framework: conjunto de codigo prehecho para armar en base al codigo que ya existe

Img

Alt: Texto por si no aparece la imagen

Src: Donde se encuentra la imagen

Form:

PlaceHolder: Texto que va a aparecer

El padding solo lo aplica a los hijos de esas sección, no al hijo del hijo

Selectores avanzados

d

Pseudo-Clases y Pseudo-Elementos

:hover – Al pasar el mouse por encima del elemento

:active – Estilos al tener el clic presionado

:focus – Al estar dentro del objeto

Absolute: Propiedad que pone el

¿Qué es un programa?

Secuencia de instrucciones y comandos para realizar una tarea concreta.

Software programación: especializados para mostrar el código de forma sencilla.

Software de aplicación: Netflix, Facebook.

Javascript: un lenguaje de programación interpretados, que se usa en sitios web para ejecutar acciones del lado de cliente, estando embebido en el código fuente de la página web

Lenguaje compilado: lee el código y si hay algún error no compila.

Lenguaje Interpretado: No va a funcionar donde hay un error.

Javascript lado servidor: Netscape Introdujo una implementación de script del lado del servidor.

Ventajas: Fácil de aprender. Suficiente maduro. Se lleva bien con los datos. Tiene aplicación en todo. Jamás de tejará la comunidad.

Condicionales.

Le preguntamos a la máquina si algo es verdadero y falso.

Operadores lógicos.

And(&&):

Or(||):

Not(!):

Console.log: Imprime en consola.

Alert: saca una alerta al navegador.

Prompt: Saca ventana para pedir un dato.

Variable(var): Para poder guardar valores:

Var=let

Let: otra manera de declarar variable

=: asignar.

==: comparar.

Variable: Un símbolo que permite identificar un elemento.

Operadores Matemáticos.

Suma (+)

Resta (-)

Multiplicación (\*)

División (/)

Residuo de una división (%), Si se hace una división, lo que queda abajo

Exponencial (\*\*)

String

El tipo de datos String es un dato especial y cuenta con diferentes métodos.

* charAt(): Tomar primer carácter de un string
* indexOf():
* replace():
* toLowerCase(): Para pasar todo a minusculas
* toUpperCase(): Para pasar todo a mayusculas
* slice(): Parte la palabra en
* split():

Si intento juntar un nombre y un texto, JS convierte el número en string.

Arrays(Arreglos)

push(): Meter un dato al final

splice(): Sustituir el que estaba

sort: Ordena de forma ascendente

reverse: Ordena de fomra descendente

Operadores de comparación

Se usan para comparar valores y devolver un resultado sea True o False.

Bucle

For (i in arr): en vez de estar poniendo las veces que queremos que se repita el texto, si ponemos una variable en (in) otra variable más, entonces nos va a repetir el proceso para la variable “arr”.

For(i in arr) = for(var i=0; i<arr.length; i++){}

Bootstrap

Framework de CSS creado por twitter

* Es fácil de instalar
* Soporte para multiples navegadores
* Facil de integrar
* Personalizable
* Bien estructurado
* Multidispositivo

Nos deja acomodar de manera sencilla en contenedores

Grid

Funciones

Una función es un grupo de instrucciones con un objetivo en particular y que se ejecuta al ser llamada desde otra función o procedimiento. Una función puede llamarse multiples veces o llamarse a si misma.

Function myFunction(){

Console.log(“MyFuncion”)

}

myFunction()

Parametros

Variavles necesarias para que la función se ejecute correctamente

document: Que entre al documento (html)

Metodos

getElementById(‘’): Mencionamos que dentro del documento, vaya a la que tiene id de ‘’

JSON(JavaScript Object Notation)

Formato para el intercambios de datos, básicamente JSON describe los datos con una sintaxis dedicada que se usa para identificar y gestionar datos, JSON nació como una alternativa a XML.

Objetos

Un objeto es una colección de atributos y métodos.

En un coche, los atributos son colores, ruedas, motor, marca, etc. y los métodos son arranca(), frena(), etc.

Los atributos es algo particular, los métodos algo general.

Ejemplo:

objeto perro.

Sus atributos son la nombre, raza, color, género.

ayudanteDeSanta = {

nombre=ayudante,

raza = galgo,

color = café,

género = macho

}

Sus métodos son corre(), come(), duerme(), ladra()

La estructura de un objeto literal está limitada por llaves, donde se encapsula cada identificador asignándole un valor literal, e un formato clave: valor.

Si vamos a hacer uso del objeto mas adelante, podemos almacenarlo en una variable asignándole un valor.

Var nombreObjeto = {

Identificador1: valor,

Identificador2: valor,

}

Todos los primitivos excepto null y undefined se tratan

Clase

El constructor es un método especial que se ejecuta automáticamente cuando se crea una instancia de esa clase.

Su función es inicializar el objeto y sirve para asegurarnos que el objeto

This: le digo que use el elemento que está en cierta posición

Herencia

Es una clase nueva que se crea a partir de una clase existente, esta clase comparte los atributos y comportamientos

Asincrono

Javascript solo hace una tarea a la vez

Call stack: donde se va a ejecutar todo nuestro código

Var algo = function(){}, se puede resumir a

Var algo => (){}

Callstack

Cola de tareas, cuando ejecuto una función asíncrona, entonces lo va a mandar al callstack

Callback

Una función que recibe como otro parámetro otra función.

Que quieres que pase después de que la función termine

Arquitectura Cliente-Servidor

**CLIENTE:** Hace peticiones, Recibe las respuestas

Quiere decir que pide los recursos al servidor, recibe la respuesta del servidor y se la muestra al usuario de manera adecuada

**SERVIDOR:** Recibe peticiones, envía respuesta

Se conecta a una base de datos para encontrar esa información solicitada

**TIPOR DE SERVIDORES:** Servidores de bases de datos, de correo electrónicos, imágenes, WEB, chat, Groupware, proxy, FTP

**TCP/IP:** El TCP es el transporte que manda la información, luego con el IP es el internet.

**URL(Uniform Resource Locator):** El link, que tiene el protocolo *http,* dominio *Google.com,* camino o path *carpeta,* recurso *index.html*

**ENVIAR DATOS AL SERVIDOR:** s

**VERBOS DE HTTP(Protocolo de transferencia de HTTP)**

Get: Recibe información.

Post: Envía información.

Put: Modifica información de un dato existente.

Delete: Borra información.

Patch: Casi lo mismo que el Put

Headers: Meta información relevante para que el cliente o el servidor conozca más acerca de la petición

**Promesas**

**Es uado para interacción asíncrona**

**Se compone de resolve cuando el objetivo de la promesa fue correcta y de reject**

Como por ejemplo, si vamos a un fast food, la promesa es el ticket con lo que dice que es lo que te van a dar, esta resuelta si te dan todo correctamente

Git

Sistema de control de versiones

Comandos básicos

Git init: Inicia una reposición

Mkdir: Crear Carpeta

Git add: agregar a git

Git add -–all: se agregan todos los archivos

Git status: estatudos

Git commit: agregar algo importante (git commit -m””)

Git log: te dice lo que ha pasado

Git Branch : Se hace una prueba de un archivo. Se crea una rama del árbol

-d (nombre): borrar una rama

Git checkout: ir a un Branch

Git merge: unifica el de master y el branch

Git clone: Se clona lo que haya en algo de github

Git push: mandar a giuyhub

**Node JS**

Npm init

Node (Archivo)

Npm install request –-save: Le digo que instela la paquetería request y diciendo --save digo que lo guarde en el json inciado con npm init

Node: Programar Javascript en consola

En el archivo de index.js(o el nombre que le hayamos puesto en el npm init, para poder leer los módulos instalados, que en el caso anterior fue request, se escribe const request = require(‘request’), y ña palabra require entra a la carpeta de node modules y lo que vaya adentro en paréntesis y comillas, la carpeta.

JSON.parse(body): Limpia el JSON adquirido y así podemos sacar por ejemplo el name, o lo que sea

Consola

Nano (archivo)

Express.js

Cookie-parser

Body-parser

Cawsey

Sudo – Permisos de super usuario

Routing

Path

Nano

vim

Programación Funcional

Es un paradigma de programación, una manera de construir software a partir de solamente funciones.

La programación funcional es declarativa.

Tiende a ser más concisa, más predecible y más fácil de leer.

Se compone de funciones puras (no utilizar librerias, funciones o procesos externos)

**Programación declarativa e imperativa**

**Declarativa:** Declaras una base de conocimiento, restricciones o afirmaciones para resolver el problema pero no dices como llegar a la solución.

**Imperativa:** Declaras un algoritmo y una serie de instrucciones o pasos para llegar a una solución.

**Función Pura**

No tiene efectos de estado que la efecten, lo que quiere decir que siempre que le pasemos un número.

A esta dará un resultado B.

**Función Impura.**

Tiene efectos de estados porque un agente externo como es Math.random() esta aectando la fiabilidad

Entre programación orientada a objetos y programación funcional no hay mejor o peor, todo depende de la complejidad del problema y que se necesita hacer.

**Funciones y métodos**

forEach(): es un for que pasa por cada elemento, es como for(i=0; i<var.length; i++)

filter(): Crea un nuevo array con todos los elementos que cumplan la condición implemantada para la función dada.

map(): Crea un nuevo array con los resultados de la llamada a la función indicada aplicados a cada uno de sus elementos.

**Base de Datos**

Es un conjunto de datos relacionados entre si

**Propiedades**

* Independencia lógica y física de datos
* Redundancia mínima
* Acceso concurrente por parte de múltiples usuarios
* Integridad de datos

**Sistema de gestión de base de datos**

Son un tipo de software muy específico, dedicando a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta

Una de la diferencia de base de datos relacional y no relacional es que en una relacional debes de tener todos los datos, mientras que las no relacionales no necesitas tener todos los datos.

Preguntas

¿Qué es el radix en parseInt?

Terminal

Es lo mismo que un sistema operativo gráficamente pero en texto.

Clear: borrar lo que hay en sistema

code .: abrir Visual Code del directorio

mkdir: crear carpeta.

Touch: crear archivo

cat: ver archivo

vi: editar visual studio code en terminal

ls: aparece todo lo que

rm -rf: borrar carpeta

rm: borra archivo

Bootstrap

Evento.listener

What most schools dont teach

Javascript Google Sheet

Cinetop.com

bash

.vue

Sopitas.com

Foundation

Wappalyzer

Build width

Blisk

Modal

Normalize

Color picker

Grid bootsrap

bulma

osana

trello

scrum

canvas

y combinator

let my google that for you

flight icon

endeavor

type script

Conferencia Code Splitting

Webpack nos ayuda

Llamar a piezas de código cuando se usa y no llama a todo.

Se usa para mejorar el perfomance de la aplicación

Menor tiempo de carga inicial

Mejor experiencia de usuario

Usarlo

Usarlo cuando JS esté pesado

Todo lo que es temporal

Por páginas (rutas)

Códigos de estado HTTP

vim