**Curso de JSON**

**Danisable (Daniel Jimenez)**

# Aula 04 - Tipos de dados

## Principais tipos de dados

STRINGS:

"maçã"

"Daniel"

"@#$!%&"

"\n", "\t" (sequências de escape)

NUMBERS:

3

9.5

-5

BOOLEAN:

true 1

false 0

NULL:

null

## Tipos de dados compostos

VALUES:

"Daniel" + "Jimenez"

5 + 10

OBJECTS:

obj1 = {

nome: "Roberto",

sobrenome: "Pinheiro",

idade: 60,

sexo: "masculino"

}

ARRAYS:

nomes = ["Carlos", "Eduardo", "Mônica"]

# Aula 05 - Repasso de Javascript

**aula\_05\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Repasso de Javascript</title>

<script>

alert("Olá, como você se chama?")

var nome;

nome = prompt("Escreva seu nome aqui", "nome");

alert("Olá, " + nome + "! Seja bem-vindo!");

</script>

</head>

<body>

Olá! Este é um texto HTML.

</body>

</html>

# Aula 06 - Estrutura de um objeto

## Básico de um objeto JSON

- Chaves definem um objeto.

{

"key1": "value1",

"key2": "value2",

"key3": "value3"

.

.

.

}

- As chaves devem ser strings válidas e sem espaços em branco.

# Aula 07 - Exemplo de objeto JSON

{

"id": 1,

"nombre": "puerta de roble",

"precio": 250.50,

"etiquetas": ["casa", "madera", "roble"]

}

# Aula 08 - Comparação entre objeto em JSON e objeto em Java

## Objeto em JSON

{

"id": 1,

"nombre": "puerta de roble",

"precio": 250.50,

"etiquetas": ["casa", "madera", "roble"]

}

## Objeto em Java

public class Puerta {

int id = 1;

String nombre = "puerta de roble";

float precio = 250.50;

String[] etiquetas = new String[] {"casa", "madera", "roble"};

}

public static void main(String[] args){

Puerta puerta1 = new Puerta(); /\* Criação de um objeto \*/

}

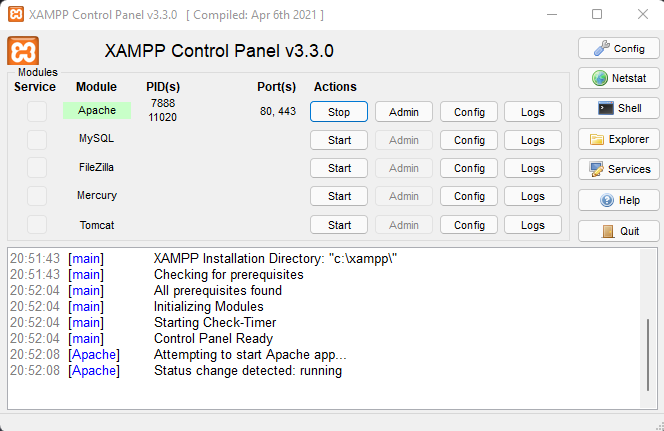
# Aula 09 - Instalação de servidor XAMPP

- Baixe e instale, em seu computador, o XAMPP

- Inicie o servidor Apache (no painel do XAMPP, clique em start).

http://localhost/json\_danisable/aula\_05/

# Aula 10 - Configurar XAMPP



**aula\_10\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<h1>Curso JSON desde cero</h1>

</body>

</html>

**http://localhost/json\_danisable/aula\_10/**



# Aula 11 - Acessar dados de um objeto

**aula\_11\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

nombre: "puerta de roble",

precio: 250.50,

etiquetas: ["casa", "madera", "roble"]

}

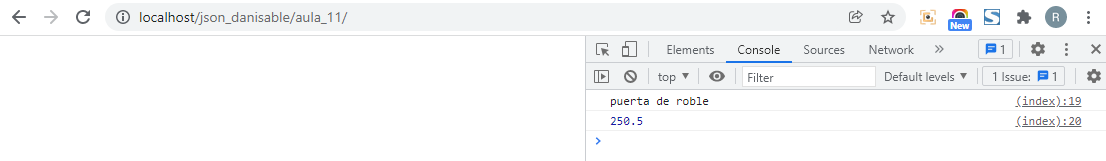
console.log(puerta["nombre"]);

console.log(puerta.precio);

</script>

</body>

</html>



# Aula 12 - Acessar dados de um array

**aula\_12\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

nombre: "puerta de roble",

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"]

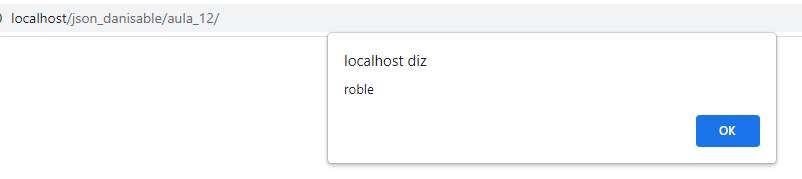
}

alert(puerta.etiquetas[2]);

</script>

</body>

</html>



# Aula 13 - Funções como atributos

**aula\_13\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

function decirHola(){

alert("Hola!");

}

var puerta = {

id: 1,

nombre: "puerta de roble",

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

decirAdios: function(){

alert("Adios!");

}

}

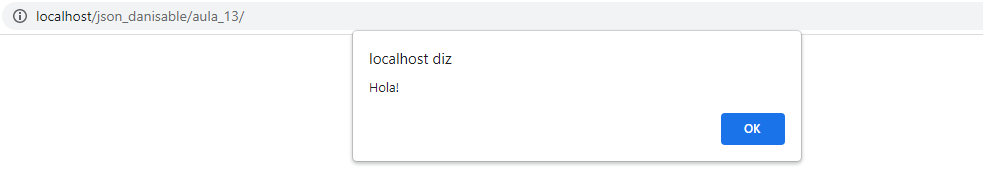
decirHola();

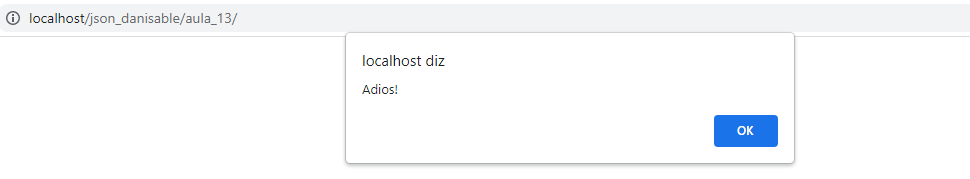
puerta.decirAdios();

</script>

</body>

</html>





# Aula 14 - Atributos dentro de funções

**aula\_14\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

nombre: "puerta de roble",

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

getDados: function(){

alert("id: " + this.id);

alert("nombre: " + this.nombre);

alert("precio: " + this.precio);

alert("etiquetas: " + this.etiquetas);

}

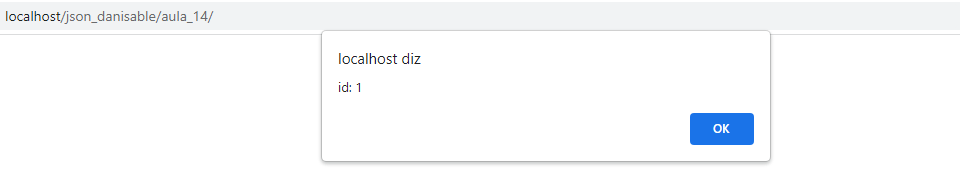
}

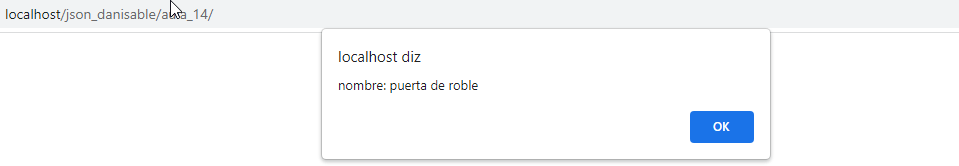
puerta.getDados();

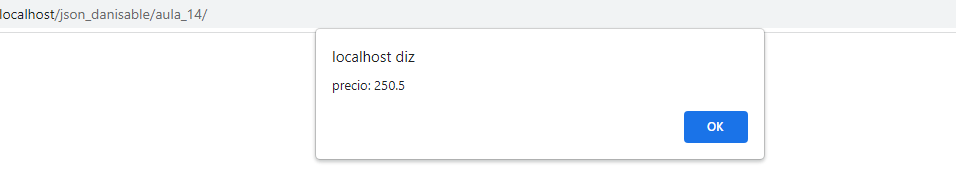
</script>

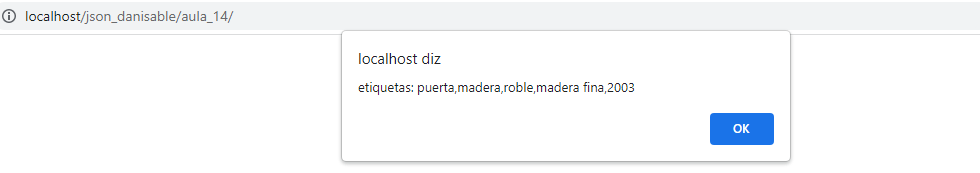
</body>

</html>









# Aula 15 - Funções múltiplas

**aula\_15\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

nombre: "puerta de roble",

saludos: function(){

alert("Saludos!");

},

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

hola: function(){

alert("Hola!");

this.saludos();

this.adios();

},

adios: function(){

alert("Adios!")

}

}

puerta.hola();

</script>

</body>

</html>







# Aula 16 - Parâmetros de funções

**aula\_16\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

alert("Resultado: " + res);

},

restaIVA: function(precio, iva){

var res = precio - iva;

alert("Resultado: " + res);

}

}

puerta.sumaIVA(puerta.precio, 20);

puerta.restaIVA(puerta.precio, 20);

</script>

</body>

</html>





# Aula 17 - Retorno de valor de funções

**aula\_17\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

return res;

}

}

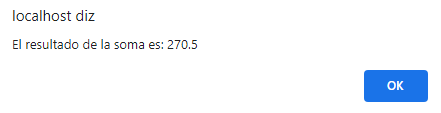
var resultado = puerta.sumaIVA(puerta.precio, 20);

alert("El resultado de la soma es: " + resultado);

</script>

</body>

</html>



# Aula 18 - Estruturas de controle com JSON

**aula\_18\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

return res;

},

venta: true,

inventario: null

}

if(puerta.venta){

alert("Si, hay puertas a la venta!");

} else {

alert("Ya no temos existencias de esta puerta!");

}

</script>

</body>

</html>



# Aula 19 - Criar classes com funções em JS

**aula\_19\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

return res;

},

venta: true,

inventario: null

}

function Casa(valor, terreno, color){

// atributos

this.valor = valor;

this.terreno = terreno;

this.color = color;

// metodos

this.darTecho = function(){

alert("Todos los habitantes de la casa tienen techo");

};

this.darProteccion = function(){

alert("Todos los habitantes de la casa estan protegidos");

};

}

</script>

</body>

</html>

# Aula 20 - Criar objetos com funções

**aula\_20\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

return res;

},

venta: true,

inventario: null

}

function Casa(valor, terreno, color){

// atributos

this.valor = valor;

this.terreno = terreno;

this.color = color;

// metodos

this.darTecho = function(){

alert("Todos los habitantes de la casa tienen techo");

};

this.darProteccion = function(){

alert("Todos los habitantes de la casa estan protegidos");

};

}

// creacion de objeto

var casa1 = new Casa(350000, 250, "Rojo");

alert(casa1.valor);

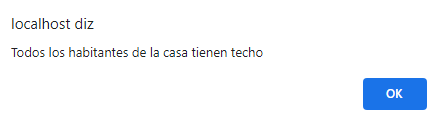
casa1.darTecho();

</script>

</body>

</html>





# Aula 21 - Prototype

**aula\_21\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

return res;

},

venta: true,

inventario: null

}

function Casa(valor, terreno, color){

// atributos

this.valor = valor;

this.terreno = terreno;

this.color = color;

// metodos

this.darTecho = function(){

alert("Todos los habitantes de la casa tienen techo");

};

this.darProteccion = function(){

alert("Todos los habitantes de la casa estan protegidos");

};

}

// creacion de objeto

var casa1 = new Casa(350000, 250, "Rojo");

console.log(casa1.\_\_proto\_\_);

console.log(puerta.\_\_proto\_\_);

var nombre = "Daniel";

console.log(nombre.\_\_proto\_\_);

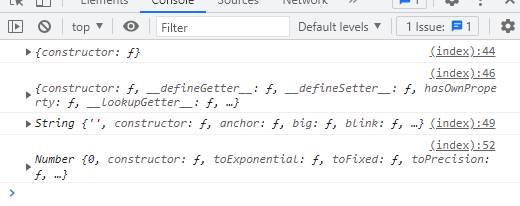
var numero = 564164;

console.log(numero.\_\_proto\_\_);

</script>

</body>

</html>



# Aula 22 - Criar atributos e métodos fora de classe

**aula\_22\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var puerta = {

id: 1,

precio: 250.50,

etiquetas: ["puerta", "madera", "roble", "madera fina", "2003"],

sumaIVA: function(precio, iva){

var res = precio + iva;

return res;

},

venta: true,

inventario: null

}

function Casa(valor, terreno, color){

// atributos

this.valor = valor;

this.terreno = terreno;

this.color = color;

// metodos

this.darTecho = function(){

alert("Todos los habitantes de la casa tienen techo");

};

this.darProteccion = function(){

alert("Todos los habitantes de la casa estan protegidos");

};

}

// creacion de objeto

var casa1 = new Casa(350000, 250, "Rojo");

casa1.noPuertas = 8;

casa1.contarHabitantes = function(){

alert("Hay 20 habitantes");

}

console.log(casa1.noPuertas);

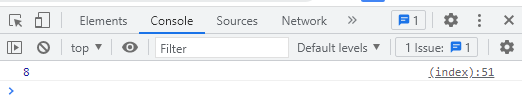
casa1.contarHabitantes();

</script>

</body>

</html>





# Aula 23 - Objetos Json dentro de Objetos Json

**aula\_23\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

automoviles: {

auto1: "Ford",

auto2: "Nissan",

auto3: "Ferrari"

},

direccion: "Cerrada Nopales No 5"

}

alert(daniel.nombre);

alert(daniel.automoviles.auto2);

</script>

</body>

</html>





# Aula 24 - Arrays dentro de Objetos Json

**aula\_24\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

automoviles: {

auto1: ["Ford", "Mustang", "2002"],

auto2: ["Nissan", "GTR", "2017"],

auto3: ["Ferrari", "Enzo", "2001"]

},

direccion: "Cerrada Nopales No 5"

}

alert(daniel.automoviles.auto2[1]);

</script>

</body>

</html>



# Aula 25 - Objetos dentro de Arrays

**aula\_25\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

automoviles: [

{

marca: "Ford",

modelo: "2020",

piezas: ["No3455", "No9845", "No988766"]

},

{

marca: "Nissan",

modelo: "2000",

piezas: ["No2345", "No78", "No64"]

},

{

marca: "Ferrari",

modelo: "1998",

piezas: ["No34565", "No23", "No608"]

}

],

direccion: "Cerrada Nopales No 5"

}

alert(daniel.automoviles[0].marca);

alert(daniel.automoviles[1].modelo);

alert(daniel.automoviles[2].piezas[1]);

</script>

</body>

</html>







# Aula 26 - Ciclo For In com JSON

**aula\_26\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5"

}

for(d in daniel){

alert(d + ": " + daniel[d]);

}

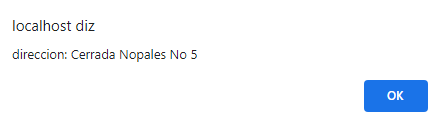
</script>

</body>

</html>







# Aula 27 - Percorrer arrays com JSON

**aula\_27\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5",

carros: ["Ford", "Nissan", "Ferrari", "Jaguar", "Chevrolet", "Lotus", "Honda"]

}

for(x in daniel.carros){

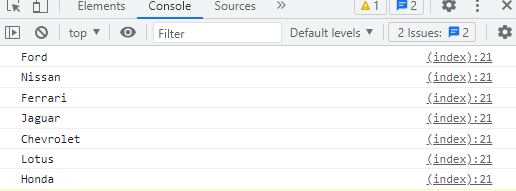
console.log(daniel.carros[x]);

}

</script>

</body>

</html>



# Aula 28 - Ciclos aninhados com JSON

**aula\_28\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5",

carros: [

{nombre: "FORD", modelos: ["Mustang", "Fusion", "F150"]},

{nombre: "NISSAN", modelos: ["GTR", "Centra", "Tsuru"]},

{nombre: "FERRARI", modelos: ["Enzo", "Maranello", "Carrera"]}

]

}

for(x in daniel.carros){

console.log(daniel.carros[x].nombre);

for(y in daniel.carros[x].modelos){

console.log(daniel.carros[x].modelos[y]);

}

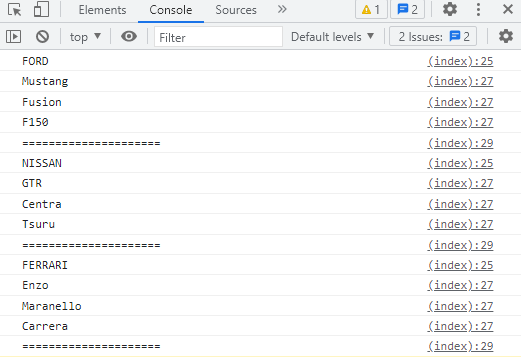
console.log("=====================");

}

</script>

</body>

</html>



# Aula 29 - Função Stringify de JSON

- Stringify transforma um objeto em uma string

**aula\_29\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5"

}

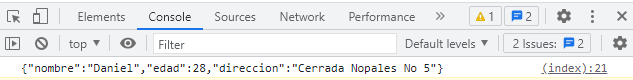
var danielString = JSON.stringify(daniel)

console.log(danielString);

</script>

</body>

</html>



# Aula 30 - Função Parse de JSON

**aula\_30\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5"

}

var danielString = JSON.stringify(daniel);

var danielObj = JSON.parse(danielString);

alert(danielObj.nombre);

</script>

</body>

</html>



# Aula 31 - JSON Local Storage

## Armazenando dados dentro do navegador

**aula\_31\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5"

}

var danielString = JSON.stringify(daniel);

// Almacenando el obj daniel en el localStorage

localStorage.setItem("ObjetoDaniel", danielString);

// Recuperando el obj daniel del localStorage

var datosStorage = localStorage.getItem("ObjetoDaniel");

alert(datosStorage);

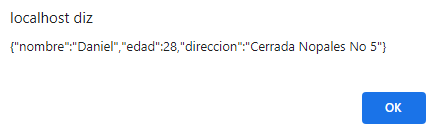
var danielObj = JSON.parse(datosStorage);

alert(danielObj.direccion);

</script>

</body>

</html>





# Aula 32 - Getters e Setters em JSON

**aula\_32\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5",

getNombre: function(){

return this.nombre;

},

setNombre: function(nombre){

this.nombre = nombre;

}

}

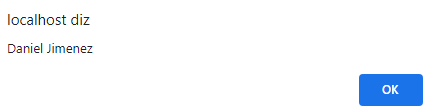
daniel.setNombre("Daniel Jimenez");

alert(daniel.getNombre());

</script>

</body>

</html>



# Aula 33 - JSON em arquivo externo

**aula\_33\index.html**

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<script>

var daniel = {

nombre: "Daniel",

edad: 28,

direccion: "Cerrada Nopales No 5",

}

alert(daniel.nombre);

</script>

</body>

</html>

**aula\_33\informaccion.json**

{

"nombre": "Daniel",

"edad": 28,

"direccion": "Cerrada Nopales No 5"

}

# Aula 34 - Comparação entre XML e JSON

// json - automovil

{

"empresa": "Honda",

"auto": "Accord",

"precio": 350000

}

// xml - automovil

<automovil>

<empresa>Honda</empresa>

<auto>Accord</auto>

<precio>350000</precio>

</automovil>