

FPT POLYTECHNIC



Bài 3: CẤU TRÚC I U KHI N, HÀM VÀ X LÝ S KI N

- Bit và ki u bit
- Ép ki u
- Khai báo bit
- Toán t
- C u trúc i u khi n
- L nh l a ch n n

- Cấu trúc i u khi n (ti p)
- Cấu trúc l a ch n
 - L nh l a ch n kép
 - L nh a l a ch n
- Cấu trúc l p
 - L p không bi t tr c s l n l p
 - L p bi t tr c s l n l p
- Hàm
- Ph m vi bi n
- M ng
- X lý s ki n

C Ứ TRÚC I Ụ KHI N

- Linh lach n n (Bài 2)
- Linh lach n kép
- Linh alach n

■ Cú pháp

```

if (dieukien) {
    //Th ả c hi ản
}
else {
    //Th ả c hi ản
}

```

■ Ví d ả :

```

var x = prompt("Hay nhap vao so be hon 100:");

```

```

if ( x > 50) {
    alert("Ban vua nhap gia tri la: " + x + ", gia tri nay lon hon 50");
}
else{
    alert("Ban vua nhap gia tri la: " + x);
}

```

■ Cú pháp

```
switch ( dieukien ) {  
    case giatri1:  
        //Thực hiện  
        break;  
    case giatri2:  
        //Thực hiện  
        break;  
    :  
    default:  
        //Thực hiện  
}
```

Sử dụng `break` thoát khỏi lệnh `switch`, nếu không sử dụng `break` thì chương trình sẽ tiếp tục thực hiện các case tiếp theo.

```
var diem = prompt("Hay nhap diem:")
switch ( diem ) {
    case "A":
        alert("Tuyet voi!");
        break;
    case "B":
    case "C":
        alert("Kha!");
        break;
    case "D":
    case "E":
    case "F":
        alert("Ban can co gang hon!");
        break;
    default:
        alert("Gia tri ban nhap vao khong hop le.");
}
```



```
var diem = prompt("Hay nhap diem:")
switch ( diem ) {
    case "A":
        alert("Tuyet voi!");
    case "B":
    case "C":
        alert("Kha!");
    case "D":
    case "E":
    case "F":
        alert("Ban can co gang hon!");
    default:
        alert("Gia tri ban nhap vao khong hop le.");
}
```

*Hãy cho biết kết quả của nó nếu nhập vào giá trị **Tuyệt vời!***

- Lệnh lặp không bị trễ cú pháp
 - While
 - Do ... While
- Lệnh lặp bị trễ cú pháp
 - For

■ Cú pháp

- Gi ng l nh Do While ... Loop

```
while (dieukien) {  
    //Th c hi n  
}
```

■ L i l p vô h n

- Thay i giá tr i u ki n

```
var x = 0;  
while ( x < 10) {  
    //Th c hi n  
    x++;  
}
```

Thay đổi giá
trị của x

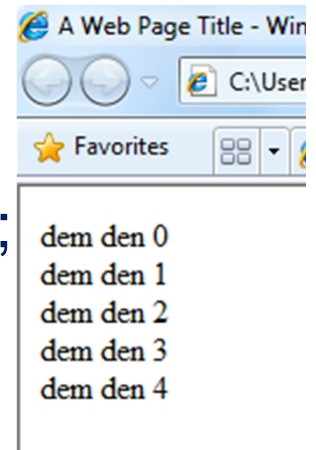
■ Cú pháp

- Gi ng l nh Do ... Loop While

```
do {
    //Th c hi n
} while ( ieukien )
```

- Ví d

```
var dem = 0;
do {
    document.write("dem den " + dem + "<br>");
    dem++;
}while (dem < 5)
```



■ Cú pháp

- Gi ng l nh For... Next

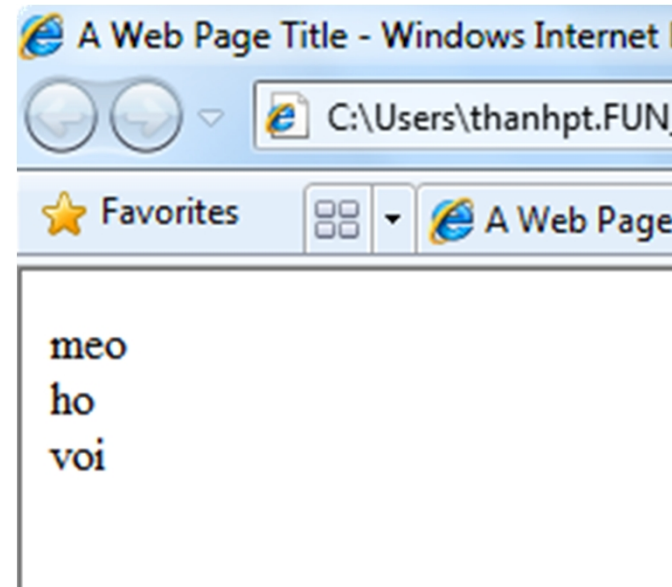
```
For (khoitao; dieukien; buoctang) {
    //Th c hi n
}
```

- Ví d

```
for (var x = 0; x < 10; x++) {
    document.write(x + "<br>");
}
```

```
var convat = new Array("meo", "ho", "voi");

for (var i = 0; i < convat.length; i++) {
    document.write(convat[i] + "<br>");
}
```



HÀM

- Hàm thể hiện một chức năng cụ thể (Giá trị trả về trong VB)

- Cú pháp

- Hàm không trả về giá trị (giá trị trả về Sub)

```
function tenham (thamso1, thamso2, ..., thamson) {  
    //Th thể hiện  
}
```

- Hàm trả về giá trị (giá trị trả về Function)

```
function tenham (thamso1, thamso2, ..., thamson) {  
    //Th thể hiện  
    return giatritrave;  
}
```


■ nh ngh a hàm

```
function myFunction (loichao1, loichao2 ) {  
    alert("Bien thu nhat la: " + loichao1);  
    alert("Bien thu hai la: " + loichao2);  
}
```

■ G i hàm

```
myFunction ("Hi", "Hello")
```

- Chú ý: L i g i hàm có th m t th script khác v i th script nh ngh a hàm

- nh nghĩa hàm

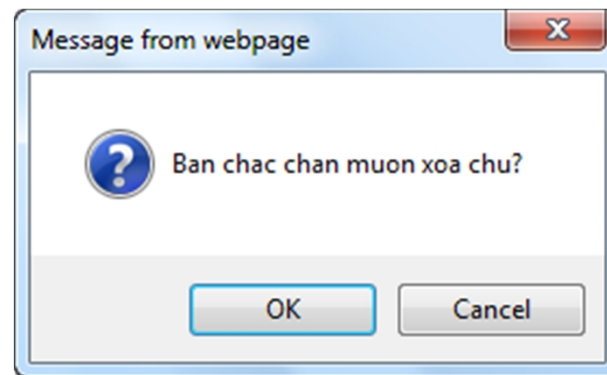
```
function nhanHaiSo (x, y) {  
    return x*y;  
};
```

- Gọi hàm

```
var tinh = nhanHaiSo (3, 4);  
alert (tinh);
```

- Là hộp thoại nhúng phía trên màn hình
- Logic hàm: Confirm(thông_ tin)
- Hàm trả về kết quả của người dùng
 - Trả về True nếu người dùng nhấn vào OK
 - Trả về False nếu người dùng nhấn vào Cancel

`confirm("Bạn chắc chắn muốn xóa chu?");`



```
function xacNhan(traloi) {
    var ketQua = "";
    if (traloi) {
        ketQua = "Tuyệt vời. Chúc bạn chiến thắng!";
    } else {
        ketQua = "Hẹn gặp lại bạn nhé!";
    }
    return ketQua;
}
```

```
var traloi = confirm("Bạn sẽ chơi game chưa?");
var thông báo = xacNhan(traloi);
alert (thông báo);
```

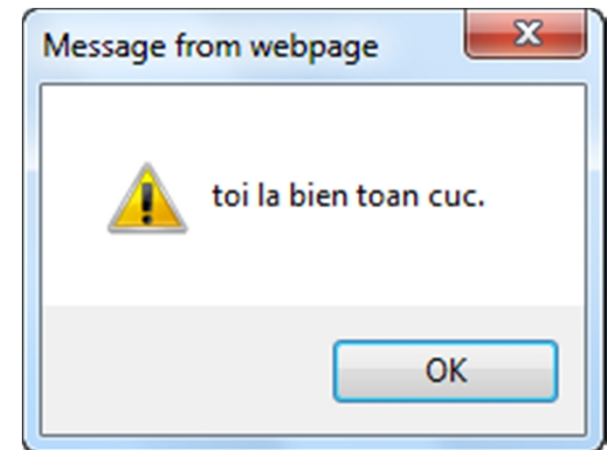
■ Bi n c c b

- Bi n c khai báo trong hàm
- Ch c tham chi u n trong ph m vi khai báo

■ Bi n toàn c c

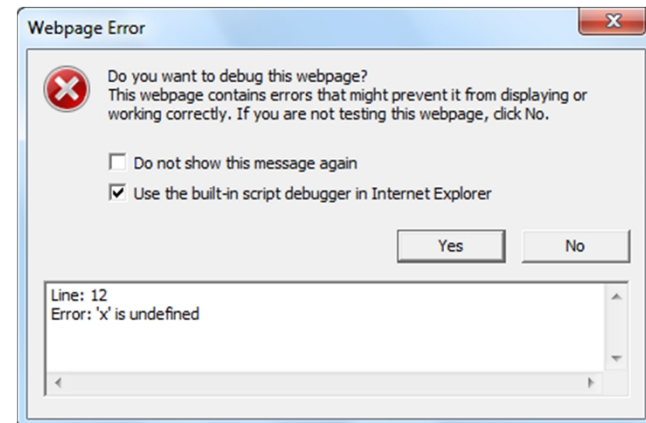
- Bi n c khai báo ngoài hàm
- Có th tham chi u n t b t c âu

```
<html>
<head>
  <title>Scoping Example</title>
  <script type="text/javascript">
    var x = "toi la bien toan cuc.";
  </script>
</head>
<body>
  <script type="text/javascript">
    alert(x);
  </script>
</body>
</html>
```



```
<html><head>  
  <title>Scoping Example</title>  
  <script type="text/javascript">  
    function() {  
      var x = "toi la bien cuc bo.";  
    }
```

```
  </script>  
</head>  
<body>  
  <script type="text/javascript">  
    alert(x);  
  </script>  
</body></html>
```



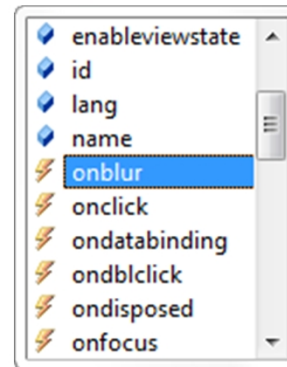
S K I N

- T t c các element trên trang web u có m t t p các s ki n t ng ng

```

1  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Trans
2  <html >
3  <head>
4  <script type="text/javascript">
5      function displayDate() {
6          alert(Date());
7      }
8  </script>
9  </head>
10 <body>
11 <button type="button" on>Display Date</button>
12 </body>
13 </html>
14

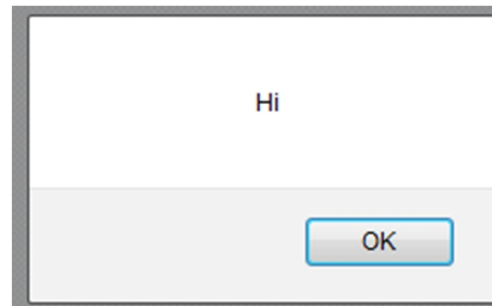
```



Chú ý: Cùng m t element, các browser khác nhau h tr các t p s ki n khác nhau

- **onClick**
 - c kích ho t khi nh n chu t vào m t element
- **onLoad và on Unload**
 - c kích ho t khi ng i dùng vào ho c thoát kh i trang web
- **onFocus, onBlur, onChange**
 - c kích ho t khi các tr ng nh n c focus, m t focus hay c thay i giá tr
- **onMouseOver**
 - c kích ho t khi ng i dùng di chu t lên m t element HTML trên form

```
<html >  
<head>  
</head>  
<body onclick="alert('Hi')">  
</body>  
</html>
```



Chú ý: IE không h tr s ki n onclick c a element body

- Có th thêm nhi u dòng l nh

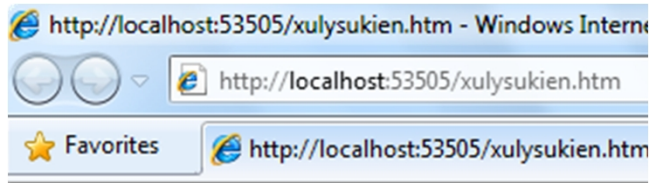
```
<html >  
<head>  
</head>  
<body onclick="alert('Hi'); alert('Hello');">  
</body>  
</html>
```

→ Trong tr ng h p x lý ph c t p cho s ki n????

- S d ng hàm th c hi n các x lý cho s ki n

```
<html >
<head>
<script type="text/javascript">
    function hienThiLoiChao() {
        alert("Hi");
        alert("Hello");
    }
</script>
</head>
<body>
<button type="button" onclick="hienThiLoiChao();">Hien thi loi chao</button>
</body>
</html>
```

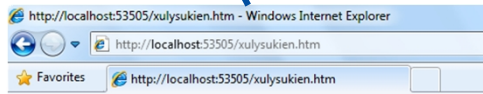
S ki n click c h tr trên c IE và Firefox



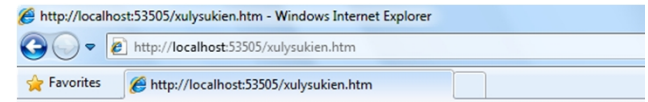
Hãy chọn đồ vật mà bạn thích:

Mũ Giày

Người dùng
nhấn vào Mũ



Người dùng
nhấn vào Giày



Xem Tai nguyen Xu ly su kien

```
<html ><head>
  <script type="text/javascript">
    function hienThiAnh(dovat) {
      if (dovat == "mu") {
        document.write("<img src = 'mu.jpg'>");
      } else {
        document.write("<img src = 'giay.jpg'>");
      }
    }
  </script>
</head>
<body>
<p> Hãy ch n v t mà b n thích:</p>
  <input type="button" value="M " onclick="hienThiAnh('mu');"/>
  <input type="button" value="Gi y" onclick="hienThiAnh('giay');"/>
</body>
</html>
```

- JavaScript cung cấp các phương thức xử lý các sự kiện thời gian
- Các phương thức này thuộc đối tượng window
- Một số phương thức quan trọng

Phương thức	Giải thích
setTimeout	Thực hiện công việc sau một khoảng thời gian trong tương lai
clearTimeout	Hủy bỏ setTimeout trước đó
setInterval	Thực hiện lặp lại công việc sau một khoảng thời gian
clearInterval	Hủy bỏ setInterval

■ Cú pháp

```
var t=setTimeout("L nh_javascript",s _mili _giây );
```

- L nh_javascript: Mã th c thi ho c l i g i hàm
- mili_giây: Sau th i gian này mã s c th c hi n
- setTimeout() tr v giá tr , giá tr c l u trong bi n t.
N u mu n h y b setTimeout, s d ng hàm clearTimeout và truy n vào i s t
- Cú pháp t ng t i v i setInterval()

```
<html>
<head></head>
<body id = "bd">
<input type="button" value = "Hãy nh ấn vào đây"
onclick="onClickEvent();" />
<script type="text/javascript">
    function onClickEvent() {
        var t = setTimeout("alert('Hi');", 1000);
    }
</script>
</body>
</html>
```

```
<html>
<head></head>
<body id = "bd">
<input type="button" value = "Hãy nh ấn vào đây"
onclick="onClickEvent();" />
<script type="text/javascript">
    function onClickEvent() {
        var t = setInterval("alert('Hi');", 1000);
    }
</script>
</body>
</html>
```

So sánh s khác nhau v ví d setTimeout

- JavaScript cung cấp 3 cấu trúc điều kiện: Cấu trúc tuần tự, cấu trúc lựa chọn và cấu trúc lặp
- Cấu trúc lựa chọn đơn (if), lựa chọn kép (if...else), lựa chọn (switch)
- Cấu trúc lặp gồm các lệnh lặp không biết số lần lặp (while, do...while) và lệnh lặp biết số lần lặp (for)
- Hàm thể hiện một chức năng cụ thể. Hàm có thể trả về giá trị hoặc không trả về giá trị
- Biến có phạm vi cục bộ và toàn cục

- Hàm Confirm là hàm được xây dựng sẵn dùng để lấy thông tin hỏi đáp từ người dùng
- JavaScript cung cấp sẵn kiểu dữ liệu cho các element của trang web. Mỗi element có một tập các thuộc tính khác nhau
- JavaScript cũng cung cấp các phương thức để xử lý các thuộc tính

XIN CHÀO MỌI NGƯỜI!