }else{
        alert('a nhỏ hơn b');
    }
</script>
```

Events in javascript

- Javascript có thể bắt được 1 số sự kiện như: onclick ,ondbclick,…
- Các sự kiện onfocus,onblur,onchange,onsubmit,… thường được dung cho các đối tượng của form
- Một số hàm thường dùng trong javascript:
  - Prompt() : nhận giá trị nhập từ bàn phím
  - Document.getElementById("id").value: lấy giá trị từ 1 thẻ html bằng id

JS Functions

- Function **là một tập hợp các đoạn mã** và nó sẽ thực thi các đoạn mã đó khi gọi hàm ra, nó sẽ được thực thi lại nhiều lần hoặc thực thi trong một trường hợp nhất định. Có nghĩa là khi bạn tạo hàm thì các đoạn mã bên trong đó sẽ không thực thi cho đến khi được gọi ra bên ngoài.
- Hàm không trả về giá trị:

```
<script>
    function tên_hàm(các tham số){
        //các câu lệnh
    }
</script>
```

JS Functions

❖ Ví dụ :

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Functions</h2>
<a href="javascript:myFunction()" > Hiển thị </a>
<script>
function myFunction() {
    alert('Fasttrack SE');
}

</script>

</body>
</html>
```

JS Functions

- Hàm trả về giá trị

```
<script>
    function tên_hàm(các tham số){
        //các câu lệnh
        return //giá trị;
    }
</script>
```

- Ví dụ:

```
<html>
<body>

<h2>JavaScript Functions</h2>

<p id="demo"></p>

<script>
function myFunction(p1, p2) {
    return p1 * p2;
}
document.getElementById("demo").innerHTML = myFunction(4, 3);
</script>

</body>
</html>
```

Javascript handling

- Tách mã javascript thành file .js
  - Tạo 1 file script.js
  - Chuyển các function javascript đã tạo vào file script.js
  - Nhúng file script.js vào file .html
    - <script type="text/javascript" src="script.js"></script>
- Cách gọi hàm từ các thẻ html
  - Gọi hàm thông qua sự kiện (event) như: onClick,… ví dụ:
    - <a href="#" onClick="tenHam()"> Text </a>
    - <a href="#" onClick="confirm('Bạn có muốn xóa ?')"Text </a>
  - Gọi hàm thông qua thuộc tính href của thẻ liên kết (<a>)
    - <a href="javascript:tenHam()">Text </a>
  - Vô hiệu hóa thẻ <a>
    - <a href="javascript:void()">Text </a>

Bài tập thực hành

- BT1 : Tạo hàm tính diện tích hình tròn
- BT2: Nhập n từ bàn phím , kiểm tra n là số chẵn hay lẻ