**JS properties** are values associated with JS object. JS object is collection of unordered properties.

JS properties are values associated with objects.

Person.name = Shruthi;

Here name is the property.

**Adding methods/property to existing class using prototype:**

Person. Nationality = “English”;

function Person(first, last, age, eyecolor) {  
    this.firstName = first;  
    this.lastName = last;  
    this.age = age;  
    this.eyeColor = eyecolor;  
}  
Person.prototype.nationality = "English";

**Creating an object:**

methodName : function() {

}

Accessing: objectName.methodName()

**Calling request xmlhttp request:**

var xhttp = new XMLHttpRequest();  
xhttp.onreadystatechange = function() {  
    if (this.readyState == 4 && this.status == 200) {  
       // Typical action to be performed when the document is ready:  
       document.getElementById("demo").innerHTML = xmlhttp.responseText;  
    }  
};  
xhttp.open("GET", "filename", true);  
xhttp.send();

**JS object:**

Var person = { firstname: ”Shruthi”,Lastname:”Mekarthi”,age:25};

**JS methods** are actions that can be performed on objects.

**Prototype:**

Every JS object has prototype. All JS objects inherit properties and methods from prototype.

**Inheritance:** If one object acquires properties and methods of parent object is known as inheritance.

**Self-invoking function**: A function that is called by itself.

**XML HTTP request** object is used to request data from server. It is developers dream. We can request the data from a server, update a page without reloading the page.

**Setintervel**: Setintervel is used to repeat particular task at the given time interval.

**Hoisting** is JavaScript’s default behavior of moving all declarations to the top of current scope.