**Functions**

* Function is a block of code.
* It is executed when it will invoked.
* It is defined with **function** keyword followed by its name and then parenthesis.

For example: function myFunction(){};

* Parentheses include parameters separated by comma.
* Function arguments are the values received by the function when it is invoked.
* When function reaches return statement, the function will stop executing.
* Return value is returned to the caller.
* Invoking function without parentheses return the function object not the value.

**Anonymous function**

* A function without name.
* They are stored in variable and called using variable name.

**Function() Constructor**

* Function can also be defined using built-in javascript function constructor called Function().
* Hoisting is applied to the function declaration that is function is called before it is declared.
* Functions are self-invoking. To invoke the function automatically, add parentheses at the end of function declaration and put function declaration between the parentheses.
* Arguments.length property returns the number of arguments received when the function was invoked.
* If an argument is less than the declared, then its value is set to undefined.

**Function Invocation**

* The code inside a function will execute when it is invoked.

**Function as a method**

* Function defined as the property of an object, is called method of an object.
* Call() method is used to call the method of an another object.
* Syntax: Ownerobj.methodname.call(anotherobj, arguments).
* Apply() method is same as call method.
* Only difference is it take argument as an array.
* Apply() method is used to apply any in built method of javascript.
* **Bind()** method is used to borrow method of other object.