Class in ES6

We can declare a class in ES6 using keyword “class” along with curly braces as its body. The properties in the class we declare is prototype to its sub class automatically we need not to make any property prototype forcefully. We Assign the properties inside a constructor() method. In class we can have static function also

Here is sample JavaScript code that showing the syntax for creating a class.

class Animal

{

constructor (name, type)

{

this. name = name;

this. type = type;

}

}

var dog = new Animal ('Dog’, ‘domestic');

var Lion = new Animal('Lion’,’ wild');

console.log (dog);

console.log (Lion);

Extending a Class

We can extend the function of a class using extend keyword. An advantageous feature of constructor functions and classes is that they can be extended into new object blueprints based off of the parent. This prevents repetition of code for objects that are similar but need some additional or more specific features.

Ex:-

class Bird extends Animal{

constructor (name, type, food)

{

super(name, type);

this.food = food;

}

}

var parrot = new Bird('parrot','domestic','peanut');

var peagon = new Bird('peagon','domestic’, ‘wheat');

console.log (parrot);

console.log (peagon);

## Object.create () function

## Object.create builds an object that inherits directly from the one passed as its first argument.

var obj = Object.create(animal);

console.log (obj);

Ajax (**A**synchronous **J**avaScript **and** **X**ML)

It is the use of the [XMLHttpRequest](https://developer.mozilla.org/en/DOM/XMLHttpRequest) object to communicate with servers. It can send and receive information in various formats, including JSON, XML, HTML, and text files. AJAX’s most appealing characteristic is its "asynchronous" nature, which means it can communicate with the server, exchange data, and update the page without having to refresh the page.

The two major features of AJAX allow you to do the following:

* Make requests to the server without reloading the page
* Receive and work with data from the server

function getStudent(e)

{

var xhr = new XMLHttpRequest()

console.log(xhr);

xhr.open('GET','student.json',true);

xhr.onload = function()

{

if(this.status === 200)

{

console.log(xhr);

const student = JSON.parse(this.responseText);

const result = `<ul>

<li> name : ${student.name}</li>

<li> name : ${student.roll}</li>

<li> name : ${student.location}</li>

</ul>`

document.getElementById('studentData').innerHTML = result;

}

}

xhr.send();}

var data = document.getElementById('getData');

data.addEventListener('click',getStudent);