**A function is a block of reusable code.** A function allows us to write code once and use it as many times as we want just by calling the function.  
  
  
  
**JavaScript function syntax**  
function functionName(parameter1, parameter2,..parameter\_n)  
{  
  //function statements  
}  
  
**Points to remember**  
1. Use the function keyword to define a function, followed by the name of the function. The name of the function should be followed by parentheses ().  
2. Function parameters are optional. The parameter names must be with in parentheses separated by commas.  
  
**Example :**JavaScript function to add 2 numbers. The following JavaScript function adds 2 numbers and return the sum  
  
function addNumbers(firstNumber, secondNumber)   
{  
    var result = firstNumber + secondNumber;  
    return result;  
}  
  
**Calling the JavaScript function :** Call the JavaScript function by specifying the name of the function and values for the parameters if any.  
  
var sum = addNumbers(10, 20);  
alert(sum);  
  
**Output :** 30  
  
**What happens when you do not specify values for the function parameters when calling the function**  
The parameters that are missing values are set to undefined  
  
**Example :**In the example below, we are passing 10 for the firstNumber parameter but the secondNumber parameter is missing a value, so this parameter will be set to undefined. When a plus (+) operator is applied between 10 and undefined we get not a number (NaN) and that will be alerted.  
  
function addNumbers(firstNumber, secondNumber)   
{  
    var result = firstNumber + secondNumber;  
    return result;  
}  
  
var sum = addNumbers(10);  
alert(sum);  
  
**Output :** NaN  
  
**What happens when you specify too many parameter values when calling the function.**  
The extra parameter values are ignored.  
  
**Example :**In the example below, 30 & 40 are ignored.  
  
function addNumbers(firstNumber, secondNumber)   
{  
    var result = firstNumber + secondNumber;  
    return result;  
}  
  
var sum = addNumbers(10, 20, 30, 40);  
alert(sum);  
  
**Should a javascript function always return a value**  
No, they don't have to. It totally depends on what you want the function to do. If an explicit return is omitted, undefined is returned automatically. Let's understand this with an example.  
  
The following function returns the sum of two numbers. We are storing the return value of the function in sum variable and writing it's value to the document.  
  
function addNumbers(firstNumber, secondNumber)   
{  
    var result = firstNumber + secondNumber;  
    return result;  
}  
  
var sum = addNumbers(10, 20);  
document.write(sum);  
  
The following function does not return any value. It simply writes the sum of two numbers to the page. However, we are assigning the return value of addNumbers() function to sum variable. Since addNumbers() function in this case does not have an explicit return statement, undefined will be returned.  
  
function addNumbers(firstNumber, secondNumber)   
{  
    var result = firstNumber + secondNumber;  
    document.write(result);  
}  
  
var sum = addNumbers(10, 20);  
alert(sum);