LEARN. DO. EARN

ACADGILD

FRONT END DEVELOPMENT (WITH ANGULARJS)

Website : http://www.acadgild.com
LinkedIn : https://www.linkedin.com/company/acadgild
Facebook : https://www.facebook.com/acadgild

Course Modules

1. Advanced JavaScript
2. jQuery
3. BootStrap
4. AngularJS

Agenda – Advanced JavaScript

1. Introduction to Object Oriented
2. Terminology
3. Functions
4. Anonymous Function
5. Immediate Function

Introduction to Object Oriented

- Object-oriented programming language or OOP is based on objects, fields and methods.
- Object-oriented programming allows you to reuse code without having to copy or recreate it.
- Many popular programming languages (such as Java, JavaScript, C#, C++, Python, PHP, Ruby and Objective-C) support object-oriented programming (OOP).

Terminology

Namespace

Namespace is a container which allows developers to bundle all functionality under a unique, application-specific name.

Property

Property is an object characteristic, such as color.

Method

Method is an object capability, such as walk. It is a subroutine or function associated with a class.

Constructor

A constructor is a method that is called at the moment of instantiation of an object. It usually has the same name as that of the class containing it.

Class

A class defines the characteristics of the object. It is a template definition of variables and methods of an object.

Terminology

Object

An object is an instance of a class where the object can be a combination of variables, functions and data structures.

Inheritance

An object or class can inherit characteristics from another object or class.

Encapsulation

Encapsulation is the a method of bundling the data and methods that use them together into a single component.

Abstraction

The conjunction of complex inheritance, methods, properties of an object must be able to simulate a reality model.

Polymorphism

Poly means "many" and morphism means "forms". Different classes might define the same method or property.

Functions

- A JavaScript function is a block of code designed to perform a particular task.
- You can define the code once and use it many times.
- A JavaScript function is executed when "something" invokes it (or calls it).

Syntax

```
function name(parameter1, parameter2, parameter3)
{
   code to be executed
   return value;
}
```

Anonymous Function

- Anonymous Function is a function without a name.
- Sometimes you need a simple function without the need of assigning it to a name. This is called anonymous function.

Example:

- **var** nums = [1, 4, 3, 2, 6, 2, 0];
- nums.sort(**function**(a,b){**return** a-b; });

- Can access the anonymous function by assigning it to a variable

```
var sayHello = function () {
        alert("Hello World !");
};
```

Immediate Function

- An immediate function executes as soon as JavaScript encounters them.
- Also called as IIFE (Immediately Invoked Function Expression)
- It is also referred to as self-invoking function.

```
var myName = ‘Acadgild‘;

(function(thisName){
   console.log( 'hello, my name is: ' + thisName );
}(myName))
```

Agenda – Advanced JavaScript

1. Inner Functions
2. Closures
3. Closure - Example

Inner Functions

- Nesting functions within functions.

```
function Ftimes() {
   var FtimesObj = new Object()
         function Ftimes3(x) {
            return x * 3
            }
         function Ftimes4(x) {
            return x * 4
            }
   FtimesObj.Ftimes3 = Ftimes3
   FtimesObj.Ftimes4 = Ftimes4
   return FtimesObj
   }

Var Multi = new Ftimes();
alert(Multi.Ftimes3(5)) // alerts 15
alert(Multi.Ftimes4(5)) //alerts 20
```

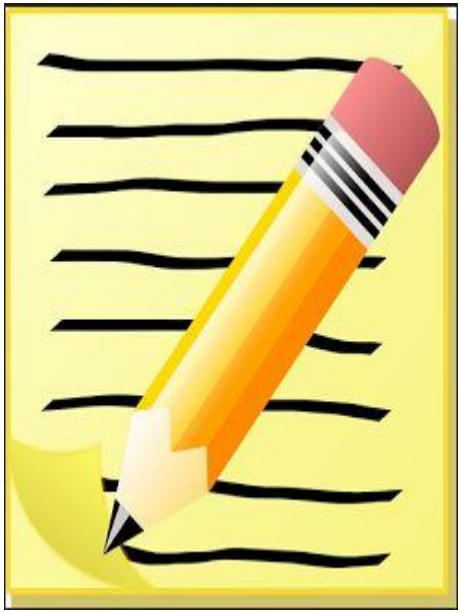
Closures

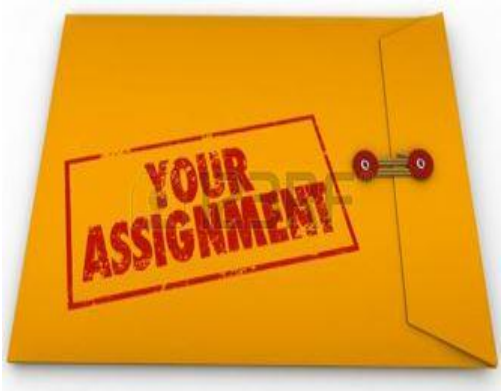
- A closure is an inner function that has access to the outer (enclosing) function's variables—scope chain.
- A closure takes place when a function creates an environment that binds local variables to it in such a way that they are kept alive after the function has returned.
- A closure is a special kind of object that combines two things: a function and any local variables that were in-scope at the time that the closure was created.

Closure - Example

```
function getNameFunction() {

        var name = “Acadgild";

        return function getName() { return name; }
        }

        var displayName = function() {
        var getName = getNameFunction();
        alert( "Hello " + getName() + "!" );
         return getName;
        }(); //holds the getName() function

alert(displayName); //call it again later...

setTimeout( 'alert( "Your name is " + displayName() )', 10 );
```

Lets Discuss Assignments





Assignment

Get in Touch with ACADGILD

Contact Info:

- **Website : http://www.acadgild.com**
- **LinkedIn : https://www.linkedin.com/company/acadgild**
- **Facebook : https://www.facebook.com/acadgild**
- **Support: support@acadgild.com**