



JavaScript အားအသုံးပြုခြင်း

onLoad Event handler in Javascript

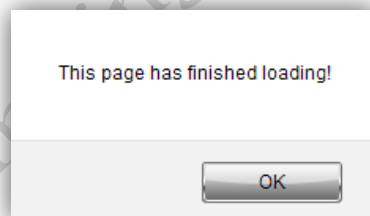
OnLoad event handler ကိုတော့ page တစ်ခုအား loading လုပ်နေချိန်၊ image အခုအား loading လုပ်နေချိန်... စတဲ့ အချိန်တွေမှာ user အတွက် message တစ်ခု ပြချင်တဲ့ အခါ၊ information တစ်ခု ပေးချင်တဲ့ အခါမျိုးမှာ အသုံးပြုနိုင်ပါတယ်။ လေ့လာကြည့်လိုက်ပါအုံးဗျာ။

```
<html>
<head><title>Body onload </title>
</head>
<body onload="alert('This page has finished loading!')">
</body>
</html>
```

(or)

```
<html>
<head><title>Body onload </title>
<script language="javascript">
function information( )
{
    window.alert('This page has finished loading!');
}
</script>
</head>
<body onload="information( )" >
</body>
</html>
```

Result:



Result က အတူတူပါပဲ။ ဒါကတော့ body loading ဥပမာလေးပါ။ ဒီလိုပါပဲ အောက်မှာကြည့်လိုက်ပါ။ မိမိ နှစ်သက်သလို အသုံးတည့်သလို အသုံးပြုနိုင်ပါတယ်။

```
<frameset onload="information( )" > //Execution of code //will begin after the current frame has loaded.
```

```
 //Execution of code will begin after the image //has loaded.
```

အဆင်ပြေသလို သုံးနိုင်ပါတယ်။ နားလည်မရဆိုးရင်တော့ ရှေ့ဆက်ကြအုံးစို့ဗျာ။

လေ့လာခြင်းဖြင့် ကျွန်ုပ်တို့၏ မနက်ဖြန်များကို ဖြတ်သန်းကြပါစို့....။



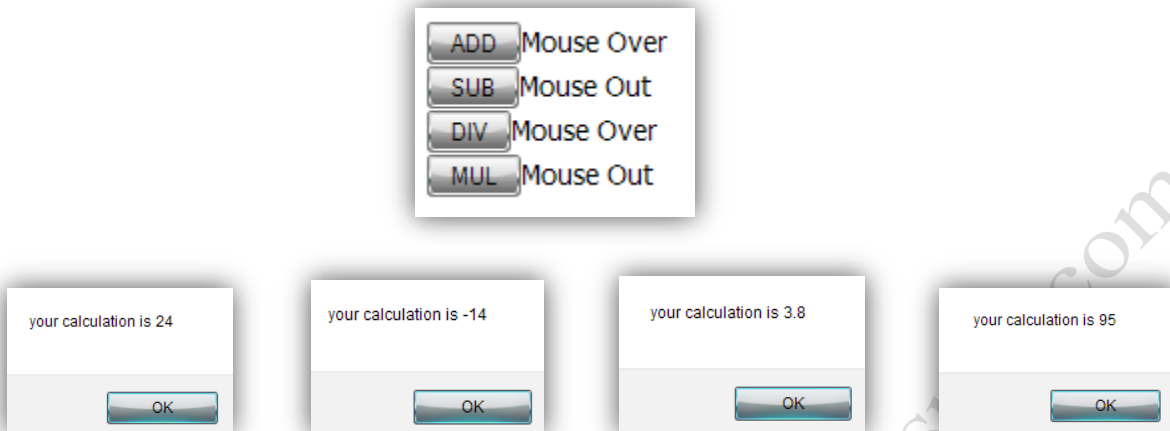
onMouseover, onMouseout Event Handlers in Javascript

Mouseover ရဲ့သဘောသဘာဝကတော့ link တခု၊ image တခု၊ button တခု၊ စသည်တို့ရဲ့ action လုပ်ဆောင်ချက်တွေ၊ ညွှန်းချက်တွေကို သိဖို့အတွက် ကလစ်နှိပ်စရာမလိုပဲ။ ၎င်းတို့ပေါ်မှာ ကာစာလေး တင်လိုက်ယုံနဲ့ အလုပ်လုပ်ပေးတဲ့ command တွေကို ဆိုလိုပါတယ်။ Mouseout ရဲ့ သဘောတရားကတော့ ကာစာလေး တင်ပြီး ကလစ်နှိပ်စရာမလိုပဲ ပြန်ထွက်လိုက်တဲ့ အချိန်မှာ သူတို့ရဲ့ လုပ်ဆောင်ချက် action တွေ၊ ညွှန်းချက်တွေကို အလုပ်လုပ်ပေးသွားတာပါ။ သဘောတရားကို နားလည်ပြီးဆိုရင်တော့ ကျနော်တို့ လေ့ကျင့်ခန်းလေး တခု လုပ်ကြည့်မယ်။

```
<html>
<head><title>Mouse Over and Mouse Out </title>
<script language="javascript">
function sub( )
{
  var a=5, b=19;
  var c=a-b;
  window.alert("your calculation is " +c);
}
function add( )
{
  var a=5, b=19;
  var c=a+b;
  window.alert("your calculation is " +c);
}
function div( )
{
  var a=5, b=19;
  var c=b/a;
  window.alert("your calculation is " +c);
}
function mul( )
{
  var a=5, b=19;
  var c=a*b;
  window.alert("your calculation is " +c);
}
</script>
</head>
<body>
<form>
<input type=button value="ADD" onMouseover="add( )">Mouse Over<br>
<input type=button value="SUB" onMouseout="sub( )">Mouse Out<br>
<input type=button value="DIV" onMouseover="div( )">Mouse Over<br>
<input type=button value="MUL" onMouseout="mul( )">Mouse Out
</form>
</body>
</html>
```



Result:



pic. Add

pic. Sub

pic. Div

pic. Mul

Result ကိုကြည့်လိုက်ပါ။ ပုံမှာ ပြထားတဲ့အတိုင်း code ကို run လျှင် run ချင်း ကျနော်တို့ page မှာ ပေါင်း၊ နှုတ်၊ မြှောက်၊ စား၊ ခလုပ်လေးခုပေါ်နေပါမယ်။ ဒီ button တွေကိုမှ ကျနော်တို့က onMouseover နဲ့ onMouseout တွေကိုသုံးပြီး calculation နဲ့ ပတ်သတ်တဲ့ function တွေကို ပြန်ခေါ်သုံးထားတဲ့ အတွက်ကြောင့် ကျနော်တို့ ADD, DIV button နှစ်ခုက ကာစာကို button ပေါ် တင်လိုက်တာနဲ့ အထပ်ပါ ပုံတွေအတိုင်း ကျနော်တို့က alert message နဲ့ result ကို ထုတ်ပြပေးပါလိမ့်မယ်။ SUB နဲ့ MUL button နှစ်ခုကတော့ mouse out event handler ကို အသုံးပြုထားတဲ့အတွက်ကြောင့် ကာစာကို button ပေါ်ကို တင်ပြီး ကလစ်နှိပ်စရာမလိုပဲ button ပေါ်မှ ကာစာကို ပြန်ဖယ်ပေးလိုက်တာနဲ့ အထပ်ပါ ပုံတွေအတိုင်း ကျနော်တို့ alert message နဲ့ result ကို ထုတ်ပေးပါလိမ့်မယ်။ ဒါတွေကတော့ Mouse Over, Mouse Out Event handlers တို့ရဲ့ လုပ်ဆောင်ပုံတွေပါ။ သဘောတရားကို သိပြီဆို ရင်တော့ ကျနော်တို့ အနေနဲ့ အဆင်ပြေသလို၊ အသုံးတည့်သလို တည့်ချင်း နှစ်သက်နဲ့ နေရာမျိုးမှာ အသုံးပြုနိုင်ပါတယ်။

onMouseDown, onMouseup Event Handlers in Javascript

Mouse Down event handler ရဲ့ သဘောသဘာဝကတော့ link တခု၊ image တခု၊ button တခု၊ စသည်တို့ရဲ့ action လုပ်ဆောင်ချက်တွေ၊ ညွှန်းချက်တွေကို သိဖို့အတွက် ၎င်းတို့အပေါ်ကို ကလစ်နှိပ်လိုက်တာနဲ့ အလုပ်လုပ်ပေးမဲ့ command ကိုဆိုလိုပါတယ်။ Mouse Up handler ကတော့ OnClick event handler နဲ့အတူတူပါပဲ။ မိမိအသုံးပြုနေသော link, image, button, ... စတာတွေအပေါ်ကို ကလစ်နှိပ်လိုက်ပြီး ပြန်လွတ်လိုက်တဲ့ အချိန်မှာ လုပ်ဆောင်ပေးမဲ့ action ပါ။ ကလစ်နှိပ်ထားပြီး မလွတ်မချင်း mouse up ရဲ့ action က အလုပ်လုပ်မှာ မလုပ်ပါဘူး။ သဘောတရားကို နားလည်ပြီဆိုရင်တော့ ကျနော်တို့ လေ့ကျင့်ခန်းလေးတွေ လေ့လာကြည့်ဖို့ဗျာ။ အောက်မှာ ကြည့်လိုက်ပါ ။

လေ့လာခြင်းဖြင့် ကျွန်ုပ်တို့၏ မနက်ဖြန်များကို ဖြတ်သန်းကြပါစို့....။



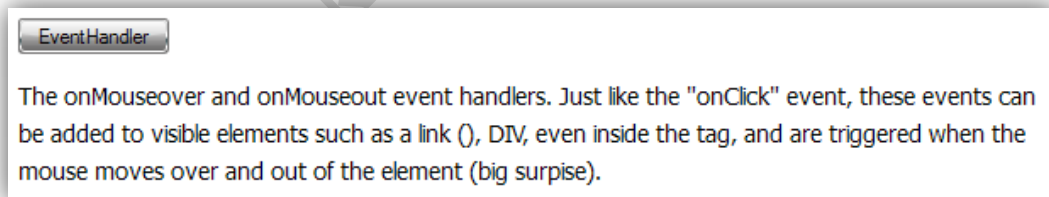
```

<html>
<head><title>Mouse Down and Mouse Up </title>
<script language="javascript">
function changetext(txt)
{
  document.getElementById("tutorial").innerHTML=txt;
}
</script>
</head>
<body >
<form>
<input type=button value= "Event handler"
onMouseDown="changetext('<b>MOUSE DOWN </b>-This is invoked when a mouse is pressed down
on the button ')"
onMouseUp="changetext('<b>MOUSE UP</b>-This is invoked when a mouse is pressed down on the
button and released')"
onMouseOver="changetext('<b>MOUSE OVER</b>-This is invoked when a mouse is moved over the
button')"
onMouseOut="changetext('<b>MOUSE OUT</b>-This is invoked when a mouse is moved over and then
out of the button')">

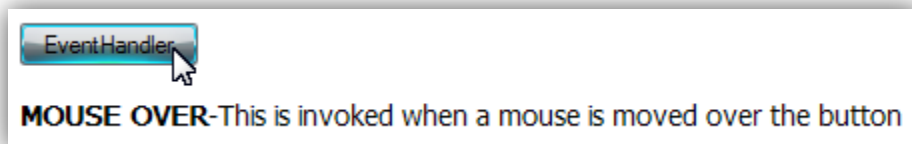
<p id="tutorial">The onMouseover and onMouseout event handlers. Just like the "onClick" event, these
events can be added to visible elements such as a link (<a>), DIV, even inside the <BODY> tag, and are
triggered when the mouse moves over and out of the element (big surprise).</p>
</form>
</body>
</html>

```

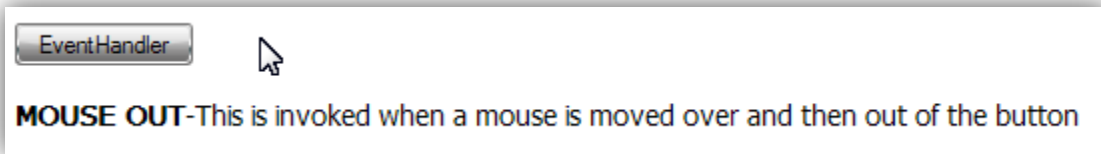
Result:



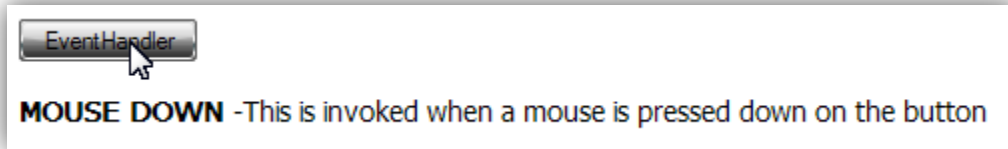
ဒါကတော့ ကျနော်တို့ ကုဒ်ကို run လျှင် run ချင်း မြင်တွေ့ရမရဲ့ မြင်ကွင်းပါ။ ဒါကိုမှာ ကျနော်တို့ အပေါ်မှာ ရေးခဲ့တဲ့ action တွေ အလုပ်လုပ်မလုပ်ဆိုတာကို စမ်းသပ်ကြည့်ကြမရဲ့။ အောက်မှာ လုပ်ဆောင်ချက် ဓါတ်ပုံလေးတွေ ကြည့်လိုက်ပါအုံးဗျာ။



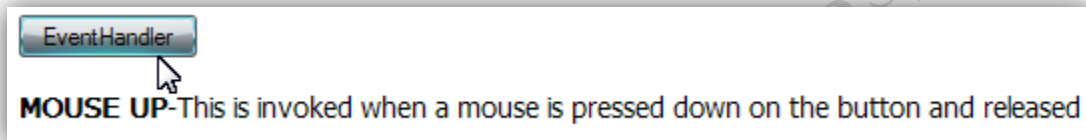
Pic. Mouse Over



Pic. Mouse Out



Pic. Mouse Down



Pic. Mouse Up

Result ပုံကိုတွေ ကြည့်လိုက်ရင်တော့ အလုပ်လုပ်သွားပုံတွေကို နားလည်မရ် ထင်ပါတရ်။ ဒီလေ့ကျင့်ခန်းကတော့ အပေါ်မှာ ကျနော်တို့ လေ့လာခဲ့ဖူးကြတဲ့ လေ့ကျင့်ခန်းတွေနဲ့ မတူပါဘူး။ ဒီလေ့ကျင့်ခန်းလေးမှာတော့ <style>tag အသုံးပြုပုံ ပုံစံနဲ့ ချည်းကပ်သွားတာပါ။ ကုဒ်ကို ကြည့်လိုက်ပါ။ ကျနော်တို့ paragraph ဖြစ်တဲ့ <p> tag ကို id=tutorial ဆိုပြီး ID ကို သတ်မှတ်ပေးခဲ့ပါတရ်။ ဒါကိုမှ ကျနော်တို့ event handlers တွေနဲ့ တွဲသုံးတဲ့နေရာမှာ document object နဲ့ id တွေကို ပြန်ခေါ်သုံးနိုင်တဲ့ method တခုဖြစ်တဲ့ getElementById("your ID") ကို အသုံးပြုပြီး ကျနော်တို့ <p>tag ကို သတ်မှတ်ပေးခဲ့တဲ့ tutorial ဆိုတဲ့ ID ကို ပြန်ခေါ်ထားပါတရ်။ ဒီမှာ တခု ထူးခြားတာက ကျနော်တို့ changetext() ဆိုတဲ့ function ကို txt ဆိုပြီး သတ်မှတ် လိုက်ပါတရ်။ ဒီတော့ ကျနော်တို့ onMouseOver(), onMouseDown,... အစရှိတဲ့ event handler တွေထဲမှာ ရိုက်ထည့်ထားတဲ့ message တွေအားလုံးဟာ changetext function ထဲကို ရောက်သွားပြီပေါ့။ ဒါကိုမှာ ကျနော်တို့က getElementById("your Id") ကိုအသုံးပြုပြီး text နဲ့ ညီပေးထားတဲ့ အတွက်ကြောင့် <p>tag paragraph နေရာကို event handler ရဲ့ message တွေဟာ ရောက်ကုန်တာပါ။ နားလည်မရ် ထင်ပါတရ်။

အခု Mousedown နဲ့ Mouseup event handler တို့ရဲ့ နောက်ဆုံးလေ့ကျင့်ခန်းလေး တပုဒ်အနေနဲ့ link tag နဲ့ image တို့နဲ့ တွဲဖတ် အလုပ်လုပ်ကြည့်ပါမရ်။ လေ့လာကြည့်လိုက်ပါအုံးဗျာ။

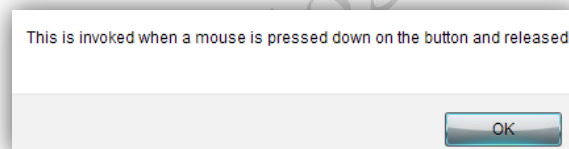
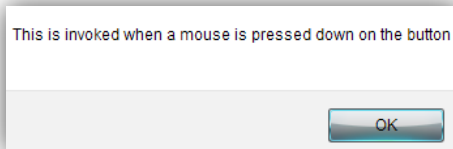
```
<html>
<head><title>Mousedown and Up handler with image & link</title>
</head>
<body>
<a href="http://google.com" onMousedown="alert('This is invoked when a mouse is pressed down on the button')">MOUSEDOWN</a><br>
```



```

</body>
</html>
```

Result:



Pic.Mouse Down with link

Pic.Mouse Up with mouse image

Result ကိုကြည့်လိုက်ရင် နားလည်သွားမရ် ထင်ပါတရ်။ ပထမ mouse down action ကို ကျနော်တို့ link ပုံစံနဲ့ ချိတ်ပေးထားပါတရ်။ Mouse Up action ကိုတော့ image နဲ့ တွဲပေးထားပါတရ်။ အလုပ်လုပ်သွားတဲ့ message တွေကိုတော့ အပေါ်မှာ ပုံနဲ့ ပြထားပါတရ်။ ကုန်နဲ့ တွဲပြီး ကြည့်လိုက်ပါ။ နားလည်သွားမရ် ထင်ပါတရ်။ မိတ်ဆွေ အပေါင်း လေ့လာခြင်းဖြင့် ကျေနပ်နိုင်ကြပါစေ။

ဒီနေ့တော့ ဒီလောက်နဲ့ပဲ နားကြအုံးစို့ဗျာ။ အားလုံးက အခြေခံလေးတွေ ဖြစ်လို့ စတင်လေ့လာသူများ အတွက်သာ ရည်ရွယ်ပါတရ်။ နောက်နေ့ အပိုင်းလေးမှာတော့ Javascript ရဲ့ Variables များအသုံးပြုခြင်း အပိုင်းလေးတွေကို ဆွေးနွေးသွားပါမရ်။

ဒါကတော့ Javascript အားအသုံးပြုခြင်း အပိုင်း(5)လေးပါ။ အားလုံးက အခြေခံလေးတွေဖြစ်လို့ စတင်လေ့လာသူများ အတွက်သာ ရည်ရွယ်ပါတရ်။ အခုမှ စဖတ်မိတဲ့ မိတ်ဆွေ၊ ညီအစ်ကို၊ မောင်နှမများအနေနဲ့ ရှေ့က ပိုစ်လေးကို အရင် ဖတ်စေချင်ပါတရ်။ ကျနော် မှားယွင်းတင်ပြမှုများ ရှိရင်လဲ နားလည်ပေးကြပေါ့ဗျာ။ ကျနော် ရည်ရွယ်ချက်ကတော့ အခြေခံလေးတွေကို မြန်မာလို ဖတ်ရှုခြင်းအားဖြင့် လျှင်မြန်စွာ နားလည် သွားနိုင်ပြီး ဒီထပ် မြင့်သော တခြားသော နယ်ပယ်များကို ကူးသွားနိုင်အောင်ပါ။ မိတ်ဆွေအပေါင်း လေ့လာခြင်းဖြင့် ကျေနပ်နိုင်ကြပါစေ။

Tay Zar Lin

Koyinmaung007@gmail.com

Programmingknowledge.blogspot.com

လေ့လာခြင်းဖြင့် ကျွန်ုပ်တို့၏ မနက်ဖြန်များကို ဖြတ်သန်းကြပါစို့....။