**Experiment 2:**  
   
**Write an HTML page including javascript that takes a given set of integer numbers and shows**

**them after sorting in descending order.**

Description:

The sort callback has to return

* a negative number if a < b
* 0 if a === b
* a positive number if a > b

Three possible return values are needed because the sort function needs to now whether a is smaller than, equal to, or larger than b in order to correctly position a in the result array.

It is very common to just return -1, 0 and 1 if you working with non-numerical data . But if you use numerical data, you can simply subtract the values

because

* if a < b then a - b < 0, i.e. a negative number
* if a === b then a - b === 0, i.e. 0
* if a > b then a - b > 0, i.e. a positive number

**The Compare Function**

The purpose of the compare function is to define an alternative sort order.

The compare function should return a negative, zero, or positive value, depending on the arguments:

function(a, b){return b-a}

When the sort() function compares two values, it sends the values to the compare function, and sorts the values according to the returned (negative, zero, positive) value.

Example:

When comparing 40 and 100, the sort() method calls the compare function(40,100).

The function calculates 100-40, and returns 60 (a positive value).

The sort function will sort 100 as a value greater than 40.

**Sortdesc.html**

<html>

<head>

<title>Number in Descending Order</title>

<script language="javascript">

function ndesc()

{

var num\_array=new Array();

var num=document.forms["frm1"].num.value;

document.forms["frm1"].desc.value="";

var nums = num.split(',');

var len=num.split(',').length;

for(var i=0;i<len;i++)

{

num\_array.push(nums[i]);

}

function sortN(a,b)

{

return b - a;

}

document.forms["frm1"].desc.value= num\_array.sort(sortN);

}

</script>

</head>

<body>

<form name="frm1">

<center>

<h3> Numbers in Descending Order</h3>

</center>

<br/>

<center>Enter Numbers separated by Comma : <input type="text" name="num"</input><br/></center>

<br/>

<center>

<input type="button" name="inwords" value="In Descending Order" onclick="ndesc()"></input>

</center>

<br/><br/>

<center>Number in Descending Order : <input type="text" name="desc"</input></center>

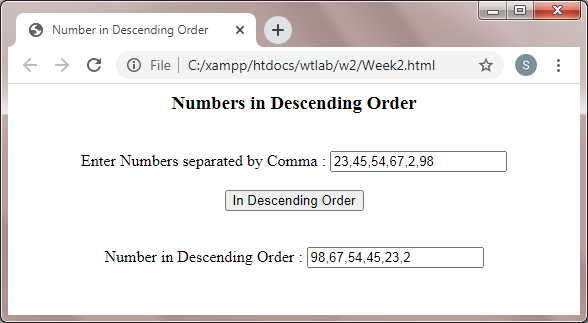
<br/>

</form>

</body>

</html>

**Output:**



**Experiment 3:**  
 **Write an HTML page including any required JavaScript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. if the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.**

**Week3.html**

<html>

<head>

<title>Number in words</title>

<script language="javascript">

function convert()

{

var num=document.forms["frm1"].num.value;

document.forms["frm1"].words.value="";

if(isNaN(num))

{

alert("Not a Number");

}

else if (num<0 || num>999)

{

alert("Out of Range");

}

else

{

var len=num.length;

var words="";

for(var i=0;i<len;i++)

{

var n=num.substr(i,1);

switch(n)

{

case '0':words+="Zero ";break;

case '1':words+="One ";break;

case '2':words+="Two ";break;

case '3':words+="Three ";break;

case '4':words+="Four ";break;

case '5':words+="Five ";break;

case '6':words+="Six ";break;

case '7':words+="Seven ";break;

case '8':words+="Eight ";break;

default:words+="Nine ";

}

}

document.forms["frm1"].words.value=words;

}

}

</script>

</head>

<body>

<form name="frm1">

<center><h3>NUMBER IN WORDS</h3></center>

<br/>

<center>Enter a Number :<input type="text" name="num"</input><br/></center>

<br/>

<center><input type="button" name="inwords" value="In Words" onclick="convert()"></input></center>

<br/><br/><center>Number in Words :<input type="text" name="words"</input></center>

<br/>

</form>

</body>

</html>

**Output:**

