

**Universidad Mariano Gálvez de Guatemala**

**Facultad de Ingeniería en Sistemas**

Proyecto final, Introducción a los Sistemas de Computo. Documentación externa del mini proyecto, juego snake en html, javascript y css

**Estudiante:**

Marlen Breneli Escobar Lima

**Carnet**:

20-15527

**Catedrático (a):**

Ing. Gerson Altamirano

**Jalapa, 9 de junio de 2020**

# Juego Snake creado en html, JavaScript y CSS

## Manual de usuario Snake Game

## Paso 1

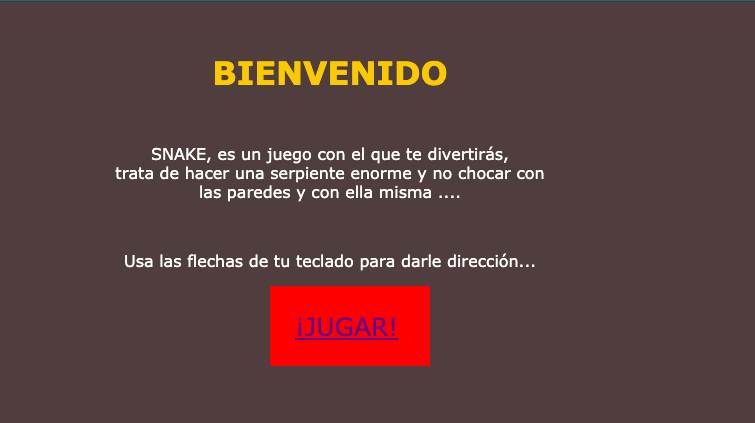
Página principal



Dar clic a la imagen de snake game, que lo direccionará a la página inicial del juego

## Paso 2

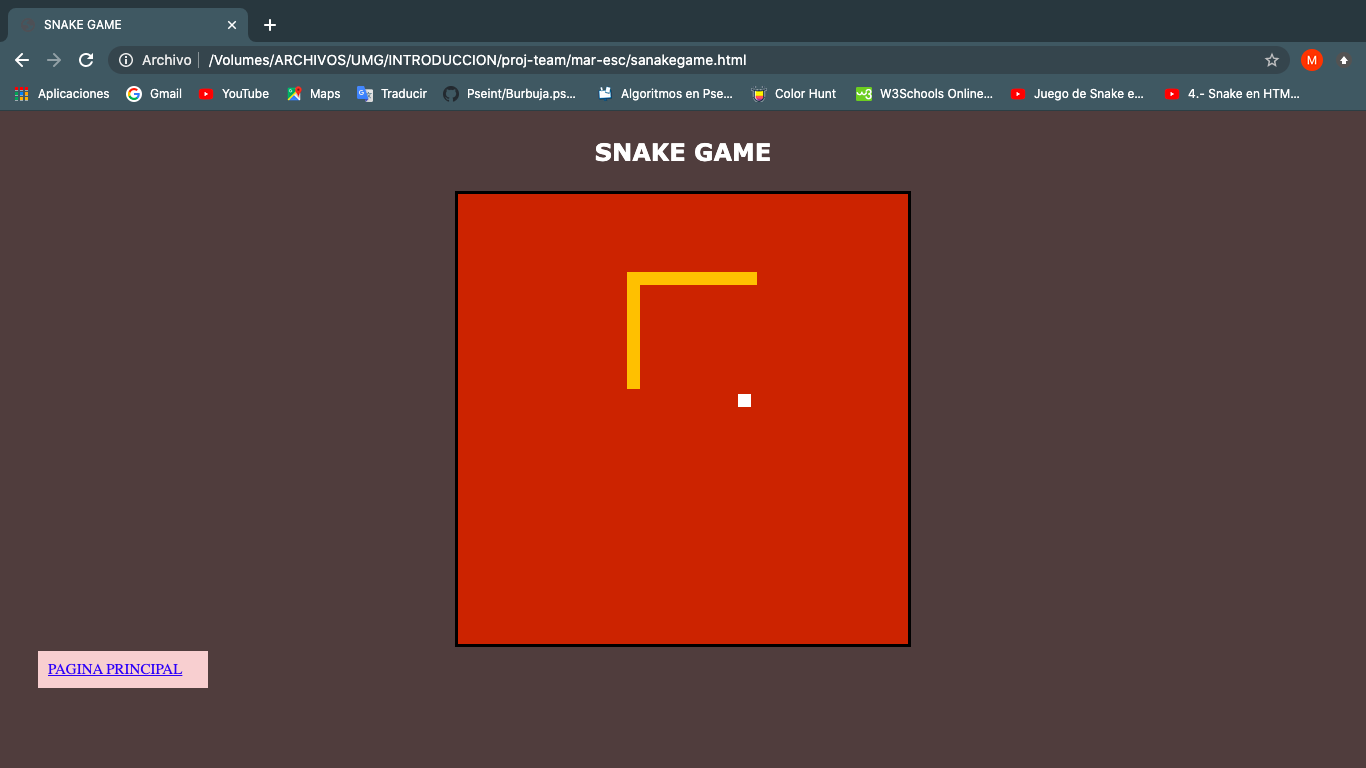
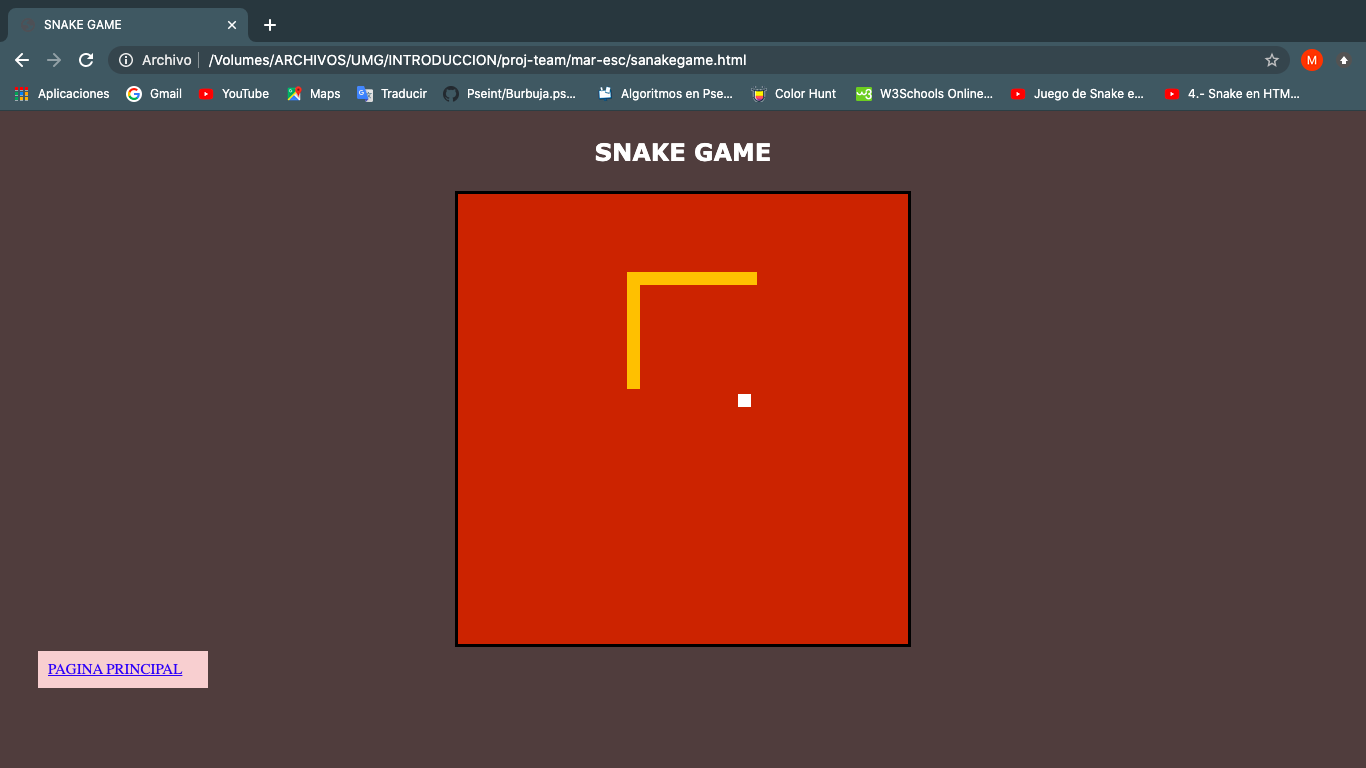
Página inicial del juego



Se encontrarán las indicaciones mínimas. Dar clic para ¡jugar!

## Paso 3

Pagina del juego



Utilizar las flechas del teclado para darle dirección la serpiente.

## Paso 4

Presionar este enlace para regresar al menú principal donde se encuentran alojados la mejor gama de juegos.

## Descripción del problema

Creación de variables, funciones y etiquetas que permitan el funcionamiento del juego snake, dado que este consiste; en la funcionalidad de un cuadrado que al momento de chocar con otro deberá ir creciendo, a su vez si se intersecta con su propio cuerpo o choque con las paredes, el juego será reiniciado.

## Autor

Marlen Breneli Escobar Lima

## Algoritmo

Diagrama de flujo

Función para llamar funciones

funciones

dibujos

Estilos CSS

Rutinas de detección de colisiones

canvas

Variables globales

Snake.html

Inicio.html

Snake.js

INICIO

SetInterval()

FIN

## Diccionario de datos

## Colisión:

Una colisión en informática, es una situación que se produce cuando dos entradas distintas producen una misma salida.

## Entrada:

Es una interacción humano-computadora, es la información que produce el usuario, con la intención de tener el control del programa.

## Canvas:

Es un elemento html, el cual sirve para dibujar gráficos usando scripts, normalmente usados en JavaScript.

## Variable:

Una variable es donde se guardan, se muestran, se asignan valores, se representan y recuperan los datos utilizados para la creación de un programa.

## Enlace:

Sirve para dirigir una página a otra.

## Función:

Son un conjunto de líneas que se encargan de realizar una tarea en concreto.

## Código fuente

## Html: snake.html

<!DOCTYPE html>

<html lang=*"es"*>

<head>

<meta charset=*"UTF-8"*>

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

<title>SNAKE GAME</title>

<script src=*"snakegame.js"*></script>

<link rel=*"stylesheet"* href=*"snakegame.css"*>

</head>

<body onkeydown=*"control(event);"*>

<h1>

</h1>

<h2>

SNAKE GAME

</h2>

<canvas id=*"canvas"* width=*"450"* height=*"450"*></canvas>

<br>

<nav>

<a href=*"/index2.html"*>PAGINA PRINCIPAL</a>

</nav>

</body>

</html>

## Snake.css

body{

background-color: #4d3e3e;

text-align: center;

}

h2{

color: #ffffff;

padding: 5px;

font-family: verdana;

}

canvas{

border: solid;

background-color: #bb3b0e;

}

nav{

text-align: left;

background-color: #f1d1d1;

padding: 10px;

margin-left: 30px;

margin-right: 1150px;

font-size: 15px;

}

## Inicio.html

<!DOCTYPE html>

<html lang=*"es"*>

<head>

<meta charset=*"UTF-8"*>

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

<title>SNAKE GAME</title>

<link rel=*"stylesheet"* href=*"inicio.css"*>

</head>

<body >

<h1>

BIENVENIDO

</h1>

<p>

SNAKE, es un juego con el que te divertirás, <br>

trata de hacer una serpiente enorme y no chocar con <br>

las paredes y con ella misma ....

</p>

<br>

<p>

Usa las flechas de tu teclado para darle dirección...

</p>

<nav>

<a href=*"sanakegame.html"*>¡JUGAR!</a>

</nav>

</body>

</html>

## Inicio.css

body{

text-align: center;

background-color: #4d3e3e;

}

h1{

font-family: Verdana;

color: #f3c623;

padding: 30px;

}

p{

font-family: Verdana;

color: white;

}

nav{

background-color:#E71414;

font-family: Verdana, Geneva, Tahoma, sans-serif;

text-align: left;

padding: 25px;

margin-left: 615px;

margin-right: 575px;

font-size: 25px;

}

## Snake.js

*//variables globales*

var velocidad = 80;

*//variable tamaño*

var tam = 13;

*//rutinas de detección de coliciones*

*//clase que hereda las demás clases y llega los métodos de tetención de lolliciones*

class objeto {

constructor(){

this.tam = tam;

}

choque(obj){

var difx = Math.abs(this.x - obj.x);

var dify = Math.abs(this.y - obj.y);

if(difx >= 0 && difx < tam && dify >= 0 && dify < tam){

return true;

} else {

return false;

}

}

}

*// dibujo de la cabeza de la serpiente (cuadrito)*

*// con extends declaro que es una subclase*

class Cola extends *objeto* {

constructor(x,y){

super();

this.x = x;

this.y = y;

this.sig = null;

}

*// método de dibujo*

*//siguiente, para indicar que los cuadritos sigan a la cabeza*

dibujar(ctx){

if(this.sig != null){

this.sig.dibujar(ctx);

}

ctx.fillStyle = *"#fcbf1e"*;

ctx.fillRect(this.x, this.y, this.tam, this.tam);

}

setxy(x,y){

if(this.sig != null){

this.sig.setxy(this.x, this.y);

}

this.x = x;

this.y = y;

}

met(){

if(this.sig == null){

this.sig = new Cola(this.x, this.y);

} else {

this.sig.met();

}

}

verSiguiente(){

return this.sig;

}

}

*//dibujo de la comida*

class Comida extends *objeto* {

constructor(){

super();

this.x = this.generar();

this.y = this.generar();

}

generar(){

var num = (Math.floor(Math.random() \* 59))\*5;

return num;

}

colocar(){

this.x = this.generar();

this.y = this.generar();

}

dibujar(ctx){

ctx.fillStyle = *"#ffffff"*;

ctx.fillRect(this.x, this.y, this.tam, this.tam);

}

}

*//Objetos del juego*

var cabeza = new Cola(0,0);

var comida = new Comida();

*//variables para lograr el movimiento*

*//variables booleanas, van a permitir habilitar o bloquear los ejes x, y*

var ejex = true;

var ejey = true;

*// variables para dar dirección al movimiento*

var xdir = 0;

var ydir = 0;

*//función para dar movimiento*

function movimiento(){

var nx = cabeza.x+xdir;

var ny = cabeza.y+ydir;

cabeza.setxy(nx,ny);

}

*//función para los controles*

function control(event){

var cod = event.keyCode;

if(ejex){

*//movimiento a la tecla de arriba*

if(cod == 38){

ydir = -tam;

xdir = 0;

ejex = false;

ejey = true;

}

*//movimiento a la tecla de abajo*

if(cod == 40){

ydir = tam;

xdir = 0;

ejex = false;

ejey = true;

}

}

if(ejey){

*//movimiento a la tecla de la derecha*

if(cod == 37){

ydir = 0;

xdir = -tam;

ejey = false;

ejex = true;

}

*//movimiento a la tecla de la izquierda*

if(cod == 39){

ydir = 0;

xdir = tam;

ejey = false;

ejex = true;

}

}

}

*// función para darle final al juego*

function findeJuego(){

xdir = 0;

ydir = 0;

ejex = true;

ejey = true;

cabeza = new Cola(0,0);

comida = new Comida();

alert(*"¡has perdido!"*);

}

*//función para el choque con el canvas*

function choquepared(){

if(cabeza.x < 0 || cabeza.x > 449 || cabeza.y < 0 || cabeza.y > 449){

findeJuego();

}

}

*//función para el choque del cuerpo de la serpiente*

function choquecuerpo(){

var temp = null;

try{

temp = cabeza.verSiguiente().verSiguiente();

}catch(err){

temp = null;

}

while(temp != null){

if(cabeza.choque(temp)){

*//fin de juego*

findeJuego();

} else {

temp = temp.verSiguiente();

}

}

}

*//motor de juego*

*//dibujar la base*

function dibujar(){

var canvas = document.getElementById(*"canvas"*);

var ctx = canvas.getContext(*"2d"*);

ctx.clearRect(0,0, canvas.width, canvas.height);

*//crear los dibujos*

cabeza.dibujar(ctx);

comida.dibujar(ctx);

}

*// llamar a todas las funciones*

function animar(){

choquecuerpo();

choquepared();

dibujar();

movimiento();

if(cabeza.choque(comida)){

comida.colocar();

cabeza.met();

}

}

setInterval(*"animar()"*, velocidad);