JavaScript Closure

**A Closure is a function having access to the parent scope, even after the parent function is closed**

A closure is a combination of functions. In JavaScript, an inner function always takes some reference from the outer function.

A closure gives you access to an outer function scope from an inner function.

Closure is created every time a function is being created, at function creation time.

When the inner function is created as a return type of the parent function a closure is also being created in JavaScript.

The parent function has been called in a variable, this variable is now a function, because the return type of the parent function is a function.

The javascript inner function creates a closure or stores the temporary data that is declared in the parent function. Thus after closing the parent function the internal function can access the variable of the parent function.

If no variable of the parent function has been used in the return type anonymous function then javascript removes all the garbage collection.

A parent function always sends the value or variable to its child function. But The parent can’t access the child’s element.