



## JavaScript အားအသုံးပြုခြင်း

### Comparison Operator in Javascript

Comparison Operator တွေကတော့ တန်ဖိုးနှစ်ခုကို နှိုင်းယှဉ်တဲ့ အခါမျိုးမှာ အသုံးပြုပါတယ်။ များသောအားဖြင့် flow chat, expression, statement တွေမှာ အသုံးများပါတယ်။ For, If, while,... စတဲ့ statement တွေမှာပေါ့ဗျာ။ Comparison operator ကို အသုံးပြုပြီး numerical, string, boolean စတာတွေကို compare လုပ်နိုင်ပါတယ်။ အောက်က Comparisinal operator လေးတွေကို လေ့လာကြည့်လိုက် ကြပါအုံးဗျာ။

Operator	Description	Example
==	Equal operator - returns <i>true</i> if both operands are equal.	x == 6 returns <i>true</i> if x is 6
!=	Not-equal operator - returns <i>true</i> if operands are not equal.	x != 6 returns <i>true</i> if x is not 6
<	Less-than operator - returns <i>true</i> if the left hand operand is less than the right hand operand.	x < 6 returns <i>true</i> if x is less than 6
>	Greater-than operator - returns <i>true</i> if the left hand operand is greater than the right.	x > 6 returns <i>true</i> if x is greater than 6
>=	Greater-than or equal operator - returns <i>true</i> if the left hand operand is greater than, or equal to the right.	x >= 6 returns <i>true</i> if x is either 6, or greater than 6
<=	Less-than or equal operator - returns <i>true</i> if the left hand operand is less than, or equal to the right.	x <= 6 returns <i>true</i> if x is either equal to 6, or less than 6

သဘောတရားကို နားလည်ပြီး ဆိုရင်တော့ ကျနော်တို့ လေ့ကျင့်ခန်းလေးတွေနဲ့ လေ့လာကြည့်ကြမယ်။

```

<html>
<head><title>Comparisinal Operator</title></head>
<body>
<script type="text/javascript">
var a = 10;
var b = 20;
var linebreak = "<br>";

document.write("(a == b) => ");
document.write(a == b);
document.write(linebreak);

document.write("(a != b) => ");
document.write(a != b);
document.write(linebreak);

document.write("(a < b) => ");
document.write(a < b);
document.write(linebreak);

document.write("(a > b) => ");
document.write(a > b);
document.write(linebreak);

document.write("(a >= b) => ");
document.write(a >= b);
document.write(linebreak);

```



```
document.write("a <= b) => ");
document.write(a <= b);
document.write(linebreak);
</script>
<p>Set the variables to different values and different operators and then try...</p>
</body>
</html>
```

Result:

```
(a == b) => false
(a != b) => true
(a < b) => true
(a > b) => false
(a >= b) => false
(a <= b) => true
Set the variables to different values and different operators and then try...
```

ဒါကတော့ comparisinal operator တွေရဲ့ အလွန်ကုဆုံး လုပ်ဆောင်ချက်လေးပါ။ ပြောပြစရာ မလိုပဲ နားလည်ကြမရဲလို့ ထင်ပါတယ်။ ဒီမှာ တခု ထူးခြားတာက ကျနော် အပေါ် logical operator မှာလဲ သုံးခဲ့ဖူးပါတယ်။ New line အောက်ဆင်းတာကို html code ဖြစ်တဲ့ <br>tag ကို script body ထဲမှာ ရေးမနေတော့ပဲ linebreak ဆိုတဲ့ variable တခုနဲ့ ပေးညှိထားတာ ထို variable ကိုပဲ script body ထဲမှာ ခေါ်သုံးထားတာကို တွေ့ရမှာပါ။ ခက်ခက်ခဲခဲတော့ မဟုတ်ပါဘူး။ နားလည်ကြမရဲ ထင်ပါတယ်။ နောက် လေ့ကျင့်ခန်းတခုလောက် ဆက်လေ့လာကြည့်ကြအုံးစို့ဗျာ။

```
<html>
<head><title>Comparisinal Operator</title></head>
<body>
<script language="javascript">
var customerName=prompt("Enter your name: ");
if(customerName=="Tayza") // (==) equal comparsional operator
{
    document.write("Hello TAYZA !!!! <br>");
}
else
{
    document.write("You are not Tayza. Where's Tayza !!! <br>");
}

var age=prompt("Enter your age: ");
if(age >=10 && age <= 50) //(>= and <=) comparsional operators
{
    if(age >= 10 && age <= 18) //(>= and <=) comparsional operators
    {
        document.write("You are a Teenager !!! <br>");
    }
    else
    {
        document.write("Your are not Teenager !!! <br>");
    }
}
```

လေ့လာခြင်းဖြင့် ကျွန်ုပ်တို့၏ မနက်ဖြန်များကို ဖြတ်သန်းကြပါစို့....။

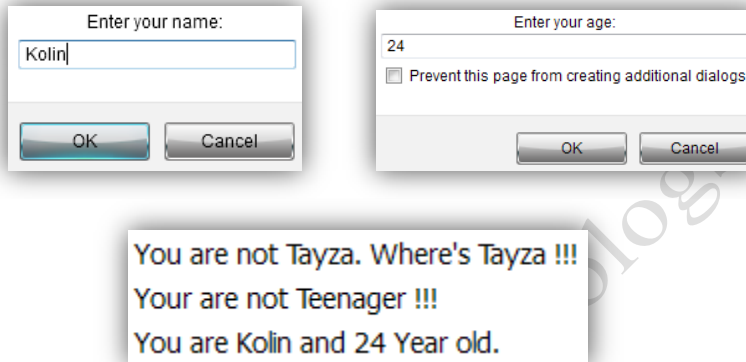


```

    }
else
{
    document.write("You are more than 50 year old !!! <br>");
}
document.write("You are "+customerName+ " and "+age+ " Year old.");
</script>
</body>
</html>

```

Result:



ဒီပုစ္ဆာမှာတော့ ကျနော်တို့ အပိုင်းနှစ်ပိုင်းခွဲပြီး ဆွေးနွေးထားပါတယ်။ ပထမ customerName ရိုက်ထည့်ခိုင်းတဲ့အပိုင်းမှာ (==)equal comparisinal operator ကို အသုံးပြုထားပြီး if...else statement နဲ့ တွဲသုံးကာ ရိုက်ထည့်လိုက်တဲ့ customerName ဟာ သတ်မှတ်ထားတဲ့ name ဟုတ်ရင် ဟုတ်တယ်။ မဟုတ်ရင် မဟုတ်ဘူးဆိုတဲ့ message ပေါ်လာမှာပါ။ ဒုတိယ age ရိုက်ထည့် ခိုင်းတဲ့ အပိုင်းမှာတော့ (>= and <=) ဆိုတဲ့ comparisinal operator နှစ်ခုကို အသုံးပြုထားပြီး Nested If condition statement ကို အသုံးပြုထားပါတယ်။ ရှင်းပြစရာ မလိုလောက်အောင်ကို ရိုးရှင်းပါတယ်။ ဒါပေမဲ့ ဒီလေ့ကျင့်ခန်းမှာ if statement တွေကို နားမလည်သေးရင်တောင် comparisinal operator တွေအကြောင်းတော့ နားလည်သွားမရလို့ ထင်ပါတယ်။ Statement တွေကို နားမလည်သေးတဲ့ မိတ်ဆွေ၊ ညီအစ်ကို၊ မောင်နှမများ အနေနဲ့ နောက်လာမဲ့ Statement သင်ခန်းစာတွေကို လေ့လာကြည့်ကြပေါ့ဗျာ။

### Assignment Operator in Javascript

Assignment operator ဆိုတာကတော့ variable တခုထဲကို တန်ဖိုးတခု ထည့်ချင် သတ်မှတ်ပေး လိုက်ချင်တဲ့ အခါမျိုးတွေမှာ အသုံးပြုပါတယ်။ Javascript မှာ အသုံးပြုနိုင်တဲ့ assignment operator ဘယ်နှစ်မျိုးရှိလဲ ဆိုတာကို အရင်လေ့လာကြည့်အုံးစို့ဗျာ။



Operator	Description	Example
=	Simple assignment operator, Assigns values from right side operands to left side operand	C = A + B will assign value of A + B into C
+=	Add AND assignment operator, It adds right operand to the left operand and assign the result to left operand	C += A is equivalent to C = C + A
-=	Subtract AND assignment operator, It subtracts right operand from the left operand and assign the result to left operand	C -= A is equivalent to C = C - A
*=	Multiply AND assignment operator, It multiplies right operand with the left operand and assign the result to left operand	C *= A is equivalent to C = C * A
/=	Divide AND assignment operator, It divides left operand with the right operand and assign the result to left operand	C /= A is equivalent to C = C / A
%=	Modulus AND assignment operator, It takes modulus using two operands and assign the result to left operand	C %= A is equivalent to C = C % A

**Note:** Same logic applies to Bitwise operators so they will become like <<=, >>=, >>=, &=, |= and ^=.

အမျိုးအစားတွေနဲ့ သဘောတရားလေးကို နားလည်ပြီးဆိုရင်တော့ ကျနော်တို့ လေ့ကျင့်ခန်းလေးတွေနဲ့ လေ့လာကြည့်ကြမရဲ့။

```

<html>
<head><title>Assignment operator</title></head>
<body>
<script language="JavaScript">
var linebreak="<br>";
var a = 4;
document.write("Value of a=>(a+=2) => "+(a+=2));      //(a+=2)→ a=4+2 → a=6
document.write(linebreak);
var b = 4;
document.write("Value of b=>(b-=2) => "+(b-=2));      //(b-=2)→ b=4-2 → b=2
document.write(linebreak);
var c = 4;
document.write("Value of c=>(c*=2) => "+(c*=2));      //(c*=2)→ c=4*2 → c=8
document.write(linebreak);
var d = 4;
document.write("Value of d=>(d/=2) => "+(d/=2));      //(d/=2)→ d=4/2 → d=2
document.write(linebreak);
var e = 4;
document.write("Value of e=>(e%=2) => "+(e%=2));      //(e%=2)→ e=4%2 → e=0
document.write(linebreak);
</script>
<p>Set the variables to different values and different operators and then try...</p>
</body>
</html>

```

Result:

```

Value of a=>(a+=2) => 6
Value of b=>(b-=2) => 2
Value of c=>(c*=2) => 8
Value of d=>(d/=2) => 2
Value of e=>(e%=2) => 0

```

Set the variables to different values and different operators and then try...

လေ့လာခြင်းဖြင့် ကျွန်ုပ်တို့၏ မနက်ဖြန်များကို ဖြတ်သန်းကြပါစို့....။



ဒီပုစ္ဆလေးကို ကြည့်လိုက်ရင်ပဲ assignment operator တွေအကြောင်းကို နားလည်သွားမရ် ထင်ပါတရ်။ ဒါကတော့ အရိုးရှင်းဆုံး ပုစ္ဆလေးပေါ့။ အခု ကျနော်တို့ for loop လေးကို အသုံးပြုပြီး ကိန်းစဉ်တန်းတခုရဲ့ မြောက်ခြင်း၊ ပေါင်းခြင်း၊ နှစ်ခုကိုလေ့လာကြည့်ကြရအောင်ဗျာ။

```
<html>
<head><title>Assignment Operator</title></head>
<body>
<script language="javascript">
var i;
var Sum=0;
var Mul=1;
for(i=1; i<=10; ++i)
{
    Sum += i;
    Mul *= i;
}
document.write("Additon of series(1 to 10)= " +Sum+ "<br>");
document.write("Multiplication of series(1 to 10)= "+Mul);
</script>
</body>
</html>
```

Reuslt:

Additon of series(1 to 10)= 55  
Multiplication of series(1 to 10)= 3628800

ဒီပုစ္ဆလေးမှတော့ for loop ကို အဓိကထားပြီး assignment operator တွေဖြစ်တဲ့ (+= and \*=) ကို အသုံးပြုထားတာပါတရ်။ For statement ရဲ့ လုပ်ဆောင်ချက် သုံးခုနဲ့ အလုပ်လုပ်ပါတရ်။ ပထမ assign လုပ်တရ်။ ဒုတိယ condition စစ်တရ်။ တတိယ increament, decrement လုပ်တရ်။ အဲ့ဒီ လုပ်ဆောင်ချက် သုံးခု မှန်နေသေးသ၍ သူရဲ့ block အတွင်းက operation တွေကို ဆက်ခါ ဆက်ခါ အလုပ်လုပ်နေအုံးမှာပါပဲ။ ကိန်းစဉ်တန်း ပေါင်းသွားပုံကို ကျနော် အနည်းငယ် ပြောပြပါမရ်။ ပထမဆုံး sum ကို သုညအဖြစ် assign လုပ်ထားတရ်။ ဒါကိုမှ for loop ထဲမှာ i=1 ကစပြီး i<=10 ဖြစ်နေတဲ့ အခြေအနေထိ i ကို တစ်ပေါင်းပေးသွားပါလိမ့်မရ်။ Looping တခုပတ်တိုင်း += ဖြစ်တဲ့ assignment operator က တခါအလုပ်လုပ်ပါတရ်။ ပထမ looping မှာ sum+=i ဖြစ်တဲ့အတွက်ကြောင့် sum=0+1=1 ဖြစ်မရ်။ ဒုတိယ looping မှာတော့ I တန်ဖိုးက 2 ဖြစ်သွားသလို sum တန်ဖိုးက 1 ဖြစ်သွားပါပြီး။ ဒီတော့ sum+=I မှာ sum=1+2=3 ဖြစ်သွားပါပြီ။ ဒီလိုနဲ့ looping ကို i<=10 ဖြစ်တဲ့ အခြေအနေထိ အလုပ်လုပ်သွားပြီး နောက်ဆုံး sum ထဲကို ထပ်ခါတလဲလဲ ပေါင်းထည့်လို့ရလာတဲ့ နောက်ဆုံးတန်ဖိုးကို ပြန်လည်ထုတ်ပြတာဖြစ်ပါတရ်ဗျာ။ Multiplication အခန်းမှာလဲ ဒီလိုပါပဲ။ ဒီလောက်ဆိုရင်တော့ assignment operator တွေရဲ့ အလုပ်လုပ်ပုံကို နားလည်လောက်ပြီ ထင်ပါတရ်။



ဒီနေ့တော့ ဒီလောက်နဲ့ပဲ နားကြအုံးစို့ဗျာ။ အားလုံးက အခြေခံလေးတွေ ဖြစ်လို့ စတင်လေ့လာသူများ အတွက်သာ ရည်ရွယ်ပါတယ်။ နောက်နေ့ အပိုင်းလေးမှာတော့ Javascript မှာ အသုံးပြုနိုင်မရဲ့ operators လေးတွေထဲကမှ bitwise operators, typeof operator, ... စတာလေးတွေရဲ့ အသုံးပြုခြင်း အပိုင်းလေးတွေကို ဆွေးနွေးသွားပါ့မယ်။

ဒါကတော့ Javascript အားအသုံးပြုခြင်း အပိုင်း(9)လေးပါ။ အားလုံးက အခြေခံလေးတွေဖြစ်လို့ စတင်လေ့လာသူများ အတွက်သာ ရည်ရွယ်ပါတယ်။ အခုမှ စဖတ်မိတဲ့ မိတ်ဆွေ၊ ညီအစ်ကို၊ မောင်နှမများအနေနဲ့ ရှေ့က ပို့စ်လေးကို အရင် ဖတ်စေချင်ပါတယ်။ 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**